# Computer Vision Pipeline for the Automated Inspection of Photovoltaic Plants

## Digitale Bildverarbeitungsmethode für die Automatische Inspektion von Photovoltaikanlagen

Der Technischen Fakultät
der Friedrich-Alexander-Universität
Erlangen-Nürnberg

zur

## Erlangung des Doktorgrades Dr.-Ing.

vorgelegt von

Lukas Bommes
aus Möchengladbach

# Contents

# Contents

# Contents

## Publication List

This thesis comprises of the following peer-reviewed journal articles.

[1]  L. Bommes, T. Pickel, C. Buerhop-Lutz, J. Hauch, C. Brabec, and I. Peters, "Computer vision tool for detection, mapping, and fault classification of photovoltaics modules in aerial IR videos," *Progress in Photovoltaics: Research and Applications*, vol. 29, no. 12, pp. 1236–1251, 2021. DOI: 10.1002/pip.3448.

[2]  L. Bommes, M. Hoffmann, C. Buerhop-Lutz, T. Pickel, J. Hauch, C. Brabec, A. Maier, and I. Peters, "Anomaly detection in IR images of PV modules using supervised contrastive learning," *Progress in Photovoltaics: Research and Applications*, vol. 30, no. 6, pp. 597–614, 2022. DOI: 10.1002/pip.3518.

[3]  L. Bommes, T. Pickel, C. Buerhop-Lutz, J. Hauch, C. Brabec, and I. Peters, "Georeferencing of photovoltaic modules from aerial infrared videos using structure-from-motion," *Progress in Photovoltaics: Research and Applications*, vol. 30, no. 9, pp. 1122–1135, 2022. DOI: 10.1002/pip.3564.

# Abstract

Due to their large size utility-scale PV plants often contain anomalous PV modules and components that lead to accelerated degradation, pose fire hazards, and reduce power output, yield, and consequently the profitability of the plant. Hence, regular inspections are recommended. A fast and contactless inspection method is drone-based IR thermography, which detects PV plant anomalies based on their increased temperatures. However, IR thermography also generates large amounts of IR videos, manual sighting of which is economically infeasible.

Thus, the goal of this thesis is to develop a computer vision pipeline for the automated processing of drone-acquired IR videos of utility-scale PV plants, which identifies anomalous PV modules in the video and localizes them in the plant to facilitate targeted repairs. To this end, we combine algorithms for instance segmentation, multi-object tracking, incremental structure from motion, and supervised image classification, employing recent deep learning methods wherever applicable.

The pipeline is fully automated, works independently of the plant layout, facilitates a high throughput of up to 45000 modules ($\sim 10.6\,\mathrm{MW}_p$) per hour, and supports both automated flights, useful for large plants, and manual zero-setup flights for small plants. Three anomaly detection methods are available, which—thanks to the use of a very large dataset with over 6.5 million IR images of 152669 PV modules from ten different PV plants—offer high accuracy, robustness to changes in environmental conditions, good generalization across plants, and sensitivity to string anomalies and all common PV module anomalies. Apart from the IR video, only a coarse GPS trajectory of the drone is required for the creation of a geographic map, on which anomaly detection results can be intuitively visualized. The pipeline successfully processes $99.3\,\%$ of all PV modules in our dataset, confirming its robustness to acquisition errors.

Apart from the theoretical contributions, the pipeline is published as an open-source software tool to accelerate research in the field.

Our contributions lower the barrier to regular inspections of utility-scale PV plants, improving their reliability, safety, durability, power output, yield, and profitability, which is essential for the future success of solar PV as a global source of clean and renewable electricity.

# Zusammenfassung

Aufgrund ihrer großen Ausmaße enthalten PV-Anlagen oft anomale PV-Module und Komponenten, die zu einer beschleunigten Degradation führen, eine Brandgefahr darstellen und die Leistung, den Ertrag und damit die Rentabilität der Anlage verringern. Daher werden regelmäßige Inspektionen empfohlen. Eine schnelle und berührungslose Inspektionsmethode ist die drohnengestützte IR-Thermografie, die Anomalien in PV-Anlagen anhand ihrer erhöhten Temperaturen erkennt. Allerdings erzeugt die IR Thermografie auch große Mengen an IR-Videos, deren manuelle Sichtung nicht wirtschaftlich ist.

Das Ziel dieser Arbeit ist daher die Entwicklung einer Computer Vision Pipeline für die automatisierte Verarbeitung von mit Drohnen aufgenommenen IR-Videos von PV-Anlagen, die anomale PV-Module im Video identifiziert und diese in der Anlage lokalisiert, um gezielte Reparaturen zu ermöglichen. Zu diesem Zweck kombinieren wir Algorithmen zur Segmentierung und zum Tracking von Objekten, zur inkrementellen Rekonstruktion auf Basis der Kamerabewegung und zur Bildklassifizierung. Hierbei verwenden wir, sofern sinnvoll, Deep-Learning-Methoden.

Die Pipeline ist vollständig automatisiert, arbeitet unabhängig vom Anlagenlayout, ermöglicht einen hohen Durchsatz von bis zu 45000 Modulen ($\sim 10.6\,\mathrm{MW}_p$) pro Stunde und unterstützt sowohl automatisierte Flüge, die für große Anlagen nützlich sind, als auch manuelle Flüge für kleine Anlagen, die keine vorherige Konfiguration erfordern. Es wurden drei Methoden zur Erkennung von Anomalien entwickelt. Diese bieten, dank der Verwendung eines sehr großen Datensatzes mit über 6 5 Millionen IR-Bildern von 152669 PV-Modulen aus zehn verschiedenen PV-Anlagen, eine hohe Genauigkeit, Robustheit gegenüber Änderungen der Umgebungsbedingungen, gute Übertragbarkeit zwischen PV-Anlagen und Sensitivität für String-Anomalien und alle gängigen Modulanomalien. Neben dem IR-Video wird nur eine grobe GPS-Trajektorie der Drohne für die Erstellung einer geografischen Karte benötigt, auf der die Ergebnisse der Anomalieerkennung intuitiv visualisiert werden können. Die Pipeline verarbeitet erfolgreich 99.3 % aller PV-Module in unserem Datensatz, was ihre Robustheit gegenüber Fehlern bei der Videoaufzeichnung bestätigt.

Abgesehen von unseren theoretischen Beiträgen wird die Pipeline als Open-Source Software-Tool veröffentlicht, um die Forschung auf diesem Gebiet voranzutreiben.

Unser Beitrag senkt die Hürde für regelmäßige Inspektionen von großen PV-Freiflächenanlagen, welche die Zuverlässigkeit, Sicherheit, Lebensdauer, Leistung, Ertrag und Rentabilität der Anlagen verbessert. Dies ist unerlässlich für den künftigen Erfolg der Photovoltaik als globale Quelle für saubere und erneuerbare Elektrizität.

# 1 Introduction

This chapter motivates the need for automated PV plant inspection and introduces the objective of a computer vision pipeline for this task. Finally, the scope and structure of this thesis are described.

## 1.1 Motivation

Solar photovoltaics (PV) plays an important role in the global energy turnaround from mainly fossil fuel-based to renewable electricity generation [4]. Continuous technology improvements, decrease in capital cost, and increase in competition have led to substantial cost reduction and increased deployments of solar PV in the past decade [5, 6]. The globally installed capacity of solar PV reached $775\,\mathrm{GW}_p$ in 2020 [7], with annual capacity additions having hit an all-time record of $160\,\mathrm{GW}_p$ in 2021 [8]. In a conservative scenario, the IEA predicts the addition of another $940\,\mathrm{GW}_p$ until 2026 [8], more than doubling the current $3.2\,\%$ share of solar PV in the global electricity generation [9].

Economic success is essential for the future growth of solar PV. As investment objects, PV plants must deliver electricity reliably and for a long time, to ensure investment targets are met and electricity is generated at a competitive price. With the increasing share of solar PV in the power mix, reliability also becomes ever more important to ensure the stability of the future power grid [10]. However, PV plants are exposed to a multitude of stresses during installation and operation, such as mechanical and thermal load cycles, ultraviolet radiation, high potential gradients, and adverse climatic and weather conditions [11, 12]. This leads to material degradation and reduces the power output by $0.5\,\%$ to $0.6\,\%$ annually [13]. Consistent with this, Buerhop et al. [14] found that about $8\,\%$ of all modules and $2\,\%$ of all strings in a large number of analyzed PV plants exhibit some form of anomaly, causing an average power loss of about $6\,\%$ [15]. While not every anomaly is critical and requires an immediate repair, some anomalies lead to high temperatures that cause accelerated degradation [16, 17] and pose fire hazards, which is especially critical for building-integrated PV [18–20]. Some common anomalies also cause $100\,\%$ power loss of the affected module or string, for instance, a faulty connection with the rest of the plant [17]. If undiscovered, such anomalies can lead to substantial monetary losses, postponing the break-even point and profitability of the investment. Hence, regular inspection of PV plants is recommended, and in some countries, even a regulatory requirement [21]. Apart from preventing power losses, an inspection is also useful for due diligence, i.e., when buying a second-hand PV plant, or as part of the commissioning of a new plant to identify installation errors and facilitate warranty claims. Inspection will also gain relevance in the near future once the currently deployed and still young PV plants have aged and degraded.

In 2022, there were nearly 22000 PV plants in Germany alone with a capacity larger than $500\,\mathrm{kW}_p$ each [9]. Assuming a typical module power of 235 W, even the smallest of these plants contain over 2000 modules. The currently largest plant in Germany has even a capacity of $187\,\mathrm{MW}_p$ and contains over 465000 modules [22]. Hence, a manual inspection, for instance, measurement of the current-voltage-characteristic of each module, is not economically feasible.

Instead, infrared (IR) thermography has proven to be a fast, simple, accurate, and economic method for the inspection of PV plants above $100\,\mathrm{kW}_p$ [23–27], which is commonly used for more than a decade [28, 29]. In IR thermography, a thermal IR camera detects module or string anomalies due to their increased temperature caused by power dissipation. IR thermography is not only sensitive to all important module anomalies [28, 30, 31], but can also detect system errors, such as faulty connections and inverters [23], and external factors, such as soiling and shading [32, 33]. This is important, as power losses due to system errors and external factors outweigh losses due to module anomalies [34]. IR thermography is non-contact and non-destructive, thus does not interfere with plant operation. This is in contrast to other techniques, such as electroluminescence (EL) imaging, which requires disconnecting and reverse-powering each string [16]. Consequently, IR thermography is cheaper, as no additional equipment apart from the IR camera is needed [31, 35], and safer, as no intervention in the electrical system is required [27]. Furthermore, IR thermography is performed in daylight instead of the night as EL, saving special work permissions and night surcharges [16, 31].

IR thermography is often performed with small consumer-market drones [15, 16, 36–40], which offer low cost, technological maturity, and the ability of automated flights [41, 42]. Drones provide a $10 - 15$ fold speedup over conventional techniques, such as walking or ground-based robots [16], and reach a throughput of up to 48000 modules ($\sim 11\,\mathrm{MW}_p$) per hour of flight [43]. Not only are drones much cheaper than an airplane or helicopter, but they also facilitate low flight altitudes and hence close-up recordings. This reduces interference with nearby infrastructures, such as buildings [16, 44], and enables recording high-resolution thermal IR videos with a lower amount of atmospheric distortion using only a simple and inexpensive consumer-market microbolometer IR camera [25, 45].

## 1.2 Problem Statement

Drone-based IR thermography of utility-scale PV plants generates large amounts of IR videos, manual sighting of which is not economic. For instance, a typical $10\,\mathrm{MW}_p$ plant has about 42500 modules (235 W each), manual sighting of which would take 11.8 hours, assuming a human expert takes one second per module and works nonstop. However, this assumes the video is already processed nicely, presenting the human expert with a single image of each PV module. Assuming, a drone-mounted IR camera with 13 mm lens scanning individual rows of the plant at 8 frames per second from a distance of 15 m, each module is visible in about 40 subsequent video frames, resulting in 1.7 million module images. Just cropping and storing each module from each frame would take a human 196 days of nonstop work if he takes ten seconds per module image. On top of that,

the identity of each module would have to be tracked over subsequent frames to know, which of the cropped images belong to which module. But doing this manually is nearly impossible, as the videos (i) are highly repetitive, (ii) contain only a few or no visual reference points for orientation, and (iii) contain short interruptions due to the flat-field calibration performed every few minutes by the microbolometer camera. Even if anyone managed to perform all these tasks, he would still not know, where exactly each module is located in the plant—an important information for repairs. The best a human could do here, is count the modules starting from the beginning of each row. But this is, again, error-prone, and time-consuming. Another problem with manual processing occurs with respect to detecting and classifying module anomalies. Humans—even experts—are not very good at this, as they are inconsistent and tend to drift over time. A small hot spot that was not considered an anomaly before may well be regarded as a full-blown anomaly just two hours later.

Luckily, all the aforementioned tasks can be automated with computer vision algorithms that are orders of magnitude faster, less error-prone, more accurate, and more consistent than a human. Hence, the objective of this thesis is the development of an automated PV plant inspection system. Providing an economically feasible inspection solution is of major importance, as otherwise plant operators would likely forego regular inspections, possibly causing underperformance of the plant, premature failures, failed investments, and safety incidents, which could ultimately curb investments and hence future deployments of solar PV. Due to the importance of this topic, several previous works have proposed solutions for automated PV plant inspection [42, 46–56] and several companies offer PV plant inspection as a service [57–59]. However, no free or open-source software is readily available for this task.

The overall objective of an automated PV plant inspection solution can be broken down into the following subtasks that are also illustrated in fig. 1.1.

**Data acquisition** The planning and manual or automatic execution of the drone flight and capturing of IR videos of the PV plant that are compatible with the remaining processing steps. The GPS trajectory of the drone is captured as well.

**PV module detection** The objective of this task is to identify and optionally extract the regions in each video frame that resemble a PV module. Optionally the identity of each PV module is tracked over the consecutive frames, in which it is visible.

**PV module localization** In this task, the location of each detected PV module in the PV plant is determined either in absolute geocoordinates or relative to other modules. This helps to incorporate external data sources and can guide the maintenance crew to a specific module for performing a repair.

**Anomaly detection** For each detected PV module, it must be predicted, whether the module functions normally, or whether it exhibits an anomaly and needs to be repaired. Optionally, different types of anomalies may be differentiated.

**Reporting** Here, the results of the previous tasks are visualized, informing the maintenance crew about the location of abnormal PV modules. Reporting may also take economic

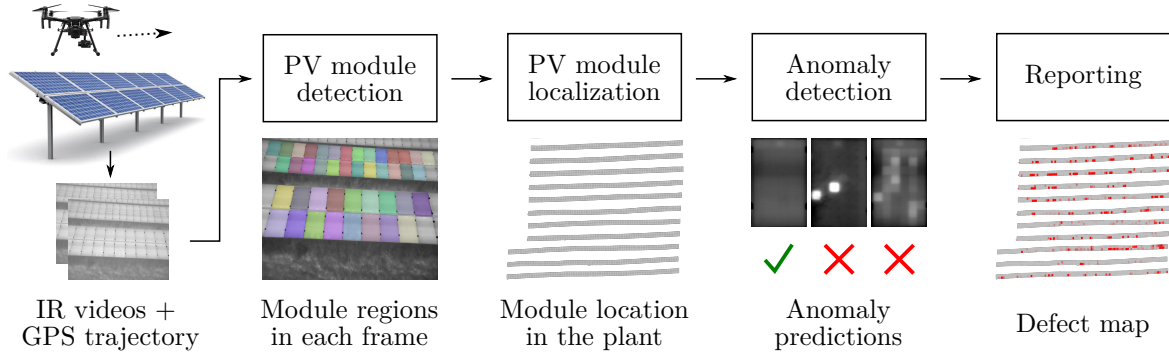factors into account to provide concrete recommendations on whether a module should be replaced or not.



Figure 1.1: Subtasks of an automated PV plant inspection system that is developed in this thesis.

There are several requirements for the developed solution: It should provide a high degree of automation, ideally being fully automated. The throughput should be as large as possible without compromising module and anomaly detection accuracy. The solution should also be flexible concerning the plant layout but work at least for utility-scale ground-mounted plants. Moreover, the solution should scale to PV plants with a multi-gigawatt capacity. Requirements on the flight trajectory of the drone should be low, facilitating both automated and manual flights. Only the flight altitude should be as low as possible to prevent obstruction of buildings and to produce high-resolution imagery for accurate anomaly detection. Finally, the solution should be fully portable, i.e., not require any setup when inspecting a new PV plant.

## 1.3 Scope of the Thesis

The publications in this thesis focus on subsets of the subtasks shown in fig. 1.1 and contribute different solutions as described in tab. 1.1.

All publications assume a manual drone flight for video acquisition because already existing manually recorded IR videos are used. The methods must adapt to the camera and drone hardware as well as the recording style of the existing IR videos. Methods for automating the drone flight are not proposed. Furthermore, reporting is limited to the indication of anomalous PV modules. The impact of these anomalies on power, yield, and economic variables is not further interpreted, and no recommendations for repairs are given. This is because solving these additional tasks in a generally applicable manner would require additional and very large amounts of data.

Publication [1] presents an initial solution for each of the subtasks, effectively solving the entire PV plant inspection problem in form of a semi-automated computer vision pipeline. Apart from demonstrating the feasibility of PV plant inspection based on the available IR videos, the work shows that common image processing algorithms function

Table 1.1: Scope of the publications in this thesis.

| Subtask | Publication [1] | Publication [3] | Publication [2] |
|---|---|---|---|
| Data Acquisition | Manual drone flight | Manual drone flight | Manual drone flight |
| Module Detection | Deep learning-based instance segmentation + multi-object tracking | Same as publication [1] | – |
| Module Localization | Association with description file by graph-matching | Georeferencing with incremental structure from motion | – |
| Anomaly Detection | Deep learning-based supervised image classification | Distribution of maximum module temperatures | Supervised contrastive representation learning + $k$-NN classifier |
| Reporting | Panorama image of each row | Visualization of results on a geographic map | – |

well in IR. In the developed pipeline PV modules are detected in each video frame by a deep learning-based instance segmentation model and tracked over subsequent frames with a multi-object tracking algorithm. IR image patches of the detected PV modules are cropped out of the frame, transformed to a rectangular shape, and stored for later use. PV module localization is realized by a graph matching procedure, which assigns a human-readable ID from a manually created plant description file to each PV module. PV module anomaly detection is framed as a supervised image classification problem with ten anomaly classes and solved by training and evaluating a ResNet-50 classifier. Anomaly detection results are reported as an overlay on a panorama image of each plant row that is assembled from extracted module images.

Publication [3] reuses the PV module extraction procedure of publication [1], but improves module localization and reporting, and proposes a simpler method for anomaly detection. Incremental structure from motion is used to obtain a georeferenced 3D reconstruction of the PV plant and absolute geocoordinates of the PV module corners. This facilitates the use of a geographic map for browsing extracted module images and reporting anomaly detection results. As opposed to the method in publication [1], this improved method is fully automated, works for plants with non-row layout, facilitates a higher throughput by parallel processing of multiple rows, and is more fault-tolerant with an extraction success rate of 99.3 % compared to 87.8 %. Publication [3] also presents another anomaly detection method based on local differences in the maximum module temperature, which is nearly as accurate, but simpler, faster, and easier to interpret than related deep learning methods.

Publication [2] exclusively addresses PV module anomaly detection. The work discovers the existence of a significant domain shift between IR images originating from different PV plants. To account for this domain shift, a more realistic problem setup is proposed, in which an anomaly classifier is trained on data of one PV plant and evaluated on data of another plant. Following this scheme, a binary classifier is developed, which uses supervised contrastive representation learning and a $k$-NN classifier to improve domain adoption.

## 1.4 Thesis Outline

The remainder of this thesis is structured as follows. Chapter 2 provides the theoretical background of PV plant anomalies, IR thermography, and computer vision methods for automated PV plant inspection. The scope of this chapter is to provide useful background knowledge needed to understand the related works and the methods developed in this thesis. In chapter 3 a thorough analysis of the state-of-the-art methods in computer vision and automated PV plant inspection is conducted. Chapter 4 discusses the contributions of our publications and compares them with each other. Afterwards, chapter 5 discusses, how our contributions improve upon the state of the art. Finally, the thesis is concluded in chapter 6 and future research opportunities are discussed in chapter 7.

## 2 Background

This chapter provides the background of automated PV plant inspection with IR thermography. After describing the typical structure of utility-scale PV plants, common PV plant anomalies and their resulting thermal patterns in IR imagery are introduced in detail. Afterwards, the basics of IR thermography for PV plant inspection are presented, and an overview of computer vision methods for the automated processing of the imagery acquired by IR thermography is given.

### 2.1 Large-Scale PV Plants

Development of a computer vision algorithm requires a solid understanding of the recorded scene, in the case of this thesis, large-scale PV plants. This understanding is not only needed to determine the best recording strategy, but also to make effective assumptions about the scene, exploit scene invariants (e.g., a PV module has always four corners), and drive algorithm design decisions. Hence, this section briefly introduces the structure and components of large-scale PV plants. The focus is on ground-mounted plants with a row-based layout as this type is very common [9] and the methods in this thesis are developed and tested on this plant type. Other types of large-scale PV plants include floating and rooftop-, or facade-mounted PV plants.

Fig. 2.1 shows a typical ground-mounted plant consisting of rows, each containing a large number of regularly arranged *PV modules*. The layout of rows can vary between plants as shown in fig. 2.2. In some plants more than two modules may be stacked vertically, modules may be oriented in landscape rather than portrait, or even rotated by 45°. There may also be regular gaps between each module or table (group of modules), and individual modules may be missing. The rows are usually facing the equator and are not parallel to the ground, but rather tilted at an angle (most common 20° to 30° [60]), to ensure an optimal incidence angle of the solar radiation and maximize annual yield. Some plants use trackers that adjust the tilt of each row over the day following the sun across the sky [61].

Fig. 2.1 also shows a PV module with crystalline silicon (c-Si) *cells*, which is currently the most common module type with a 95 % share in the 2020 global solar energy production [9]. A typical module contains 60 or 72 cells [62], and has a rated peak power of 250 W to 400 W. Typical dimensions of a 60 and 72 cell module are 1.0 m × 1.65 m and 1.0 m × 2.0 m, respectively [63, 64]. The module is a sandwich of an insulating polymer backsheet, the cells encapsulated in a polymer matrix, and a highly transmissive glass front that,

---
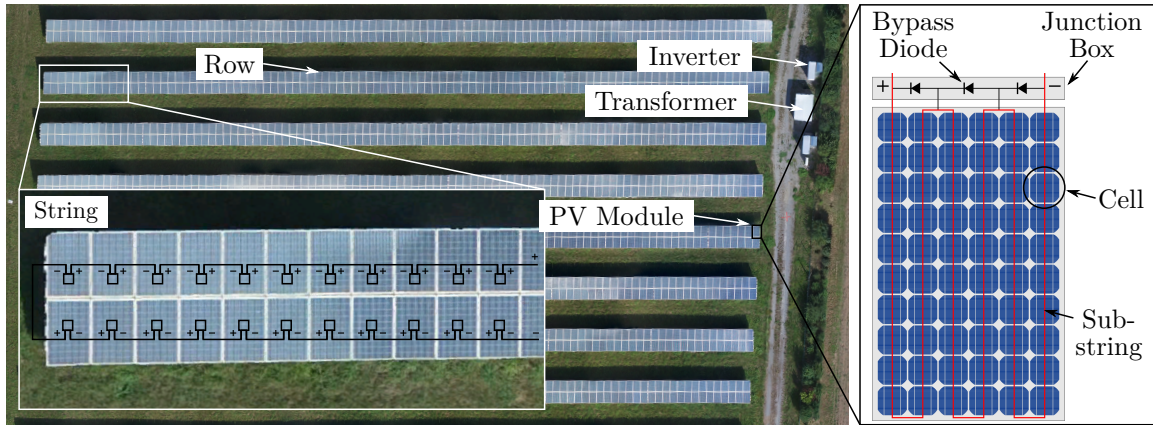
Figure 2.1: Elements of a large-scale photovoltaics plant.
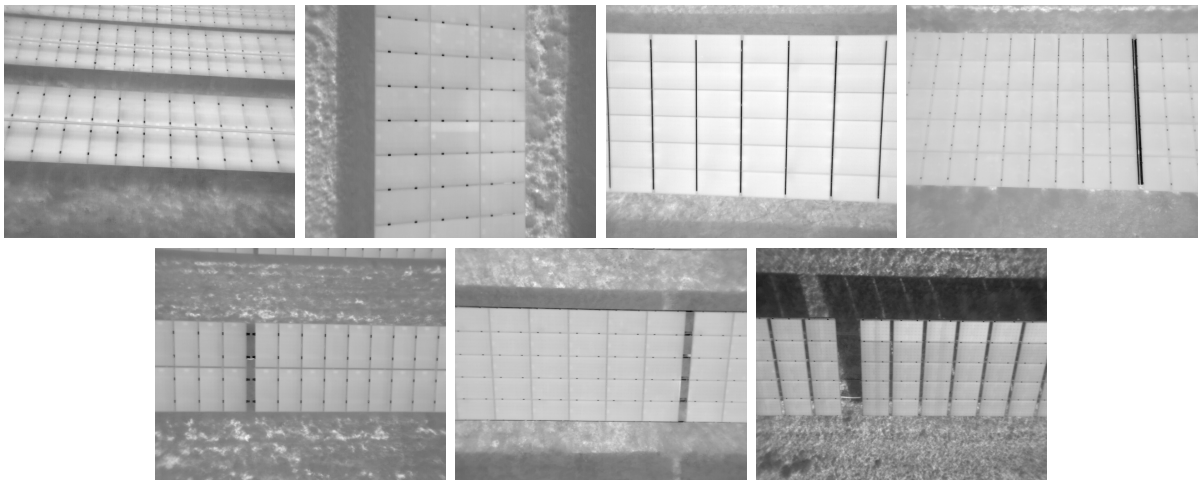


Figure 2.2: Different row layouts of ground-mounted PV plants.[1]

together with an aluminium frame, provides structural stability. The silicon cells are the active elements, which convert incoming solar energy into electrical energy. Each cell has a characteristic *open-circuit voltage* (typically $0.5\,V$ to $0.6\,V$) and a *short-circuit current* (typically between $5.8\,A$ and $13.3\,A$) [65]. To reduce resistive losses and increase inverter efficiency, a high module voltage and low module current are favorable. This is achieved by connecting all cells in series, summing up their voltages while maintaining the current of a single cell [65]. The module is split into three *substrings* with an equal number of cells, each equipped with a *bypass diode*. A bypass diode prevents reverse currents and excessive energy dissipation (heat) in cells of a substring in case of a mismatch of individual cell currents, which can occur, for example, due to partial shading of the module [66–68].

For the same reason that the cells of a module are connected in series, multiple PV modules (here 22 modules) are connected in series to form a *string*. An *inverter* converts the direct current of the string to an alternating current, which is stepped up in voltage by a *transformer* before being fed into the power grid [62]. The inverter also ensures that each string operates in its maximum power point, where power transfer efficiency is highest

[69]. Both string-level inverters and central inverters, which are connected to many strings in parallel, are used [62].

One aspect to keep in mind when inspecting large-scale PV plants is that the electrical and spatial layout of a plant do not always coincide. For instance, a string may be split up over multiple rows. Furthermore, it happens that modules or entire strings are accidentally swapped during installation or repairs, causing deviations from the original circuit plan.

## 2.2 PV Plant Anomalies

Anomalies in PV plants can occur due to problems in the manufacturing process, mishandling during installation, or due to exposure to thermal and mechanical stresses, ultraviolet radiation, high potential gradients, and adverse climatic and weather conditions, during operation in the field [11, 12, 31]. The term *anomaly* is chosen intentionally instead of *defect* or *fault* as not all anomalies cause drastic power losses or pose safety hazards, and, consequently, may not require repairs or exchange of the affected component.

The following section first describes common PV plant anomalies, focusing on their physical causes. Afterwards, thermal anomaly patterns that can be observed by IR thermography are shown and linked to their physical causes. In addition, a classification scheme for the thermal patterns is provided, which allows for automatic classification using computer vision algorithms. Note, that throughout the rest of this thesis we use the term *anomaly* interchangeably for the thermal pattern and the underlying physical cause.

### 2.2.1 Physical Causes

The following list of common PV plant anomalies is based on several related articles and technical reports [17, 24, 31, 70, 71]. Fig. 2.3 shows some of the anomalies discussed here. For additional details, such as typical temperature differences, see tab. 2.1 below. Note, that this overview is limited to anomalies of c-Si modules. Other technologies, such as thin-film or half-cell modules, share some of the anomalies, but also have their specific anomalies [23, 30].

**Electrical mismatch**   As pointed out in sec. 2.1, PV plants contain serial connections both on module-level and string-level to increase voltages, and consequently decrease currents and resistive losses. In a series connection, the same current flows through each component (cell or module), while the voltages across the components differ, summing up to the total voltage of the series connection. Due to a mismatch in the electrical properties of components, voltages across individual components can become negative (reverse bias),

---

[2]Reprinted from "Investigating the impact of shading ffect on the characteristics of a large-scale grid-connected PV power plant in northwest China," by S. Yunlin, S. Chen, L. Xie et al., 2014, International Journal of Photoenergy, vol. 2014, no. 3, p. 763106 [77]. CC BY.

[3]Reprinted from "Designing new materials for photovoltaics: Opportunities for lowering cost and increasing performance through advanced material innovations," by G. Oreski, J. Stein, G. Eder et al., 2021 [78]. © 2021 International Energy Agency.
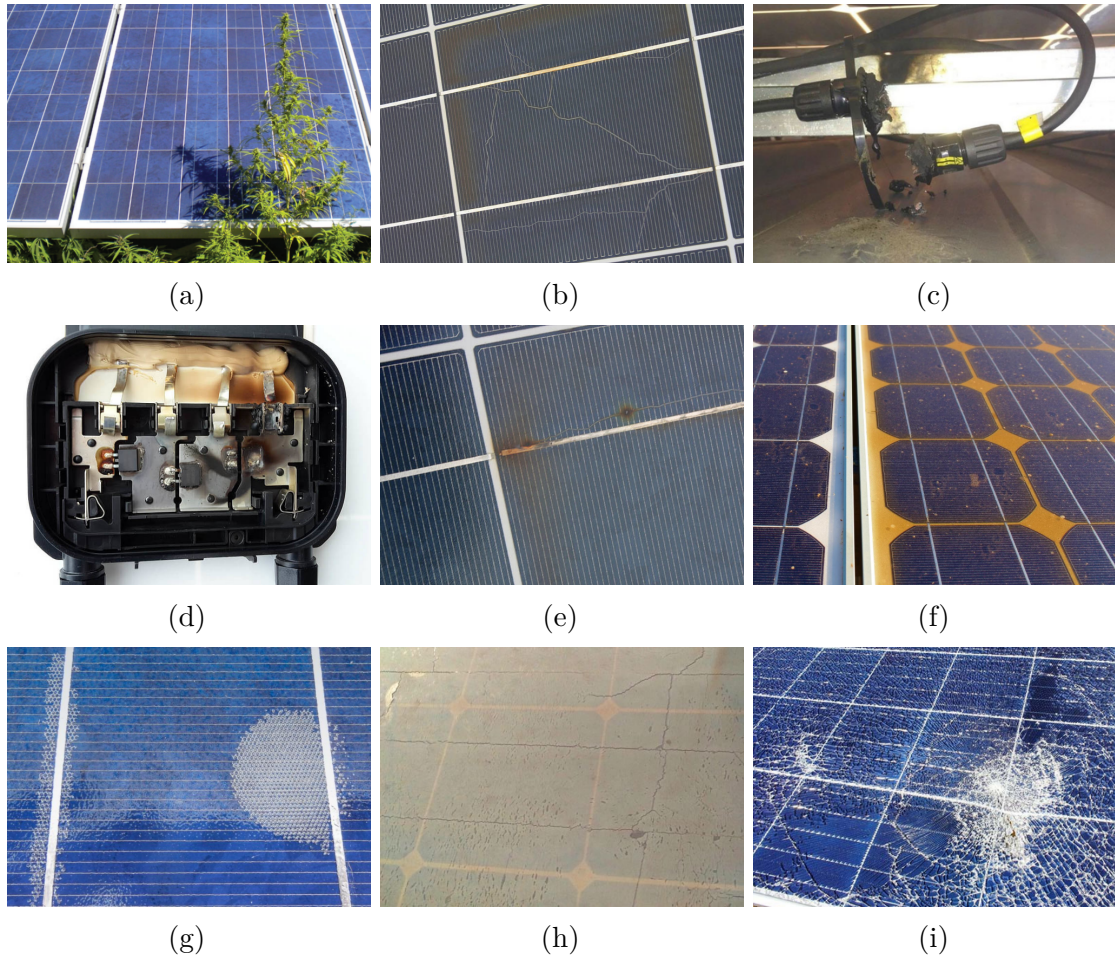
Figure 2.3: Common PV plant anomalies. (a) Partial shading[2], (b) cracked cell [72], (c) broken module connector causing open-circuited modules and strings [73], (d) defective bypass diode [72], (e) high-resistance connection (cell interconnect ribbon) [72], (f) encapsulant discoloration[3], (g) delamination [74], (h) backsheet degradation [75], and (i) front glass breakage [76].

turning the component from a producer into a consumer, which dissipates energy as heat causing a local temperature increase [79]. Several of the following anomalies induce an electrical mismatch in the series connection of cells or modules.

**Partial shading**   An electrical mismatch occurs for example, when some of the cells in a module are shaded, a situation referred to as partial shading (see fig. 2.3a). In this case, the voltage generated by the shaded cell drops and the cell may become reverse-biased and heat up. Partial shading is no anomaly per se, as it is reversible. However, it still leads to power loss. Common causes for shading are vegetation overgrowth and local soiling, e.g., bird droppings.

**Cracked cell**   Cell cracks are a common type of anomaly, caused due to mishandling of modules during installation and thermomechanical stresses during operation (see fig. 2.3b).

Cell cracks can lead to power losses, as they may cause leakage currents within the cell, or cause parts of the cell area to become disconnected, i.e., inactive. How much power is lost, depends strongly on the size, geometry, and orientation of the crack. Cell cracks can worsen over time due to repeated thermal and mechanical load cycles in the field.

**Shunted cell**   Shunts are cell areas of lower crystal quality that have low ohmic resistance. Possible causes are the presence of impurities during cell manufacturing or ineffective isolation of the cell edge. Shunts lead to high local current densities and consequently resistive heating, which ultimately reduces the power output of the cell.

**Open-circuited substring, module, or string**   If a substring, module, or string is in open circuit, generated electricity cannot be transported away, but instead must be dissipated in the form of heat, which causes a homogenous temperature increase of the affected substring, module (see Mh and Sh anomalies in fig. 2.4) or string (see fig. 2.5a). Typical causes are broken module connectors (see fig. 2.3c), cables, soldering joints, or interconnecting ribbons between cells. As no current can flow 100 % of the power generated by the substring, module, or string is lost.

**Short-circuited substring, module, or string**   A substring, module, or string can be short-circuited, for example, due to a short circuit inside a connector or cable. A module may also be short-circuited if all three bypass diodes are short-circuited. In the case of a short circuit, the same short-circuit current flows through all cells in the short-circuited path. Electrical mismatches cause some of the cells to become reverse-biased and heat up. The result is a patchwork pattern (see Mp and Sp anomalies in fig. 2.4). However, compared to the operation near the maximum power point (MPP), the temperature differences of cells in a short circuit are much larger. This is because near the cell's characteristic short-circuit current a small change in current is accompanied by a much larger change in voltage and power as compared to near the MPP. A short circuit leads to 100 % power loss of the affected component as the total voltage is zero.

**Defective bypass diode**   Each of the three substrings in a typical c-Si module is equipped with a bypass diode (see fig. 2.1), which reduces power losses in case of partial shading of the module by providing an alternative path for the substring current. The bypass diode also limits the reverse bias voltage of individual cells to the allowed maximum. Bypass diodes can fail due to an electrostatic discharge, thermal runaway, or mechanical stresses, becoming either high-ohmic or low-ohmic (short-circuited). A short-circuited bypass diode causes a short circuit of the respective substring, resulting in a patchwork pattern within the substring as explained above. A high-ohmic bypass diode acts as if it was absent (see fig. 2.3d). Hence, it becomes only apparent when an electrical mismatch occurs in the substring, e.g., due to partial shading or cracked cells. In this case, the bypass diode cannot limit the reverse bias voltage, which can become very large, leading to excessive heating of the mismatched cell (see Cs+ and Cm+ anomalies in fig. 2.4).

**High-resistance connections**   Connections between the cells, modules, and strings in a PV plant, such as soldering joints, interconnect ribbons, cables and connectors may exhibit a high contact resistance (see fig. 2.3e), causing local resistive heating of the connection

(see Chs anomaly in fig. 2.4) and consequently power losses. Root causes for high contact resistance are poor solderings, thermomechanical stresses, corrosion of metallic contacts, or installation errors.

**Potential induced degradation**   Large potential differences between cells and ground can cause the migration of ions within cells, leading to many shunted cells. This effect is called potential induced degradation (PID) and can lead to power losses of up to 100 % in affected modules. PID typically affects multiple neighboring modules towards the end of a string, where the potential difference is highest. PID becomes visible in IR as a non-uniform patchwork pattern that expands over multiple modules (see Pid anomaly in fig. 2.4 and fig. 2.5b).

**Packaging degradation**   The packaging materials of PV modules degrade over time, which negatively impacts the optical and electrical properties of the module. A degraded module packaging is prone to the intrusion of moisture, which can lead to corrosion and accelerate the degradation. Common modes of packaging degradation are the discoloration of encapsulation materials (see fig. 2.3f), delamination (see fig. 2.3g), backsheet degradation (see fig. 2.3h), and front glass breakage (see fig. 2.3i). Discoloration and delamination reduce the amount of light that reaches the cells and consequently lead to reduced power output. Delamination and backsheet degradation may lead to module leakage currents and short-circuiting of the module. Possible causes are exposure to ultraviolet radiation, moisture, mechanical stresses, and high temperatures caused by other types of anomalies.

## 2.2.2  Thermal Patterns

This section highlights the difference between the thermal patterns of anomalies observed in thermographic IR images and their physical causes. It also provides a classification scheme for the thermal patterns, which facilitates automatic classification using computer vision algorithms.

As we want to use computer vision algorithms to automatically classify anomalies, we need to consider, that computer vision algorithms base their predictions solely on the visual appearance of the image that is fed into the algorithm. Hence, the thermal pattern of an anomaly in the IR image is much more relevant to us than its physical cause. This would be no problem if a bijective mapping between the different physical causes and the resulting thermal patterns existed. However, no such mapping exists, because several of the above-described anomalies can lead to multiple different thermal patterns. For instance, a broken bypass diode can cause any number of substrings to either heat up homogeneously or exhibit a patchwork pattern. Hence, to allow for automated classification of the anomalies, we provide an appearance-based anomaly classification scheme, which provides ten disjunct classes, each with a unique thermal pattern. This scheme is shown in fig. 2.4. Tab. 2.1 contains further details and maps each of the thermal patterns to possible physical causes.
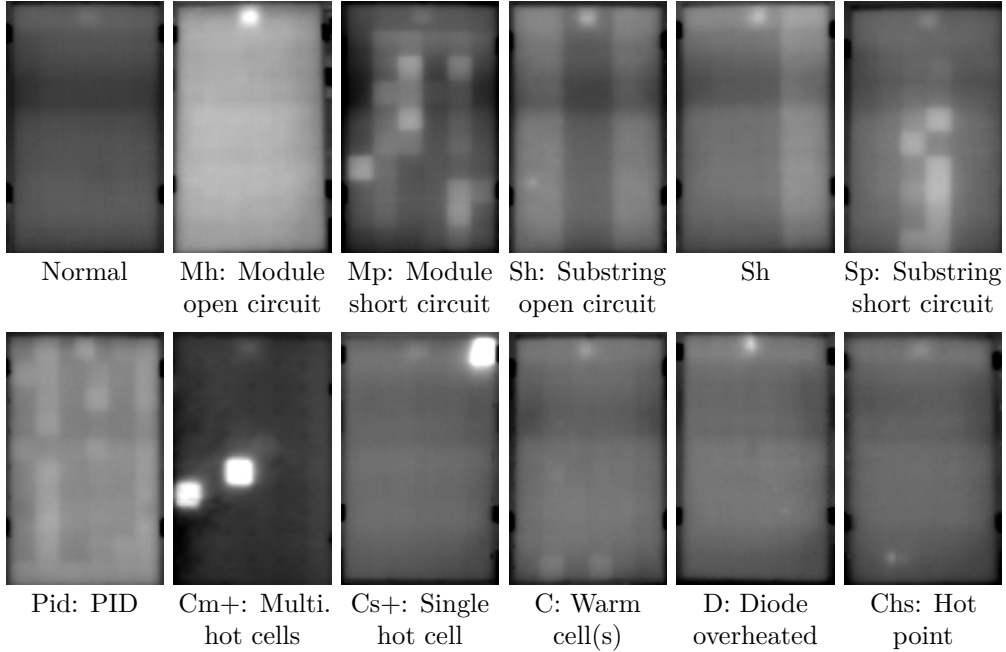
Figure 2.4: Thermal patterns of common PV module anomalies. Temperature is clipped at 30 °C (black) and 60 °C (white).[4]

Some of the thermal patterns can only be caused by one specific anomaly, such as Pid. This allows for direct inference of the underlying cause from the thermal pattern. However, for the anomaly patterns with multiple possible causes, additional on-site inspection, such as visual inspection or measurement of the current-voltage-characteristic may be required to disambiguate possible causes [23].

While our appearance-based classification scheme agrees with the literature for most of the thermal anomaly patterns [17, 30], there are additional special and rarely occurring anomalies that would require a more granular classification scheme. For our scheme, we started with 26 anomaly classes. However, this scheme was not suitable for downstream image classification by deep learning for two reasons: (i) Our dataset did not contain enough example images for some of the classes to fully describe the visual variance of that class, and (ii) some classes had the same or very similar thermal patterns. To resolve both issues, we grouped some of the 26 classes together, which resulted in the ten classes presented in fig. 2.4 and tab. 2.1. These ten classes have distinctive thermal patterns and there are enough example images per class. Dunderdale et al. approach the problem in the same way but resort to only five classes as they have fewer data [80].

Despite the simplifications of the classification scheme some challenges remain. For example, a single module may simultaneously exhibit different anomalies, which leads to a superposition of the corresponding thermal patterns. For instance, it is common, that an open-circuited substring also causes the bypass diode to overheat. Furthermore, the

---

[4]Reprinted from "Computer vision tool for detection, mapping, and fault classification of photovoltaics modules in aerial IR videos," by L. Bommes, T. Pickel, C. Buerhop-Lutz, et al., 2021, Progress in Photovoltaics: Research and Applications, vol. 29, no. 12, pp. 1236–1251 [1]. CC BY.

thermal patterns are not always as clear as the ones in fig. 2.4. If, for example, irradiance is lower, less power is dissipated in the anomaly, which leads to smaller temperature gradients and lower contrast in the IR image.

While the scheme in fig. 2.4 focuses on module-level anomalies, it is also applicable to anomalies that affect multiple modules or entire strings as shown in fig. 2.5. This way, defects in other components of the plant, such as the inverters, electrical connectors, or cables are detectable. For example, multiple neighboring modules with Mh and Mp anomaly patterns indicate an open-circuited and short-circuited string, respectively [30, 81].
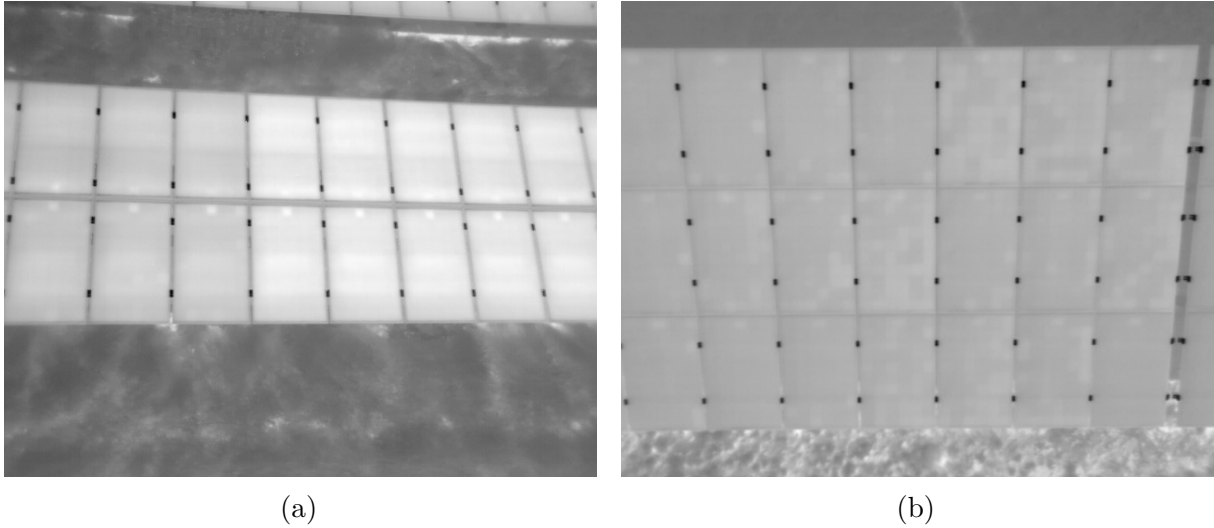


(a)                                              (b)

Figure 2.5: Thermal anomaly patterns of string-level anomalies. (a) Open-circuited string (b) Potential induced degradation.

Table 2.1: Details of the PV module anomalies in fig. 2.4. Values for $\Delta T$ and power loss are from [30] and [17], respectively. $\Delta T$ relates to a normally operating module at $1000\,\mathrm{W\,m^{-2}}$.

| **Anomaly** | **Description** | **$\Delta$T** | **Physical Causes** | **Power Loss** |
|---|---|---|---|---|
| Mh | Module homogeneously warmer than others | 2 K to 7 K | • Module in open circuit<br>• Failed system connection | 100 % |
| Mp | Patchwork pattern in entire module | 2 K to 7 K | • Short-circuited module (all bypass diodes in short circuit, erroneous connection) | 100 % |
| Sh | Substring(s) homogeneously warmer + heated bypass diode | 2 K to 7 K | • Substring(s) in open circuit (disconnected cell, erroneous connection) | 33.3 % per substring |
| Sp | Patchwork pattern in substring(s) | 2 K to 7 K | • Substring in short circuit (bypass diode in short circuit) | 33 % per substring |
| Pid | Module patchwork pattern, neighbour modules also affected | 2 K to 7 K | • Shunts caused by potential induced degradation | up to 100 % |
| Cs+/Cm+ | Single or multiple cell(s) much warmer than others | $\gg$ 10 K | • Cracked cell<br>• Shaded cell<br>• Bypass diode absent | proportional to number of affected cells |
| C | Single or multiple cell(s) slightly warmer than others | 2 K to 7 K | • Shunted cell<br>• Delaminated cell | negligible |
| D | Junction box (diode) overheated | $\geq$ 3 K w.r.t. nearby junction box | • Defective bypass diode<br>• High contact resistance in junction box | negligible |
| Chs | Overheated point | > 10 K | • Cracked cell<br>• High-resistance connections<br>• Broken interconnect ribbon<br>• Soiling (e.g., bird dropping) | negligible |

## 2.3 IR Thermography for PV Plant Inspection

IR thermography is a cheap, fast, simple, safe, and accurate method frequently used for the contactless detection of all common PV plant anomalies (see sec. 2.2) based on their elevated temperatures [25]. It is often performed by drones to speed up recording [16]. The computer vision pipeline developed in this thesis is tailored to IR thermography. Hence, a basic understanding of IR thermography is required to comprehend the developed methods.

The following section introduces the measurement principle of IR thermography and specifies the requirements on the camera, external conditions, and recording procedure. These requirements must be carefully implemented when recording IR videos for use with the developed computer vision pipeline to ensure compatibility and the best possible results.

### 2.3.1 Measurement Principle

According to Planck's law, [82] every body in thermal equilibrium with a temperature above $0\,\mathrm{K}$ emits electromagnetic radiation with a wavelength specific to the temperature. For typical ambient temperatures, this wavelength lies in the thermal IR band of the spectrum and can be detected by a thermal IR camera.

In the context of IR thermography, the body may be a PV module. The overall radiant power emitted from the module per unit surface area (called radiant exitance) is given by the Stefan-Boltzmann law [83] and amounts to

$$M_{PV}\left(T_{PV}\right) = \epsilon \sigma T_{PV}^4, \tag{2.1}$$

where $T_{PV}$ is the module surface temperature, $\sigma$ the Stefan-Boltzmann constant, and $\epsilon$ the emissivity of the module surface material, usually glass.

The IR camera receives not only the radiant exitance emitted by the module $M_{PV}$, but also the fraction of the environment's radiant exitance $M_E$ that is reflected by the module

$$M_{IR} = \epsilon M_{PV}\left(T_{PV}\right) + \left(1 - \epsilon\right) M_E\left(T_E\right). \tag{2.2}$$

This shows, that the measurement of the IR camera depends solely on the constant emissivity $\epsilon$ of the surface material, the module temperature $T_{PV}$, and the radiant exitance of the environment $M_E$ [28]. As the emissivity of glass is close to 1 and the temperature of the environment is typically lower than the module temperature, the second term in equation 2.2 has only a small influence on the measurement. This enables the accurate measurement of the surface temperature of the PV module that is needed to detect anomalies.

### 2.3.2 Camera Requirements

Due to the limited payload capacity of the drone, thermography of PV plants is performed exclusively with compact and lightweight uncooled microbolometer cameras [25]. However,

several aspects have to be considered when choosing a camera model to ensure sharp and accurate images that are useful for inspection [84]. Recommended values for the thermographic inspection of PV plants in the discussion below are taken from the IEC TS 62446-3 standard [30].

The *spectral range* is the portion of the radiation spectrum which can be sensed by the camera. For PV plant inspection, a spectral range of 8 μm to 14 μm is recommended, corresponding to object temperatures of about $-20\,°C$ to $350\,°C$ [25].

The *temperature resolution* is the smallest temperature difference that can be resolved by the camera and is usually expressed as noise-equivalent temperature (NETD). The maximum NETD recommended for PV plant inspection is 0.1 K, which is easily achieved by current uncooled microbolometer cameras that offer a NETD of about 50 mK.

The *temperature accuracy* is the maximum error between the measured and true temperature of the object. A value of $\pm 2\,K$ is recommended for PV plant inspection.

The *temperature range* defines the minimum and maximum measurable object temperatures. For PV plant inspection, it should be at least $-20\,°C$ to $120\,°C$. This is easily achieved by most IR cameras, which have much larger temperature ranges, e.g., $-40\,°C$ to $550\,°C$ for the DJI Zenmuse XT2. Some cameras also have a variable temperature gain allowing for higher temperature resolution at the cost of a smaller temperature range.

The *spatial resolution* is the minimum distance of two points in the scene that can be resolved by the camera. The spatial resolution depends on the detector resolution, the lens parameters, and the distance between the camera and the object. The minimum spatial resolution recommended for PV plant inspection is 3 cm, which is equivalent to $5 \times 5$ pixels per module cell.

The *frame rate* is the number of images acquired per second. For aerial inspection with drones, a high frame rate is important to facilitate the capture of blur-free videos at high flight velocities. While some cameras achieve up to 50 Hz [84], others may be artificially rate-limited due to export restrictions. For example, the U.S. allows the export of microbolometer cameras only as long as they are rate-limited to at most 9 Hz [85].

Other aspects to consider are cost, weight, size, and the possible presence of a secondary visual camera.

### 2.3.3 External Conditions

Apart from the hardware requirements, IR thermography of PV plants has further requirements on the state of the plant and environmental conditions that are specified in the IEC TS 62446-3 standard [30].

The inspected PV plant must be under operating conditions, in a thermal steady state, and not partially shaded, e.g., due to nearby infrastructure. Soiling must be low, incurring at most 10 % reduction in operating current, and homogeneous. No partial shading due to bird droppings, leaves, or vegetation must be present.

In terms of the environmental conditions, successful IR thermography requires a minimum irradiance of $600\,\mathrm{W\,m^{-2}}$ in the module plane, wind speeds of at most $28\,\mathrm{km\,h^{-1}}$, and a maximum of 2 okta of the sky covered by cumulus clouds.

### 2.3.4 Recording Procedure with Drones

Further requirements of IR thermography relate to the recording procedure. Generally, there must be no sun reflections on the surface of the PV modules. Moreover, the viewing angle of the camera should ideally be perpendicular to the module surface, and must not deviate more than 60° from the module normal to prevent reflections of the background radiation on the module surface [30].

When using drones, there are additional requirements on the flight path and the camera trajectory [38, 86], which are usually specific to the inspection software used. For example, our software requires plant rows to be scanned individually, or in groups of two or more rows as shown in fig. 2.6. The scanning order is uncritical; however, the same row should not be scanned multiple times. Furthermore, the heading of the drone should be always kept constant, the drone must move a sufficient distance in at least two orthogonal directions, and the camera must not be tilted relative to the drone. The viewing angle does not need to be nadir but should be kept vertical enough to prevent additional rows from becoming visible in the background. The scanned row(s) should not be cropped, and no neighboring rows should become visible at the top and bottom of the video except when changing to the next row(s). Abrupt movements should be avoided. Instead, the flight velocity should be constant and slow enough to not cause motion blur. Moreover, the camera should move monotonically along each row, i.e., never move backward. In case there is no accurate altitude measurement available, the flight altitude should be kept constant. However, if the altitude is measured, a constant height over the modules should be maintained and elevation changes in the terrain should be followed.

## 2.4 Computer Vision

Computer vision plays a central role in all of our methods and the related works, as it is used to automate the subtasks of a PV plant inspection system introduced in sec. 1.2. Hence, the following section provides an overview of several computer vision methods from the subfields of classic image processing, traditional machine learning, deep learning, and geometric computer vision, which are relevant to automated PV plant inspection.

### 2.4.1 Computer Vision Tasks

Fig. 2.7 introduces some computer vision tasks commonly encountered in the automatic PV plant inspection literature.

PV modules and anomalies are often localized in an image by detecting relevant edges, axis-aligned bounding boxes (object detection), or pixel-level masks (semantic / instance
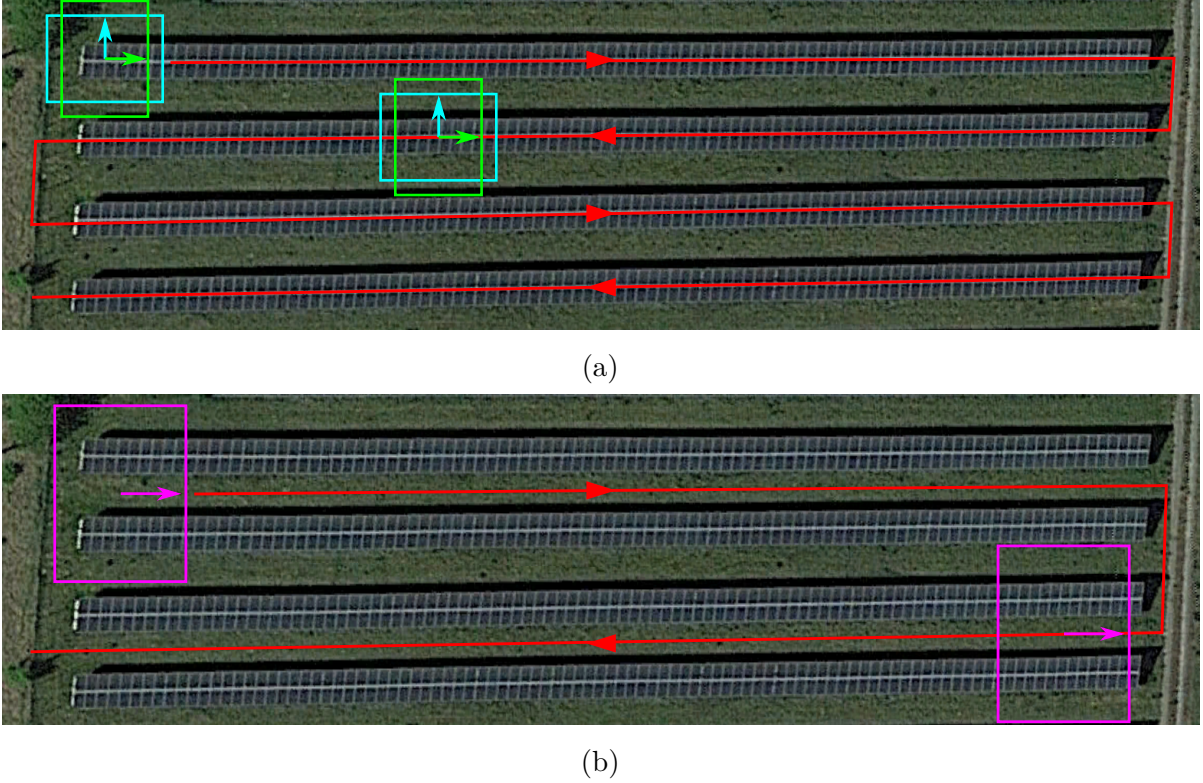
(a)



(b)

Figure 2.6: Flight procedure for PV plant inspection. Rows are scanned individually (a) or in groups of two or more (b). Boxes and arrows indicate the viewport and up-direction of the image, in which rows lie either horizontal (cyan box) or vertical (green box).

segmentation). Both object detection and instance segmentation differentiate individual instances of an object, whereas semantic segmentation provides only a single mask for the entire image.

Image classification is the task of assigning a single class label out of a limited set of class labels to an image. This is useful for categorizing PV module anomalies based on images showing a single module. In binary classification, there are only two classes, whereas in multi-class classification there are more than two classes.

Multi-object tracking aims at connecting bounding boxes or instance masks of the same physical object in the scene over subsequent images in a sequence (or video). A unique ID is assigned to each set of boxes or masks serving as a reference to the physical object.

## 2.4.2 Computational Representation of Images and Video

In computer vision, color images are represented as three-dimensional integer- or real-valued tensors $\mathbf{x} \in \mathbb{R}^{h \times w \times d}$. The first two dimensions correspond to the height $h$ and width $w$ of the image and the third dimension contains the red, green, and blue color channels ($d = 3$) and an optional alpha-channel for transparency ($d = 4$). Monochromatic images, such
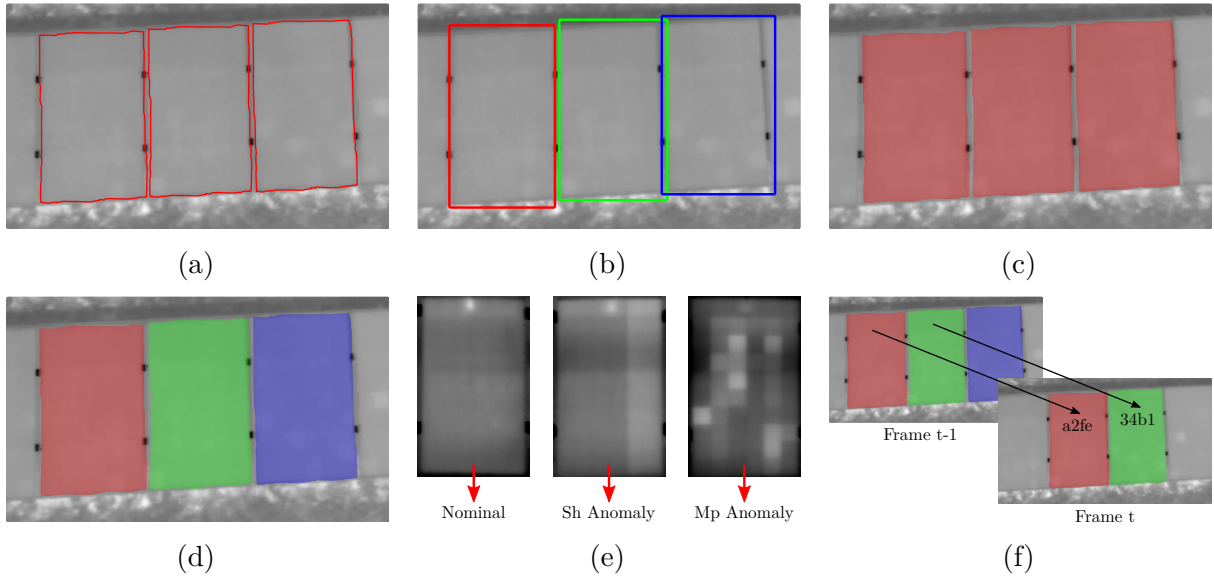
Figure 2.7: Common computer vision tasks encountered in automatic PV plant inspection systems are (a) edge detection, (b) object detection, (c) semantic segmentation, (d) instance segmentation, (e) image classification, and (f) multi-object tracking.

as those produced by an IR camera, have only one color channel ($d = 1$). Consequently, they can be represented by a two-dimensional tensor (matrix). Image values resemble the intensity of each pixel. Visual images typically use 8-bit values ($0 \ldots 255$), whereas most IR images use 16 bits ($0 \ldots 65535$) for more accurate temperature quantization. Videos resemble a sequence of images concatenated along an additional time dimension.

### 2.4.3 Classic Image Processing

The following section presents classic image processing methods for image enhancement, edge detection, line detection, and semantic segmentation. These algorithms are frequently used by the related works to detect PV modules (see sec. 3.3.1) and PV module anomalies (see sec. 3.4.1) in IR and visual images. Our methods use classic image processing only for contrast enhancement.

In classic image processing, one or more explicitly programmed algorithms transform a digital image to perform the tasks presented in sec. 2.4.1.

Image enhancement methods improve an image to make it more suitable for subsequent processing [87]. This includes contrast enhancements, for example, using histogram equalization (see fig. 2.8b), the application of Gaussian blur, median blur, or averaging to smoothen the image and remove high-frequency noise, or the conversion of color spaces, e.g., color to grayscale, or RGB to HSV.

Semantic segmentation can be performed by image binarization via thresholding [87]. Here, pixel values of a grayscale image are set to zero or full intensity if they are below or above a threshold value. In global thresholding (see fig. 2.8c) a single threshold value is applied to

the entire image, which is either selected manually or, as in the popular Otsu's method [88], computed automatically from the intensity histogram. However, global thresholding does not work well if brightness varies across the image. In such cases, adaptive thresholding (see fig. 2.8d) is favorable, as it computes an optimal threshold for each pixel based on the brightness of the surrounding pixels. Binarization may also involve two threshold values, which define an interval of pixel intensities to be set to full intensity.

Edge detection in classic image processing can be achieved by computing the first or second order image derivatives with the Sobel or Laplacian operator, respectively. The Sobel operator [89] uses two $3 \times 3$ kernels, $K_x = \left( \begin{smallmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{smallmatrix} \right)$ and $K_y = \left( \begin{smallmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{smallmatrix} \right)$, which are convolved with the input image to obtain the horizontal and vertical derivative images $G_x$ and $G_y$. Here, convolution means the kernel is moved over the image in a sliding window fashion and at each pixel the cross-correlation between kernel and the pixels underneath is computed. The gradient magnitude $G = \sqrt{G_x^2 + G_y^2}$ highlights edges independently of their direction (see fig. 2.8e). The Laplacian operator [87] computes second order derivatives and highlights regions with quickly varying intensities, such as edges (see fig. 2.8f). The Laplacian of an image with pixel intensities $I(x, y)$ is defined as $L(x, y) = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2}$ and can be approximated by convolution with the kernel $K = \left( \begin{smallmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{smallmatrix} \right)$. Since the Laplacian operator is very sensitive to noise, often a Gaussian filter is applied beforehand. The most common method for edge detection is the Canny edge detector [90], which combines multiple simpler image processing algorithms as follows: (i) smoothing with a Gaussian filter, (ii) computation of gradient magnitude and direction with the Sobel operator, (iii) edge thinning with non-maximum suppression based on gradient direction, and (iv) filtering of edges based on thresholding of the gradient magnitude and analysis of edge connectivity. An example of Canny edge detection is shown in fig. 2.8g.

Edge detection is often followed by line detection, which identifies predominant lines formed by sets of colinear points or edge segments in the binary image (see fig. 2.8i). A popular method for line detection is the Hough transform [91], which uses polar coordinates to represent a line through a point in image space as a single point in Hough space and vice versa. This allows transforming the set of all lines passing through a pixel into a single sinusoid in Hough space. If two or more sinusoids intersect, then the associated pixels are colinear in image space. Hence, predominant lines in the image can be detected by identifying those points in Hough space, where enough (greater than a threshold) sinusoids intersect.

Morphological operations are used to modify the shape of objects in binary images [87]. The two basic morphological operations are erosion and dilation. Here, a structuring element (a binary kernel) is slid over the input image, and a pixel in the input image is set to 1 only if all pixels below the kernel are 1 (erosion) or if at least one pixel below the kernel is 1 (dilation). Erosion erodes shape boundaries, hence is useful for noise removal or edge thinning. Dilation inflates shape boundaries, and thus can be used to thicken edges (see fig. 2.8h) or close holes and gaps in an object.
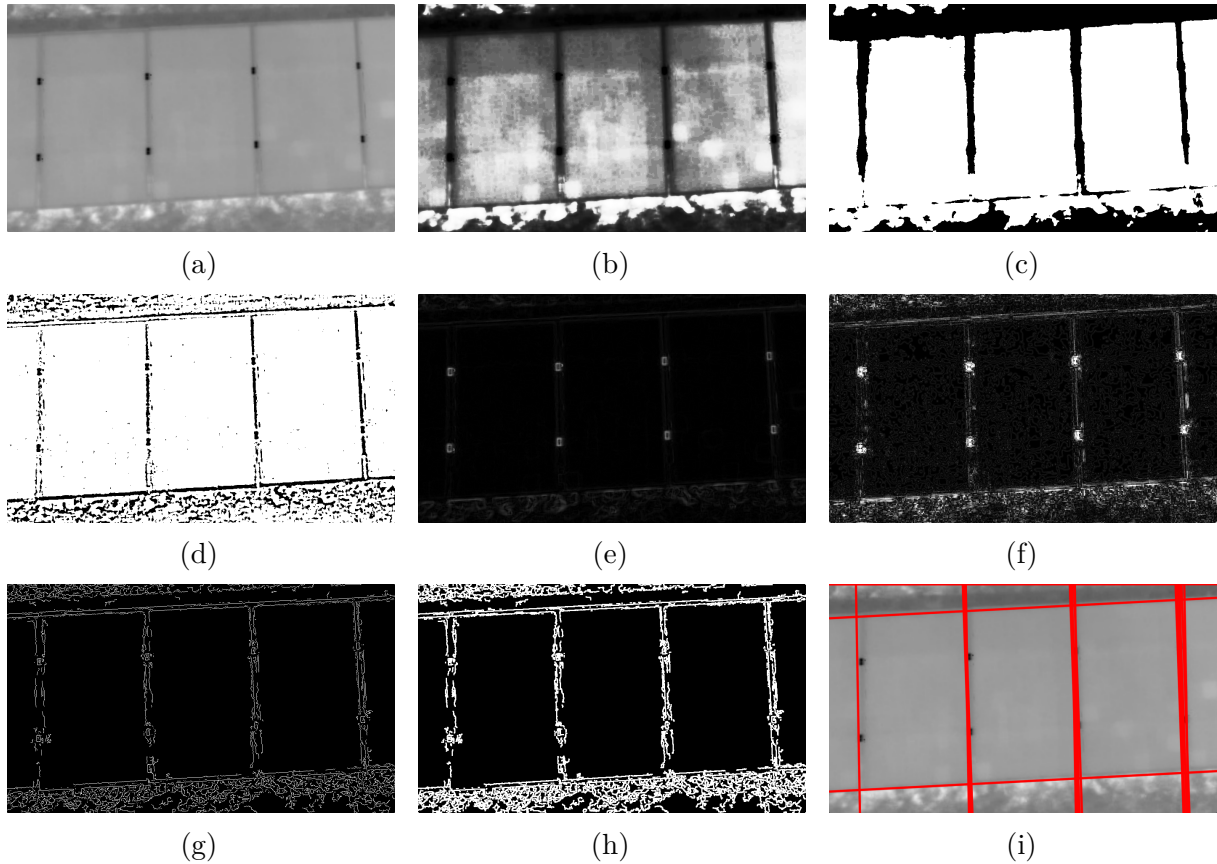
Figure 2.8: Common classic image processing methods applied to the exemplary IR image in (a): Histogram equalization (b). Segmentation with global (c) and adaptive threshold (d). Edge detection with Sobel operator (e), Laplacian operator (f), and Canny edge detector (g). Morphological dilation of the Canny edge image (h). Line detection in the Canny edge image with Hough transform (i).

## 2.4.4 Machine Learning

This section briefly introduces several core concepts of machine learning, such as datasets, feature extraction, predictive models, and model training, using the example of supervised image classification. These concepts are relevant for the understanding of the subsequent chapter on deep learning. There are also several related works, which use machine learning for PV module detection (see sec. 3.3.2) and PV module anomaly detection (see sec. 3.4.2).

As opposed to classic image processing, machine learning systems are not explicitly programmed to perform a task, but instead, learn a predictive model based on training data that encodes the task [92].

In PV inspection systems, machine learning is used primarily for supervised image classification of thermal anomaly patterns (see sec. 2.2.2) [80, 93]. Here, training data consists of $N$ pairs of thermal IR images $\mathbf{x}$ with the corresponding (numerically encoded) anomaly class $y \in \{1, 2, \ldots, K\}$. The goal is to learn optimal parameters $\theta^*$ of a predictive model $f_\theta$ so that after training, the model can accurately predict the anomaly class of a novel

thermal image, which may not even be part of the training data. This is achieved through iterative optimization of an objective function [94]

$$\theta^* = \arg\min_{\theta} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(y, \hat{y}), \qquad (2.3)$$

where $\mathcal{L}$ is the *loss function* that compares the predicted class $\hat{y} = f_{\theta}(\mathbf{x})$ to the ground truth class $y$ of a single training sample.

A major challenge in machine learning is the high dimensionality of the input space. For example, a typical $150 \times 100$ pixel monochromatic IR image lies in a 15000-dimensional vector space. Most machine learning models do not work well in such high-dimensional spaces [95, 96]. To overcome this problem, the dimensionality of the data must be reduced. This can be achieved by computing descriptive statistics (called *features*), such as mean and standard deviation, of each training image [97]. The model is then trained on the extracted features instead of the images. This is possible as the training data lies, according to the manifold hypothesis, on a low-dimensional manifold in the input space [98, 99]. Each point on that manifold can effectively be described by a much smaller number of local coordinates, which correspond to the extracted features.

Fig. 2.9 illustrates the principle of machine learning for supervised image classification. There are many different machine learning classifiers, such as support vector machines [100], decision trees [101], nearest neighbors [102], and Gaussian processes [103]. They share the same principle but differ in the concrete realization of the model, objective, and optimization procedure.
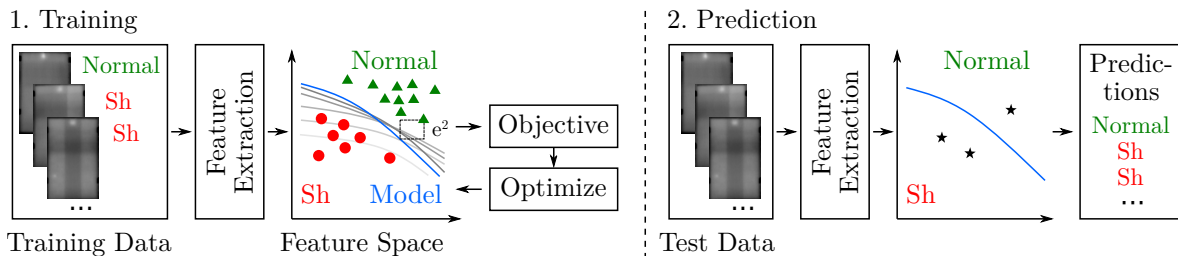


Figure 2.9: Supervised (binary) image classification with machine learning.

For accurate image classification simple global features, such as mean and standard deviation, are in most cases not sufficient. Instead, features are used that describe the local pixel neighborhood at multiple points of interest (called *keypoints*) in the image [104]. Commonly used combined keypoint detectors and descriptors are SIFT [105], SURF [106], ORB [107], and BRISK [108]. FAST [109] and Hessian-Affine [110] are exclusive keypoint detectors, and LIOP [111], BRIEF [112], FREAK [113], and HoG [114] exclusive descriptors. A study by Johanson found the combination of Hessian-Affine keypoint detector with LIOP descriptor to work best for IR images [115].

Two problems prevent the direct use of the so extracted features: (i) The number of keypoints differs in each image, and (ii) a large number of keypoints in each image results in a high dimensionality of the feature space. A typical solution is the encoding of features into a short fixed-length vector by the bag-of-visual-words method [116].

## 2.4.5 Deep Learning

Deep learning is a subset of machine learning that has gained a strong interest in the recent decade and led to impressive advances in many fields [117]. More recent works on automated PV plant inspection as well as most of our methods, use deep learning to perform tasks, such as instance segmentation and image classification (see sec. 3.3.3 and sec. 3.4.3). The following section introduces the most important theoretical concepts of deep learning. State-of-the-art topics, such as convolutional neural networks, supervised image classification, instance segmentation, domain shift, and supervised contrastive losses, will be explained later in sec. 3.1.

The key difference between deep learning and the more traditional machine learning methods described in sec. 2.4.4 is the use of artificial neural networks as a model to learn a task *end-to-end*. This means, that instead of having to manually specify the features to extract, the neural network can operate directly on the high-dimensional input data and learn the optimal features for the task at hand.

Artificial neural networks are parametric functions $f_\theta$, which are composed of stacks of simple linear constituent functions (called *layers*) with intermediate non-linear *activation functions*. Training is usually performed by stochastic gradient descent [118], which iteratively computes the gradient of a loss function $\mathcal{L}(y, f_\theta(\mathbf{x}))$ with respect to the network parameters $\theta$ for a single training sample $(\mathbf{x}, y)$ (or batch of samples), and then updates the network parameters to descent in the direction of the gradient towards the global minimum of the loss function. A single update step of stochastic gradient descent can be written as

$$\theta := \theta - \eta \nabla_\theta \mathcal{L}(y, f_\theta(\mathbf{x})), \qquad (2.4)$$

where $\eta$ is the *learning rate*. The gradient can be efficiently computed with the backpropagation of error algorithm [119].

Different tasks have given rise to different types of neural networks, such as fully connected networks, recurrent networks [120], attention-based networks [121], and adversarial networks [122]. For computer vision, the by far most popular network type is the convolutional neural network (CNN), which uses layers comprised of filters that are convolved with the inputs [123].

## 2.4.6 Comparison of Classic Image Processing, Machine Larning, and Deep Learning

The following section briefly covers the main advantages and disadvantages of classic methods, machine learning, and deep learning for image processing. These differences are important for the comparison of state-of-the-art PV plant inspection methods later in chapter 3.

**Classic Image Processing**

Classic image processing methods are often easy to implement, computationally cheap, and require no labeled training data. However, they do not leverage data to learn from. Instead, a human domain expert must identify a suitable combination of explicit processing algorithms and select appropriate hyperparameters to solve the task at hand. The design space can be large, requiring significant experimental efforts and experience. Furthermore, the lack of labeled data prevents a systematic evaluation of the obtained algorithm. Thus, chances are high that the developed solution performs sub-optimally and has a poor ability to generalize beyond the dataset used for development. This is especially problematic for PV inspection, where many factors, such as lighting, perspective, image background, module type, and weather, can vary greatly between inspections.

**Machine Learning**

Traditional machine learning methods improve upon classic image processing by learning a task from training data. While this requires creating a labeled training dataset, the obtained predictive models typically achieve higher task accuracy and generalize better to unseen data. As compared to classic image processing, the development process of a machine learning method is more principled, and due to the use of validation data also more quantitative. This reduces the amount of experimentation and human expertise required. However, a decent amount of human domain expertise is still needed for the definition of suitable hand-crafted features.

**Deep Learning**

Apart from the requirement of labeled training data, deep learning has the disadvantage of being compute-intensive, typically requiring hardware acceleration for training and prediction. Neural networks can also be more difficult to interpret than classical image processing algorithms. However, many methods, such as the grand tour [124] and class activation maps [125, 126], have been developed for the visualization of trained models. The major advantage of deep learning is its ability to leverage large quantities of training data to automatically learn the optimal features for the task at hand. This reduces not only the amount of human domain expertise and experimentation required, but also yields more expressive features, as no simplifying assumptions need to be made and neural networks can learn arbitrarily complex features [127–129]. Consequently, this leads to predictive models that achieve even higher task accuracy and generalize even better to unseen data than traditional machine learning models.

## 2.4.7 Multi-object Tracking

This section covers the common computer vision task of (multi-)object tracking, which we employ in our publications [1, 3] to group extracted PV module images by module identity. The objective of object tracking is to connect bounding boxes or instance masks of the

same physical object, such as a PV module, over consecutive frames of a video [130]. A unique ID is assigned to each object and the associated instance masks or bounding boxes. Multi-object tracking performs this task for multiple objects in the scene simultaneously.

Multi-object tracking is a broad topic and there are many ways to accomplish the described functionality. Thus, we refer the reader to the excellent surveys in [131–134] and limit the explanation to our method from publication [1] for the tracking of PV module instance masks.

Fig. 2.10 shows a single tracking step from frame $t_1$ to $t_2$. The explanation assumes the tracker is already initialized, i.e., each segmentation mask in $t_1$ already has a unique ID assigned to it. To propagate these IDs to $t_2$, first, keypoints (see sec. 2.4.4) are detected, described and matched. Based on these matches, a perspective transformation matrix $H_{12}$, which aligns both frames, is estimated. Next, the center points of the segmentation masks in $t_1$ are projected into $t_2$ by means of the perspective transform $H_{12}$. Each projected center is then matched with the nearest segmentation mask center in $t_2$ and its ID is propagated. This step is generally referred to as *data association*. A frequently used optimal method for data association is the Hungarian method [135]. However, our method uses a simpler nearest neighbor search with an upper limit for the distance. Finally, the *track update* creates IDs for all new objects in $t_2$ and ensures objects that left the frame are no longer tracked in the next step.

The projection may also be repeated for several frames, before performing the comparatively slower instance segmentation and data association again. This leads to an overall speedup, however at the cost of reduced accuracy.
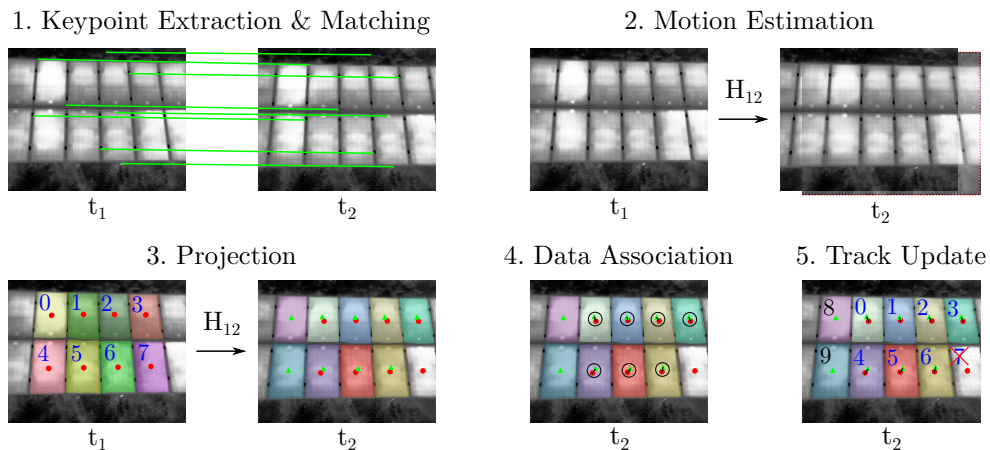


Figure 2.10: Principle of multi-object tracking.

## 2.4.8 Geometric Computer Vision

Geometric computer vision deals with the projection of 3D objects into 2D images, a process commonly modeled with the pinhole camera model, whose parameters are obtained through camera calibration. One well-established problem of geometric computer vision

is structure from motion, which is concerned with the reconstruction of a 3D scene from multiple observing images. Structure from motion has a large relevance for PV plant inspection, with many related works using it to create digital PV plant models and orthophotos from aerial imagery (see sec. 3.5.2 and sec. 3.5.3). In publication [3], we also use structure from motion for the georeferencing of PV modules.

### Pinhole Camera Model

A camera projects 3D scene points onto a 2D image plane. In computer vision, this projection is typically modeled with the pinhole camera model, which assumes a point-like camera aperture and no lens [130, 136]. Non-linear distortions introduced by the lens are modeled in a second step, for example with the Brown-Conrady radial distortion model [137].

In the pinhole camera model the projection of a $3D$ scene point $\mathbf{X} = (X, Y, Z, 1)^\intercal$ onto a 2D image point $\mathbf{x} = (u, v, 1)^\intercal$ is given as

$$\mathbf{x} = \pi\left(\mathbf{K}, \mathbf{P}, \mathbf{X}\right) = \mathbf{K}\left[\mathbf{R}|\mathbf{t}\right]\mathbf{X}. \tag{2.5}$$

Here, $\mathbf{P} = [\mathbf{R}|\mathbf{t}]$ is the camera pose described by the rotation matrix $\mathbf{R} \in \mathbb{R}^{3\times3}$ and translation vector $\mathbf{t} \in \mathbb{R}^3$ that transform a point from scene to camera coordinates. They are referred to as the *extrinsic camera parameters*. The camera matrix

$$\mathbf{K} = \begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix} \tag{2.6}$$

contains the focal lengths $f_x, f_y$ and the camera center $c_x, c_y$, which are the *intrinsic parameters*. Note, that the projective relation above uses homogeneous point coordinates [138].

### Camera Calibration

The objective of camera calibration is the estimation of the intrinsic camera parameters from several views of a calibration target. Additionally, the extrinsic parameters for each view are obtained as well as the parameters of an optional distortion model. Knowledge of these parameters facilitates the extraction of metric information from 2D images [139].

A calibration target is a planar object with known geometry and easily detectable features. Often, a chessboard pattern of black and white squares printed onto a sheet of paper or cardboard is used. Here, the corners between squares form distinctive features that can be detected automatically with an image processing algorithm.

In the following, we will briefly outline the calibration algorithm by Zhang et al. [139], which is implemented in the popular OpenCV library [140]. This algorithm takes as inputs the known 3D positions of the chessboard corners (in the local coordinate system of the calibration target) and the corresponding 2D projections in the images. A closed-form

solution exists for the computation of initial parameter estimates. These are further refined by iterative minimization of the total reprojection error

$$\sum_i \sum_j \|\mathbf{x}_{ij} - \pi\left(\mathbf{K}, \mathbf{P}_i, \mathbf{X}_j\right)\|_2^2. \tag{2.7}$$

Here, $\mathbf{x}_{ij}$ is the $j$th corner point detected in image $i$ and $\pi\left(\mathbf{K}, \mathbf{P}_i, \mathbf{X}_j\right)$ is the same corner projected as per eq. 2.5 using the current parameter estimates. This optimization is often performed with the Levenberg-Marquardt algorithm [141].

The outlined calibration procedure also applies to thermal IR cameras. However, special care must be taken to ensure the visibility of the calibration target in IR. To this end, a printed standard target can be exposed to sunlight, which causes the black squares to absorb light and heat up more than the white squares [142]. Alternatively, materials of different thermal emissivity, such as a metal plate and polymer foils, can be combined to produce a high-contrast IR image [143, 144]. Finally, parts of the target can be actively illuminated with IR LEDs [145, 146], or actively heated or cooled [142]. Since IR images are more blurry than visual images, automatic detection of the calibration target features is challenging and may require special algorithms [147].

**Incremental Structure from Motion**

Structure from motion (SfM) aims at reconstructing a 3D scene from multiple unordered 2D images of the scene taken from different viewpoints [148]. SfM exploits the motion of a monocular camera to recover the depth information of the scene that was lost during projection [149]. Often additional GPS measurements are considered to recover also the scale of the reconstruction [150]. SfM is typically an offline procedure, requiring the complete set of images to be available, and can handle millions of images [151, 152].

There are different strategies to solve SfM, such as global [153–155] and hierarchical SfM [156, 157]. However, the most popular one is incremental SfM [152, 158–160], where one image at a time is added to the reconstruction in an incremental procedure. There are several mature implementations available for incremental SfM, such as OpenSfM [161], Theia [162], VisualSFM [160], and COLMAP [163].

In the context of PV plant inspection, SfM is frequently used for creating orthophotos of the plant (see sec. 3.5.2) and for georeferencing of PV modules (see sec. 3.5.3).

Fig. 2.11 illustrates a typical pipeline for incremental SfM, which takes as input an unordered set of images $\mathcal{I} = \{I_0, I_1, \ldots, I_{N_I}\}$, the intrinsic camera parameters, and optionally their measured GPS positions $\mathcal{S} = \{\mathbf{s}_0, \mathbf{s}_1, \ldots, \mathbf{s}_{N_S}\}$. Outputs are a sparse 3D point cloud of triangulated image keypoints $\mathcal{X} = \{\mathbf{X}_0, \mathbf{X}_1, \ldots, \mathbf{X}_{N_X}\}$ and the camera poses of all images $\mathcal{P} = \{\mathbf{P}_0, \mathbf{P}_1, \ldots, \mathbf{P}_{N_P}\}$. The following briefly outlines each step of the pipeline.

**Image Matching**    First, point correspondences are established between pairs of images by detecting, describing, and matching keypoints, such as SIFT or SURF (see sec. 2.4.4). Since matching has a prohibitive complexity of $O(N_I^2)$ [164], often candidate pairs are
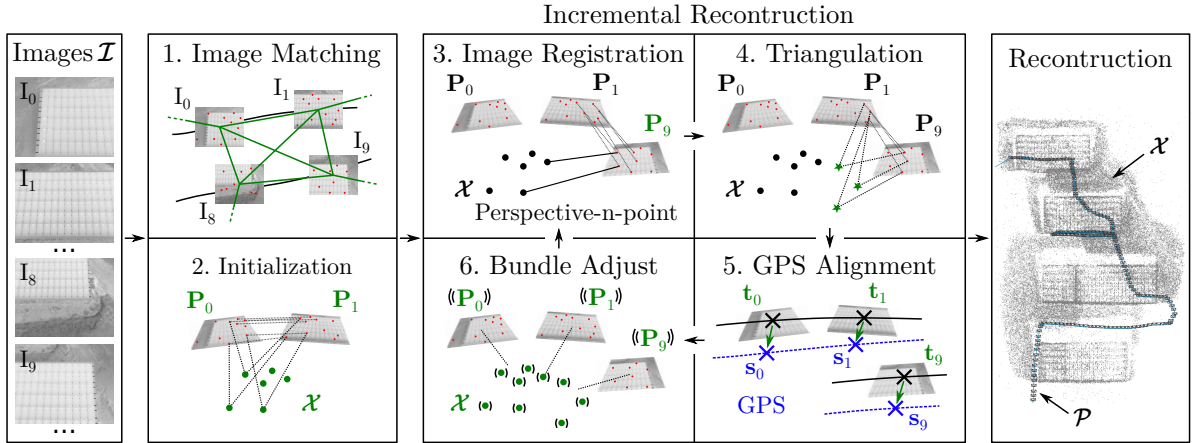
Figure 2.11: An incremental structure from motion pipeline for the 3D reconstruction of a scene from an unordered set of images. Outputs of each step are colored in green.

obtained beforehand by a fast method based on global image appearance [152, 165], bag-of-visual-words descriptors [159, 164, 166], preemptive matches [160], or GPS positions and timestamps [152]. Afterwards, matches are geometrically verified in a RANSAC [167] scheme, which estimates a transformation that explains the majority of matches and discards outlier matches [168].

**Initialization**  Next, the reconstruction is initialized from a suitable pair of images, e.g., an image pair with a large number of matches and sufficient parallax [169]. Initialization obtains the camera poses of the two images and triangulates an initial set of 3D points from the keypoints matched between both images. Since the scene structure is unknown, often multiple two-view reconstruction methods are employed, such as the five-point algorithm [170] for general 3D scenes or homography decomposition [171] for planar scenes. The more appropriate initialization is then selected based on a quality criterion, such as the reprojection error or transfer error [172].

**Incremental Reconstruction**  Now, the remaining images are added incrementally to the reconstruction. Typically, the next image to be added is the one with the most matches to any of the already reconstructed images. Its pose is estimated from correspondences between reconstructed 3D points and their 2D image projections, which is known as the perspective-n-point problem [173]. Subsequently, the 3D point cloud is extended by triangulating keypoints shared between the newly registered image and other images in the reconstruction [174, 175]. If GPS positions of the images are available, the entire reconstruction is rigidly transformed, i.e., rotated, shifted, and scaled, to best align reconstructed and measured camera positions. Finally, bundle adjustment [176] jointly optimizes all camera poses, intrinsic camera parameters, and 3D scene points by reducing the total reprojection error as defined in eq. 2.7. This prevents non-recoverable drift of the reconstruction due to uncertainties in the image registration and triangulation [163]. To reduce the computational complexity, bundle adjustment may not be performed in every iteration and may optimize only a local subset of camera poses and scene points [172].

# 3 State of the Art

This chapter is an overview of the state of the art in automated PV plant inspection. First, state-of-the-art deep learning-based computer vision methods, which are frequently used in PV plant inspection systems, are introduced. Afterwards, state-of-the-art works on automated PV plant inspection are presented and their shortcomings are discussed. The structure of the second part of the chapter corresponds to the subtasks of an automatic inspection system identified in sec. 1.2. While several of the state-of-the-art works present an entire automatic inspection system, most works focus only on a subset of tasks, such as PV module detection, PV module anomaly detection, and PV module localization.

## 3.1 Computer Vision

Due to its superior performance on tasks like instance segmentation and image classification, many of our methods and the more recent related works on automated PV plant inspection use computer vision methods that are based on deep learning. Exemplary uses of deep learning, discussed later in this chapter, are the detection of PV modules (see sec. 3.3.3) and PV module anomalies (see sec. 3.4.3), and the automatic derivation of the drone flight trajectory from satellite imagery (see sec. 3.2).

This section describes state-of-the-art aspects of deep learning-based computer vision. For theoretical aspects of deep learning see sec. 2.4.5. First, convolutional neural networks and their use for supervised image classification and instance segmentation are described. Afterwards, the common problem of domain shift is elucidated and the supervised contrastive loss as one possible solution is introduced.

### 3.1.1 Convolutional Neural Networks

Nowadays, convolutional neural networks (CNNs) are the most frequently used type of neural network in computer vision. CNNs have recently found their way into PV plant inspection systems, mainly for two tasks: (i) the detection or semantic/instance segmentation of PV modules and hot spots in IR images [177–179], and (ii) the supervised classification of thermal anomaly patterns of individual PV modules [80, 180].

CNNs are stacks of convolutional layers with intermediate non-linearities and normalization and pooling layers that extract increasingly abstract feature maps of the input image.

As exemplarily shown in fig. 3.1, each convolutional layer implements a cross-correlation between the 3-dimensional input tensor and a bank of learnable filter matrices. There are as many filters as there are channels in the input tensor. Typically, filters are of a

small spatial extent, such as $3 \times 3$. To obtain multi-channel outputs, one separate filter bank is used for each output channel, resulting in a 4-dimensional filter tensor. Inputs are optionally padded with zeros to control the size of the output.
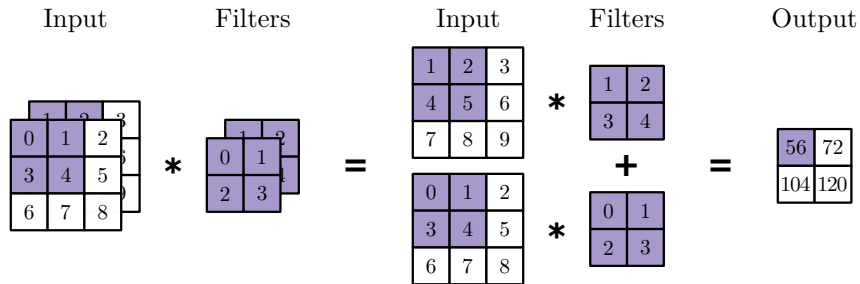


Figure 3.1: Cross-correlation performed by a convolutional layer on a two-channel input generating a single-channel output.[1]

Convolutional layers are well-suited to computer vision because they can deal with variable-sized inputs, act locally on a small neighborhood of input values, are invariant to translation of the input, and are computationally efficient due to parameter sharing [181].

Apart from the standard convolutional layer, there are special layers, such as $1 \times 1$ convolutions [182], dilated (atrous) convolutions [183], and depthwise separable convolutions [184].

Many successful CNN architectures increase the number of output channels in each layer while simultaneously downsampling the spatial size of the feature maps [181]. Down-sampling is achieved by pooling layers, which compute the mean or maximum over local neighborhoods (e.g., of size $2 \times 2$) in the output feature map [185]. Alternatively, strided convolutions can be used that skip intermediate locations during the cross-correlation when sliding the filter matrix over the input tensor [186].

Another feature of many modern CNNs are skip connections, which were first introduced in ResNet [187]. Skip connections connect layers that are spaced further apart in the network, improving the backward flow of error gradients during training and effectively preventing the vanishing gradient problem [188].

As usual in neural networks, a non-linear activation function is applied elementwise to the outputs of each layer. Due to its simplicity and computational efficiency, the rectified linear unit $\text{ReLU}(x) = \max(0, x)$ is a popular choice for CNNs. However, there exist many more activation functions [189], for example Sigmoid [190], Leaky ReLU [191], Softplus [192], GELU [193], ELU [194], and Maxout [195].

---

[1]Reprinted from "Dive into deep learning," by A. Zhang, Z. C. Lipton, M. Li et al., 2021, arXiv preprint, arXiv: 2106.11342 [181]. CC BY-SA.

## 3.1.2 Supervised Image Classification with CNNs

Supervised image classification with CNNs is one of the earliest and most ubiquitous tasks in the deep learning literature and many different models, such as AlexNet [196], VGGNet [197], ResNet [187], ResNeXt [198], MobileNet [199], EfficientNet [200], and ConvNext [201], have been proposed. A popular training and benchmarking dataset for image classification is ImageNet-1K, which contains 1.2 million natural images of 1000 different classes [202]. We employ an ImageNet-pretrained ResNet-50 for supervised classification of ten PV module anomalies in publication [1].

A typical CNN classifier uses a CNN backbone $g_\phi$ to extract features $\mathbf{z} = g_\phi(\mathbf{x})$ with $\mathbf{z} \in \mathbb{R}^d$ of the input image $\mathbf{x}$. A subsequent linear layer $h_\psi$ of the form

$$h_\psi(\mathbf{z}) = \mathbf{W}\mathbf{z} + \mathbf{b} \tag{3.1}$$

with weight matrix $\mathbf{W}$ and bias vector $\mathbf{b}$ transforms the features $\mathbf{z}$ into a *logits* vector $\mathbf{o} \in \mathbb{R}^K$, where $K$ is the number of classes. Alternatively, a pooling operation can be used instead of the linear layer. The softmax activation function

$$\sigma(\mathbf{o})_j = \frac{\exp(o_j)}{\sum_{\ell=1}^{K} \exp(o_\ell)} \quad \text{for } j = 1, 2, \ldots, K \tag{3.2}$$

converts the logits vector into a discrete probability distribution $\mathbf{p} \in \mathbb{R}^K$ over the $K$ possible classes, i.e., the outputs of the softmax function are always in the interval $(0, 1)$ and sum to 1. The predicted class for the input image is then simply $\hat{y} = \arg\max_j(p_j)$.

For training, integer class labels $y \in \{1, 2, \ldots, K\}$ of the training data are one-hot encoded into binary vectors $\mathbf{q} \in \mathbb{R}^K$ with

$$q_j = \begin{cases} 1, & \text{if } j = y \\ 0, & \text{otherwise.} \end{cases} \tag{3.3}$$

For example, for a classification problem with three classes, one-hot encoding yields the binary vectors $(1, 0, 0)^\intercal$, $(0, 1, 0)^\intercal$, and $(0, 0, 1)^\intercal$ for classes 0, 1, and 2, respectively.

This allows to define the categorical cross-entropy loss between the one-hot encoded ground truth vector $\mathbf{q}$ and the predicted probability distribution $\mathbf{p}$ of a single training sample as

$$\mathcal{L}(\mathbf{q}, \mathbf{p}) = -\sum_{j=1}^{K} q_j \log p_j. \tag{3.4}$$

This loss is then used to train both the CNN backbone and the linear layer by stochastic gradient descent as described in sec. 2.4.5.

In practice, *transfer learning* or *fine-tuning* is a frequently used technique [203, 204], which speeds up training and yields well-generalizing models even for small datasets. Here, a model that was previously trained on ImageNet, is trained on a task-specific dataset with a learning rate that is smaller than the pretraining learning rate. Sometimes, parameters of the lower CNN layers are fixed and only the higher layers are trained.

## 3.1.3 Instance Segmentation with CNNs

Instance segmentation is the task of localizing individual objects in an image by means of a binary segmentation mask and additionally classifying each object. We use the popular Mask R-CNN framework [205] to perform instance segmentation of PV modules in our publications [1, 3]. Apart from Mask R-CNN, other instance segmentation models have been proposed, such as Cascade Mask R-CNN [206], PANet [207], Hybrid Task Cascade [208], DetectorRS [209] and many more [210, 211]. CNNs play an essential role in all these methods. Some common benchmarking datasets for instance segmentation are Microsoft COCO [212], Cityscapes [213], and Mapillary Vistas [214].

Mask R-CNN belongs to a group of instance segmentation methods that first detect and classify objects with an object detector (in this case Faster R-CNN [215]) and then generate a segmentation mask for each object bounding box [216–218].

As shown in fig. 3.2 Mask R-CNN employs a CNN backbone to extract a feature map from the input image. Different backbones, such as ResNet, ResNeXt, or Feature Pyramid Networks (FPN) [219], can be used. A separate region proposal network takes the feature map as input and generates a large number of bounding box candidates (RoIs). A special RoiAlign layer extracts a small feature map (e.g., of size $7 \times 7$ pixels) for each RoI from the backbone feature map. Each RoI feature map is then fed into a CNN that classifies whether the RoI contains only background or an object and of which class the object is (as described in sec. 3.1.2). The CNN further refines the bounding box via regression. In parallel, each RoI feature map is fed into another CNN, which outputs a binary segmentation mask of size $m \times m$ for each of the $K$ possible classes. For training, a multi-task loss is defined as the sum of classification loss, bounding box regression loss, and mask loss. The mask loss is the average binary cross-entropy loss between the predicted and ground truth mask after applying a per-pixel sigmoid function. If the ground truth class of the RoI is $k$, only the $k$th mask channel contributes to the loss. At inference time, each object mask is scaled up via bilinear interpolation and shifted to align with the regressed bounding box.
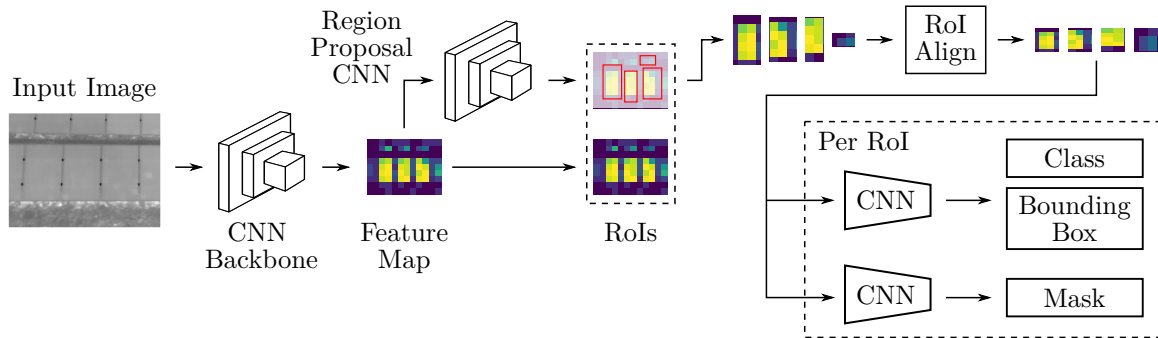


Figure 3.2: Mask R-CNN instance segmentation framework.

### 3.1.4 Domain Shift and Unsupervised Domain Adaption

*Domain shift* is a frequently observed phenomenon in real-world datasets and refers to the distributional shift between data used to train a deep learning model and data encountered during prediction [220, 221].

In publication [2], we find significant domain shift in our dataset of PV module images as illustrated in fig. 3.3. As can be seen in fig. 3.3a, the images form five distinct clusters (domains) in feature space that correspond to the PV plants from which the images originate. The distributional shift between these clusters is substantially larger than the shift between normal and anomalous images as highlighted in fig. 3.3b. Likely reasons for this domain shift are differences in (i) ambient conditions, such as irradiance, air temperature, and cloud cover, (ii) external conditions, such as plant load, soiling, and shading, (iii) recording conditions, such as camera model, spatial resolution, camera angle, and (iv) module geometry, cell count, cell technology, orientation, location within the row, and mounting structure.

It would require an impractically large dataset to densely sample all these factors of variation. Hence, for datasets the size of ours, every newly added PV plant is very likely to produce yet another cluster separated from all other clusters by domain shift [222].

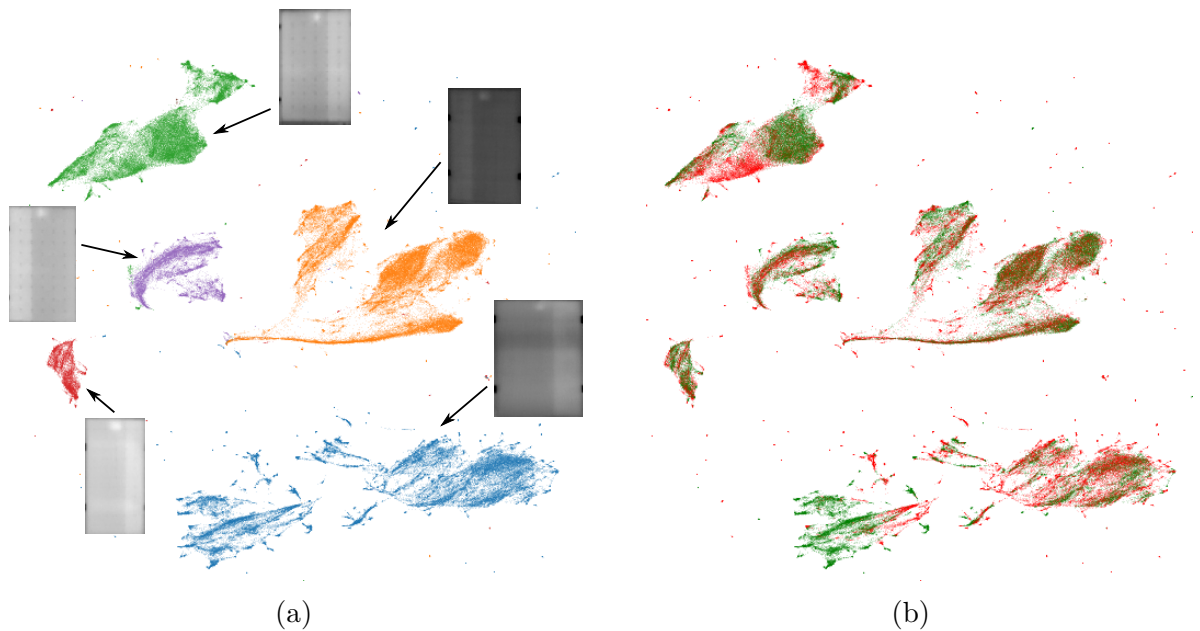

|  |  |
|:--:|:--:|
| (a) | (b) |

Figure 3.3: Illustration of domain shift in a dataset of 620780 IR images of PV modules from five different PV plants. Half of the modules exhibit a thermal anomaly. Colors in (a) indicated the plant and in (b) green and red refer to normal and abnormal PV modules. The figure is based on our publication [2].

In the standard setting of deep learning, training and testing data are sampled uniformly from all domains [223]. This way, training and testing data share the same underlying distribution, which results in predictive models that perform well on the testing data [224]. However, this setting ignores domain shift.

A more realistic setting, which we propose for PV module anomaly detection in publication [2], is given by *unsupervised domain adaption* [223, 225]. Here, a deep learning model is trained on labelled data of one (or multiple) source domains and makes predictions on another target domain, for which no labelled data is available [223]. This setting is well suited to PV plant inspection, where labelled training data from several PV plants is available, and inspection is to be performed on another PV plant, for which no labelled data can be obtained, as this would be too time-consuming and expensive. In this case, training and testing data exhibit a distributional shift. Many methods have been proposed to actively overcome this shift [226–228]. However, a simpler method is to learn more informative and domain-agnostic features with a supervised contrastive loss [229–232].

## 3.1.5 Supervised Contrastive Loss for Image Classification

Supervised contrastive losses have recently been proposed as an alternative to the categorical cross-entropy loss (see sec. 3.1.2) for the training of CNN classifiers [233]. In publication [2], we use a supervised contrastive loss to learn domain-agnostic features for binary classification of PV module anomalies. Supervised contrastive losses are a subset of the broader category of *contrastive losses* [234, 235] that replace the older triplet [236], max-margin [237] and $N$-pair losses [238] for deep metric learning. As opposed to the cross-entropy loss, which acts on the probability distribution of predicted class labels, contrastive losses directly optimize the feature space, so that visually similar images lie close to another in feature space and as far away as possible from visually dissimilar images. To this end, the contrastive loss groups the images in each iteration into *positives* and *negatives*. Positives are attracted to each other and repelled from all negatives, which, in addition, are spread evenly over the feature space.

The most popular contrastive loss is the non-parametric softmax classifier [239] with its variants, such as InfoNCE [240] and NT-Xent [241], which has the following form for the $i$th training sample in a batch of $M$ images

$$\mathcal{L} = -\log \frac{\exp\left(\mathbf{z}_i \cdot \mathbf{z}_p / \tau\right)}{\sum_{m=0}^{M} \exp\left(\mathbf{z}_i \cdot \mathbf{z}_m / \tau\right)}. \tag{3.5}$$

Here, $\cdot$ denotes the dot product between two feature vectors $\mathbf{z} \in \mathbb{R}^d$ and $\tau \in \mathbb{R}^+$ is a commonly used scalar temperature parameter [239, 242]. If there are multiple positives, $\mathbf{z}_p$ may be replaced by the mean embedding of all positives [243] or the loss may be computed separately for each positive and then summed up [233].

In the recently very popular self-supervised setting, individual images [239, 240] and optionally their augmentations [241, 242, 244, 245] are used as positives, while all other images are negatives. This yields a feature space that discriminates individual images. As opposed to this, supervised contrastive losses use images with the same class label (and optional augmentations) as positives, resulting in a feature space that clusters images based on their class membership. Since the contrastive loss maximizes the margin between clusters, the obtained features are more informative than cross-entropy features [233, 246], which improves domain adaption [229–232] and generalization to unseen classes [246–250].

Other limitations of the cross-entropy loss are also overcome, such as the poor margin [237, 251, 252] and low robustness to label noise [253–255], corrupted data [247] and adversarial perturbations [251, 256, 257].

## 3.2 Data Acquisition

Most works on automated PV plant inspection simply specify requirements on the data acquisition procedure to ensure recorded videos are suitable for downstream processing. Many works can utilize both manually and automatically recorded videos. But some works also require individual images with a specific overlap and sidelap that can only be recorded with an automated waypoint flight, which is a feature provided by many current consumer-market drones. To this end, the operator manually specifies the desired drone trajectory, which the drone then follows automatically.

Several works automate the waypoint planning. For example, Henry et al. [41] detects the outlines of all rows in satellite imagery of a PV plant using binary thresholding and places waypoints at the ends of each detected row (see fig. 3.4a). Similarly, Xi et al. [258] and Sizkouhi et al. [42] extract the boundary of the entire PV plant and compute the shortest trajectory needed to cover the entire plant (see fig. 3.4b).

Typically, there is an error of several meters between satellite imagery and the actual geographic feature. Hence, prior calibration is required. Furthermore, GPS measurement errors can cause the drone to deviate from the desired trajectory. To mitigate these issues, several works [42, 258–260] propose vision-based row-tracking controllers, which are executed in real-time on the flight controller and ensure optimal alignment of the scanned row with the camera center (see fig. 3.4c). In addition, Sizkouhi et al. [42] actively steer the drone towards abnormal PV modules that are detected directly on the flight controller.

While these methods are promising, they require specialized hardware and embedded software, increasing the complexity of the inspection system. Hence, they are useful and economic only for very large plants, where a manual flight would take prohibitively long.
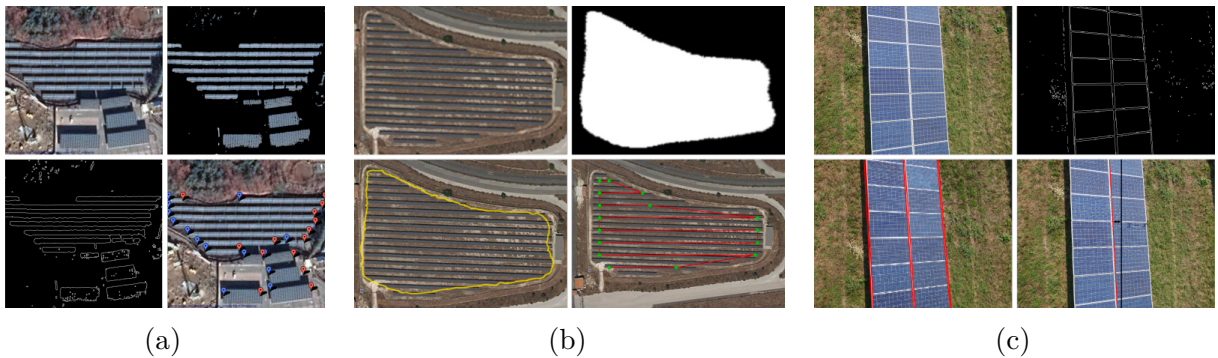


(a)  (b)  (c)

Figure 3.4: Automation of the data aqusition (a) by waypoint planning from plant rows detected in satellite imagery,[2] (b) by waypoint planning from plant outlines detected in satellite imagery,[3] and (c) by visual tracking of the scanned row.[4]

## 3.3 PV Module Detection

Many works on automated PV plant inspection first detect individual PV modules before examining each module for anomalies in a subsequent step. We use the term *detection* to refer to all methods that find regions in an image associated with a PV module. This includes edge detection, semantic segmentation, object detection, and instance segmentation (see sec. 2.4.1). PV modules are detected either directly in each camera image [52, 53, 55, 177, 261, 262], or in a composite image (orthophoto or stitched panorama) of the entire PV plant or parts of it [49–51, 263, 264]. PV module detection is mostly performed in the thermal IR images [46, 177, 178, 261], but sometimes also in visual images [49], or both [53, 263]. Some works post-process detected PV module areas by projecting them onto a rectangle to correct for perspective distortion [53, 55, 265]. While earlier works utilize exclusively classic image processing methods (see sec. 2.4.3), more recent works have shown promising results using traditional machine learning (see sec. 2.4.4) and state-of-the-art deep learning methods (see sec. 3.1).

### 3.3.1 Classic Image Processing Methods

Many works detect PV modules by combining several classic image processing algorithms, such as binary thresholding, edge detection, and morphological operations. While each work uses a slightly different combination of these algorithms, three broader categories emerge.

Works of the first category isolate the edges of each PV module using Canny edge detection [42], morphological operations [266], or Sobel and Laplacian operator for computation of the first and second horizontal and vertical image derivatives [264]. No further processing is applied to obtain a refined representation of the PV modules.

Methods in the second category detect lines by applying the Hough transform directly on the image [52], or on the image resulting from prior Canny edge detection [46, 262, 265] (see fig. 3.5a). As the detected lines may extend beyond the PV modules and there may be many false positive lines, a post-processing step is required. Here, clusters of horizontal and vertical lines are formed and lines that deviate too much from the two clusters are removed. Afterwards, line segments are merged by hierarchical clustering, long lines are truncated, and missing line segments are added. This yields a line grid representing the module outlines.

The third category comprises methods that embrace thresholding to segment PV modules. One group of works utilizes color information by separately thresholding each channel of

---

[2]Reprinted from "Automatic detection system of deteriorated PV modules using drone with thermal camera," by C. Henry, S. Poudel, S.-W. Lee et al., 2020, Applied Sciences, vol. 10, p. 3802 [41]. CC BY.

[3]Reprinted from "RoboPV: An integrated software package for autonomous aerial monitoring of large scale PV plants," by A. Moradi Sizkouhi, S. Esmailifar, M. Aghaei et al., 2022, Energy Conversion and Management, vol. 254, p. 115217 [42]. CC BY-NC-ND.

[4]Reprinted from "A computer vision line-tracking algorithm for automatic UAV photovoltaic plants monitoring applications," by G. Roggi, A. Niccolai, F. Grimaccia et al., 2020, Energies, vol. 13, no. 4, p. 838 [259]. CC BY.

a color image [49–51] (see fig. 3.5b). The obtained masks are filtered based on area and aspect ratio to remove spurious detections. The other works in this category do not directly segment modules, but instead, segment entire rows. Modules are then isolated either by detecting horizontal and vertical lines using a Sobel operator [261, 267] (see fig. 3.5c) or morphological operations [55], or by extracting and filtering rectangular module candidate regions with the maximally stable extremal regions (MSER) algorithm [53].

All of these methods rely on classic image processing, which has several major disadvantages (see sec. 2.4.6). Most importantly, due to the small amount of data used in most works, it is questionable, whether the developed methods generalize across different PV plants.
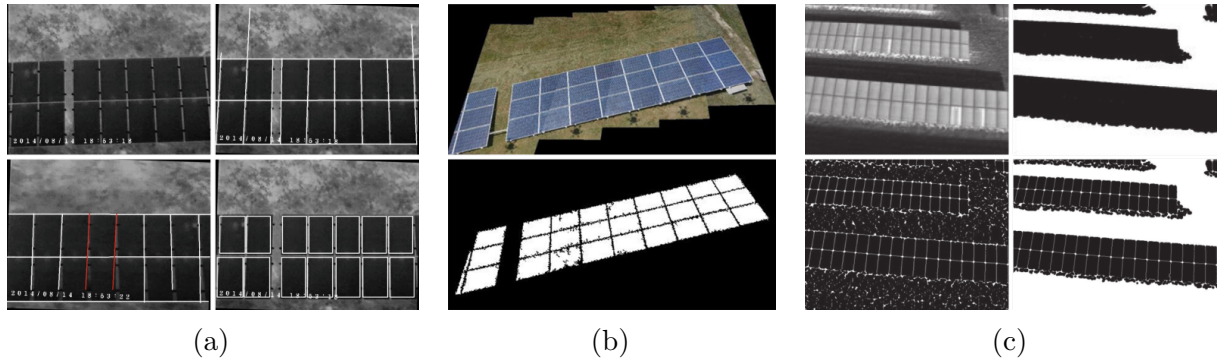


(a) (b) (c)

Figure 3.5: PV module detection (a) by line detection,[5] (b) by color thresholding,[6] and (c) by thresholding of rows with subsequent isolation of modules.[7]

### 3.3.2 Machine Learning Methods

To overcome the problems of classic image processing, a few works propose traditional machine learning methods for PV module detection. For instance, Díaz et al. [177] detect rectangular candidate regions with classic image processing methods, extract 440 different texture features, and then use an SVM to classify whether a region contains a PV module or not. A disadvantage of this method is that it still relies partially on classic image processing to extract candidate regions.

Template matching is another data-driven method proposed for module detection by Addabbo et al. [54]. Here, a template image of a single PV module is moved over the input image in a sliding-window fashion. The normalized cross-correlation between template and input image is computed at each position and maxima are selected as potential locations of

---

[5]Reprinted by permission from Springer Nature Customer Service Centre GmbH: Springer Nature, Image Analysis and Recognition by Fakhri Karray, Aurélio Campilho and Farida Cheriet [52]. © 2017, all rights reserved.

[6]Reprinted from "PV plant digital mapping for modules' defects detection by unmanned aerial vehicles," by A. Niccolai, S. Leva, F. Grimaccia, 2017, IET Renewable Power Generation, vol. 11, no. 10, pp. 1221–1228 [49]. © 2017 John Wiley and Sons.

[7]Reprinted from "Automatic photovoltaic panel area extraction from UAV thermal infrared images," by D. Kim, J. Youn, C. Kim, 2016, Journal of the Korean Society of Surveying, Geodesy, Photogrammetry and Cartography, vol. 34, pp. 559–568 [267]. © 2016 Korea Institute of Science and Technology Information.

PV modules. A disadvantage of template matching is that it does not effectively leverage larger amounts of available training data, as only a single template image is used. This also leads to problems when the appearance of PV modules in the image changes, for example, due to changes in perspective, module type, or module orientation. Furthermore, it is unclear which image to select as the template.

### 3.3.3 Deep Learning Methods

Several recent works have applied CNN-based models to PV module detection. Among the first were Zhang et al. [268], who combine a ResNet-34 with UNet for semantic segmentation of PV modules (see fig. 3.6a). However, as the method cannot distinguish individual modules, its applicability is limited. To overcome this problem, Greco et al. [178] and Pérez et al. [269] use the YOLO object detector to obtain bounding boxes of individual modules (see fig. 3.6b). However, as the bounding boxes are aligned with the image axes, they do not accurately represent the outlines of possibly rotated modules. Díaz et al. [177, 263] solve this problem by using Mask R-CNN for instance segmentation, which not only obtains the bounding box of each module, but also a mask indicating the pixels belonging to the module (see fig. 3.6c).

There are also works that detect entire rows or panels of modules using Mask R-CNN for instance segmentation [270] and R-CNN for object detection [271] (see fig. 3.6d). The problem of this approach is the strong dependency on the spatial arrangement of the modules in the panel or row, which can vary between different PV plants. Consequently, a R-CNN or Mask R-CNN trained in this way may not generalize to a PV plant with a module arrangement that differs from that of the plant used for training.
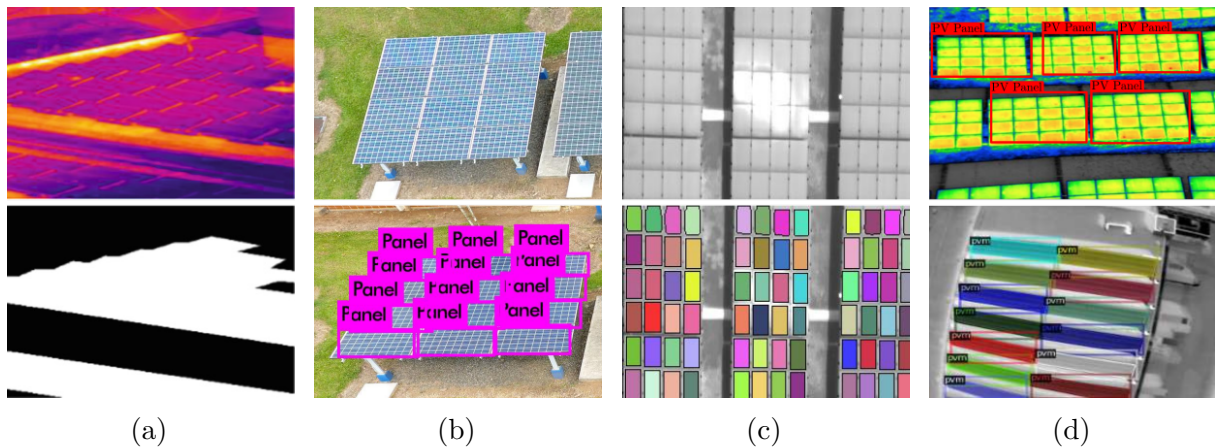


| (a) | (b) | (c) | (d) |

Figure 3.6: PV module detection with (a) UNet,[8] (b) YOLO,[9] and (c) Mask R-CNN.[10] In (d) entire module tables are detected.[11]

---

## 3.4 PV Module Anomaly Detection

Detection of PV module anomalies is at the core of every automated PV plant inspection system. Consequently, a vast number of methods for this task have been published in recent years. Older methods are based exclusively on classic image processing (see sec. 2.4.3), while more recent ones use machine learning (see sec. 2.4.4) and deep learning (see sec. 3.1). Despite their differences, all methods make use of the characteristic thermal patterns (see sec. 2.2.2) to detect anomalous modules in thermal IR images. The term *detection* is used generically for all methods. Different methods may frame the anomaly detection task as semantic segmentation, object detection, binary image classification, or multiclass image classification. Most works perform anomaly detection on IR images of individual PV modules, hence requiring preceding PV module detection. Only a few works operate on the entire camera image [179, 262, 271–273].

## 3.4.1 Classic Image Processing Methods

The main building blocks of classic methods for PV module anomaly detection are, again, edge detection and binary thresholding. One of the most simplistic methods by Tsanakas et al. [56] applies merely Canny edge detection to the IR image to obtain the outlines of hot regions in the module. No additional processing is performed to handle noise or spurious edges. They improve on this in a follow-up work by filtering out edges with short contour lengths [274] (see fig. 3.7a). Wei et al. [262] go even further and analyze a variety of geometric properties of the detected edges. However, the authors report that the method fails to distinguish between actual hot spots and sun reflections on the module surface.

Another simplistic, yet impractical approach is the segmentation of hot spots in the IR image by non-adaptive binary thresholding [49, 50, 52] (see fig. 3.7b). This is impractical, as the fixed threshold value must be constantly manually adjusted to account for changes in the module temperature, for example, due to changes in the momentary irradiance or air temperature. A possible fix is the use of adaptive threshold values, which are automatically computed from the statistics of the image intensity histogram [53, 275, 276]. However, one problem remains. Binary segmentation considers only pixel values and ignores spatial connectivity of the segmented regions, which can lead to noise and spurious blobs in the segmentation mask that require further cleaning.

---

[9]Reprinted from "Solar panels recognition based on machine learning," by R. M. Pérez, J. Solano Arias, A. Méndez-Porras, 2019 [269]. © 2019 IEEE.

[10]Reprinted from "Solar panel detection within complex backgrounds using thermal images acquired by UAVs," by J. J. Vega Díaz, M. Vlaminck, D. Lefkaditis et al., 2020, Sensors, vol. 20, no. 21, p. 6219 [177]. CC BY.

[11](Top) Reprinted from "Photovoltaic plant condition monitoring using thermal images analysis by convolutional neural network-based structure," by Á. Huerta Herraiz, A. Pliego Marugán, F. P. García Márquez, 2020, Renewable Energy, vol. 153, pp. 334–348 [271]. © 2020, with permission from Elsevier. (Bottom) Reprinted from "Using Mask R-CNN to isolate PV panels from background object in images," by M. Sait, A. Erguzen, E. Erdal, 2020, International Journal of Trend in Scientific Research and Development, vol. 5, no. 1, pp. 1191–1195 [270]. CC BY.

As a solution, some works grow segmentation masks starting from the hottest pixels in the image and iteratively include pixels adjacent to already segmented regions if their intensities meet a criterion [46, 277].

All of these works share the general disadvantages of classic image processing (see sec. 2.4.6). Another problem is the use of small datasets (median image count of 34, mostly from a single PV plant) for development and validation (see tab. 3.1), which raises doubts about the general applicability of the methods. The only exceptions in this regard are the works by Arenella et al. [52] and Carletti et al. [46], which use 1171 images and 14215 images of multiple PV plants, respectively. As most methods are validated on only a few images of anomalies with large temperature gradients, such as Cs+, Cm+, Sh, Sp, and Mp anomalies (see fig. 2.4), it is also unclear how sensitive the methods are to anomalies with smaller temperature gradients, such as the important Mh and Pid anomalies. While the methods may be sensitive to different types of anomalies, most of them do not perform the additional processing, such as shape analysis, needed to differentiate anomaly patterns. Hence, they can make at most a binary prediction. However, many of the works do not even make an actual prediction in the form of a compact single-word or single-digit class label. Instead, they merely transform the IR image into a different image, such as a binary mask or edge image.
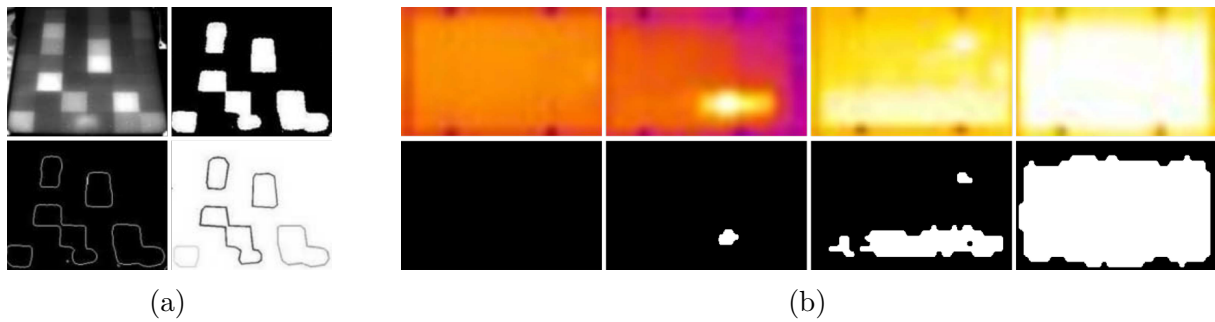


Figure 3.7: PV module anomaly detection by (a) edge detection,[12] and (b) binary thresholding.[13]

## 3.4.2 Machine Learning Methods

A variety of works have explored traditional machine learning for PV module anomaly detection. Niazi et al. [93] extract global image features, such as the histogram of oriented gradients and texture features (homogeneity, contrast, energy, and local binary pattern), and use these to train a binary naive Bayes classifier. Using the same features, Ali et

---

[12]Reprinted from "Fault diagnosis of photovoltaic modules through image processing and Canny edge detection on field thermographic measurements," by J. Tsanakas, D. Chrysostomou, P. Botsaris et al., 2015, International Journal of Sustainable Energy, vol. 34, no. 6, pp. 351–372 [274]. © 2015 Taylor & Francis.

[13]Reprinted from "PV plant digital mapping for modules' defects detection by unmanned aerial vehicles," by A. Niccolai, S. Leva, F. Grimaccia, 2017, IET Renewable Power Generation, vol. 11, no. 10, pp. 1221–1228 [49]. © 2017 John Wiley and Sons.

Table 3.1: Dataset sizes of related works on PV module anomaly detection. *Binary* indicates that the method is sensitive to multiple types of anomalies but can not differentiate them.

| Category | Work | Year | Images | Modules | Plants | | Anomaly Classes |
|---|---|---|---|---|---|---|---|
| Classic | [276] | 2015 | < 10 | 3 | 1 | 1 | Hot spot |
| | [274] | 2015 | < 10 | 8 | 1 | 1 | Hot spot |
| | [56] | 2016 | < 10 | – | 1 | 1 | Hot spot |
| | [275] | 2016 | 1 | 1 | 1 | 1 | Hot spot |
| | [52] | 2017 | 1171 | 84 | >1 | 1 | Hot spot |
| | [49] | 2017 | 34 | – | 1 | 3 | Mh, Sh, Cs+ |
| | [50] | 2018 | 34 | – | 1 | 3 | Mh, Sh, Cs+ |
| | [277] | 2018 | < 10 | 2 | 1 | 1 | Hot spot |
| | [46] | 2019 | 14215 | 14 215 | >1 | 1 | Hot spot |
| | [53] | 2020 | 40 | 240 | 1 | 1 | Hot spot |
| Classic, DL | [262] | 2019 | 110 | – | 1 | 2 | Hot spot, Sun reflection |
| ML | [55] | 2016 | 37 | 1544 | 1 | 3 | Mh, Sh, Cs+ |
| | [278] | 2017 | 120 | 120 | 1 | 5 | Mh, Sh, Cs+, Sp, Soiling |
| | [261] | 2017 | 3 | 204 | 1 | 1 | Binary |
| | [54] | 2018 | 270 | – | 1 | 1 | Hot spot |
| | [93] | 2019 | 375 | 375 | 1 | 2 | (Non-)defective hot spot |
| | [279] | 2020 | 315 | 315 | 1 | 2 | (Non-)defective hot spot |
| | [280] | 2020 | 4 | 4 | 1 | 1 | Hot spot |
| | [281] | 2020 | 3 | 3 | 1 | 2 | Mh, Cs+ |
| | [282] | 2021 | 120 | – | 1 | 1 | Binary |
| ML, DL | [80] | 2020 | 783 | 783 | 1 | 4 | Sh, Mp, Sp, Cs+ |
| | [283] | 2021 | 1428 | 480 | 1 | 5 | Cs+, Chs, Sh, Cracking, Shading by another row |
| | [284] | 2021 | 20000 | 20 000 | >1 | 11 | Mh, Cs, Cm, Cs+, Cm+, Sh, multiple Sh, Soiling, Cracking, Vegetation, Shadowing |
| DL | [272] | 2018 | 3336 | – | 1 | 1 | Binary |
| | [179] | 2019 | < 10 | – | 1 | 3 | Sh, Cs+, Open-circuit string |
| | [273] | 2020 | 1009 | – | 1 | 3 | Mh, multiple Mh, Open-circuit string |
| | [271] | 2020 | 800 | – | 1 | 1 | Hot spot |
| | [285] | 2020 | 893 | 893 | >1 | 1 | Binary |
| | [180] | 2021 | 1000 | 1000 | >1 | 1 | Hot spot |
| | [286] | 2021 | 20000 | 20 000 | >1 | 11 | Mh, Cs, Cm, Cs+, Cm+, Sh, multiple Sh, Soiling, Cracking, Vegetation, Shadowing |
| | [42] | 2022 | 3584 | – | >1 | 1 | Bird droppings |
| | [263] | 2022 | 9000 | 9000 | 6 | 1 | Hot spot |
| | [47] | 2022 | 93220 | 93 220 | 28 | 5 | Mh, Sh, Cs+, Mp, Chs |

al. [279] train multiple binary classifiers, such as an SVM, *k*-nearest neighbor, naive Bayes, quadratic discriminant analysis, and bagging ensemble, and Le et al. [284] train an SVM for the classification of 11 different anomalies. Deitsch et al. [55] also extract global features, such as module medians, cell-wise medians, histogram skewness, and vertical projections, and classify three types of anomalies with a cascaded Grubb's test and Dixon's Q test. As global image features lack expressiveness, Dunderdale et al. [80] extract local SIFT features, encode them with the bag of words method, and train both an SVM and random forest to classify five types of anomalies. The approach by Su et al. [283] is similar, but classifies seven anomalies.

Other data-driven methods proposed for anomaly detection are *k*-means clustering [280], principle component analysis [282], template matching [54], and fuzzy rule based classifiers [278, 281].

Machine learning overcomes some of the limitations of classic image processing (see sec. 2.4.6). Although being larger than those of the classic image processing methods, the datasets used still have a relatively low median image count of 195 (see tab. 3.1). And, with exception of the recent work by Le et al. [284], still only a single PV plant is considered. Training on such small datasets can lead to predictive models that are overfitted and, therefore, generalize poorly. Furthermore, not all works utilize the ability of machine learning models to perform fine-grained classification and instead make only a binary prediction.

### 3.4.3 Deep Learning Methods

At the start of this thesis, only a few works used deep learning for PV module anomaly detection [80, 179, 262, 272]. But in the meantime the body of literature on CNN-based methods for this task has grown rapidly.

Several methods perform anomaly detection directly in the camera image, i.e., without performing PV module detection beforehand. For instance, Pierdicca et al. [272] perform binary image classification with a VGG-16 CNN-classifier, predicting whether the camera image contains any anomalous module or not. The problem is that this method can neither localize the anomalous module in the image nor does it differentiate between a single and multiple anomalous modules in an image. These problems can be solved by using semantic segmentation (see fig. 3.8a), as shown by Oliveira et al. [179], who use a model based on VGG-16, and a newer work by Pierdicca et al. [273], which uses UNet, FPN, and LinkNet on top of an EfficientNet backbone. However, semantic segmentation has two problems. First, it provides only a single binary mask for the entire image, which does not distinguish between individual modules. Second, it does not classify different types of anomalies. Object detection solves both problems by detecting the bounding box and predicting the anomaly class of each anomalous PV module in the image. To this end, the R-CNN and newer Faster R-CNN object detectors are employed by Huerta Herraiz et al. [271] and Wei et al. [262] (see fig. 3.8b). However, Huerta Herraiz et al. detect only hot cells without performing any further classification, and Wei et al. only distinguish between hot cells and sun reflections. One issue of object detection is the approximation of the

anomalous region by a rectangular and axes-aligned bounding box. To solve this problem Pierdicca et al. [273] use Mask R-CNN for instance segmentation, which provides not only a bounding box but also an accurate binary mask for each anomalous module. They also classify three different types of anomalies.

Skipping PV module detection, and detecting anomalous modules directly in the camera image, facilitates the detection of string anomalies. In the case of object detection and instance segmentation, both PV module detection and anomaly classification are performed jointly. This means, that only a single neural network needs to be trained. However, it also reduces the amount of control over each of both tasks. Furthermore, additional module-wise preprocessing, such as perspective warping or normalization, cannot easily be applied. Joint learning is also not likely to increase accuracy, as the module location in the image and anomaly class are not correlated.

Hence, many works first detect PV modules, as described in sec. 3.3, and then perform anomaly detection on images of individual modules (see fig. 3.8d). The most common method is supervised binary or multiclass image classification with a CNN-classifier based on categorical cross-entropy loss (see sec. 3.1.2) [47, 80, 180, 284–286]. Contrary to recommended practice, many of these works build custom CNN models, which perform well on the task-specific dataset but may generalize worse to unseen data than the established CNN backbones, such as VGGNet, ResNet, or MobileNet.

To precisely localize hot spots in the PV module image, Sizkouhi et al. [42] perform semantic segmentation with a model based on VGG-16. Similarly, Su et al. [283] and Vlaminck et al. [263] use the R-CNN and Faster R-CNN object detectors (see fig. 3.8c). The problem with these approaches is that not all anomaly patterns are localized but instead spread over the entire module. Examples are the Mh, Mp, and Pid anomalies (see fig. 2.4). The three works do not encounter this problem, as they consider only those anomalies that are localized, such as hot cells and pointlike hot spots.

Several works find by quantitative comparison that deep learning outperforms traditional machine learning [80, 283, 284] and classic image processing [262] for the task of anomaly detection. A small weakness of CNNs is the lower sensitivity to anomalies with a small spatial extent in the image [283, 286].

The works existing at the start of this thesis used small datasets of 110 [262], 783 [80], and 3336 [272] images of one PV plant. Even considering recent publications, the median dataset size has grown to only 1220 images and seven out of the 14 reviewed works still use data from only a single plant (see tab. 3.1). As mentioned before, such small and low-variance datasets are likely to produce models that do not generalize well. This is even more critical for deep learning than for machine learning, as model capacity is much larger, requiring larger amounts of data to prevent overfitting. However, three very recent works use sufficiently large datasets. Zefri et al. [47] use 93220 images of five different anomaly patterns from 28 PV plants, and both Alves et al. [286] and Le et al. [284] use a dataset published by Millendorf et al. [287], which contains 20000 images of 11 anomaly types (see fig. 3.8d). Another important aspect of the dataset is the range of anomaly classes considered. Only a few works use datasets containing all major anomaly classes

[47, 283, 284, 286]. All other works use incomplete datasets, which may result in models that are insensitive to most anomalies occurring in practice. Nevertheless, training on incomplete datasets is an important topic that requires attention, as there will always be yet another rare anomaly class not considered in the training dataset. However, none of the reviewed works addresses this issue.

Another important shortcoming of all works in this section is that they sample training and validation data uniformly from the entire dataset, which disregards the problem of domain shift explained in sec. 3.1.4.



(a)      (b)      (c)      (d)
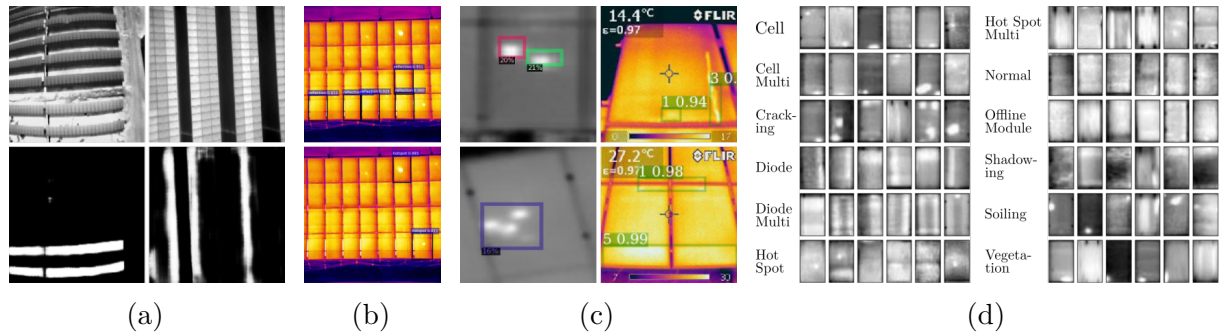
Figure 3.8: Detection of anomalous PV modules in the camera image by (a) semantic segmentation [179], and (b) object detection.[14] In (c) anomalies are localized within a module image by object detection.[15] (d) Shows the 11 classes of the anomaly dataset by Millendorf et al. [287] used in several works.[16]

## 3.5 Localization of PV Modules in the Plant

Localization is the task of obtaining the position of each PV module in the plant, which is essential for guiding maintenance crews when performing repairs. Localizing PV modules from only an aerial video or a set of aerial images is challenging as the images are highly repetitive and typically show only a few PV modules at a time. Several approaches for localization have been proposed in the literature. While a few works provide only a relative module location inside a plant row [49, 50, 264], most works obtain either absolute geolocations [41, 47, 54, 56, 70, 288, 289], or absolute locations based on a schematic site plan of the plant [51]. For absolute geolocations, the GPS trajectory of the drone is used in combination with the imagery.

---

[14]Reprinted from "Hotspots infrared detection of photovoltaic modules based on hough line transformation and Faster-RCNN approach," by S. Wei, X. Li, S. Ding et al., 2019 [262]. © 2019 IEEE.

[15](Left) Reprinted from "Region-based CNN for anomaly detection in PV power plants using aerial imagery," by M. Vlaminck, R. Heidbuchel, W. Philips et al., 2022, Sensors, vol. 22, no. 3, p. 1244 [263]. CC BY.

(Right) Reprinted from "Automated overheated region object detection of photovoltaic module with thermography image," by Y. Su, F. Tao, J. Jin et al., 2021, IEEE Journal of Photovoltaics, vol. 11, no. 2, pp. 535–544 [283]. © 2021 IEEE.

[16]Reprinted from "Remote anomaly detection and classification of solar photovoltaic modules based on deep neural network," by M. Le, V. S. Luong, D. K. Nguyen et al., 2021, Sustainable Energy Technologies and Assessments, vol. 48, p. 101545 [284]. © 2021, with permission from Elsevier.

### 3.5.1 Direct Georeferencing

Direct georeferencing is used for PV module localization by [54, 271, 288]. The method detects PV modules in each video frame and directly projects the module center from image coordinates into a geographic coordinate system using the known camera parameters, and measured position, height, and heading of the drone [290].

Direct georeferencing is simple and computationally cheap but requires centimeter-accurate estimates of the position, heading, and altitude of the drone, which are difficult to obtain. This as well as the missing consideration of visual cues makes direct georeferencing prone to GPS measurement errors. Furthermore, the camera needs to always point vertically downwards, as any roll or pitch movements lead to errors in the position estimate, making drone operation more complicated. Furthermore, site elevation is not considered, leading to problems with PV plants in hilly terrain.

### 3.5.2 Orthophoto

A common approach for PV module localization is the creation of a georeferenced orthophoto of the entire PV plant (see fig. 3.9) from a smaller number of nadiral images acquired at a higher altitude of 25 m to 150 m (depending on the camera) [47, 56, 70, 263, 289]. To this end, usually, a software like OpenSfM [161], Pix4D [291] or OpenDroneMap [292] is used. PV modules and module anomalies are detected directly in the orthophoto.

To create an orthophoto, first, a georeferenced 3D reconstruction (point cloud) of the PV plant as well as the 6D camera poses of all images are obtained for example with incremental SfM (see sec. 2.4.8). Next, a triangular mesh is computed from the 3D point cloud. For each mesh vertex, the image whose normal is closest to the vertex normal is selected. Projecting these images onto the mesh vertices yields the orthophoto, which is further refined by texture and color adjustments [293].

The orthophoto is advantageous for PV module localization as it provides the absolute geocoordinates of each module, is flexible regarding the plant layout, and facilitates fast inspections as images are captured at a high altitude. As opposed to direct georeferencing, both GPS and visual cues are considered, which increases robustness towards GPS measurement errors. Often standard GPS is sufficient, and no special RTK-GPS is required. On the downside, the method is computationally expensive and high flight altitudes may not be feasible if there are nearby streets, train tracks, or power lines. Furthermore, images with well-defined overlap and sidelap are required limiting the approach to automated flights. The high flight altitude also results in a lower image resolution, which, in combination with potential visual artifacts in the orthophoto, can negatively impact downstream module and anomaly detection. As each module is visible in only a few images, the amount of data resulting from an orthophoto is comparably small, limiting its usefulness for the creation of large machine learning datasets.

---

[17]Reprinted from "Smart inspection of a solar farm using drones," 2015, `https://www.pix4d.com/blog/smart-inspection-of-a-solar-farm/` [294]. © 2015 Pix4D SA, all rights reserved.
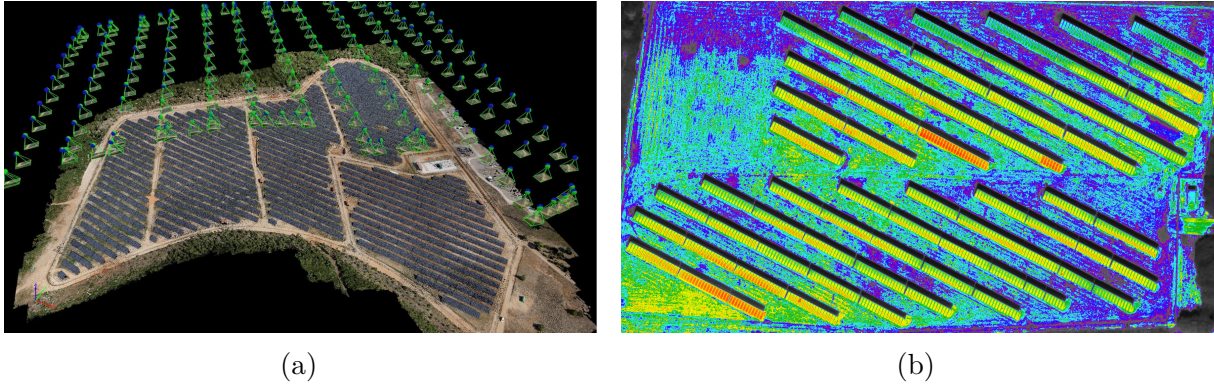
Figure 3.9: Orthophoto for PV module localization. (a) Reconstructed camera poses and point cloud with projected textures.[17] (b) Orthophoto created from thermal IR images.[18]

### 3.5.3 Georeferencing with Structure from Motion

Similar to the orthophoto-method, López-Fernández et al. [48] use SfM to obtain a georeferenced 3D reconstruction of the PV plant in the form of a dense point cloud. They complement the point cloud with temperature measurements from the IR image and intensity values from the visual image and perform PV module detection and anomaly detection directly in the resulting 5D point cloud (see fig. 3.10).

This method shares most of the advantages and disadvantages of the orthophoto method. As an additional advantage, basing the SfM procedure on visual instead of IR images, is likely to improve reconstruction accuracy and robustness, due to the higher resolution of visual images. However, this also requires a camera that simultaneously records visual and thermal IR images. The biggest disadvantage of this method is the need for specialized algorithms to perform PV module and anomaly detection directly in the 5D point cloud. This means the method cannot benefit from the large corpus of readily available image-based methods for these tasks.

### 3.5.4 Panorama Stitching

Several works propose panorama stitching for PV module localization [49, 50, 264]. Here, an image series with specified overlap and sidelap is recorded for each plant row from a low altitude of about 15 m. Keypoints are extracted from each image and matched between subsequent images. Based on matched point pairs homographies are computed that align subsequent images and transform all images into a common panorama image (see fig. 3.11). Constituent images of the panorama are blended to create smooth boundaries and reduce

---

[18]Reprinted from "Photovoltaic plant inspection with thermal camera," 2019, `https://www.drone-thermal-camera.com/photovoltaic-plant-inspection-with-thermal-camera/` [295]. © 2019 Workswell, all rights reserved.

[19]Reprinted from "Automatic evaluation of photovoltaic power stations from high-density RGB-T 3D point clouds," by L. López Fernández, S. Lagüela, J. Fernandez-Hernandez et al., 2017, Remote Sensing, vol. 9, no. 6, p. 631 [48]. CC BY.
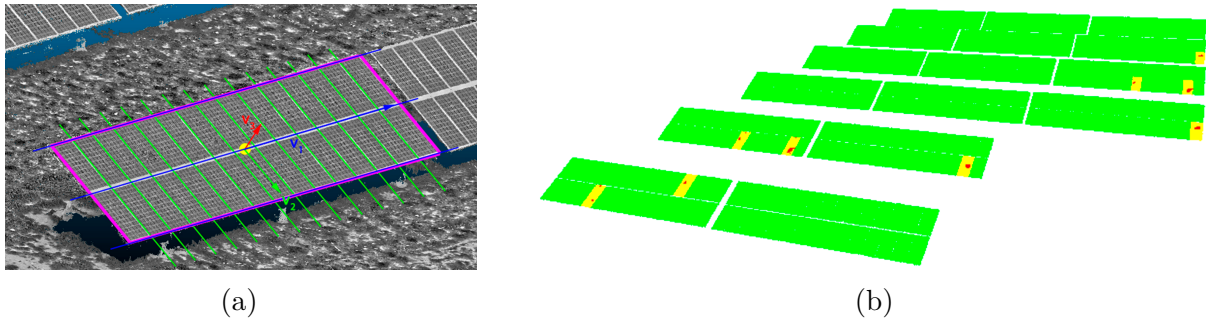
(a)          (b)

Figure 3.10: Detection and localization of (a) PV modules and (b) module anomalies in a dense 3D point cloud of a PV plant that is obtained with SfM.[19]

visual artifacts. Finally, PV modules are detected in the panorama image and enumerated, yielding the relative position of each module in the current row.

Panorama stitching has a moderate computational complexity, requires no GPS trajectory, and produces high-resolution imagery due to the low flight altitude. However, panorama stitching yields only relative locations, requires manual grouping of the images belonging to each plant row, and needs images with well-defined overlap and sidelap that can only be acquired with an automated flight. Furthermore, due to the sequential nature of the approach, errors add up and can lead to large deviations in the image alignment for long sequences. Together with visual artifacts introduced by the image blending, this can negatively impact downstream module and anomaly detection.
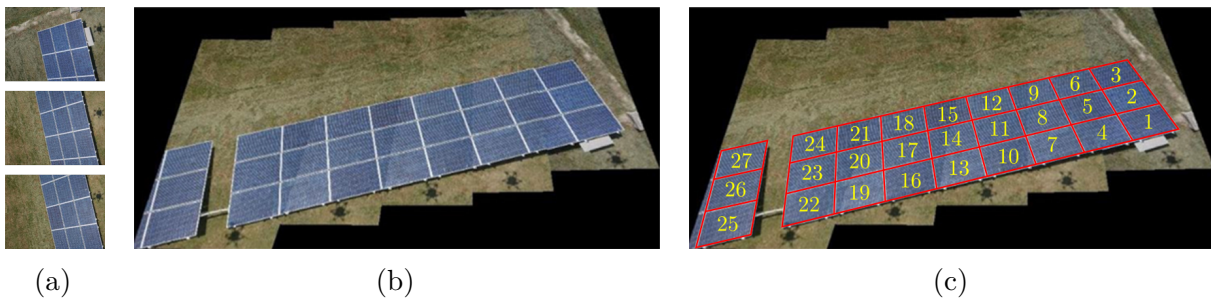


(a)          (b)          (c)

Figure 3.11: Individual images (a) are stitched into a panorama (b), in which PV modules are detected (c).[20]

## 3.5.5 Panorama Stitching with CAD-Plan Matching

To obtain absolute module locations, Niccolai et al. [51] match panorama images of each PV plant row with a CAD plan of the plant (see fig. 3.12). The matching is based on the mean geolocation of all images in a panorama and the geolocation of the plant row as specified in the CAD plan.

---

[20]Reprinted from "PV plant digital mapping for modules' defects detection by unmanned aerial vehicles," by A. Niccolai, S. Leva, F. Grimaccia, 2017, IET Renewable Power Generation, vol. 11, no. 10, pp. 1221–1228 [49]. © 2017 John Wiley and Sons.

While this approach yields absolute module locations with respect to the CAD plan, it requires a standardized CAD plan, which is often not available.
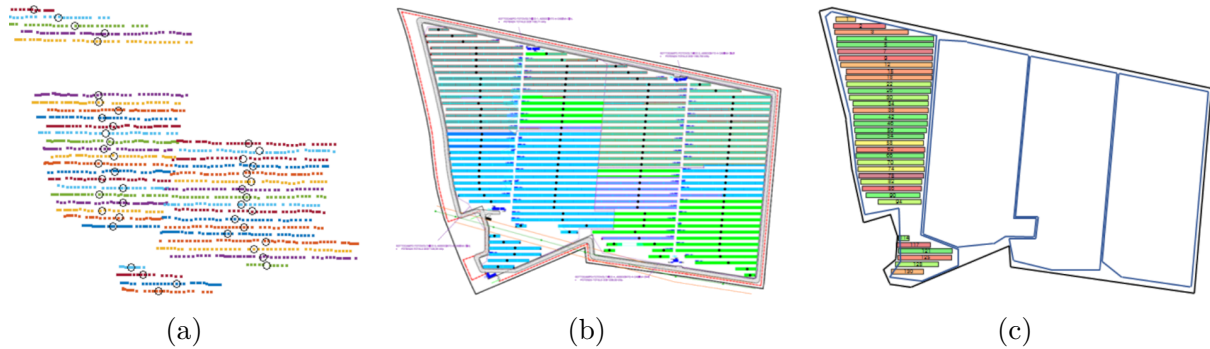


(a)             (b)             (c)

Figure 3.12: Matching of (a) image positions to (b) a CAD plan to obtain (c) absolute PV module locations.[21]

## 3.5.6 Naive Drone Position Logging

A straightforward localization method is to mark the GPS position of the drone on a map whenever the corresponding video frame contains an anomalous PV module [41, 56] (see fig. 3.13). This method is very simple and requires no additional computation but provides only an approximate location of the anomalous module. A manual search is required to identify, which module in the plant refers to the anomalous module in the video frame.
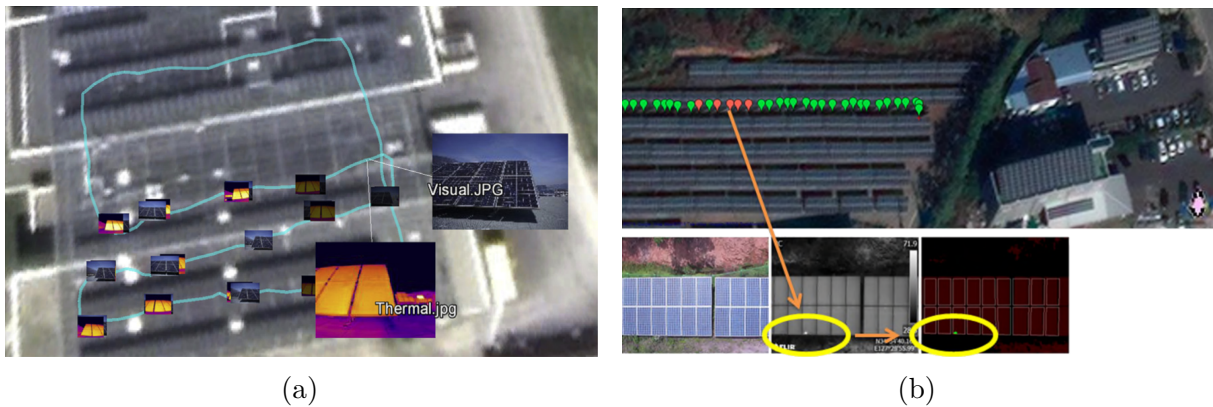


(a)             (b)

Figure 3.13: Naive logging of the drone position for images with anomalous PV modules.[22]

---

[21]Reprinted from "Advanced asset management tools in photovoltaic plant monitoring: UAV-based digital mapping," by A. Niccolai, F. Grimaccia, and S. Leva, 2019, Energies, vol. 12, no. 24, p. 4736 [51]. CC BY.

[22](a) Reprinted from "Advanced inspection of photovoltaic installations by aerial triangulation and terrestrial georeferencing of thermal/visual imagery," by I. A. Tsanakas, L. Ha, and F. Al Shakarchi, 2016, Renewable Energy, vol. 102 (Part A), pp. 224–233 [56]. © 2016, with permission from Elsevier. (b) Reprinted from "Automatic detection system of deteriorated PV modules using drone with thermal camera," by C. Henry, S. Poudel, S.-W. Lee et al., 2020, Applied Sciences, vol. 10, p. 3802 [41]. CC BY.

# 4 Scientific Contributions

This chapter summarizes and discusses our main contributions to the field of automated PV plant inspection. In addition, rationales are provided for many of the method and design choices made, and different methods developed for the same task are compared with each other. The chapter structure is based on the subtasks of an automatic inspection system introduced in sec. 1.2.

## 4.1 Data Acquisition

For the development of our methods, we recorded IR videos of ten different utility-scale PV plants in Germany, containing a total of 152669 PV modules (97.7 % of which are c-Si modules). The videos were recorded with two different models of the DJI Zenmuse XT2 IR camera, one with 13 mm focal length and 8 Hz frame rate and one with 19 mm focal length and 30 Hz frame rate. The use of videos allowed us to collect about 40 images per module, resulting in a total of 6561419 extracted module images, which we used to train our PV module anomaly detection methods.

## 4.2 PV Module Detection

In publication [1] we propose a pipeline for PV module detection, which we reuse in publication [3]. Fig. 4.1 illustrates this detection pipeline.
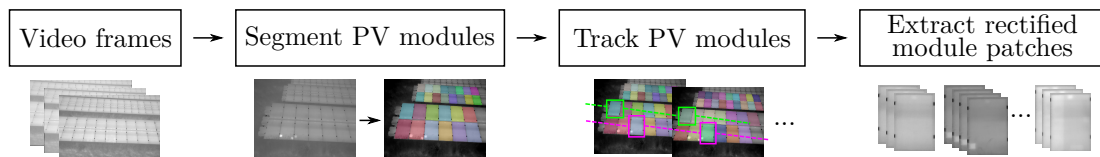


Figure 4.1: Our proposed pipeline for PV module detection based on instance segmentation and multi-object tracking.

We make use of the advantages of deep learning and use the Mask R-CNN instance segmentation framework (with a ResNet-101-FPN backbone) to segment PV modules in each IR video frame. Training and validation of Mask R-CNN are performed on a large dataset of 1165 IR images with a total of 35463 PV modules originating from seven different PV plants. We opt for instance segmentation, as it differentiates individual modules and provides a segmentation mask for each module, which accurately describes the corner points of each module. The corner points are needed for the cropping and downstream localization of each module. Even though there were already newer instance segmentation

frameworks available, we still chose Mask R-CNN, due to its popularity and availability of several mature implementations, which are more performant, less buggy, and easier to use than those of the cutting-edge methods. A quantitative analysis in publication [1] shows that our trained Mask R-CNN achieves a high segmentation accuracy (F1-score of 90.51 % on the validation set) and generalizes well across different ground-mounted PV plants, even those not considered in the training dataset.

We do not only segment PV modules in each image, but we also employ multi-object tracking to associate detections of the same module over subsequent video frames. This is the main enabler of the huge datasets generated by our methods. While it would be possible to extract the same large number of module images from the video without tracking, the resulting dataset would be of little use, as there would be no information about which of the images belong to which PV module. Hence, tracking helps to increase the accuracy of anomaly reports, as ambiguities, in which the same module is analyzed multiple times, are resolved. Tracking also enables aggregation of downstream anomaly detection results over multiple images of the same module, which increases prediction accuracy. Furthermore, it clears the way for filtering of module images with sun reflections, since sun reflections are non-stationary over subsequent module images, whereas the thermal anomaly pattern is stationary.

## 4.3 PV Module Anomaly Detection

Due to the importance of anomaly detection for automatic PV plant inspection, we developed three different methods for this task. The first two methods, published in publication [1] and [2], utilize CNNs for supervised image classification, while the last method from publication [3] uses temperature differences in a local neighborhood of modules to identify anomalies. All three methods work on IR images of individual PV modules that are extracted beforehand as described in sec. 4.2.

## 4.3.1 Supervised Image Classification with ResNet

In publication [1], we perform supervised image classification of thermal anomaly patterns using a ResNet-50 classifier that is trained with categorical cross-entropy loss (see sec. 3.1.2). Since this is a standard problem in deep learning, we follow the recommended practice in terms of data preprocessing, data sampling, data augmentation, optimization procedure, and evaluation metrics. We also follow the common practice of using a model that is pre-trained on the ImageNet-1K dataset, which saves training time. We choose ResNet, as it is very popular and old enough so that there are mature implementations available, which are performant, bug-free, and easy to use. But ResNet is also novel enough, to have solved many of the initial problems of earlier CNNs, such as AlexNet, GoogLeNet, and VGGNet. ResNet is available with different model capacities. While the highest-capacity model may have improved classification accuracy, we use the intermediate model, ResNet-50, due to hardware limitations and to achieve faster training. We train on a large dataset containing images of 453511 PV modules from seven different plants. Furthermore, we aggregate class

predictions over multiple images of the same PV module, which is possible as our dataset contains about 40 images per module. As we show, this improves classification accuracy from 84 % to 90.9 %. Finally, we propose to use class activation maps and inspect the dataset embedded in the learned feature space, to ensure sensible behavior of the classifier.

## 4.3.2 Unsupervised Domain Adaption with Supervised Contrastive Learning

When inspecting our anomaly classification dataset from publication [1], we noted significant domain shift between IR images originating from different PV plants (see sec. 3.1.4). Hence, sampling training and testing datasets uniformly from all domains (PV plants), as done in our publication [1] and in all related works until today, is not a realistic scenario for practical applications. Instead, we propose in publication [2] to frame anomaly classification as an unsupervised domain adaption problem, where the classifier is trained on data of one plant (source) and evaluated on data of another plant (target). This is a more practical setting, as the anomaly detection method must perform accurately on a new PV plant without requiring fine-tuning on labeled images of that plant.

In publication [2], we reuse the anomaly dataset from publication [1]. However, we perform only a binary anomaly classification, to account for the more difficult problem setting and facilitate a more fundamental investigation of the domain shift. We train a ResNet-34 CNN with a supervised contrastive loss (see sec. 3.1.5) and employ a $k$-NN classifier on top of the learned contrastive features (see fig. 4.2). We choose a supervised contrastive loss, as it provides more informative features than the categorical cross-entropy loss [233, 246], which improves domain adaption. Another reason for our choice of supervised contrastive features in combination with a $k$-NN classifier is the improved sensitivity to out-of-distribution samples [248, 296], which facilitates the detection of unknown types of anomaly, i.e., anomalies which are not part of the training dataset. While the cross-entropy loss trains both the CNN backbone and classifier simultaneously, the contrastive loss exclusively trains the CNN backbone. Hence, the classifier needs to be trained in a subsequent step. We opt for a $k$-NN classifier due to its simplicity and ease of interpretation. Like in publication [1], we find that aggregating predictions over multiple images of the same PV module increases prediction accuracy. Furthermore, we propose to analyze how sensitive the anomaly detection method is to changes in the hyperparameters since a low sensitivity to changes is important for a truly practical method.

## 4.3.3 Temperature Distribution for Anomaly Detection

In publication [3], we propose an anomaly detection method that is not data-driven but instead based on the spatial distribution of mean or maximum module temperatures (see fig. 4.3). The method requires preceding PV module localization, as it needs to know which modules are adjacent to each other. This information allows computing the difference between each module's maximum temperature and the median of the maximum
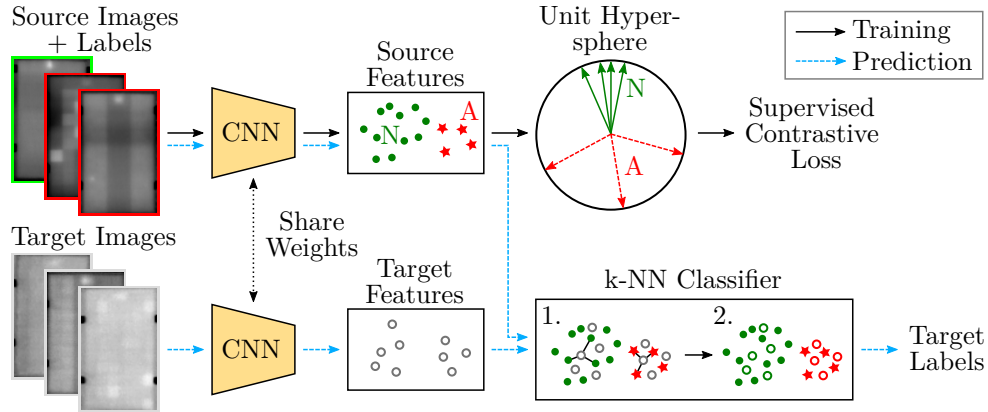
Figure 4.2: Our proposed binary classifier for PV module anomalies trained with a supervised contrastive loss in an unsupervised domain adaption setting.

temperatures of the neighboring modules, effectively highlighting modules with thermal anomalies.



Figure 4.3: Map of a PV plant revealing anomalous PV modules by plotting trend-corrected maximum module temperatures.[1]

This temperature-based anomaly detection method solves a few problems of the CNN-based methods presented above. Above all, it can detect string anomalies. This is important, as string anomalies have a large impact on the overall power and yield of the plant. The temperature method is also more sensitive to anomalies with a small spatial extent in the image (D and Chs anomalies). Furthermore, open-circuited modules (Mh anomaly) can be detected accurately. This is something the CNN classifiers cannot do, as they consider only the relative temperature differences within each module but not between modules. On top of that, the temperature method is easier to interpret, as it is based on a physical model, and requires no hardware acceleration, no labeled training data, and no training phase. The absence of data also means there is no problem with domain shift.

However, the temperature method also comes with several important disadvantages. For instance, it cannot differentiate multiple types of anomalies and is less sensitive to anomalies

---

[1]Reprinted from "Georeferencing of photovoltaic modules from aerial infrared videos using structure-from-motion," by L. Bommes, T. Pickel, C. Buerhop-Lutz, et al., 2022, Progress in Photovoltaics: Research and Applications [3]. CC BY.

with small temperature gradients, such as the important Pid anomaly. There is also no easy way of improving the sensitivity to certain anomaly classes or the overall classification accuracy, which is possible for the CNN classifiers simply by extending the training data or using a higher-capacity model. Because of these disadvantages, the temperature method should not replace the CNN classifier but instead complement it.

## 4.4 Localization of PV Modules in the Plant

We developed two different approaches for the localization of PV modules in the PV plant. In publication [1], we propose a graph matching procedure, which assigns a human-readable ID from a manually created plant description file to each PV module. As this method has several disadvantages, we develop a more capable method in publication [3], which georeferences PV modules using incremental structure from motion.

### 4.4.1 Graph Matching with Plant Description File

The graph matching procedure (see fig. 4.4) proposed in publication [1] creates two graphs for each row of the PV plant, which encode the spatial relationship of PV modules. The first graph contains random tracking IDs of the PV modules that are segmented and tracked in the video. The second graph contains human-readable module IDs as specified in a manually created plant description file. Matching the two graphs yields an assignment between the tracking IDs and human-readable module IDs, which facilitates the localization of each module in the plant.
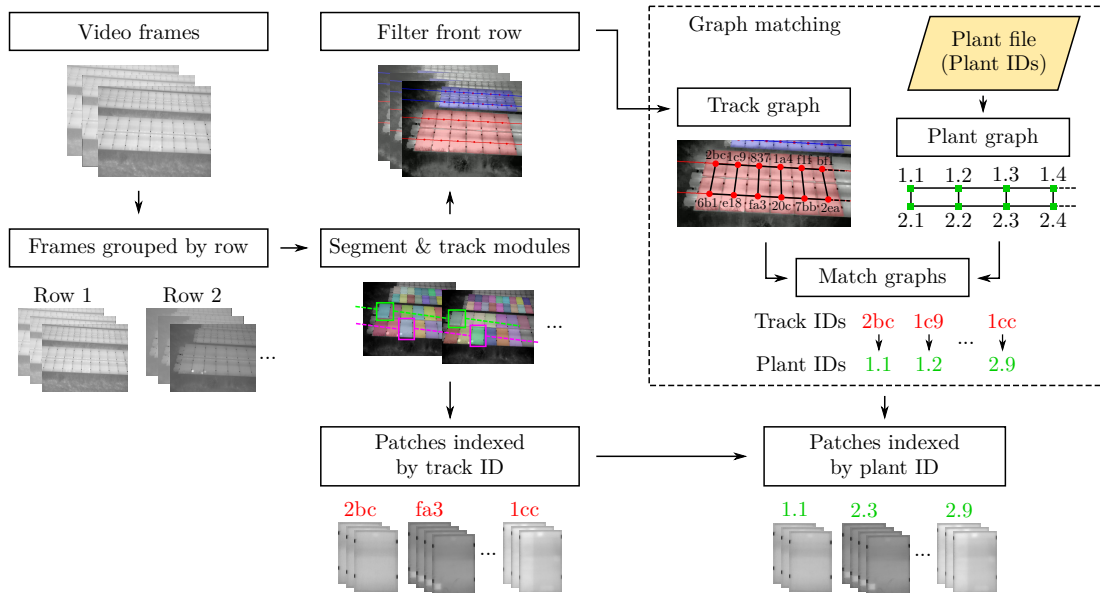


Figure 4.4: Graph matching for PV module localization.

Advantages of this method are the low computational complexity, and the low required accuracy of the measured GPS trajectory. Furthermore, the method can directly operate on
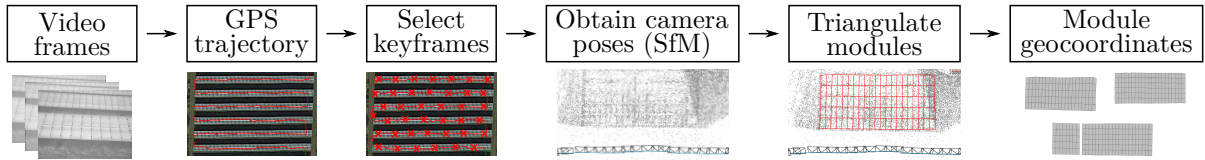
Figure 4.5: SfM-based georeferencing for PV module localization.

videos and does not require nadiral images or images with a specific overlap. However, the greatest advantage of this method is that it provides not only the location of PV modules in the plant but also their electrical connectivity if this information is encoded in the module IDs in the plant description file. Additionally, assigning IDs to each detected module also facilitates easy integration of other data sources, such as electrical measurements. Such additional data is needed to interpret the influence of detected anomalies on the power and yield of the plant.

Despite its advantages, graph matching has several important disadvantages. Most importantly, it has a low degree of automation, involving several manual steps. For instance, the plant description file must be created in a time-consuming manual process, as no standardized file formats for the description of PV plants have been established yet. Moreover, the method operates on each PV plant row individually, requiring the video to be manually split into sections according to the rows of the plant. Finally, graph matching has four possible solutions, which can only be disambiguated with a manually provided seed match. Another important disadvantage of the method is that it works only for plants with a regular row-based layout, which excludes most rooftop plants and floating PV plants. Also, irregularities within the rows, e.g., missing or shifted modules, cannot be handled. Other cases that cannot be handled are tracking errors, e.g., when the current row drifts out of the video frame, and the intrusion of additional rows at the top or bottom of the video frame. The latter necessitates an additional filtering algorithm to discard rows that become visible in the background at low camera angles. Due to the sequential nature of the method, a single error means that the entire plant row cannot be localized. Hence, data acquisition needs to be performed carefully. Nonetheless, the method manages to localize 87.8 % of the 122865 PV modules used in publication [1].

## 4.4.2 Georeferencing with Incremental Structure from Motion

In publication [3], we propose an improved PV module localization method that uses incremental SfM to obtain absolute geocoordinates of all PV modules detected in the video (see fig. 4.5). This allows to plot analysis results, such as the results of anomaly detection, on a geographic map, which is intuitive and shows the distribution over the entire PV plant, possibly providing further insights.

The method uses SfM to obtain a georeferenced 3D point cloud and 6D camera pose of each video frame in a subset of previously selected keyframes. The point cloud is then discarded and the camera poses are used to triangulate the corner points of detected and tracked PV modules into the reconstruction, yielding their geocoordinates (see fig. 4.6).
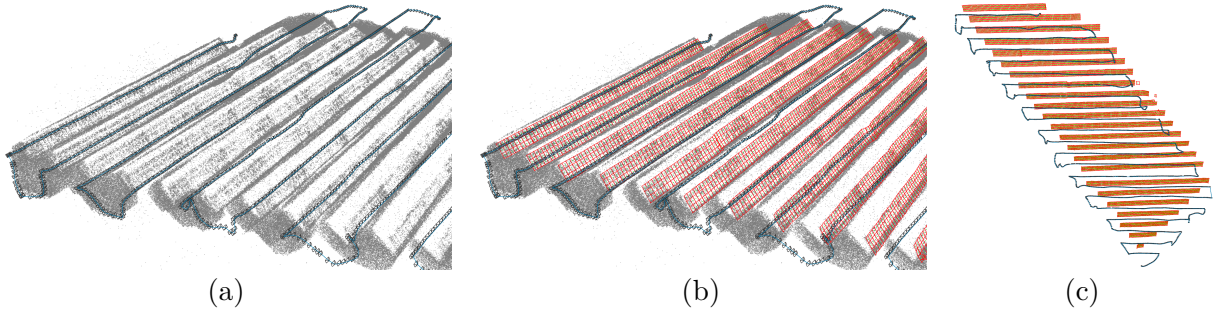
(a)  (b)  (c)

Figure 4.6: SfM reconstruction of a PV plant.[2] (a) Reconstructed feature points (grey) and camera poses (blue line and black camera frustrums), (b) with triangulated PV modules, and (c) top-down view on the triangulated modules of an entire PV plant.

Table 4.1: PV module throughput and resolution for simultaneous scanning of one, two, and three rows. Results are taken from our publication [3] and are measured on 2376 PV modules ($\sim 0.56\,\mathrm{MW}_p$) distributed over 12 rows of a PV plant.

|  | One Row | Two Rows | Three Rows |
|---|---|---|---|
| Flight distance | 1307 m | 681 m | 461 m |
| Flight duration | 707 s | 338 s | 189 s |
| Average module resolution | 141 px × 99 px | 73 px × 50 px | 46 px × 33 px |
| Module throughput | $3.36\,\mathrm{s}^{-1}$ | $7.03\,\mathrm{s}^{-1}$ | $12.57\,\mathrm{s}^{-1}$ |

Georeferencing with SfM overcomes many of the disadvantages of the graph matching-based PV module localization from publication [1]. For instance, it is fully automated, requiring no previous splitting of the video or creation of a plant description file. The new method is also independent of the plant layout, hence can deal with irregularities in the plant rows and can be applied to other types of plants, such as rooftop and floating plants. It is also more robust towards data acquisition errors, not only simplifying the drone operation, but also increasing the success rate of module localization to 99.3 % in our experiments. Another advantage is that multiple plant rows can be processed simultaneously (see fig. 4.7). This facilitates a trade-off between throughput and image resolution of PV modules (see tab. 4.1).

---

[2]Reprinted from "Georeferencing of photovoltaic modules from aerial infrared videos using structure-from-motion," by L. Bommes, T. Pickel, C. Buerhop-Lutz, et al., 2022, Progress in Photovoltaics: Research and Applications [3]. CC BY.

[3]Reprinted from "Georeferencing of photovoltaic modules from aerial infrared videos using structure-from-motion," by L. Bommes, T. Pickel, C. Buerhop-Lutz, et al., 2022, Progress in Photovoltaics: Research and Applications [3]. CC BY.
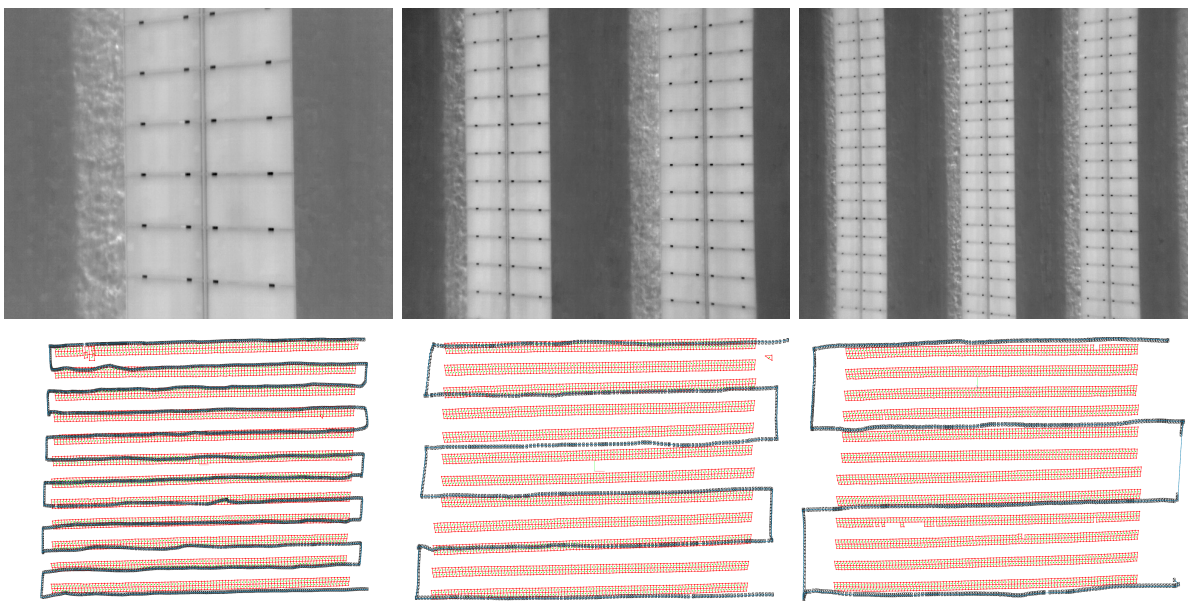
Figure 4.7: Top row: Scanning one, two and three PV plant rows simultaneously. Bottom row: Resulting SfM reconstructions of modules and drone trajectory.[3]

# 5 Discussion

This chapter discusses the improvements made by our contributions, presented in chapter 4, over the state-of-the-art methods for automated PV plant inspection, which were introduced in chapter 3.

## 5.1 Data Acquisition

Like the majority of the related works, all of our methods operate solely on thermal IR imagery. Opposed to some related works that rely on both visual and IR imagery [48, 49, 53, 263], this allows for the use of an IR-only camera instead of a more expensive dual IR/visual camera and renders complex registration of visual and IR imagery needless.

While most of the related works are based on the acquisition of individual images, our methods work with IR videos instead, which has several advantages. Most importantly, videos have a large redundancy, i.e., each PV module is visible in many subsequent video frames. This significantly reduces the effort required to generate the very large datasets that are needed for the training of deep learning methods. Videos also facilitate faster and easier data acquisition than images, as they can be captured when flying at a constant, and possibly high, velocity. Furthermore, videos can be acquired both with an automated and a manual drone flight. In contrast, to take individual images, the drone must either stop, or the image capture has to be synchronized with the current position of the drone to ensure the correct overlap of the images. This always requires an automated flight, which needs to be planned beforehand. However, automated flights make sense only for large plants, whereas for small plants the flight planning and required GPS calibration can take more time than a manual flight performed ad-hoc. The large redundancy of videos is also advantageous for localizing PV modules via panorama stitching (see sec. 3.5.4), orthophoto (see sec. 3.5.2), or incremental structure from motion (see sec. 3.5.3). Typically, for this, individual images are captured with a predefined overlap and sidelap. If at any point the overlap or sidelap is insufficient, there is no way to fix this apart from repeating the data acquisition. However, when using video, one can solve this problem simply by subsampling another set of video frames.

In terms of size, our acquired dataset (see sec. 4.1) is the current state of the art with 6561419 IR images of 152669 PV modules from ten different plants. The extent of our dataset compared to the related works is illustrated in fig. 5.1. As compared to the second-largest dataset by Zefri et al. [47], our dataset contains 70 times as many images and 63.7 % more PV modules. And compared to Carletti et al. [46], which was the state of the art at the time of our publication, our dataset contains 461 times as many images and more than ten times as many modules. As we use deep learning for PV module detection

and anomaly detection, such large training datasets are essential to ensure trained models generalize across different PV plants and are sensitive to all types of thermal anomalies. Large datasets also facilitate the detection of rare types of anomalies, which affect only a handful out of thousands of modules.
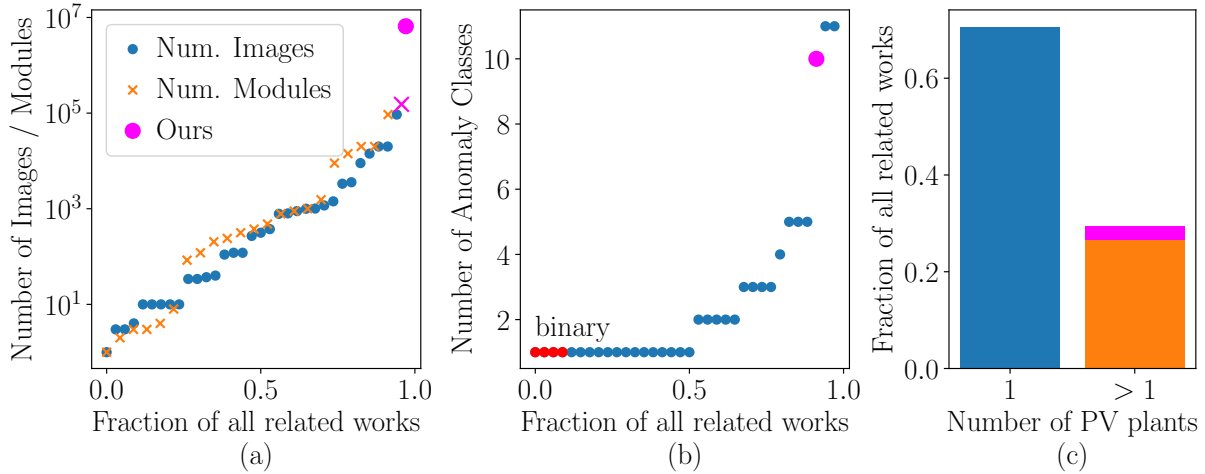


Figure 5.1: Size of our dataset (colored magenta) versus the related works in terms of the numbers of (a) images and PV modules, (b) anomaly classes, and (c) PV plants. The plot is based on data from tab. 3.1. *Binary* indicates that the method is sensitive to multiple types of anomalies but cannot differentiate them.

Another important aspect is the information content of each image, described, for instance, by the spatial resolution. The average image resolution in our dataset is $141 \, \text{px} \times 99 \, \text{px}$ (see tab. 4.1). Assuming module dimensions of $160 \, \text{cm} \times 100 \, \text{cm}$, this translates to a spatial resolution of $1.13 \, \text{cm}$, which is almost three times smaller than the maximum of $3 \, \text{cm}$ recommended by the IEC TS 62446-3 standard [30] (see sec. 2.3.2). Compared to this, the datasets by Zefri et al. [47] and Millendorf et al. [287] (used by Alves et al. [286] and Le et al. [284]) do not fullfill the IEC standard with spatial resolutions of $5 \, \text{cm}$ and $3 \, \text{cm}$ to $15 \, \text{cm}$, respectively. Furthermore, the dataset by Millendorf et al. contains only preprocessed images without absolute temperature values, limiting its usefulness.

## 5.2 PV Module Detection

The majority of related works use classic image processing and traditional machine learning methods for PV module detection (see sec. 3.3.1 and sec. 3.3.2), which have many disadvantages and generally perform worse than deep learning-based methods (see sec. 2.4.6). Hence, several more recent works, including this thesis, employ deep learning for PV module detection (see sec. 3.3.3). Among these, the work by Díaz et al. [177] is most similar to our method presented in sec. 4.2. Díaz et al. also use the popular Mask R-CNN instance segmentation framework to predict segmentation masks for individual PV modules in each image. They were the first to use instance segmentation for PV

module detection. This improved upon works by Zhang et al. [268], Greco et al. [178], and Pérez et al. [269], who use object detection or semantic segmentation, which either cannot distinguish different PV modules or provide only inaccurate bounding boxes instead of pixel-accurate segmentation masks.

Both our work and the state-of-the-art work by Díaz et al. achieve very high segmentation accuracy with F1-scores of 90.5 % and 98.9 % on their respective datasets. However, our method improves upon Díaz et al. in several aspects. The biggest drawback of the method by Díaz et al. is the use of a complex post-processing stage to remove truncated modules and fill in missing modules. This post-processing stage assumes a regular grid layout of modules and uses classic image processing and machine learning methods, reintroducing problems, such as reliance on heuristics, poor generalization, and the need for manual hyperparameter tuning. As opposed to this, we do not perform any post-processing, which maintains the full flexibility of Mask R-CNN and ensures independence of the PV module detection from the plant layout. Moreover, instead of filtering truncated modules in the post-processing stage, we train Mask R-CNN only on fully visible PV modules, which makes for a more powerful filter that is directly embedded into the Mask R-CNN. Similarly, Díaz et al. correct outlier images, i.e., images containing sun reflections, using a filter based on a heuristic that introduces new hyperparameters. We, on the other hand, do not rely on any heuristic filter and perform only parameterless histogram equalization to enhance image contrast. A further drawback of the method by Díaz et al. is that they approximate each PV module in the image as a rotated rectangle. This works only because they use nadiral images, but would fail when using oblique images, where modules exhibit a perspective distortion. Our method is more flexible and accounts for perspective distortion by fitting a quadrilateral to each module and mapping the module onto a rectangle using a homography. This idea stems from several other works [53, 55]. Another important improvement of our work is the use of a much larger training and validation dataset. While Díaz et al. use only 100 IR images showing 18244 PV modules from three different plants, our dataset contains 1165 IR images with a total of 35463 PV modules originating from seven different PV plants. This reduces the risk of overfitting and allows us to use a larger and more capable ResNet-101-FPN backbone inside Mask R-CNN (as compared to the ResNet-50-FPN used by Díaz et al.). Our large dataset also enabled us to analyze both the generalization ability of Mask R-CNN and the correlation between training dataset size and segmentation accuracy, which is something no other related work has done before. Furthermore, compared to Díaz et al. our dataset is not only larger but contains also higher-quality labels. This is because, Díaz et al. label each PV module with a separate polygon, whereas we annotate grid spines between the PV modules in a table. Consequently, our mask labels have perfectly colinear edges.

While some related works detect PV modules in a composite image, such as an orthophoto or stitched panorama [49–51, 263, 264], our method directly detects modules in each image, which is similar to several other works [49–51, 263, 264]. This sidesteps the problem of visual artifacts that may be present in the composite image, improving module and anomaly detection accuracy.

We do not only segment PV modules in each frame, we also track modules over consecutive video frames (see sec. 4.2). The only other related work that employs tracking is the one by Carletti et al. [46]. Like our work, Carletti et al. use tracking to aggregate anomaly predictions for the same PV module over consecutive frames. However, different from us, they do not leverage tracking to extract multiple images for each module. Hence, their dataset contains only a single image for each module, whereas ours contains on average 40 images per module. Furthermore, Carletti et al. do not leverage tracking for the filtering of module images with sun reflections like we do.

## 5.3 PV Module Anomaly Detection

Similar to PV module detection, the field of PV module anomaly detection is currently moving from classic image processing and traditional machine learning (see sec. 3.4.1 and sec. 3.4.2) towards deep learning (see sec. 3.4.3). We were among the first to employ deep learning for PV module anomaly detection by performing supervised multi-class image classification of thermal anomaly patterns with a CNN classifier (see sec. 4.3.1). The only existing work at that time with a similar approach was the one by Dunderdale et al. [80].

However, our work improves upon Dunderdale et al. in several ways. For instance, we use a more recent ResNet-50 CNN, whereas Dunderdale et al. use the outdated VGG-16 architecture and MobileNet, which is optimized for low-computational complexity and consequently less capable. But most importantly, we use a much larger dataset containing 453511 IR images of 11644 PV modules from seven different plants (a subset of our entire dataset presented in sec. 4.1), whereas Dunderdale et al. use only one image for each of 783 modules from three different plants. Our larger dataset reduces the risk of overfitting, facilitates the use of larger and more capable CNN architectures, and results in models that generalize better across different PV plants. However, the biggest advantage of our large dataset is that it covers all ten of the important anomaly classes shown in fig. 2.4. Compared to this, the dataset used by Dunderdale et al. is incomplete, covering only four of the anomaly patterns (Sh, Mp, Sp, Cs+). Leaving out some of the common anomaly classes is problematic, as the resulting model may be insensitive to those left-out anomalies. As these anomaly classes are not part of the validation dataset, there is also no way to quantify, how accurately they can be detected by the model. Distinguishing many anomaly classes is of value for plant operators, as it facilitates a more detailed cataloging of anomalies in a plant. This is important because some anomalies can worsen over time and eventually cause power losses, outages, or even fires. Another fundamental advantage of our method, compared to that of Dunderdale et al., is the availability of multiple images per module, which enables us to boost classification accuracy by aggregating predictions over multiple images of the same PV module. While our method achieves a classification accuracy of 90.9 %, Dunderdale et al. achieve 85.8 % and 89.5 % for the VGG-16 and MobileNet classifiers, respectively. The similar classification accuracies do not mean our improvements are ineffective, though. This is because our validation dataset, compared to the one used by Dunderdale et al., poses a much harder classification problem due to its larger variance and the larger number of anomaly classes and PV plants. The greater

size of our dataset also increases model stability. While our classification accuracy has a standard deviation of only 0.23 % over three training runs, Dunderdale et al. report a much larger standard deviation of 5.2 % over five training runs.

In the meanwhile, several newer works have also used CNNs for supervised image classification of thermal anomaly patterns [47, 180, 284–286]. The works by Manno et al. [180] and Akram et al. [285] still use small datasets of 1000 and 893 images, with, in the case of Manno et al., only covering a single anomaly class (Cs+). However, the other works follow our example and use much larger datasets. Le et al. [284] and Alves et al. [286] use 20000 images with 11 anomaly classes and Zefri et al. [47] use 93220 images with five anomaly classes. While containing more PV modules than our dataset, these newer works have access to only a single image per module, which prevents them from boosting accuracy by aggregating predictions.

Apart from these improvements, we also go a step further than the related works in the way we analyze our trained classification models. While the related works only compute an aggregate classification accuracy or at best a confusion matrix, we additionally visualize learned embeddings and class activation maps to interpret the behavior of the classifier (see sec. 4.3.1). We also analyze the sensitivity of the model with regard to changes in the hyperparameters (see sec. 4.3.2). This is important in practice, where it is desirable to use a single set of hyperparameters for inspecting a wide range of PV plants.

In publication [2], we propose viewing PV module anomaly detection as unsupervised domain adaption problem, to account for the domain shift between different PV plants (see sec. 4.3.2). To date, none of the related works has adopted this view. Instead, even the most recent works still sample training and testing data uniformly from all PV plants in their dataset. This is an unrealistic scenario, because, in practice, the anomaly classifier is trained once on a labeled dataset of one or more PV plants, but later must make predictions on data of another PV plant. Thus, to further advance the state of the art in the field, future works should view PV module anomaly detection as an unsupervised domain adaption problem and actively compensate for the domain shift. Furthermore, no work apart from ours considers the detection of out-of-distribution samples. Being able to accurately detect out-of-distribution samples is important because there is a high chance of encountering novel and rare types of anomalies when just looking at enough different PV plants.

In publication [3], we develop an anomaly detection method that is based on the distribution of module temperatures to overcome some of the remaining problems of the deep learning-based methods presented in sec. 4.3.3. This method is rather unique and nothing comparable has been proposed by any of the related works. We believe this method is important for the field of PV module anomaly detection because it complements deep learning methods in detecting anomalies with a small spatial extent and string-level anomalies. Detecting string-level anomalies is essential for PV plant inspection, as underlined by a study by Weinreich et al., who analyzed 185 c-Si PV plants with a total capacity of $702\,\mathrm{MW_p}$, and found that 83.6 % of the overall power loss could be attributed to string failures [23]. Our work on this method also highlighted that deep learning is no miracle

cure for PV module anomaly detection, but also has several unique disadvantages that must be compensated with other classic techniques.

Similar to most of the related works (see sec. 3.4), all three of our anomaly detection methods perform anomaly detection after having performed PV module detection in a previous step. This is opposed to some of the related works, which combine both module and anomaly detection in a single deep learning algorithm, e.g., Pierdicca et al. [272, 273], Oliveira et al. [179], Huerta Herraiz et al. [271], and Wei et al. [262]. Tackling module and anomaly detection independently, not only simplifies method development and ensures optimal performance on both tasks, but also allows us to easily combine different anomaly detection methods with the same PV module detection method. Most importantly, performing both tasks sequentially facilitates the creation of a reusable dataset of extracted PV module images, which can be shared with the community to advance the state of the art in PV module anomaly detection.

## 5.4 Localization of PV Modules in the Plant

Many methods for the automated localization of modules in the PV plant have been proposed in the related works (see sec. 3.5). Some of these methods localize PV modules relative to each other by means of panorama stitching (see sec. 3.5.4). Others obtain absolute geocoordinates by direct georeferencing (see sec. 3.5.1), or by creating an orthophoto (see sec. 3.5.2) or a 3D reconstruction of the plant (see sec. 3.5.3) with the help of SfM. In this thesis, we developed two methods for PV module localization, one based on graph matching (see sec. 4.4.1) and one based on georeferencing with SfM (see sec. 4.4.2).

Our graph matching procedure differs substantially from most of the related works, but shares some similarities with the method by Niccolai et al. [51]. Niccolai et al. first stitch images of each plant row into a panorama image, which is then matched to a CAD plan of the plant (see sec. 3.5.5). Like our method, this yields an index for the PV modules, which allows for absolute localization in the plant and integration of external data sources. Also like our method, images need to be grouped manually according to the scanned PV plant row, only plants with regular grid-like row layouts can be processed, and only one row can be scanned at a time, limiting the throughput. In addition to these limitations, Niccolai et al. require nadiral images with a well-defined overlap and sidelap that can only be acquired with an automated flight. As opposed to this, our requirements for the data acquisition are less stringent, thanks to the use of videos (see sec. 5.1). Furthermore, we need only coarse GPS positions for the manual grouping of frames into rows. Niccolai et al., on the other hand, utilize GPS positions for the CAD plan matching, thus, requiring more accurate GPS measurements. Moreover, the CAD plan must contain the GPS positions of each PV module. However, this is often not the case, and manually capturing the GPS position of each module is too time-consuming to be economically feasible. Different from this, our method does not require the GPS positions of the modules, but, instead, uses a text file that encodes the spatial relation between modules in each row. If a CAD plan is available, this text file can be derived automatically, else it can be created manually in a reasonable amount of time. Since Niccolai et al. stitch images sequentially into a

panorama without performing any subsequent global optimization, there is a gradual error drift that causes visual artifacts in the resulting panorama. This limits the applicability to shorter image sequences and negatively impacts downstream PV module detection and anomaly detection, which Niccolai et al. perform directly on the panorama. Our method, on the other hand, performs module and anomaly detection directly in the images and encodes adjacent modules in a graph without relying on visual stitching. Hence, there is no error drift and no risk of introducing visual artifacts. Graph matching also has a lower computational complexity than panorama stitching leading to shorter processing times.

Our SfM-based method for PV module localization differs from the previous works that have used SfM in a few key aspects. The related works use SfM to obtain a georeferenced 3D point cloud [48] and optionally an orthophoto of the PV plant, on which they perform module and anomaly detection [47, 56, 70, 263, 289]. However, our method performs module and anomaly detection in each video frame and uses SfM solely for localizing modules in the plant. To this end, the 3D point cloud can be discarded and only the georeferenced 6D camera pose of each image is used to triangulate the corner points of detected PV modules, yielding their geocoordinates. This approach is beneficial because it can easily be combined with existing module and anomaly detection algorithms that operate on individual camera images, and, hence, does not require the development of special methods that operate on the orthophoto. Furthermore, our method provides better robustness and flexibility, as module and anomaly detection do not depend on the (sometimes unsuccessful) SfM reconstruction of the PV plant. Finally, by performing module and anomaly detection directly in the video frames instead of the orthophoto or 3D point cloud, there is no performance degradation due to visual artifacts, larger datasets can be extracted, and standard computer vision algorithms can be employed. Not having to create an orthophoto also reduces the computational burden of the method. We also fly at lower altitudes than common for orthophoto capture, resolving some of the associated issues, such as low spatial resolution and obstruction of nearby infrastructure.

The integration of external data sources, such as electrical data, is a very important topic that is often overlooked in the related works on PV module localization. In fact, apart from our works and those of Niccolai et al. [51] and López-Fernández et al. [48], no other work considers external data sources, let alone proposes a solution. We believe the community should consider this aspect more frequently, because it allows answering questions, such as: (i) How do different anomalies impact the power and yield of a PV plant? (ii) When does it make (economic) sense to replace an anomalous module? (iii) How do anomalies impact string performance and other modules in the same string? The topic will also become more important in the near future, when digital versions of the plant layout and electrical connectivity become more easily available, and possibly even standardized.

Another important aspect of PV module localization is the achieved module throughput, which is determined by the flight altitude, flight velocity, optical camera parameters, and the number of plant rows scanned in parallel. Here, both of our PV module localization methods achieve state-of-the-art results (see tab. 4.1). While Niccolai et al. take 30 min to 90 min for a $1\,\mathrm{MW}_p$ plant, our graph matching method takes only 21.1 min (assuming 235 W modules) for the same plant. Since our SfM-based method can process two or three

rows at a time, the time for the inspection of a $1\,\mathrm{MW}_p$ plant can be reduced even further to only $10.1\,\mathrm{min}$ and $5.6\,\mathrm{min}$, respectively. This is more than twice as fast as the method by López-Fernández et al., which takes $12.7\,\mathrm{min}$ per $\mathrm{MW}_p$.

# 6 Conclusion

In 2020 the global installed capacity of solar photovoltaics reached $775\,\mathrm{GW}_p$ [7] and is predicted to increase by another $1100\,\mathrm{GW}_p$ until 2026 [8]. Despite their robustness, most PV plants contain anomalous modules and components [14]. Anomalies lead to accelerated degradation [16, 17], can pose fire hazards [18–20], and reduce power output and yield of the plant [17], which in turn causes monetary losses and reduces profitability. Hence, regular inspection is recommended and in some countries, even a regulatory requirement [21]. However, utility-scale PV plants contain thousands to millions of PV modules, rendering manual inspection infeasible. Instead, IR thermography has been employed since over a decade as a fast, simple, and accurate method for the detection of PV system anomalies based on their characteristic thermal patterns [23, 24, 26, 27]. Typically, IR thermography is performed with low-cost consumer-market drones [25], which offer a $10-15$ fold speedup over conventional techniques, such as walking or ground-based robots [16].

However, drone-based thermography of utility-scale PV plants produces large amounts of IR videos, manual sighting of which is not economically feasible. Thus, at the beginning of this thesis, we set the goal of developing a solution for the automatic processing of the IR videos acquired during the inspection of utility-scale PV plants.

We achieved this goal by proposing a pipeline composed of computer vision algorithms for instance segmentation, multi-object tracking, incremental structure from motion, and supervised image classification. This pipeline detects PV modules in the IR videos, localizes each module in the plant, and identifies modules with thermal anomalies, effectively solving all major steps of automated PV plant inspection.

The developed pipeline is fully automated, facilitates a high throughput of up to 45000 modules ($\sim 10.6\,\mathrm{MW}_p$) per hour of flight, and is fully portable, i.e., requires no setup when inspecting a new PV plant. The pipeline features three different anomaly detection methods, which offer high accuracy, robustness to changes in environmental conditions, good generalization across different plants, and sensitivity to string anomalies and all common PV module anomalies. This is enabled by our very large training dataset with over 6.5 million IR images of 152669 PV modules from ten different PV plants. Furthermore, the pipeline facilitates intuitive visualization of anomaly detection results on a geographic map, and has low hardware requirements, requiring only a consumer-market drone with standard GPS and an uncooled microbolometer camera without any visual camera. The pipeline achieves not only a low mapping error of $\pm 0.22\,\mathrm{m}$ to $\pm 0.82\,\mathrm{m}$ thanks to the use of incremental SfM, but is also robust towards errors made during data acquisition and manages to successfully extract $99.3\,\%$ of all PV modules in our dataset. Videos can be recorded with both an automatic and a manual flight, reducing the complexity of drone operation and making our method useful for PV plants of all sizes. Furthermore,

the pipeline works independently of the plant layout, which allows for providing a single solution for multiple types of plants. This is not only cheaper than multiple plant-specific solutions, but it also simplifies software maintenance and usage. Finally, we publish our computer vision pipeline as free-of-charge and open-source software with a graphical user interface for intuitive visualization and exploration of inspection results.

Our contributions make the inspection of utility-scale PV plants economically feasible. Hence, regular inspections are enabled, which in turn will lead to better-maintained PV plants that are more reliable, safer, last longer, and provide higher power, yield, and returns on investment. These are vital aspects, which could decide the future success or failure of solar PV as a global source of clean and renewable electricity.

# 7 Outlook

The following chapter discusses the limitations of the developed PV plant inspection pipeline and outlines opportunities for future research.

**PV Module Detection**  While our approach for PV module detection based on Mask R-CNN (see sec. 4.2) is capable, there are still some options for improvements. For instance, Mask R-CNN could be replaced by a more recent and more accurate instance segmentation framework, such as Cascade Mask R-CNN [206]. Similarly, a newer backbone model, such as ConvNext [201] or SwinTransformer [297, 298], could be employed. The backbone may also be pretrained with a self-supervised contrastive loss to enhance accuracy even further [299].

Moreover, our training dataset comprises only images taken from utility-scale ground-mounted PV plants. While this leads to good accuracy on such plants, many false positives occur for other types of plants, such as rooftop plants, where the image contains additional obstructions, e.g., chimneys or windows. Extending the training dataset with images of other types of plants could resolve this issue.

Finally, our module detection method comprises not only Mask R-CNN for instance segmentation but two additional algorithms for multi-object tracking and estimation of the module corner points from the segmentation masks. It is imaginable to combine all three tasks in a single deep learning model that operates directly on the video and is trained end-to-end. While this is very challenging, it reduces the amount of code that needs to be maintained, and likely improves segmentation accuracy, as the video model can leverage the temporal information present in the video.

**PV Module Anomaly Detection**  We developed two anomaly detection methods based on CNNs (see sec. 4.3.1 and 4.3.2) and one based on the distribution of module temperatures (see sec. 4.3.3). While each of these methods overcomes some of the shortcomings of the other two methods, new limitations are introduced. For instance, our supervised ResNet-50 classifier from publication [1] classifies ten different anomaly patterns, however, does not take domain shift into account. Our method from publication [2] overcomes the domain shift problem and the temperature classifier is not affected by domain shift at all. However, both methods sacrify the capability to differentiate multiple anomaly patterns. Similarly, the CNN methods cannot detect string anomalies and are not very sensitive to open-circuited modules and anomalies with a small spatial extent. The temperature classifier overcomes these issues but is insensitive to anomalies with a small temperature gradient. Hence, a method, which simultaneously overcomes all these limitations, is yet to be developed.

Another issue of the CNN methods is the large class imbalance present in the training dataset, which could be resolved by undersampling majority classes or oversampling minority classes, e.g., with SMOTE [300]. Furthermore, the dataset has a large bias, as it was labeled by only a single human expert. To reduce the bias and increase the quality of the dataset, multiple experts should label the data and a consensus label should be obtained. Moreover, we only use small CNNs to speed up experimentation. Hence, the accuracy of the CNN methods could be easily improved by using larger models, e.g., ResNet-152 instead of ResNet-34, or by using state-of-the-art architectures, such as ConvNext [201].

In publication [2], we employed a supervised contrastive loss to learn informative features, which reduce the impact of domain shift. Future works could additionally align the domains and sub-domains with special domain adaption losses, such as Maximum Mean Discrepancy [301–303], L2- or cosine distance [304, 305], Rényi divergence [306], KL-divergence [307], or contrastive alignment losses [308–310]. Furthermore, we investigated only single domain adaption, where the CNN classifier is trained on a single source PV plant and tested on another target plant. However, it may be beneficial to perform multi-domain adaption, where the classifier is trained on multiple source plants.

Another important limitation of our CNN-based methods is that they operate on images of individual PV modules, ignoring the contexts of neighboring modules and electrical connectivity. However, these contexts are important to accurately detect string anomalies and other anomalies, which spread over multiple adjacent modules, such as potential induced degradation. Future works could employ multi-stream CNNs [311, 312] or Graph-CNNs [313] to perform prediction based on multiple images, possibly considering their relations and additional data modalities, such as electrical measurements.

While we framed anomaly detection as supervised image classification, future works could explore numerous other methods for this task [314–316]. Most of these methods learn features of normal data using generative adversarial networks [317, 318], autoencoders [319, 320], one-class losses [321, 322], self-supervised learning [323–326], or metric learning [327, 328] and detect anomalies based on their high reconstruction error or large distance to the normal feature. There are also methods that both detect and segment anomalies in the image, for example, PaDiM [329], Patch Core [330], and Fast Flow [331]. This could be useful for cases, in which a PV module exhibits multiple anomalies simultaneously.

Other research opportunities for anomaly detection are (i) the extension to thin-film, bifacial, and half-cell modules, (ii) the quantification of the impact of changes in external factors, such as solar irradiance, cloud cover, air temperature, wind speed, camera temperature, and plant load, and (iii) the exploration of the temporal evolution of anomalies. These research directions require an extension of our anomaly detection dataset with appropriate data, e.g., a time series of IR images from repeated inspections of the same PV plant.

Future works could also assemble an even larger anomaly detection dataset using our methods, make it publicly available, or even host a public competition for this task, to

encourage community contributions and facilitate quantitative comparison of published methods.

Finally, future works could investigate the detection of non-thermal anomalies, such as geometric misalignment, which are not detectable in IR but have a significant impact on power and yield. Such anomalies could be detected by the geometric analysis of a 3D reconstruction of the plant obtained by SfM.

**PV Module Localization**  Our SfM-based method for PV module localization (see sec. 4.4.2) solves most problems of the graph matching-based method (see sec. 4.4.1). However, it still has several limitations that need to be addressed. The biggest issue of the SfM method is that the 3D reconstruction sometimes fails, preventing module localization altogether. Further experiments are needed to identify a flight configuration and hyperparameter settings of the SfM library that lead to an increased robustness. Other possible measures are (i) the use of centimeter-accurate RTK-GPS instead of standard GPS, (ii) modification of the image preprocessing to enhance keypoint detection and matching in IR videos, e.g., by performing local as opposed to global contrast enhancement, and (iii) utilization of the visual video for SfM, which has a higher resolution, wider viewing angle, color information, and lower variation of image intensities [332]. However, incorporating the visual video requires accurate temporal synchronization and spatial registration with the IR video, which is a challenging task.

Moreover, additional experimentation is needed to validate the applicability of the SfM-based module localization to PV plants with non-row layouts, such as rooftop and floating plants.

Another issue of the SfM method is that it requires manual entry of the electrical connectivity of the localized PV modules. Assuming standardized CAD files will become more ubiquitous in the future, this step could be automated by combining the SfM method and graph matching method to automatically associate the localized modules with a plant description file. This would also facilitate the alignment of multiple SfM reconstructions of the same PV plant, for example, obtained by repeated inspections.

**Reporting**  Currently, reporting of our inspection pipeline is limited to an interactive 2D geographic map showing each PV module in the plant overlaid with the results of one of the developed anomaly detection methods. In addition to this, it would be highly beneficial to rate the severity of the detected anomalies in terms of safety risk and impact on power, yield, and economic variables, such as annual monetary losses, and cost of repair. Based on this, the report should provide a clear statement on which PV modules are to be repaired and which anomalies can be safely ignored. Furthermore, the reporting should account for repeated inspections of the same PV plant, i.e., store and utilize historical data to make predictions. Another useful feature of the reporting software would be to localize and guide the maintenance crew to those modules in the PV plant that must be repaired. Without this feature, localizing a particular module in the field requires error-prone counting. Finally, instead of visualizing results on a map, the SfM reconstruction of the plant could be rendered into an immersive 3D model of the plant

with overlaid textures, module images, and interactive reports for each module, enabling virtual walkabouts and exploration with a virtual reality headset.

**Others** Apart from the mentioned aspects, future works could transfer our inspection pipeline to electro- and photoluminescence imaging, or even use these imaging modalities in conjunction with IR thermography. Another important and often overlooked aspect is the scalability of the inspection solution to multi-gigawatt plants. While our inspection pipeline is fully scalable in theory, the concrete implementation needs to be improved in terms of vertical and horizontal scalability to effectively run on a distributed computing cluster.

# References

[4] "World energy outlook," International Energy Agency, Paris, Tech. Rep., 2021.

[5] "Lazard's levelized cost of energy analysis – Version 14.0," Lazard, New York, NY, USA, Tech. Rep., 2020.

[6] "The power to change: Solar and wind cost reduction potential to 2025," International Renewable Energy Agency, Abu Dhabi, Tech. Rep., 2016.

[7] "Renewables 2020," International Energy Agency, Paris, Tech. Rep., 2020.

[8] "Renewables 2021," International Energy Agency, Paris, Tech. Rep., 2021.

[9] A. Bett *et al.*, "Photovoltaics report," Fraunhofer Institute for Solar Energy Systems, ISE, Tech. Rep., 2022.

[10] Q. Navid, A. Hassan, A. A. Fardoun, R. Ramzan, and A. Alraeesi, "Fault diagnostic methodologies for utility-scale photovoltaic power plants: A state of the art review," *Sustainability*, vol. 13, no. 4, pp. 1629.1–1629.22, 2021. DOI: 10.3390/su13041629.

[11] J. Kim, M. Rabelo, S. P. Padi, H. Yousuf, E.-C. Cho, and J. Yi, "A review of the degradation of photovoltaic modules for life expectancy," *Energies*, vol. 14, no. 14, p. 4278, 2021. DOI: 10.3390/en14144278.

[12] J. A. Tsanakas and P. N. Botsaris, "On the detection of hot spots in operating photovoltaic arrays through thermal image analysis and a simulation model," *Materials evaluation*, vol. 71, no. 4, pp. 457–465, 2013.

[13] D. C. Jordan, S. R. Kurtz, K. VanSant, and J. Newmiller, "Compendium of photovoltaic degradation rates," *Progress in Photovoltaics: Research and Applications*, vol. 24, no. 7, pp. 978–989, 2016. DOI: 10.1002/pip.2744.

[14] C. Buerhop, T. Pickel, H. Scheuerpflug, C. Camus, J. Hauch, and C. J. Brabec, "Statistical overview of findings by IR-inspections of PV-plants," in *Reliability of Photovoltaic Cells, Modules, Components, and Systems IX*, vol. 9938, SPIE, 2016, pp. 100–108. DOI: 10.1117/12.2237821.

[15] C. Buerhop, R. Weißmann, H. Scheuerpflug, R. Auer, and C. Brabec, "Quality control of PV-modules in the field using a remote-controlled drone with an infrared camera," in *European Photovoltaic Solar Energy Conference and Exhibition (EUPVSEC)*, Frankfurt, Germany, 2012, pp. 3370–3373. DOI: 10.4229/27thEUPVSEC2012-4BV.2.43.

[16] S. A. Rahaman, T. Urmee, and D. A. Parlevliet, "PV system defects identification using remotely piloted aircraft (RPA) based infrared (IR) imaging: A review," *Solar Energy*, vol. 206, pp. 579–595, 2020. DOI: 10.1016/j.solener.2020.06.014.

## References

[17] J. A. Tsanakas, L. Ha, and C. Buerhop, "Faults and infrared thermographic diagnosis in operating c-Si photovoltaic modules: A review of research and future challenges," *Renewable and Sustainable Energy Reviews*, vol. 62, pp. 695–709, 2016. DOI: 10.1016/j.rser.2016.04.079.

[18] W. Vaassen *et al.*, "PV fire hazard: Analysis and assessment of fire incidents," in *European Photovoltaic Solar Energy Conference and Exhibition (EUPVSEC)*, Villepinte, France, 2013, pp. 4304–4311. DOI: 10.4229/28thEUPVSEC2013-5BV.7.71.

[19] L. Fiorentini, L. Marmo, E. Danzi, V. Rossini, and V. Puccia, "Fire risk assessment of photovoltaic plants," *Chemical Engineering Transactions*, vol. 48, no. 10, pp. 427–432, 2016. DOI: DOI:10.3303/CET1648072.

[20] H. Wirth, "Recent facts about photovoltaics in germany," Fraunhofer Institute for Solar Energy Systems, ISE, Tech. Rep., 2021.

[21] "Photovoltaic (PV) systems – Requirements for testing, documentation and maintenance – Part 2: Grid connected systems – Maintenance of PV systems," International Organization for Standardization, Geneva, Standard ISO/IEC TS 62446-2:2020, 2021.

[22] EnBW Energie Baden-Württemberg AG, *EnBW Weesow-Willmersdorf solar park*, https://www.enbw.com/renewable-energy/solar/solarpark_weesow/, Accessed: 05/03/2022, 2021.

[23] B. Weinreich, B. Schauer, S. Gürzing, and R. Hass, "Feldstudie 2.0 zur Modul- und Anlagenqualität auf Basis thermographischer Messungen von 1 GW," HaWe Engineering GmbH, Gauting-Hausen, Tech. Rep., 2019.

[24] G. Schirripa Spagnolo, P. Del Vecchio, G. Makary, D. Papalillo, and A. Martocchia, "A review of IR thermography applied to PV systems," in *International Conference on Environment and Electrical Engineering*, Venice, Italy, 2012, pp. 879–884. DOI: 10.1109/EEEIC.2012.6221500.

[25] S. Gallardo-Saavedra, L. Hernández-Callejo, and O. Duque-Perez, "Technological review of the instrumentation used in aerial thermographic inspection of photovoltaic plants," *Renewable and Sustainable Energy Reviews*, vol. 93, pp. 566–579, 2018. DOI: 10.1016/j.rser.2018.05.027.

[26] B. Du, Y. He, Y. He, and C. Zhang, "Progress and trends in fault diagnosis for renewable and sustainable energy system based on infrared thermography: A review," *Infrared Physics & Technology*, vol. 109, p. 103 383, 2020. DOI: 10.1016/j.infrared.2020.103383.

[27] A. Kandeal, M. Elkadeem, A. Kumar Thakur, G. B. Abdelaziz, R. Sathyamurthy, A. Kabeel, N. Yang, and S. W. Sharshir, "Infrared thermography-based condition monitoring of solar photovoltaic systems: A mini review of recent advances," *Solar Energy*, vol. 223, pp. 33–43, 2021. DOI: 10.1016/j.solener.2021.05.032.

[28] C. Buerhop, U. Jahn, U. Hoyer, B. Lerche, and S. Wittmann, "Abschlussbericht der Machbarkeitsstudie zur Überprüfung der Qualität von Photovoltaik-Modulen mittels Infrarot-Aufnahmen," ZAE Bayern e.V., Tech. Rep., 2007.

[29] C. Buerhop, D. Schlegel, M. Niess, C. Vodermayer, R. Weißmann, and C. Brabec, "Reliability of IR-imaging of PV-plants under operating conditions," *Solar Energy Materials and Solar Cells*, vol. 107, pp. 154–164, 2012. DOI: `10.1016/j.solmat.2012.07.011`.

[30] "Photovoltaic (PV) systems – Requirements for testing, documentation and maintenance – Part 3: Photovoltaic modules and plants – Outdoor infrared thermography," International Organization for Standardization, Geneva, Standard ISO/IEC TS 62446-3:2017, 2017.

[31] "Review of failures of photovoltaic modules," International Energy Agency, Paris, Tech. Rep. IEA-PVPS T13-01:2014, 2014.

[32] A. Dolara, G. C. Lazaroiu, and E. Ogliari, "Efficiency analysis of PV power plants shaded by MV overhead lines," *International Journal of Energy and Environmental Engineering*, vol. 7, no. 4, pp. 115–123, 2016. DOI: `10.1007/s40095-016-0208-2`.

[33] Z. Dawood, A. H. Numan, and H. Hussein, "Hot spot analysis of photovoltaic module under partial shading conditions by using IR-imaging technology," *Journal of Power and Energy Engineering*, vol. 39, no. 9, 2020. DOI: `10.30684/etj.v39i9.841`.

[34] I. Lillo-Bravo, P. González-Martínez, M. Larrañeta, and J. Guasumba-Codena, "Impact of energy losses due to failures on photovoltaic plant energy balance," *Energies*, vol. 11, no. 2, p. 363, 2018. DOI: `10.3390/en11020363`.

[35] "Review on infrared and electroluminescence imaging for PV field applications," International Energy Agency, Paris, Tech. Rep. Report IEA-PVPS T13-10:2018, 2018.

[36] H. Scheuerpflug and C. Buerhop-Lutz, "Field inspection of PV-modules using aerial, drone-mounted thermography," in *European Photovoltaic Solar Energy Conference and Exhibition (EUPVSEC)*, Amsterdam, Netherlands, 2014, pp. 2975–2979. DOI: `10.4229/EUPVSEC20142014-5BV.2.53`.

[37] P. B. Quater, F. Grimaccia, S. Leva, M. Mussetta, and M. Aghaei, "Light unmanned aerial vehicles (UAVs) for cooperative inspection of PV plants," *IEEE Journal of Photovoltaics*, vol. 4, no. 4, pp. 1107–1113, 2014. DOI: `10.1109/JPHOTOV.2014.2323714`.

[38] A. Niccolai, A. Gandelli, F. Grimaccia, R. Zich, and S. Leva, "Overview on photovoltaic inspections procedure by means of unmanned aerial vehicles," in *IEEE Milan PowerTech*, Milan, Italy, 2019, pp. 1–6. DOI: `10.1109/PTC.2019.8810987`.

[39] N. M. Kumar, K. Sudhakar, M. Samykano, and V. Jayaseelan, "On the technologies empowering drones for intelligent monitoring of solar photovoltaic power plants," in *International Conference on Robotics and Smart Manufacturing (ICROSMA)*, Chennai, India, 2018, pp. 585–593. DOI: `10.1016/j.procs.2018.07.087`.

[40] F. Bizzarri, S. Nitti, and G. Malgaroli, "The use of drones in the maintenance of photovoltaic fields," *E3S Web of Conferences*, vol. 119, 2019. DOI: `10.1051/e3sconf/201911900021`.

[41] C. Henry, S. Poudel, S.-W. Lee, and H. Jeong, "Automatic detection system of deteriorated PV modules using drone with thermal camera," *Applied Sciences*, vol. 10, p. 3802, 2020. DOI: `10.3390/app10113802`.

[42] A. Moradi Sizkouhi, S. Esmailifar, M. Aghaei, and M. Karimkhani, "RoboPV: An integrated software package for autonomous aerial monitoring of large scale PV plants," *Energy Conversion and Management*, vol. 254, p. 115 217, 2022. DOI: 10.1016/j.enconman.2022.115217.

[43] U. Vision, *Drones cut cost of thermographic PV panel inspections*, https://www.uasvision.com/2016/09/15/drones-cut-cost-of-thermographic-pv-panel-inspections/, Accessed: 05/03/2022, 2016.

[44] T. Rakha and A. Gorodetsky, "Review of unmanned aerial system (UAS) applications in the built environment: Towards automated building inspection procedures using drones," *Automation in Construction*, vol. 93, pp. 252–264, 2018. DOI: 10.1016/j.autcon.2018.05.002.

[45] S. Leva, M. Aghaei, and F. Grimaccia, "PV power plant inspection by UAS: Correlation between altitude and detection of defects on PV modules," in *IEEE International Conference on Environment and Electrical Engineering (EEEIC)*, Rome, Italy, 2015, pp. 1921–1926. DOI: 10.1109/EEEIC.2015.7165466.

[46] V. Carletti, A. Greco, A. Saggese, and M. Vento, "An intelligent flying system for automatic detection of faults in photovoltaic plants," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, pp. 2027–2040, 2019. DOI: 10.1007/s12652-019-01212-6.

[47] Y. Zefri, I. Sebari, H. Hajji, and G. Aniba, "Developing a deep learning-based layer-3 solution for thermal infrared large-scale photovoltaic module inspection from orthorectified big UAV imagery data," *International Journal of Applied Earth Observation and Geoinformation*, vol. 106, no. 5, p. 102 652, 2022. DOI: 10.1016/j.jag.2021.102652.

[48] L. López Fernández, S. Lagüela, J. Fernandez-Hernandez, and D. González-Aguilera, "Automatic evaluation of photovoltaic power stations from high-density RGB-T 3D point clouds," *Remote Sensing*, vol. 9, no. 6, p. 631, 2017. DOI: 10.3390/rs9060631.

[49] F. Grimaccia, S. Leva, and A. Niccolai, "PV plant digital mapping for modules' defects detection by unmanned aerial vehicles," *IET Renewable Power Generation*, vol. 11, no. 10, pp. 1221–1228, 2017. DOI: 10.1049/iet-rpg.2016.1041.

[50] G. Francesco, L. Sonia, and N. Alessandro, "A semi-automated method for defect identification in large photovoltaic power plants using unmanned aerial vehicles," in *IEEE Power Energy Society General Meeting (PESGM)*, Portland, OR, USA, 2018, pp. 1–5. DOI: 10.1109/PESGM.2018.8586506.

[51] A. Niccolai, F. Grimaccia, and S. Leva, "Advanced asset management tools in photovoltaic plant monitoring: UAV-based digital mapping," *Energies*, vol. 12, no. 24, p. 4736, 2019. DOI: 10.3390/en12244736.

[52] A. Arenella, A. Greco, A. Saggese, and M. Vento, "Real time fault detection in photovoltaic cells by cameras on drones," in *International Conference Image Analysis and Recognition (ICIAR)*, Montréal, Canada, 2017, pp. 617–625. DOI: 10.1007/978-3-319-59876-5_68.

## References

[53] H. Jeong, G.-R. Kwon, and S.-W. Lee, "Deterioration diagnosis of solar module using thermal and visible image processing," *Energies*, vol. 13, no. 11, p. 2856, 2020. DOI: 10.3390/en13112856.

[54] P. Addabbo, A. Angrisano, M. L. Bernardi, G. Gagliarde, A. Mennella, M. Nisi, and S. L. Ullo, "UAV system for photovoltaic plant inspection," *IEEE Aerospace and Electronic Systems Magazine*, vol. 33, no. 8, pp. 58–67, 2018. DOI: 10.1109/MAES.2018.170145.

[55] S. Dotenco, M. Dalsass, L. Winkler, T. Würzner, C. Brabec, A. Maier, and F. Gallwitz, "Automatic detection and analysis of photovoltaic modules in aerial infrared imagery," in *IEEE Winter Conference on Applications of Computer Vision (WACV)*, Lake Placid, NY, USA, 2016, pp. 1–9. DOI: 10.1109/WACV.2016.7477658.

[56] I. Tsanakas, L. Ha, and F. Al Shakarchi, "Advanced inspection of photovoltaic installations by aerial triangulation and terrestrial georeferencing of thermal/visual imagery," *Renewable Energy*, vol. 102 (Part A), pp. 224–233, 2016. DOI: 10.1016/j.renene.2016.10.046.

[57] *Above Surveying Ltd.* https://www.abovesurveying.com/, Accessed: 05/03/2022, Colchester, United Kingdom, 2022.

[58] *Raptor Maps*, https://raptormaps.com/, Accessed: 05/03/2022, Somerville, Massachusetts, 2022.

[59] *Aerial PV Inspection GmbH*, http://www.aepvi.com/, Accessed: 05/03/2022, Aachen, Germany, 2022.

[60] "Eia-860 annual electric generator report," U.S. Energy Information Administration (EIA), Tech. Rep., 2021.

[61] A. Awasthi, A. K. Shukla, M. M. S.R., C. Dondariya, K. Shukla, D. Porwal, and G. Richhariya, "Review on sun tracking technology in solar PV system," *Energy Reports*, vol. 6, pp. 392–405, 2020. DOI: 10.1016/j.egyr.2020.02.004.

[62] "Photovoltaic plants: Cutting edge technology. From sun to socket," ABB Limited, Tech. Rep., 2019.

[63] *Duomax dual glass 60-cell module*, TSM-PEG5, Rev. TSM_EN_2017_A, Trina Solar Limited, 2017.

[64] *Tallmax framed 72-cell module*, TSM-PD14, Rev. TSM_EN_2018_B, Trina Solar Limited, 2017.

[65] K. Jäger, O. Isabella, A. H. Smets, R. A. van Swaaij, and M. Zeman, *Solar Energy: Fundamentals, Technology, and Systems*. Delft University of Technology, 2014.

[66] F. Lu, S. Guo, T. M. Walsh, and A. G. Aberle, "Improved PV module performance under partial shading conditions," *Energy Procedia*, vol. 33, pp. 248–255, 2013. DOI: 10.1016/j.egypro.2013.05.065.

[67] L. Fialho, R. Melicio, V. Mendes, J. Figueiredo, and M. Collares-Pereira, "Effect of shading on series solar modules: Simulation and experimental results," *Procedia Technology*, vol. 17, pp. 295–302, 2014. DOI: 10.1016/j.protcy.2014.10.240.

## References

[68]  A. Abramovitz and D. Shmilovitz, "Short survey of architectures of photovoltaic arrays for solar power generation systems," *Energies*, vol. 14, no. 16, p. 4917, 2021. DOI: 10.3390/en14164917.

[69]  J. P. Ram, T. S. Babu, and N. Rajasekar, "A comprehensive review on solar PV maximum power point tracking techniques," *Renewable and Sustainable Energy Reviews*, vol. 67, pp. 826–847, 2017. DOI: 10.1016/j.rser.2016.09.076.

[70]  Y. Zefri, A. ElKettani, I. Sebari, and S. A. Lamallam, "Thermal infrared and visual inspection of photovoltaic installations by UAV photogrammetry—Application case: Morocco," *Drones*, vol. 2, no. 4, p. 41, 2018. DOI: 10.3390/drones2040041.

[71]  "Assessment of photovoltaic module failures in the field," International Energy Agency, Paris, Tech. Rep. Report IEA-PVPS T13-09:2017, 2017.

[72]  WINAICO, *11 common solar panel defects and how to avoid them*, https://winaico.com/blog/common-solar-panel-defects/, Accessed: 26/05/2022, 2021.

[73]  Infinity Energy, *Why quality matters*, https://www.infiniteenergy.com.au/products/why-quality-matters/, Accessed: 26/05/2022, 2022.

[74]  H. Schwarzburger, *Polymeranalyse für eva-folie*, https://www.photovoltaik.eu/solarmodule/polymeranalyse-fuer-eva-folie/, Accessed: 26/05/2022, 2016.

[75]  Clean Energy Revies, *Solar panel problems and degradation explained*, https://www.cleanenergyreviews.info/solar-panel-failure-degradation/, Accessed: 26/05/2022, 2022.

[76]  M. Shaibani, *Solar panel recycling: Turning tickng time bombs into opportunities*, https://www.greenbuildingafrica.co.za/solar-panel-recycling-turning-ticking-time-bombs-into-opportunities/, Accessed: 26/05/2022, 2020.

[77]  S. Yunlin, S. Chen, L. Xie, R. Hong, and H. Shen, "Investigating the impact of shading effect on the characteristics of a large-scale grid-connected PV power plant in northwest China," *International Journal of Photoenergy*, vol. 2014, no. 3, p. 763 106, 2014. DOI: 10.1155/2014/763106.

[78]  "Designing new materials for photovoltaics: Opportunities for lowering cost and increasing performance through advanced material innovations," International Energy Agency, Paris, Tech. Rep. Report IEA-PVPS T13-13:2021, 2021.

[79]  W. Herrmann, W. Wiesner, and W. Vaassen, "Hot spot investigations on PV modules: New concepts for a test standard and consequences for module design with respect to bypass diodes," in *IEEE Photovoltaic Specialists Conference (PVSC)*, Anaheim, CA, USA, 1997, pp. 1129–1132. DOI: 10.1109/PVSC.1997.654287.

[80]  C. Dunderdale, W. Brettenny, C. Clohessy, and E. E. van Dyk, "Photovoltaic defect classification through thermal infrared imaging using a machine learning approach," *Progress in Photovoltaics: Research and Applications*, vol. 28, no. 3, pp. 177–188, 2020. DOI: 10.1002/pip.3191.

[81]  M. Aghaei, U. Madukanya, A. Oliveira, and R. Rüther, "Fault inspection by aerial infrared thermography in a PV plant after a meteorological tsunami," in *Congresso Brasileiro de Energia Solar (CBENS)*, Gramado, Brazil, 2018.

# References

[82] M. Planck and M. Masius, *The Theory of Heat Radiation*, 2nd. Philadelphia, PA, USA: P. Blakiston's Son & Co., 1914, Translated by Masius, M.

[83] L. Boltzmann, "Ableitung des stefan'schen gesetzes, betreffend die abhängigkeit der wärmestrahlung von der temperatur aus der electromagnetischen lichttheorie," *Annalen der Physik*, vol. 258, no. 6, pp. 291–294, 1884. DOI: 10.1002/andp.18842580616.

[84] S. Bagavathiappan, B. Lahiri, T. Saravanan, J. Philip, and T. Jayakumar, "Infrared thermography for condition monitoring: A review," *Infrared Physics & Technology*, vol. 60, pp. 35–55, 2013. DOI: 10.1016/j.infrared.2013.03.006.

[85] R. International, *Exportation conditions*, https://www.flircameras.com/export_conditions, Accessed: 14/03/2022.

[86] F. Grimaccia, M. Aghaei, M. Mussetta, S. Leva, and P. B. Quater, "Planning for PV plant performance monitoring by means of unmanned aerial systems (UAS)," *International Journal of Energy and Environmental Engineering*, vol. 6, no. 1, pp. 47–54, 2015. DOI: 10.1007/s40095-014-0149-6.

[87] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Boston, MA, USA: Addison-Wesley Publishing Company, 1992, ISBN: 9780201180756.

[88] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979. DOI: 10.1109/TSMC.1979.4310076.

[89] I. Sobel, "An isotropic 3x3 image gradient operator," *Presentation at Stanford A.I. Project 1968*, 2014.

[90] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679–698, 1986. DOI: 10.1109/TPAMI.1986.4767851.

[91] R. O. Duda and P. E. Hart, "Use of the Hough transformation to detect lines and curves in pictures," *Communications of the ACM*, vol. 15, no. 1, pp. 11–15, 1972. DOI: 10.1145/361237.361242.

[92] T. Mitchell, *Machine Learning*. New York, NY, USA: McGraw-Hill Education, 1997, ISBN: 9780070428072.

[93] K. A. K. Niazi, W. Akhtar, H. A. Khan, Y. Yang, and S. Athar, "Hotspot diagnosis for solar photovoltaic modules using a naive Bayes classifier," *Solar Energy*, vol. 190, pp. 34–43, 2019. DOI: 10.1016/j.solener.2019.07.063.

[94] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of Machine Learning*, 2nd. Cambridge, MA, USA: The MIT Press, 2018, ISBN: 9780262039406.

[95] G. V. Trunk, "A problem of dimensionality: A simple example," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-1, no. 3, pp. 306–307, 1979. DOI: 10.1109/TPAMI.1979.4766926.

[96] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*, 4th. Boston, MA, USA: Academic Press, 2009, ISBN: 9781597492720.

## References

[97]  W. K. Mutlag, S. K. Ali, Z. M. Aydam, and B. H. Taher, "Feature extraction methods: A review," *Journal of Physics: Conference Series*, vol. 1591, no. 1, p. 012028, 2020. DOI: 10.1088/1742-6596/1591/1/012028.

[98]  I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016, ISBN: 9780262035613.

[99]  C. Fefferman, S. Mitter, and H. Narayanan, "Testing the manifold hypothesis," *Journal of the American Mathematical Society*, vol. 29, pp. 983–1049, 2016. DOI: 10.1090/jams/852.

[100]  C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995. DOI: 10.1023/A:1022627411411.

[101]  L. Breiman, J. Friedman, R. Olshen, and C. Stone, *Classification And Regression Trees*, 1st. New York, NY, USA: Routledge, 1984, ISBN: 9781315139470.

[102]  N. S. Altman, "An introduction to kernel and nearest-neighbor nonparametric regression," *The American Statistician*, vol. 46, no. 3, pp. 175–185, 1992. DOI: 10.1080/00031305.1992.10475879.

[103]  C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA, USA: The MIT Press, 2005, ISBN: 9780262182539.

[104]  A. I. Awad and M. Hassaballah, Eds., *Image Feature Detectors and Descriptors*, 1st. Cham, Switzerland: Springer International Publishing, 2016, ISBN: 9783319288529.

[105]  D. Lowe, "Object recognition from local scale-invariant features," in *IEEE International Conference on Computer Vision (ICCV)*, Kerkyra, Greece, 1999, pp. 1150–1157. DOI: 10.1109/ICCV.1999.790410.

[106]  H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008. DOI: 10.1016/j.cviu.2007.09.014.

[107]  E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *IEEE International Conference on Computer Vision (ICCV)*, Barcelona, Spain, 2011, pp. 2564–2571. DOI: 10.1109/ICCV.2011.6126544.

[108]  S. Leutenegger, M. Chli, and R. Y. Siegwart, "BRISK: Binary robust invariant scalable keypoints," in *IEEE International Conference on Computer Vision (ICCV)*, Barcelona, Spain, 2011, pp. 2548–2555. DOI: 10.1109/ICCV.2011.6126542.

[109]  E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," in *European Conference on Computer Vision (ECCV)*, Graz, Austria, 2006, pp. 430–443. DOI: 10.1007/11744023_34.

[110]  K. Mikolajczyk and C. Schmid, "Scale & affine invariant interest point detectors," *International Journal of Computer Vision*, vol. 60, no. 10, pp. 63–86, 2004. DOI: 10.1023/B:VISI.0000027790.02288.f2.

[111]  Z. Wang, B. Fan, and F. Wu, "Local intensity order pattern for feature description," in *IEEE International Conference on Computer Vision (ICCV)*, Barcelona, Spain, 2011, pp. 603–610. DOI: 10.1109/ICCV.2011.6126294.

# References

[112] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "BRIEF: Binary robust independent elementary features," in *European Conference on Computer Vision (ECCV)*, Hersonissos, Greece, 2010, pp. 778–792. DOI: 10.1007/978-3-642-15561-1_56.

[113] A. Alahi, R. Ortiz, and P. Vandergheynst, "FREAK: Fast retina keypoint," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Providence, RI, USA, 2012, pp. 510–517. DOI: 10.1109/CVPR.2012.6247715.

[114] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, San Diego, CA, USA, 2005, pp. 886–893. DOI: 10.1109/CVPR.2005.177.

[115] J. Johansson, "Interest point detectors and descriptors for IR images," M.S. thesis, KTH Royal Institute of Technology, Stockholm, Sweden, 2015.

[116] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *IEEE International Conference on Computer Vision (ICCV)*, Nice, France, 2003, pp. 1470–1477. DOI: 10.1109/ICCV.2003.1238663.

[117] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 5, pp. 436–44, 2015. DOI: 10.1038/nature14539.

[118] L. Bottou, "Online algorithms and stochastic approximations," in *Online Learning and Neural Networks*, Cambridge, UK: Cambridge University Press, 1998, ISBN: 9780521652636.

[119] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagation errors," 6088, vol. 323, 1986, pp. 533–536. DOI: 10.1038/323533a0.

[120] Y. Yu, X. Si, C. Hu, and J. Zhang, "A review of recurrent neural networks: LSTM cells and network architectures," *Neural Computation*, vol. 31, no. 7, pp. 1235–1270, 2019. DOI: 10.1162/neco_a_01199.

[121] M.-H. Guo, T.-X. Xu, J.-J. Liu, Z.-N. Liu, P.-T. Jiang, T.-J. Mu, S.-H. Zhang, R. R. Martin, M.-M. Cheng, and S.-M. Hu, "Attention mechanisms in computer vision: A survey," *arXiv preprint*, 2021. arXiv: 2111.07624.

[122] A. Jabbar, X. Li, and B. Omar, "A survey on generative adversarial networks: Variants, applications, and training," *ACM Computing Surveys*, vol. 54, no. 8, pp. 157.1–157.49, 2021. DOI: 10.1145/3463475.

[123] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, "A survey of convolutional neural networks: Analysis, applications, and prospects," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–21, 2021. DOI: 10.1109/TNNLS.2021.3084827.

[124] D. Asimov, "The grand tour: A tool for viewing multidimensional data," *SIAM Journal on Scientific and Statistical Computing*, vol. 6, no. 1, pp. 128–143, 1985. DOI: 10.1137/0906011.

[125] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *IEEE International Conference on Computer Vision (ICCV)*, Venice, Italy, 2017, pp. 618–626. DOI: 10.1109/ICCV.2017.74.

# References

[126] A. Chattopadhay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, "Grad-CAM++: Generalized gradient-based visual explanations for deep convolutional networks," in *IEEE Winter Conference on Applications of Computer Vision (WACV)*, Lake Tahoe, NV, USA, 2018, pp. 839–847. DOI: 10.1109/WACV.2018.00097.

[127] D.-X. Zhou, "Universality of deep convolutional neural networks," *Applied and Computational Harmonic Analysis*, vol. 48, no. 6, 2019. DOI: 10.1016/j.acha.2019.06.004.

[128] M. Leshno, V. Y. Lin, A. Pinkus, and S. Schocken, "Multilayer feedforward networks with a nonpolynomial activation function can approximate any function," *Neural Networks*, vol. 6, no. 6, pp. 861–867, 1993. DOI: 10.1016/S0893-6080(05)80131-5.

[129] A. Pinkus, "Approximation theory of the MLP model in neural networks," *Acta Numerica*, vol. 8, pp. 143–195, 1999. DOI: 10.1017/S0962492900002919.

[130] D. A. Forsyth and J. Ponce, *Computer Vision - A Modern Approach*, 2nd. Hoboken, NJ, USA: Pearson Education Inc., 2012, ISBN: 9780136085928.

[131] G. Ciaparrone, F. Luque Sánchez, S. Tabik, L. Troiano, R. Tagliaferri, and F. Herrera, "Deep learning in video multi-object tracking: A survey," *Neurocomputing*, vol. 381, pp. 61–88, 2020. DOI: 10.1016/j.neucom.2019.11.023.

[132] Y. Park, L. M. Dang, S. Lee, D. Han, and H. Moon, "Multiple object tracking in deep learning approaches: A survey," *Electronics*, vol. 10, no. 19, p. 2406, 2021. DOI: 10.3390/electronics10192406.

[133] M. Fiaz, A. Mahmood, and S. K. Jung, "Tracking noisy targets: A review of recent object tracking approaches," *arXiv preprint*, 2018. arXiv: 1802.03098.

[134] W. Luo, J. Xing, A. Milan, X. Zhang, W. Liu, X. Zhao, and T.-K. Kim, "Multiple object tracking: A literature review," *Artificial Intelligence*, vol. 293, p. 103 448, 2014. DOI: 10.1016/j.artint.2020.103448.

[135] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955. DOI: 10.1002/nav.3800020109.

[136] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd. Cambridge, UK: Cambridge University Press, 2004, ISBN: 9780511811685.

[137] D. C. Brown, "Decentering distortion of lenses," *Photogrammetric Engineering*, vol. 32, no. 3, pp. 444–462, 1966.

[138] J. Bloomenthal and J. Rokne, "Homogeneous coordinates," *The Visual Computer*, vol. 11, pp. 15–26, 1994. DOI: 10.1007/BF01900696.

[139] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000. DOI: 10.1109/34.888718.

[140] G. Bradski, "The OpenCV library," *Dr. Dobb's Journal of Software Tools*, vol. 120, pp. 122–125, 2000.

[141] D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *Journal of the Society for Industrial and Applied Mathematics*, vol. 11, no. 2, pp. 431–441, 1963. DOI: 10.1137/0111030.

# References

[142] S. S. Ruixuan Liu Hengrui Zhang, *Multiple methods of geometric calibration of thermal camera and a method of extracting thermal calibration feature points*, https://henryzh47.github.io/Thermal-Camera-Calibration/, Accessed: 11/03/2022.

[143] R. Usamentiaga, C. Ibarra-Castanedo, and X. P. V. Maldague, "Comparison and evaluation of geometric calibration methods for infrared cameras to perform metric measurements on a plane," *Applied optics*, vol. 57, no. 18, pp. D1–D10, 2018. DOI: 10.1364/AO.57.0000D1.

[144] T. Herrmann, C. Migniot, and O. Aubreton, "Thermal camera calibration with cooled down chessboard," in *Quantitative Infrared Thermography Conference (QIRT)*, Tokyo, Japan, 2019. DOI: 10.21611/qirt.2020.010.

[145] T. Luhmann, J. Ohm, J. Piechel, and T. Roelf, "Geometric calibration of thermographic cameras," in *ISPRS International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Newcastle upon Tyne, UK, 2010. DOI: 10.1007/978-94-007-6639-6_2.

[146] R. Yang, W. Yang, Y. Chen, and X. Wu, "Geometric calibration of IR camera using trinocular vision," *Journal of Lightwave Technology*, vol. 29, no. 24, pp. 3797–3803, 2011. DOI: 10.1109/JLT.2011.2170812.

[147] S. Vidas, R. Lakemond, S. Denman, C. Fookes, S. Sridharan, and T. Wark, "A mask-based approach for the geometric calibration of thermal-infrared cameras," *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 6, pp. 1625–1635, 2012. DOI: 10.1109/TIM.2012.2182851.

[148] O. Ozyesil, V. Voroninski, R. Basri, and A. Singer, "A survey of structure from motion," *arXiv preprint*, 2017. arXiv: 1701.08493.

[149] S. Ullman, "The interpretation of structure from motion," *Proceedings of the Royal Society of London. Series B, Biological Sciences*, vol. 203, no. 1153, pp. 405–426, 1979. DOI: 10.1098/rspb.1979.0006.

[150] M. Lhuillier, "Incremental fusion of structure-from-motion and gps using constrained bundle adjustment," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 12, pp. 2489–2495, 2012. DOI: 10.1109/TPAMI.2012.157.

[151] J. Heinly, J. L. Schönberger, E. Dunn, and J.-M. Frahm, "Reconstructing the world* in six days *(as captured by the Yahoo 100 million image dataset)," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, 2015, pp. 3287–3295. DOI: 10.1109/CVPR.2015.7298949.

[152] J.-M. Frahm *et al.*, "Building rome on a cloudless day," Hersonissos, Greece, 2010, pp. 368–381. DOI: 10.1007/978-3-642-15561-1_27.

[153] Z. Cui and P. Tan, "Global structure-from-motion by similarity averaging," in *IEEE International Conference on Computer Vision (ICCV)*, Santiago, Chile, 2015, pp. 864–872. DOI: 10.1109/ICCV.2015.105.

[154] D. Crandall, A. Owens, N. Snavely, and D. Huttenlocher, "Discrete-continuous optimization for large-scale structure from motion," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Washington, DC, USA, 2011, pp. 3001–3008. DOI: 10.1109/CVPR.2011.5995626.

*References*

[155] K. Wilson and N. Snavely, "Robust global translations with 1DSfM," in *European Conference on Computer Vision (ECCV)*, Zurich, Switzerland, 2014. DOI: 10.1007/978-3-319-10578-9_5.

[156] R. Gherardi, M. Farenzena, and A. Fusiello, "Improving the efficiency of hierarchical structure-and-motion," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, San Francisco, CA, USA, 2010, pp. 1594–1600. DOI: 10.1109/CVPR.2010.5539782.

[157] L. Zhao, S. Huang, and G. Dissanayake, "Linear SFM: A hierarchical approach to solving structure-from-motion problems by decoupling the linear and nonlinear components," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 141, pp. 275–289, 2018. DOI: 10.1016/j.isprsjprs.2018.04.007.

[158] N. Snavely, S. M. Seitz, and R. Szeliski, "Photo tourism: Exploring photo collections in 3D," *ACM Transactions on Graphics*, vol. 25, no. 3, pp. 835–846, 2006. DOI: 10.1145/1141911.1141964.

[159] S. Agarwal, N. Snavely, I. Simon, S. M. Seitz, and R. Szeliski, "Building rome in a day," in *IEEE International Conference on Computer Vision (ICCV)*, Kyoto, Japan, 2009, pp. 72–79. DOI: 10.1109/ICCV.2009.5459148.

[160] C. Wu, "Towards linear-time incremental structure from motion," in *International Conference on 3D Vision (3DV)*, Seattle, WA, USA, 2013, pp. 127–134. DOI: 10.1109/3DV.2013.25.

[161] Mapillary, *OpenSfM: Open source structure-from-motion pipeline*, https://opensfm.org/, Accessed: 25/02/2022, 2021.

[162] C. Sweeney, *Theia multiview geometry library: Tutorial & reference*, http://theia-sfm.org, Accessed: 11/03/2022.

[163] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 2016, pp. 4104–4113. DOI: 10.1109/CVPR.2016.445.

[164] M. Havlena and K. Schindler, "VocMatch: Efficient multiview correspondence for structure from motion," in *European Conference on Computer Vision (ECCV)*, Zurich, Switzerland, 2014, pp. 46–60. DOI: 10.1007/978-3-319-10578-9_4.

[165] J. L. Schönberger, A. C. Berg, and J.-M. Frahm, "PAIGE: PAirwise Image Geometry Encoding for improved efficiency in structure-from-motion," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, 2015, pp. 1009–1018. DOI: 10.1109/CVPR.2015.7298703.

[166] Y. Lou, N. Snavely, and J. Gehrke, "MatchMiner: Efficient spanning structure mining in large image collections," in *European Conference on Computer Vision (ECCV)*, Firenze, Italy, 2012. DOI: 10.1007/978-3-642-33709-3_4.

[167] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981. DOI: 10.1145/358669.358692.

[168]  R. Raguram, J. Tighe, and J.-m. Frahm, "Improved geometric verification for large scale landmark image collections," in *British Machine Vision Conference (BMVC)*, Surrey, UK, 2012, pp. 77.1–77.11. DOI: 10.5244/C.26.77.

[169]  C. Beder and R. Steffen, "Determining an initial image pair for fixing the scale of a 3d reconstruction from an image sequence," in *DAGM German Conference on Pattern Recognition (GCPR)*, Berlin, Germany, 2006, pp. 657–666. DOI: 10.1007/11861898_66.

[170]  D. Nister, "An efficient solution to the five-point relative pose problem," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 6, pp. 756–770, 2004. DOI: 10.1109/TPAMI.2004.17.

[171]  O. D. Faugeras and F. Lustman, "Motion and structure from motion in a piecewise planar environment," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 02, no. 03, pp. 485–508, 1988. DOI: 10.1142/S0218001488000285.

[172]  R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM: A versatile and accurate monocular SLAM system," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, 2015. DOI: 10.1109/TRO.2015.2463671.

[173]  V. Lepetit, F. Moreno-Noguer, and P. Fua, "EPnP: An accurate O(n) solution to the PnP problem," *International Journal of Computer Vision*, vol. 81, no. 2, pp. 1573–1405, 2008. DOI: 10.1007/s11263-008-0152-6.

[174]  R. I. Hartley and P. Sturm, "Triangulation," *Computer Vision and Image Understanding*, vol. 68, no. 2, pp. 146–157, 1997. DOI: 10.1006/cviu.1997.0547.

[175]  L. Kang, L. Wu, and Y.-H. Yang, "Robust multi-view l2 triangulation via optimal inlier selection and 3D structure refinement," *Pattern Recognition*, vol. 47, pp. 2974–2992, 2014. DOI: 10.1016/j.patcog.2014.03.022.

[176]  B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon, "Bundle adjustment: A modern synthesis," in *International Workshop on Vision Algorithms*, Corfu, Greece, 1999, pp. 298–372. DOI: 10.1007/3-540-44480-7_21.

[177]  J. J. Vega Díaz, M. Vlaminck, D. Lefkaditis, S. A. Orjuela Vargas, and H. Luong, "Solar panel detection within complex backgrounds using thermal images acquired by UAVs," *Sensors*, vol. 20, no. 21, p. 6219, 2020. DOI: 10.3390/s20216219.

[178]  A. Greco, C. Pironti, A. Saggese, M. Vento, and V. Vigilante, "A deep learning based approach for detecting panels in photovoltaic plants," in *International Conference on Applications of Intelligent Systems (APPIS)*, Las Palmas de Gran Canaria, Spain, 2020, pp. 1.1–1.7. DOI: 10.1145/3378184.3378185.

[179]  A. Oliveira, M. Aghaei, and R. Rüther, "Automatic fault detection of photovoltaic arrays by convolutional neural networks during aerial infrared thermography," in *European Photovoltaic Solar Energy Conference and Exhibition (EUPVSEC)*, Marseille, France, 2019, pp. 1302–1307. DOI: 10.4229/EUPVSEC20192019-5BO.6.4.

[180]  D. Manno, G. Cipriani, G. Ciulla, V. Di Dio, S. Guarino, and V. Lo Brano, "Deep learning strategies for automatic fault diagnosis in photovoltaic systems by thermographic images," *Energy Conversion and Management*, vol. 241, p. 114 315, 2021. DOI: 10.1016/j.enconman.2021.114315.

## References

[181]  A. Zhang, Z. C. Lipton, M. Li, and A. J. Smola, "Dive into deep learning," *arXiv preprint*, 2021. arXiv: 2106.11342.

[182]  M. Lin, Q. Chen, and S. Yan, "Network in network," *arXiv preprint*, 2014. arXiv: 1312.4400.

[183]  L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *arXiv preprint*, 2017. arXiv: 1706.05587.

[184]  L. Sifre, "Rigid-motion scattering for image classification," Ph.D. dissertation, École Polytechnique, Palaiseau France, 2014.

[185]  L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M. A. Fadhel, M. Al-Amidie, and L. Farhan, "Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions," *Journal of Big Data*, vol. 8, no. 1, p. 53, 2021. DOI: 10.1186/s40537-021-00444-8.

[186]  C. Kong and S. Lucey, "Take it in your stride: Do we need striding in CNNs?" *arXiv preprint*, 2017. arXiv: 1712.02502.

[187]  K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 2016, pp. 770–778. DOI: 10.1109/CVPR.2016.90.

[188]  J. F. Kolen and S. C. Kremer, "Gradient flow in recurrent nets: The difficulty of learning longterm dependencies," in *A Field Guide to Dynamical Recurrent Networks*. Hoboken, NJ, USA: Wiley-IEEE Press, 2001, pp. 237–243. DOI: 10.1109/9780470544037.ch14.

[189]  A. Apicella, F. Donnarumma, F. Isgrò, and R. Prevete, "A survey on modern trainable activation functions," *Neural Networks*, vol. 138, pp. 14–32, 2021. DOI: 10.1016/j.neunet.2021.01.026.

[190]  J. Han and C. Moraga, "The influence of the sigmoid function parameters on the speed of backpropagation learning," in *From Natural to Artificial Neural Computation*, Malaga-Torremolinos, Spain, 1995, pp. 195–201. DOI: 10.1007/3-540-59497-3_175.

[191]  A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *ICML Workshop on Deep Learning for Audio, Speech and Language Processing*, Atlanta, GA, USA, 2013.

[192]  C. Dugas, Y. Bengio, F. Bélisle, C. Nadeau, and R. Garcia, "Incorporating second-order functional knowledge for better option pricing," in *Advances in Neural Information Processing Systems (NIPS)*, Denver, CO, USA, 2000, pp. 472–478.

[193]  D. Hendrycks and K. Gimpel, "Gaussian error linear units (GELUs)," *arXiv preprint*, 2016. arXiv: 1606.08415.

[194]  D. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (ELUs)," in *International Conference on Learning Representations ICLR*, San Juan, Puerto Rico, 2016.

[195]  I. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, and Y. Bengio, "Maxout networks," in *International Conference on Machine Learning (ICML)*, Atlanta, GA, USA, 2013, pp. 1319–1327.

## References

[196]  A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems (NIPS)*, Lake Tahoe, NV, USA, 2012. DOI: 10.1145/3065386.

[197]  K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations ICLR*, San Diego, CA, USA, 2015.

[198]  S. Xie, R. Girshick, P. Dollar, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 2017, pp. 5987–5995. DOI: 10.1109/CVPR.2017.634.

[199]  A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint*, 2017. arXiv: 1704.04861.

[200]  M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *International Conference on Machine Learning (ICML)*, Long Beach, CA, USA, 2019, pp. 6105–6114.

[201]  Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A convnet for the 2020s," *arXiv preprint*, 2022. arXiv: 2201.03545.

[202]  J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Miami, FL, USA, 2009, pp. 248–255. DOI: 10.1109/CVPR.2009.5206848.

[203]  M. Huh, P. Agrawal, and A. A. Efros, "What makes imagenet good for transfer learning?" *arXiv preprint*, 2016. arXiv: 1608.08614.

[204]  F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, and Q. He, "A comprehensive survey on transfer learning," *Proceedings of the IEEE*, vol. 109, no. 1, pp. 43–76, 2021. DOI: 10.1109/JPROC.2020.3004555.

[205]  K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *IEEE International Conference on Computer Vision (ICCV)*, Venice, Italy, 2017, pp. 2980–2988. DOI: 10.1109/ICCV.2017.322.

[206]  Z. Cai and N. Vasconcelos, "Cascade R-CNN: High quality object detection and instance segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 5, pp. 1483–1498, 2021. DOI: 10.1109/TPAMI.2019.2956516.

[207]  S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, 2018. DOI: 10.1109/CVPR.2018.00913.

[208]  K. Chen *et al.*, "Hybrid task cascade for instance segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, 2019, pp. 4969–4978. DOI: 10.1109/CVPR.2019.00511.

[209]  S. Qiao, L. Chen, and A. Yuille, "DetectoRS: Detecting objects with recursive feature pyramid and switchable atrous convolution," in *IEEE Conference on Computer*

*Vision and Pattern Recognition (CVPR)*, Los Alamitos, CA, USA, 2021, pp. 10 208–10 219. DOI: 10.1109/CVPR46437.2021.01008.

[210] A. M. Hafiz and G. M. Bhat, "A survey on instance segmentation: State of the art," *International Journal of Multimedia Information Retrieval*, vol. 9, pp. 171–189, 2020. DOI: 10.1007/s13735-020-00195-x.

[211] S. Minaee, Y. Y. Boykov, F. Porikli, A. J. Plaza, N. Kehtarnavaz, and D. Terzopoulos, "Image segmentation using deep learning: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2021. DOI: 10.1109/TPAMI.2021.3059968.

[212] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *European Conference on Computer Vision (ECCV)*, Zurich, Switzerland, 2014, pp. 740–755. DOI: 10.1007/978-3-319-10602-1_48.

[213] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 2016. DOI: 10.1109/CVPR.2016.350.

[214] G. Neuhold, T. Ollmann, S. R. Bulò, and P. Kontschieder, "The mapillary vistas dataset for semantic understanding of street scenes," in *IEEE International Conference on Computer Vision (ICCV)*, Venice, Italy, 2017, pp. 5000–5009. DOI: 10.1109/ICCV.2017.534.

[215] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems (NIPS)*, Montréal, Canada, 2015, pp. 91–99.

[216] Y. Li, H. Qi, J. Dai, X. Ji, and Y. Wei, "Fully convolutional instance-aware semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 2017, pp. 4438–4446. DOI: 10.1109/CVPR.2017.472.

[217] J. Dai, K. He, and J. Sun, "Instance-aware semantic segmentation via multi-task network cascades," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 2016, pp. 3150–3158. DOI: 10.1109/CVPR.2016.343.

[218] S. Zagoruyko, A. Lerer, T.-Y. Lin, P. O. Pinheiro, S. Gross, S. Chintala, and P. Dollar, "A multipath network for object detection," in *British Machine Vision Conference (BMVC)*, York, UK, 2016, pp. 15.1–15.12. DOI: 10.5244/C.30.15.

[219] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 2017, pp. 936–944. DOI: 10.1109/CVPR.2017.106.

[220] J. G. Moreno-Torres, T. Raeder, R. Alaiz-Rodríguez, N. V. Chawla, and F. Herrera, "A unifying view on dataset shift in classification," *Pattern Recognition*, vol. 45, no. 1, pp. 521–530, 2012. DOI: 10.1016/j.patcog.2011.06.019.

[221] W. M. Kouw and M. Loog, "An introduction to domain adaptation and transfer learning," *arXiv preprint*, 2019. arXiv: 1812.11806.

# References

[222] P. W. Koh *et al.*, "WILDS: A benchmark of in-the-wild distribution shifts," in *International Conference on Machine Learning (ICML)*, virtual, 2021, pp. 5637–5664.

[223] G. Wilson and D. J. Cook, "A survey of unsupervised deep domain adaptation," *ACM Transactions on Intelligent Systems and Technology*, vol. 11, no. 5, pp. 51.1–51.46, 2020. DOI: 10.1145/3400066.

[224] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. Vaughan, "A theory of learning from different domains," *Machine Learning*, vol. 79, pp. 151–175, 2010. DOI: 10.1007/s10994-009-5152-4.

[225] G. Csurka, R. Volpi, and B. Chidlovskii, "Unsupervised domain adaptation for semantic image segmentation: A comprehensive survey," *arXiv preprint*, 2021. arXiv: 2112.03241.

[226] K. Zhou, Z. Liu, Y. Qiao, T. Xiang, and C. C. Loy, "Domain generalization in vision: A survey," *arXiv preprint*, 2021. arXiv: 2103.02503.

[227] S. Zhao, B. Li, C. Reed, P. Xu, and K. Keutzer, "Multi-source domain adaptation in the deep learning era: A systematic survey," *arXiv preprint*, 2020. arXiv: 2002.12169.

[228] M. Wang and W. Deng, "Deep visual domain adaptation: A survey," *Neurocomputing*, vol. 312, pp. 135–153, 2018. DOI: 10.1016/j.neucom.2018.05.083.

[229] K. Shen, R. M. Jones, A. Kumar, S. M. Xie, and P. Liang, *How does contrastive pre-training connect disparate domains?* https://openreview.net/forum?id=vBn2OXZuQCF, note = Accessed: 14/03/2022, 2022.

[230] K. Rombach, G. Michau, and O. Fink, "Contrastive learning for fault detection and diagnostics in the context of changing operating conditions and novel fault types," *Sensors*, vol. 21, no. 10, p. 3550, 2021. DOI: 10.3390/s21103550.

[231] C. Hu, J. Wu, C. Sun, R. Yan, and X. Chen, "Robust supervised contrastive learning for fault diagnosis under different noises and conditions," in *IEEE International Conference on Sensing, Measurement & Data Analytics in the era of Artificial Intelligence (ICSMD)*, Nanjing, China, 2021, pp. 1–6. DOI: 10.1109/ICSMD53520.2021.9670794.

[232] P. O. Pinheiro, "Unsupervised domain adaptation with similarity learning," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, 2018, pp. 8004–8013. DOI: 10.1109/CVPR.2018.00835.

[233] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, "Supervised contrastive learning," in *Advances in Neural Information Processing Systems (NIPS)*, virtual, 2020, pp. 18 661–18 673.

[234] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, New York, NY, USA, 2006, pp. 1735–1742. DOI: 10.1109/CVPR.2006.100.

[235] P. H. Le-Khac, G. Healy, and A. F. Smeaton, "Contrastive representation learning: A framework and review," *IEEE Access*, vol. 8, pp. 193 907–193 934, 2020. DOI: 10.1109/ACCESS.2020.3031549.

[236] K. Q. Weinberger and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," *Journal of Machine Learning Research (JMLR)*, vol. 10, no. 9, pp. 207–244, 2009.

[237] W. Liu, Y. Wen, Z. Yu, and M. Yang, "Large-margin softmax loss for convolutional neural networks," *arXiv preprint*, 2016. arXiv: 1612.02295.

[238] K. Sohn, "Improved deep metric learning with multi-class n-pair loss objective," in *Advances in Neural Information Processing Systems (NIPS)*, Barcelona, Spain, 2016, pp. 1857–1865.

[239] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, "Unsupervised feature learning via non-parametric instance discrimination," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, 2018, pp. 3733–3742. DOI: 10.1109/CVPR.2018.00393.

[240] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint*, 2018. arXiv: 1807.03748.

[241] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," *arXiv preprint*, 2020. arXiv: 2002.05709.

[242] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, 2020, pp. 9726–9735. DOI: 10.1109/CVPR42600.2020.00975.

[243] C. Wei, Y. Tang, C. Niu, H. Hu, Y. Wang, and J. Liang, "Self-supervised representation learning for evolutionary neural architecture search," *arXiv preprint*, 2020. arXiv: 2011.00186.

[244] M. Ye, X. Zhang, P. C. Yuen, and S.-F. Chang, "Unsupervised embedding learning via invariant and spreading instance feature," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, 2019, pp. 6203–6212. DOI: 10.1109/CVPR.2019.00637.

[245] X. Chen, H. Fan, R. Girshick, and K. He, "Improved baselines with momentum contrastive learning," *arXiv preprint*, 2020. arXiv: 2003.04297.

[246] J. Winkens *et al.*, "Contrastive training for improved out-of-distribution detection," *arXiv preprint*, 2020. arXiv: 2007.05566.

[247] J. Li, C. Xiong, and S. C. Hoi, "Learning from noisy data with robust representation learning," in *IEEE International Conference on Computer Vision (ICCV)*, virtual, 2021, pp. 9465–9474. DOI: 10.1109/ICCV48922.2021.00935.

[248] O. Köpüklü, J. Zheng, H. Xu, and G. Rigoll, "Driver anomaly detection: A dataset and contrastive learning approach," in *IEEE Winter Conference on Applications of Computer Vision (WACV)*, Waikoloa, HI, USA, 2021, pp. 91–100. DOI: 10.1109/WACV48630.2021.00014.

[249] K. Sohn, C.-L. Li, J. Yoon, M. Jin, and T. Pfister, "Learning and evaluating representations for deep one-class classification," in *International Conference on Learning Representations (ICLR)*, virtual, 2021.

[250] J. Tack, S. Mo, J. Jeong, and J. Shin, "CSI: Novelty detection via contrastive learning on distributionally shifted instances," in *Advances in Neural Information Processing Systems (NIPS)*, virtual, 2020, pp. 11 839–11 852.

[251] G. Elsayed, D. Krishnan, H. Mobahi, K. Regan, and S. Bengio, "Large margin deep networks for classification," in *Advances in Neural Information Processing Systems (NIPS)*, Montréal, Canada, 2018.

[252] K. Cao, C. Wei, A. Gaidon, N. Arechiga, and T. Ma, "Learning imbalanced datasets with label-distribution-aware margin loss," in *Advances in Neural Information Processing Systems (NIPS)*, Vancouver, Canada, 2019, pp. 1565–1576.

[253] Y. Xue, K. Whitecross, and B. Mirzasoleiman, "Investigating why contrastive learning benefits robustness against label noise," *arXiv preprint*, 2022. arXiv: 2201.12498.

[254] Z. Zhang and M. R. Sabuncu, "Generalized cross entropy loss for training deep neural networks with noisy labels," in *Advances in Neural Information Processing Systems (NIPS)*, Montréal, Canada, 2018, pp. 8792–8802.

[255] S. Sukhbaatar, J. Bruna, M. Paluri, L. Bourdev, and R. Fergus, "Training convolutional networks with noisy labels," *arXiv preprint*, 2014. arXiv: 1406.2080.

[256] T. Pang, K. Xu, Y. Dong, C. Du, N. Chen, and J. Zhu, "Rethinking softmax cross-entropy loss for adversarial robustness," *arXiv preprint*, 2019. arXiv: 1905.10626.

[257] K. Nar, O. Ocal, S. S. Sastry, and K. Ramchandran, "Cross-entropy loss and low-rank features have responsibility for adversarial examples," *arXiv preprint*, 2019. arXiv: 1901.08360.

[258] Z. Xi, Z. Lou, Y. Sun, X. Li, Q. Yang, and W. Yan, "A vision-based inspection strategy for large-scale photovoltaic farms using an autonomous UAV," in *International Symposium on Distributed Computing and Applications for Business Engineering and Science (DCABES)*, Wuxi, China, 2018, pp. 200–203. DOI: 10.1109/DCABES.2018.00059.

[259] G. Roggi, A. Niccolai, F. Grimaccia, and M. Lovera, "A computer vision line-tracking algorithm for automatic UAV photovoltaic plants monitoring applications," *Energies*, vol. 13, no. 4, p. 838, 2020. DOI: 10.3390/en13040838.

[260] L. Morando, C. T. Recchiuto, J. Calla, P. Scuteri, and A. Sgorbissa, "Thermal and visual tracking of photovoltaic plants for autonomous UAV inspection," *arXiv preprint*, 2022. arXiv: 2202.01003.

[261] D. Kim, J. Youn, and C. Kim, "Automatic fault recognition of photovoltaic modules based on statistical analysis of UAV thermography," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences (ISPRS)*, vol. XLII-2/W6, pp. 179–182, 2017. DOI: 10.5194/isprs-archives-XLII-2-W6-179-2017.

[262] S. Wei, X. Li, S. Ding, Q. Yang, and W. Yan, "Hotspots infrared detection of photovoltaic modules based on hough line transformation and Faster-RCNN approach," in *International Conference on Control, Decision and Information Technologies (CoDIT)*, Paris, France, 2019, pp. 1266–1271. DOI: 10.1109/CoDIT.2019.8820333.

[263] M. Vlaminck, R. Heidbuchel, W. Philips, and H. Luong, "Region-based CNN for anomaly detection in PV power plants using aerial imagery," *Sensors*, vol. 22, no. 3, p. 1244, 2022. DOI: 10.3390/s22031244.

[264] M. Aghaei, S. Leva, and F. Grimaccia, "PV power plant inspection by image mosaicing techniques for IR real-time images," in *IEEE Photovoltaic Specialists Conference (PVSC)*, Portland, OR, USA, 2016, pp. 3100–3105. DOI: 10.1109/PVSC.2016.7750236.

[265] E. Alfaro Mejia, H. Loaiza-Correa, E. Franco, and L. Hernández-Callejo, "Segmentation of thermography image of solar cells and panels," in *Ibero-American Congress of Smart Cities (ICSC-CITIES)*, San José, Costa Rica, 2020, pp. 1–8. DOI: 10.1007/978-3-030-38889-8_1.

[266] F. Wu, D. Zhang, X. Li, X. Luo, J. Wang, W. Yan, Z. Chen, and Q. Yang, "Aerial image recognition and matching for inspection of large-scale photovoltaic farms," in *International Smart Cities Conference (ISC2)*, Wuxi Shi, China), 2017, pp. 1–6. DOI: 10.1109/ISC2.2017.8090792.

[267] D. Kim, J. Youn, and C. Kim, "Automatic photovoltaic panel area extraction from UAV thermal infrared images," *Journal of the Korean Society of Surveying, Geodesy, Photogrammetry and Cartography*, vol. 34, pp. 559–568, 2016. DOI: 10.7848/ksgpc.2016.34.6.559.

[268] H. Zhang, X. Hong, S. Zhou, and Q. Wang, "Infrared image segmentation for photovoltaic panels based on Res-UNet," in *Pattern Recognition and Computer Vision (PRCV)*, Xi'an, China, 2019, pp. 611–622. DOI: 10.1007/978-3-030-31654-9_52.

[269] R. M. Pérez, J. Solano Arias, and A. Méndez-Porras, "Solar panels recognition based on machine learning," in *Jornadas Costarricenses de Investigación en Computación e Informática (JoCICI)*, San Pedro, Costa Rica, 2019, pp. 1–5. DOI: 10.1109/JoCICI48395.2019.9105311.

[270] M. Sait, A. Erguzen, and E. Erdal, "Using Mask R-CNN to isolate PV panels from background object in images," *International Journal of Trend in Scientific Research and Development (IJTSRD)*, vol. 5, no. 1, pp. 1191–1195, 2020.

[271] Á. Huerta Herraiz, A. Pliego Marugán, and F. P. García Márquez, "Photovoltaic plant condition monitoring using thermal images analysis by convolutional neural network-based structure," *Renewable Energy*, vol. 153, pp. 334–348, 2020. DOI: 10.1016/j.renene.2020.01.148.

[272] R. Pierdicca, E. Malinverni, F. Piccinini, M. Paolanti, A. Felicetti, and P. Zingaretti, "Deep convolutional neural network for automatic detection of damaged photovoltaic cells," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences (ISPRS)*, vol. XLII-2, pp. 893–900, 2018. DOI: 10.5194/isprs-archives-XLII-2-893-2018.

[273] R. Pierdicca, M. Paolanti, A. Felicetti, F. Piccinini, and P. Zingaretti, "Automatic faults detection of photovoltaic farms: solAIr, a deep learning-based system for thermal images," *Energies*, vol. 13, no. 24, p. 6496, 2020. DOI: 10.3390/en13246496.

## References

[274]    J. Tsanakas, D. Chrysostomou, P. Botsaris, and A. Gasteratos, "Fault diagnosis of photovoltaic modules through image processing and Canny edge detection on field thermographic measurements," *International Journal of Sustainable Energy*, vol. 34, no. 6, pp. 351–372, 2015. DOI: 10.1080/14786451.2013.826223.

[275]    L. Jiang, J. Su, and X. Li, "Hot spots detection of operating PV arrays through IR thermal image using method based on curve fitting of gray histogram," *MATEC Web of Conferences*, vol. 61, p. 06 017, Jan. 2016. DOI: 10.1051/matecconf/20166106017.

[276]    M. Aghaei, F. Grimaccia, C. A. Gonano, and S. Leva, "Innovative automated control system for PV fields inspection and remote control," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 11, pp. 7287–7296, 2015. DOI: 10.1109/TIE.2015.2475235.

[277]    M. Alsafasfeh, I. Abdel-Qader, B. Bazuin, Q. Alsafasfeh, and W. Su, "Unsupervised fault detection and analysis for large photovoltaic systems using drones and machine vision," *Energies*, vol. 11, no. 9, p. 2252, 2018. DOI: 10.3390/en11092252.

[278]    Z. A. Jaffery, A. K. Dubey, Irshad, and A. Haque, "Scheme for predictive fault diagnosis in photovoltaic modules using thermal imaging," *Infrared Physics & Technology*, vol. 83, pp. 182–187, 2017. DOI: 10.1016/j.infrared.2017.04.015.

[279]    M. U. Ali, H. F. Khan, M. Masud, K. D. Kallu, and A. Zafar, "A machine learning framework to identify the hotspot in photovoltaic module using infrared thermography," *Solar Energy*, vol. 208, pp. 643–651, 2020. DOI: 10.1016/j.solener.2020.08.027.

[280]    A. Et-taleby, M. Boussetta, and M. Benslimane, "Faults detection for photovoltaic field based on k-means, elbow, and average silhouette techniques through the segmentation of a thermal image," *International Journal of Photoenergy*, vol. 2020, p. 6 617 597, 2020. DOI: 10.1155/2020/6617597.

[281]    G. Balasubramani, V. Thangavelu, M. Chinnusamy, U. Subramaniam, S. Padmanaban, and L. Mihet-Popa, "Infrared thermography based defects testing of solar photovoltaic panel with fuzzy rule-based evaluation," *Energies*, vol. 13, no. 6, p. 1343, 2020. DOI: 10.3390/en13061343.

[282]    Q. Wang, K. Paynabar, and M. Pacella, "Online automatic anomaly detection for photovoltaic systems using thermography imaging and low rank matrix decomposition," *Journal of Quality Technology*, pp. 1–14, 2021. DOI: 10.1080/00224065.2021.1948372.

[283]    Y. Su, F. Tao, J. Jin, and C. Zhang, "Automated overheated region object detection of photovoltaic module with thermography image," *IEEE Journal of Photovoltaics*, vol. 11, no. 2, pp. 535–544, 2021. DOI: 10.1109/JPHOTOV.2020.3045680.

[284]    M. Le, V. S. Luong, D. K. Nguyen, V.-D. Dao, N. H. Vu, and H. H. T. Vu, "Remote anomaly detection and classification of solar photovoltaic modules based on deep neural network," *Sustainable Energy Technologies and Assessments*, vol. 48, p. 101 545, 2021. DOI: 10.1016/j.seta.2021.101545.

[285]    M. W. Akram, G. Li, Y. Jin, X. Chen, C. Zhu, and A. Ahmad, "Automatic detection of photovoltaic module defects in infrared images with isolated and

develop-model transfer deep learning," *Solar Energy*, vol. 198, pp. 175–186, 2020. DOI: 10.1016/j.solener.2020.01.055.

[286] R. H. Fonseca Alves, G. A. de Deus Júnior, E. G. Marra, and R. P. Lemos, "Automatic fault classification in photovoltaic modules using convolutional neural networks," *Renewable Energy*, vol. 179, pp. 502–516, 2021. DOI: 10.1016/j.renene.2021.07.070.

[287] M. Millendorf, E. Obropta, and N. Vadhavkar, "Infrared solar module dataset for anomaly detection," in *International Conference on Learning Representations (ICLR)*, Adis Ababa, Ethiopia, 2020, pp. 1–5.

[288] M. Nisi, F. Menichetti, V. Bramante, T. Tr, B. Muhammad, and R. Prasad, "EGNSS high accuracy system improving photovoltaic plant maintenance using RPAS integrated with low-cost RTK receiver," in *Global Wireless Summit (GWS)*, Aarhus, Denmark, 2016.

[289] D. H. Lee and J. H. Park, "Developing inspection methodology of solar energy plants by thermal infrared sensor on board unmanned aerial vehicles," *Energies*, vol. 12, no. 15, p. 2928, 2019. DOI: 10.3390/en12152928.

[290] M. Cramer, "Performance of GPS/inertial solutions in photogrammetry," in *Photogrammetric Week*, Stuttgart, Germany, 2001, pp. 49–62.

[291] *Pix4D: Professional photogrammetry and drone mapping software*, https://www.pix4d.com/, Accessed: 25/02/2022, 2022.

[292] *ODM: A command line toolkit to generate maps, point clouds, 3D models and DEMs from drone, balloon or kite images.* https://www.opendronemap.org/, Accessed: 25/02/2022, 2020.

[293] L. Barazzetti, R. Brumana, D. Oreni, M. Previtali, and F. Roncoroni, "True-orthophoto generation from UAV images: Implementation of a combined photogrammetric and computer vision approach," *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. II, no. 5, pp. 57–63, 2014. DOI: 10.5194/isprsannals-II-5-57-2014.

[294] Pix4D, *Smart inspection of a solar farm using drones*, https://www.pix4d.com/blog/smart-inspection-of-a-solar-farm/, Accessed: 21/04/2022, 2015.

[295] Workswell, *Photovoltaic plant inspection with thermal camera*, https://www.drone-thermal-camera.com/photovoltaic-plant-inspection-with-thermal-camera/, Accessed: 21/04/2022, 2019.

[296] L. Bergman, N. Cohen, and Y. Hoshen, "Deep nearest neighbor anomaly detection," *arXiv preprint*, 2020. arXiv: 2002.10445.

[297] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," *arXiv preprint*, 2021. arXiv: 2103.14030.

[298] Z. Liu *et al.*, "Swin transformer V2: Scaling up capacity and resolution," *arXiv preprint*, 2021. arXiv: 2111.09883.

[299] E. Xie, J. Ding, W. Wang, X. Zhan, H. Xu, P. Sun, Z. Li, and P. Luo, "DetCo: Unsupervised contrastive learning for object detection," *arXiv preprint*, 2021. arXiv: 2102.04803.

# References

[300] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, no. 1, pp. 321–357, 2002. DOI: `10.1613/jair.953`.

[301] M. Long, Y. Cao, J. Wang, and M. I. Jordan, "Learning transferable features with deep adaptation networks," in *International Conference on Machine Learning (PMLR)*, Lille, France, 2015, pp. 97–105.

[302] A. Rozantsev, M. Salzmann, and P. Fua, "Beyond sharing weights for deep domain adaptation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 4, pp. 801–814, 2019. DOI: `10.1109/TPAMI.2018.2814042`.

[303] Y. Zhu, F. Zhuang, and D. Wang, "Aligning domain-specific distribution and classifier for cross-domain classification from multiple sources," in *AAAI Conference on Artificial Intelligence*, Honolulu, HI, USA, 2019, pp. 5989–5996. DOI: `10.1609/aaai.v33i01.33015989`.

[304] H. Guo, R. Pasunuru, and M. Bansal, "Multi-source domain adaptation for text classification via DistanceNet-bandits," *arXiv preprint*, 2020. arXiv: `2001.04362`.

[305] S. Rakshit, B. Banerjee, G. Roig, and S. Chaudhuri, "Unsupervised multi-source domain adaptation driven by deep adversarial ensemble learning," in *Pattern Recognition*. Cham, Switzerland: Springer International Publishing, 2019, pp. 485–498. DOI: `10.1007/978-3-030-33676-9_34`.

[306] J. Hoffman, M. Mohri, and N. Zhang, "Algorithms and theory for multiple-source adaptation," in *Advances in Neural Information Processing Systems (NIPS)*, Montréal, Canada, 2018, pp. 8256–8266.

[307] F. Zhuang, X. Cheng, P. Luo, S. J. Pan, and Q. He, "Supervised representation learning: Transfer learning with deep autoencoders," in *International Joint Conference on Artificial Intelligence (IJCAI)*, Buenos Aires, Argentina, 2015, pp. 4119–4125.

[308] G. Kang, L. Jiang, Y. Yang, and A. G. Hauptmann, "Contrastive adaptation network for unsupervised domain adaptation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, 2019, pp. 4888–4897. DOI: `10.1109/CVPR.2019.00503`.

[309] S. Dai, Y. Cheng, Y. Zhang, Z. Gan, J. Liu, and L. Carin, "Contrastively smoothed class alignment for unsupervised domain adaptation," *arXiv preprint*, 2020. arXiv: `1909.05288`.

[310] C. Park, J. Lee, J. Yoo, M. Hur, and S. Yoon, "Joint contrastive learning for unsupervised domain adaptation," *arXiv preprint*, 2020. arXiv: `2006.10297`.

[311] S. M. and M. P., "Multi-view classification with convolutional neural networks," *PLOS ONE*, vol. 16, no. 1, pp. 1–17, 2021. DOI: `10.1371/journal.pone.0245230`.

[312] W. Zhao, X. Chen, Y. Tang, and Q. Liu, "Learning deep feature fusion for group images classification," in *Chinese Conference on Computer Vision (CCCV)*, Tianjin, China, 2017, pp. 566–576. DOI: `10.1007/978-981-10-7302-1_47`.

[313] J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, "Graph neural networks: A review of methods and applications," *AI Open*, vol. 1, pp. 57–81, 2020. DOI: `10.1016/j.aiopen.2021.01.001`.

# References

[314] G. Pang, C. Shen, L. Cao, and A. V. D. Hengel, "Deep learning for anomaly detection: A review," *ACM Computing Surveys*, vol. 54, no. 2, pp. 38.1–38.38, 2021. DOI: 10.1145/3439950.

[315] S. Bulusu, B. Kailkhura, B. Li, P. K. Varshney, and D. Song, "Anomalous example detection in deep learning: A survey," *IEEE Access*, vol. 8, pp. 132 330–132 347, 2020. DOI: 10.1109/ACCESS.2020.3010274.

[316] R. Chalapathy and S. Chawla, "Deep learning for anomaly detection: A survey," *arXiv preprint*, 2019. arXiv: 1901.03407.

[317] S. Akcay, A. Atapour-Abarghouei, and T. P. Breckon, "GANomaly: Semi-supervised anomaly detection via adversarial training," in *Asian Conference on Computer Vision (ACCV)*, Auckland, New Zealand, 2019, pp. 622–637. DOI: 10.1007/978-3-030-20893-6_39.

[318] H. Zenati, M. Romain, C.-S. Foo, B. Lecouat, and V. Chandrasekhar, "Adversarially learned anomaly detection," in *IEEE International Conference on Data Mining (ICDM)*, Singapore, 2018, pp. 727–736. DOI: 10.1109/ICDM.2018.00088.

[319] J. Chen, S. Sathe, C. Aggarwal, and D. Turaga, "Outlier detection with autoencoder ensembles," in *SIAM International Conference on Data Mining (SDM)*, Houston, TX, USA, 2017, pp. 90–98. DOI: 10.1137/1.9781611974973.11.

[320] J. An and S. Cho, "Variational autoencoder based anomaly detection using reconstruction probability," vol. 2, no. 1, 2015.

[321] L. Ruff, R. A. Vandermeulen, N. Görnitz, A. Binder, E. Müller, K.-R. Müller, and M. Kloft, "Deep semi-supervised anomaly detection," in *International Conference on Learning Representations (ICLR)*, Addis Ababa, Ethiopia, 2020.

[322] L. Ruff, R. Vandermeulen, N. Goernitz, L. Deecke, S. A. Siddiqui, A. Binder, E. Müller, and M. Kloft, "Deep one-class classification," in *International Conference on Machine Learning (PMLR)*, Stockholm, Sweden, 2018, pp. 4393–4402.

[323] D. Hendrycks, M. Mazeika, S. Kadavath, and D. Song, "Using self-supervised learning can improve model robustness and uncertainty," in *Advances in Neural Information Processing Systems (NIPS)*, Vancouver, Canada, 2019, pp. 15 663–15 674.

[324] I. Golan and R. El-Yaniv, "Deep anomaly detection using geometric transformations," in *Advances in Neural Information Processing Systems (NIPS)*, Montréal, Canada, 2018, pp. 9781–9791.

[325] S. Wang, Y. Zeng, X. Liu, E. Zhu, J. Yin, C. Xu, and M. Kloft, "Effective end-to-end unsupervised outlier detection via inlier priority of discriminative network," in *Advances in Neural Information Processing Systems (NIPS)*, Vancouver, Canada, 2019.

[326] L. Bergman and Y. Hoshen, "Classification-based anomaly detection for general data," in *International Conference on Learning Representations (ICLR)*, Adis Ababa, Ethiopia, 2020.

[327] P. Perera and V. M. Patel, "Learning deep features for one-class classification," *IEEE Transactions on Image Processing*, vol. 28, no. 11, pp. 5450–5463, 2019. DOI: 10.1109/TIP.2019.2917862.

[328] S. F. Yilmaz and S. S. Kozat, "Unsupervised anomaly detection via deep metric learning with end-to-end optimization," *arXiv preprint*, 2020. arXiv: 2005.05865.

[329] T. Defard, A. Setkov, A. Loesch, and R. Audigier, "PaDiM: A patch distribution modeling framework for anomaly detection and localization," in *International Conference on Pattern Recognition (ICPR)*, virtual, 2021, pp. 475–489. DOI: 10.1007/978-3-030-68799-1_35.

[330] K. Roth, L. Pemula, J. Zepeda, B. Schölkopf, T. Brox, and P. Gehler, "Towards total recall in industrial anomaly detection," *arXiv preprint*, 2021. arXiv: 2106.08265.

[331] J. Yu, Y. Zheng, X. Wang, W. Li, Y. Wu, R. Zhao, and L. Wu, "FastFlow: Unsupervised anomaly detection and localization via 2d normalizing flows," *arXiv preprint*, 2021. arXiv: 2111.07677.

[332] S.-S. Lin, "Review: Extending visible band computer vision techniques to infrared band images," *Technical Reports (CIS)*, 2001.

[333] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "g²o: A general framework for graph optimization," in *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, 2011, pp. 3607–3613. DOI: 10.1109/ICRA.2011.5979949.

[334] D. Merkel, "Docker: Lightweight linux containers for consistent development and deployment," *Linux journal*, vol. 2014, no. 239, p. 2, 2014.

[335] The Qt Company Ltd., *Qt for Python*, https://doc.qt.io/qtforpython, Accessed: 15/03/2022, 2021.

[336] V. Agafonkin, *Leaflet: An open-source JavaScript library for mobile-friendly interactive maps*, https://leafletjs.com, Accessed: 15/03/2022, 2010.

[337] I. Labelbox, *Labelbox*, https://labelbox.com, Accessed: 15/03/2022, 2022.

[338] B. Sekachev *et al.*, *Computer Vision Annotation Tool (CVAT)*, https://github.com/openvinotoolkit/cvat, Accessed: 15/03/2022, 2020.

[339] M. Grinberg, *Flask web development: Developing web applications with Python*, 2nd. Boston, MA, USA: O'Reilly Media, Inc., 2018, ISBN: 9781491991732.

[340] M. Bostock, *D3.js – Data-Driven Documents*, https://d3js.org/, Accessed: 15/03/2022, 2021.

# Appendices

# A  Publications

The following is an overview of the journal articles, in which the methods and results of this dissertation have been published.

## Publication 1

**Abstract**  Increasing deployment of photovoltaics (PV) plants demands for cheap and fast inspection. A viable tool for this task is thermographic imaging by unmanned aerial vehicles (UAV). In this work, we develop a computer vision tool for the semi-automatic extraction of PV modules from thermographic UAV videos. We use it to curate a dataset containing 4.3 million IR images of 107842 PV modules from thermographic videos of seven different PV plants. To demonstrate its use for automated PV plant inspection, we train a ResNet-50 to classify ten common module anomalies with more than $90\,\%$ test accuracy. Experiments show that our tool generalizes well to different PV plants. It successfully extracts PV modules from 512 out of 561 plant rows. Failures are mostly due to an inappropriate UAV trajectory and erroneous module segmentation. Including all manual steps our tool enables inspection of $3.5\,\mathrm{MW_p}$ to $9\,\mathrm{MW_p}$ of PV installations per day, potentially scaling to multi-gigawatt plants due to its parallel nature. While we present an effective method for automated PV plant inspection, we are also confident that our approach helps to meet the growing demand for large thermographic datasets for machine learning tasks, such as power prediction or unsupervised defect identification.

**Author Contributions**  L.B. is the principal author of this work. He developed, implemented, and tested the proposed computer vision pipeline, conducted all experiments, and wrote the initial manuscript. He developed software tools for data annotation and used the pipeline to assemble the presented image dataset. The defect annotation scheme for the dataset was developed by L.B., T.P., and C.B-L., and C.B-L. manually annotated the dataset. T.P. piloted the drone and acquired the IR videos used in this work. C.B-L., C.B., J.H, and I.P. procured funding for this project. Supervision was provided by C.B. and I.P. All authors edited the manuscript.

## Publication 2

**Abstract**    Increasing deployment of photovoltaic (PV) plants requires methods for automatic detection of faulty PV modules in modalities, such as infrared (IR) images. Recently, deep learning has become popular for this. However, related works typically sample train and test data from the same distribution ignoring the presence of domain shift between data of different PV plants. Instead, we frame fault detection as more realistic unsupervised domain adaptation problem where we train on labelled data of one source PV plant and make predictions on another target plant. We train a ResNet-34 convolutional neural network with a supervised contrastive loss, on top of which we employ a $k$-nearest neighbor classifier to detect anomalies. Our method achieves a satisfactory area under the receiver operating characteristic (AUROC) of $73.3\%$ to $96.6\%$ on nine combinations of four source and target datasets with 2.92 million IR images of which $8.5\%$ are anomalous. It even outperforms a binary cross-entropy classifier in some cases. With a fixed decision threshold this results in $79.4\%$ and $77.1\%$ correctly classified normal and anomalous images, respectively. Most misclassified anomalies are of low severity, such as hot diodes and small hot spots. Our method is insensitive to hyperparameter settings, converges quickly and reliably detects unknown types of anomalies making it well suited for practice. Possible uses are in automatic PV plant inspection systems or to streamline manual labelling of IR datasets by filtering out normal images. Furthermore, our work serves the community with a more realistic view on PV module fault detection using unsupervised domain adaptation to develop more performant methods with favorable generalization capabilities.

**Author Contributions**    L.B. is the principal author of this work. He developed the proposed deep learning method, conducted all experiments, and wrote the initial manuscript. In several discussions, M.H. provided feedback on the deep learning algorithm. This work made used the image dataset created in cooperation with T.P. and C.B-L. in the first publication. C.B-L., C.B., J.H, A.M., and I.P. procured funding for this project. Supervision was provided by C.B. and I.P. All authors edited the manuscript.

# Publication 3

**Abstract**   To identify abnormal photovoltaic (PV) modules in large-scale PV plants economically, drone-mounted infrared (IR) cameras and automated video processing algorithms are frequently used. While most related works focus on the detection of abnormal modules, little has been done to automatically localize those modules within the plant. In this work, we use incremental structure-from-motion to automatically obtain geocoordinates of all PV modules in a plant based on visual cues and the measured GPS trajectory of the drone. In addition, we extract multiple IR images of each PV module. Using our method, we successfully map 99.3 % of the 35084 modules in four large-scale and one rooftop plant and extract over 2.2 million module images. As compared to our previous work, extraction misses 18 times less modules (one in 140 modules as compared to one in eight). Furthermore, two or three plant rows can be processed simultaneously, increasing module throughput and reducing flight duration by a factor of 2.1 and 3.7, respectively. Comparison with an accurate orthophoto of one of the large-scale plants yields a root mean square error of the estimated module geocoordinates of 5.87 m and a relative error within each plant row of 0.22 m to 0.82 m. Finally, we use the module geocoordinates and extracted IR images to visualize distributions of module temperatures and anomaly predictions of a deep learning classifier on a map. While the temperature distribution helps to identify disconnected strings, we also find that its detection accuracy for module anomalies reaches, or even exceeds, that of a deep learning classifier for seven out of ten common anomaly types. The software is published at https://github.com/LukasBommes/PV-Hawk.

**Author Contributions**   L.B. is the principal author of this work. He developed the proposed method, conducted all experiments, and wrote the initial manuscript. Furthermore, he implemented and published the method in form of a software tool. He also published a graphical software tool for browsing the results of the method. T.P. piloted the drone and acquired additional IR videos for this work. C.B-L., C.B., J.H, and I.P. procured funding for this project. Supervision was provided by C.B. and I.P. All authors edited the manuscript.

# B  Published Software

The methods in this dissertation are purely computational and can be accurately formalized as software. The following is a brief overview of the software tools that were developed during this dissertation. All software is published under the permissive MIT license to benefit the research community and accelerate further work in the field.

## B.1  PV Hawk

PV Hawk[1] is a Python command line tool (see fig. B.1a), which implements the computer vision pipeline as presented in publication [3]. The tool contains all algorithms for the extraction and mapping of PV modules based on IR videos. PV Hawk relies extensively on OpenCV [140] for image processing and uses the OpenSfM [161] library for incremental structure from motion. The g2o [333] graph optimization framework is used to refine triangulated PV module corners during the mapping procedure. The application is packaged as a Docker container [334] to ensure fast and easy setup on other machines.

## B.2  PV Hawk Viewer

The PV Hawk Viewer[2] is a desktop app (see fig. B.1b) for viewing reconstructions of large-scale PV plants created with PV Hawk. The user can browse individual PV modules on an interactive map and corresponding IR video frames and extracted module images are shown. Functionality for annotating module anomalies and electrical connectivity of modules is provided. Furthermore, PV Hawk Viewer contains the algorithms for PV module anomaly detection from publication [3] and visualizes anomaly detection results on the map. PV Hawk Viewer is built with the Qt for Python [335] desktop framework following the model-view-controller architecture to separate data models, application logic, and user interface. The map component in PV Hawk Viewer is based on the Leaflet [336] library, written in JavaScript and embedded with a QT web engine view. Python distutils are used for packaging and distribution of the application via the Python package index (PyPI).

---

[1]https://github.com/LukasBommes/PV-Hawk
[2]https://github.com/LukasBommes/PV-Hawk-Viewer

## B.3  Grid Annotation Tool

The Grid Annotation Tool[3] is a web app for annotating image datasets for instance segmentation (see fig. B.1c). It was built for annotating the PV module segmentation dataset needed to train and evaluate the Mask R-CNN model in publication [1]. As opposed to existing labeling tools, such as Labelbox [337] or CVAT [338], grid-like structures can be annotated faster and more accurately by annotating the grid spines instead of individual polygons. The backend of the Grid Annotation Tool is based on Python Flask [339] and the frontend is written in JavaScript relying on the D3 [340] library for drawing.

---

[3]https://github.com/LukasBommes/Grid-Annotation-Tool
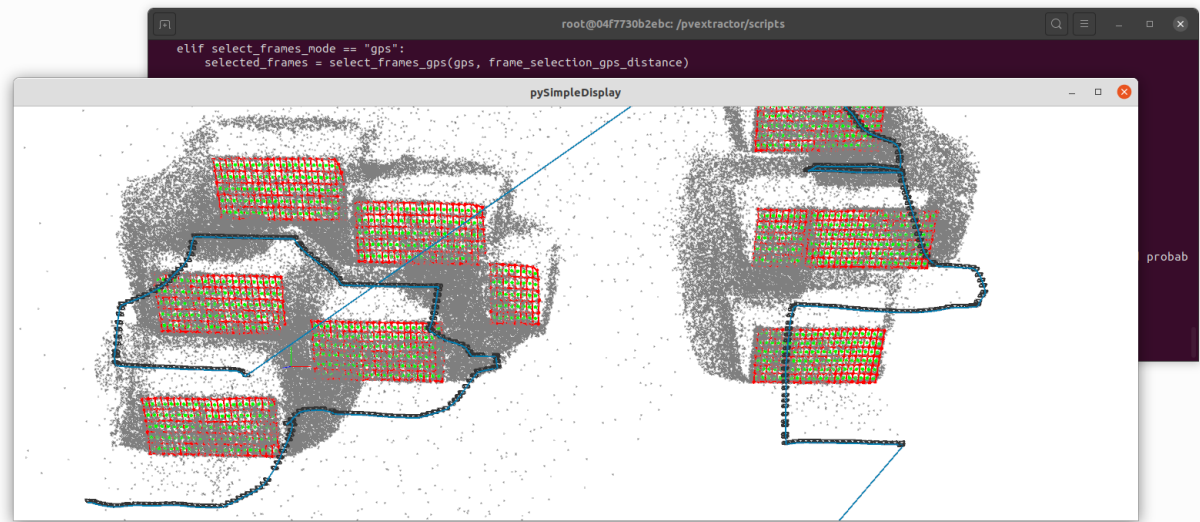
(a) *PV Hawk*: Extract and map PV modules from aerial IR videos of large-scale PV plants.
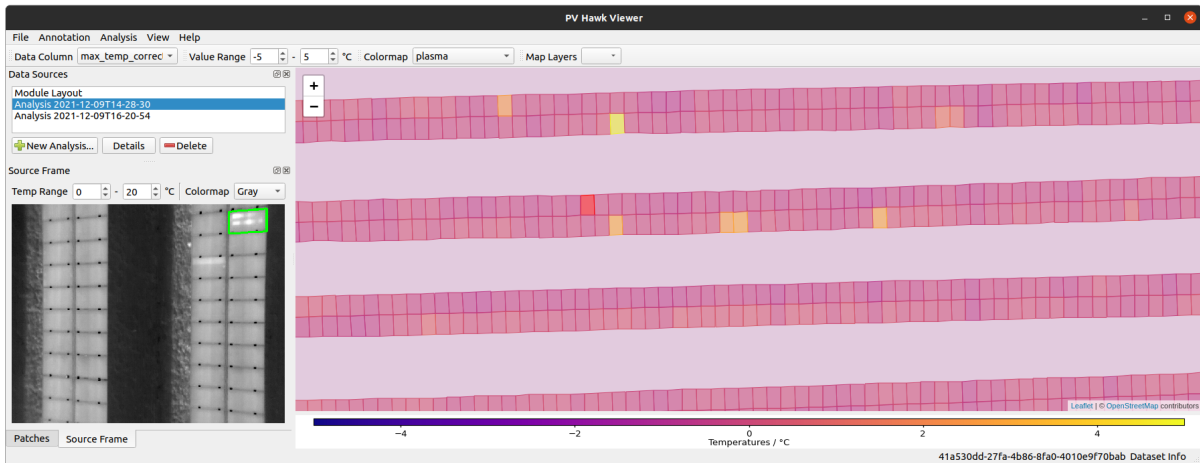


(b) *PV Hawk Viewer*: View and analyze PV plant reconstructions created with PV Hawk.
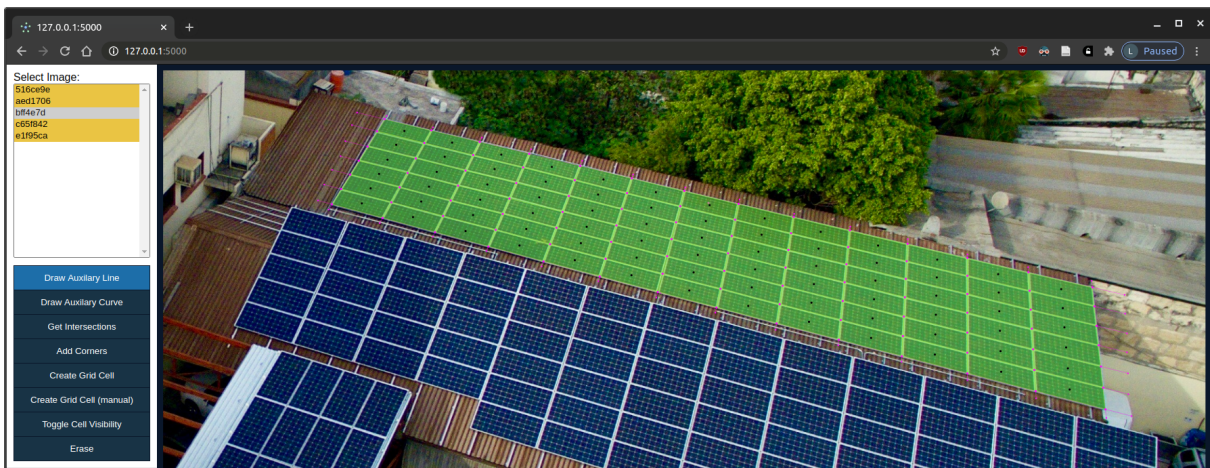


(c) *Grid Annotation Tool*: Annotate datasets for the instance segmentation of PV modules.

Figure B.1: Software published with this thesis.

# Curriculum Vitae

| | |
|---|---|
| 06/2020 – today | **PhD Candidate at Helmholtz Institute Erlangen-Nürnberg for Renewable Energy** <br> Thesis: *"Computer Vision Pipeline for the Automated Inspection of Photovoltaic Plants"* <br> • Development of a software for automated detection, georeferencing and defect identification of photovoltaic modules in aerial infrared videos <br> • Application of my tool to many real-world inspection videos and creation of a dataset of over 6 million thermal images for defect detection <br> • Implementation of a desktop app with Qt and web apps with Python Flask for annotating, browsing and analyzing datasets <br> • Deployment of my software in collaboration with industry partners |
| 03/2019 - 03/2020 | **Research Software Engineer at A*STAR SIMTech** <br> • Development of a Python/C++ app for video-based detection and tracking of staff and objects on the shopfloor <br> • Realization of a fast object tracker which exploits motion vectors in encoded videos <br> • Implementation of a C++ library with Python bindings to synchronize multiple RTSP video streams <br> • Supervision of interns |
| 10/2016 - 03/2019 | **Master Mechanical Engineering at TU Braunschweig** |
| 06/2018 - 03/2019 | Master's Thesis at A*STAR SIMTech <br> • Topic: *"Shopfloor Monitor: Multi-Camera-Based Detection and Tracking System for a Manufacturing Environment"* |
| 10/2017 - 06/2018 | Student Research Assistant at IWF <br> • Development of deep learning methods for predictive maintenance of machine tools |
| 04/2017 - 10/2017 | Study Thesis at A*STAR Singapore Institute of Manufacturing Technology (SIMTech) <br> • Thesis: *"Application of Machine Learning Algorithms for Operating State Classification and Fault Detection of a CNC Milling Machine"* <br> • Project work: Development of a software for monitoring of the beam trajectory in a laser welding machine |
| 10/2016 - 04/2017 | Student Research Assistant at Institute of Machine Tools and Production Technology (IWF) TU Braunschweig <br> • Implementation of a C++ microcontroller firmware for a workpiece transport system |
| 10/2013 - 07/2016 | **Bachelor Mechanical Engineering at TU Braunschweig** <br> • Thesis: *"Entwicklung und Aufbau einer Regelung zum Spannen von Werkstücken mittels Adhäsionskräften in der Mikrobearbeitung"* |

# Computer Vision Tool for Detection, Mapping and Fault Classification of PV Modules in Aerial IR Videos

Lukas Bommes[1], Tobias Pickel[1], Claudia Buerhop-Lutz[1], Jens Hauch[1], Christoph Brabec[1,2], and Ian Marius Peters[1]

[1]*Forschungszentrum Jülich GmbH, Helmholtz-Institute Erlangen-Nuremberg for Renewable Energies (HI ERN)*
[2]*Institute Materials for Electronics and Energy Technology, Universität Erlangen-Nürnberg (FAU)*
Correspondence to i.peters@fz-juelich.de

## Abstract

*Increasing deployment of photovoltaics (PV) plants demands for cheap and fast inspection. A viable tool for this task is thermographic imaging by unmanned aerial vehicles (UAV). In this work, we develop a computer vision tool for the semi-automatic extraction of PV modules from thermographic UAV videos. We use it to curate a dataset containing 4.3 million IR images of 107842 PV modules from thermographic videos of seven different PV plants. To demonstrate its use for automated PV plant inspection, we train a ResNet-50 to classify ten common module anomalies with more than 90 % test accuracy. Experiments show that our tool generalizes well to different PV plants. It successfully extracts PV modules from 512 out of 561 plant rows. Failures are mostly due to an inappropriate UAV trajectory and erroneous module segmentation. Including all manual steps our tool enables inspection of $3.5\,MW_p$ to $9\,MW_p$ of PV installations per day, potentially scaling to multi-gigawatt plants due to its parallel nature. While we present an effective method for automated PV plant inspection, we are also confident that our approach helps to meet the growing demand for large thermographic datasets for machine learning tasks, such as power prediction or unsupervised defect identification.*

Figure 1: High-level overview of our tool for semi-automatic inspection of PV plants using thermographic videos acquired by an UAV.

## I. Introduction

Deployment of solar photovoltaics (PV) has increased exponentially in the past years. At the end of 2019, globally installed capacity reached $586\,GW_p$ [1]. Many PV plants contain defective PV modules which pose safety hazards and reduce power output, yield and as a consequence, the profitability of the plant. Defects occur during manufacturing, installation or due to aging. To identify defective modules PV plants need to be inspected regularly.

A valuable tool for defect identification in PV modules is thermographic imaging which uses a thermal IR camera to visualize defects based on their increased temperature. To speed up the inspection process thermography is typically performed by unmanned aerial vehicles (UAV) [2–5]. Many works have explored the use of UAVs for PV plant inspection. A high-level overview of the inspection process and the challenges involved is given in [6, 7]. [8] compares available camera and drone technologies and [9] performs an economical analysis. [10, 11] analyze the influence of the image resolution on the detectability of defects.

UAV thermography of PV plants with millions of modules produces so many images that manual sighting is infeasible. This raises the need for image processing tools which automatically detect PV modules in each image and identify thermal anomalies. To enable repairs or exchange of defective modules the automated processing tool needs to further determine the exact location of each module in the plant. Instead of taking individual images at predetermined positions, we simply fly along each row of the PV plant and acquire videos. This renders expensive and time consuming flight planning unnecessary and allows for faster inspection on-site. However, it increases the amount of data as each PV module occurs in multiple consecutive video frames. It further introduces perspective distortion and other artefacts, such as sun reflections, which need to be handled by the processing tool to make the images usable for downstream anomaly classification and other machine learning algorithms. The large number of acquired thermographic images is key to accurate anomaly classification as some anomalies are very seldom and machine learning algorithms used for anomaly classification require many examples to achieve high accuracy and good generalization.

In this work we develop such an image processing tool for the semi-automatic extraction and localization of PV modules in UAV thermographic videos of large-scale PV plants (see fig. 1). It can be used to automate inspection of PV plants and to curate large datasets for downstream machine-learning tasks. While there are several works on automated PV plant inspection systems [12–19], they rely heavily on classic image processing techniques, such as intensity thresholding (see tab. 1). These techniques are based on heuristics, need extensive manual tuning, do not generalize well and are not very accurate. Further, many of the related works can distinguish at most three different thermal anomalies or perform only a binary classification. First works have shown promising results using deep learning for these tasks [20, 21]. Following this recent trend, we use the Mask R-CNN instance segmentation framework [22] to robustly extract PV modules from thermal IR videos. A ResNet-50 deep convolutional classifier [23] is used

for fine-grained classification of ten thermal anomalies. Further, we exploit the large redundancy and temporal context present in the video data to efficiently build a large-scale dataset of thermographic images of PV modules for downstream machine learning tasks. To summarize, our contributions are as follows:

- A tool for semi-automatic extraction and localization of PV modules in UAV thermographic videos of large-scale PV plants which can be used for automated plant inspection and to curate large datasets for downstream machine-learning tasks.
- A dataset of 4.3 million thermographic images of 107842 PV modules from seven PV plants with fine-grained labels of ten common thermal anomalies.
- Training and evaluation of a ResNet-50 classifier on our dataset.
- A quantitative analysis of generalization ability, processing time and failure cases of our tool.

## II. RELATED WORKS

The following is an overview of related methods for semi-automatic thermographic PV plant inspection by UAVs. We compare them in terms of module detection, thermal anomaly detection and localization of modules in the plant. Tab. 1 summarizes methods and dataset sizes of the related works.

### A. PV Module Detection

Most works employ classic computer vision algorithms to detect PV modules in both visual and thermographic images. The most popular method used by [13–15, 19, 24, 25] is binary thresholding of image intensities to obtain segmentation masks of the PV modules. [21] detects rectangular candidate contours by thresholding, extracts texture features and classifies them with a Support Vector Machine (SVM). Other works find edges of PV modules using morphological operations [26, 27] or the Hough transform [12, 16]. More exotic techniques are template matching [18] and maximally stable extremal regions [17]. Main issue of all these works is their reliance on classic image processing which is based on manual priors and heuristics, needs extensive manual tweaking of hyper parameters and generalizes poorly to unseen imagery.

Deep learning overcomes these problems and is applied to PV module detection by [21, 28, 29]. [28] performs semantic segmentation with a combination of a ResNet-34 [23] and a U-Net [30]. A weakness of semantic segmentation is that it does not distinguish between individual PV modules. [29] employs the YOLO object detector [31] which does not have this problem. However, it suffers from the imprecise representation of PV modules by bounding boxes instead of segmentation masks. Similar to our work [21] solves both problems by utilizing the Mask R-CNN instance segmentation model. It outputs an individual segmentation masks for each PV module which allow for accurate localization of PV modules in thermographic images.

### B. Thermal Anomaly Detection

Similar to the PV module detection many works [13, 14, 16, 17, 32] use binary thresholding to segment hot regions of PV modules in thermographic images which correspond to thermal anomalies. The works in [12, 33] iteratively grow segmentation masks of hot spots starting from local intensity maxima. In [18] hot spots are found by template matching. Another approach is to extract features, such as mean and standard deviation, for each PV module and finding outliers with statistical tests [19] or by comparing with neighbouring modules [25].

Several recent works explore deep learning for anomaly detection to overcome the limitations of classic image processing [20, 34, 35]. In [34] a segmentation model based on VGG-16 is used to segment three different anomalies directly in the thermographic image. VGG-16 is also used by [35] to classify whether an image contains an anomalous module or not. Problem of this method is the inability to accurately localize the anomalous module. In [20] four different anomalies are classified using MobileNet and VGG-16. The authors find that both deep learning methods outperform a SVM and a Random Forest classifier using SIFT features.

Problem of the current methods is that the list of anomalies classified is by no means complete. Further, small datasets with only 360 to 3336 images are used.

Similar to [20] we utilize a deep convolutional classifier, in our case ResNet-50. However, we obtain a significantly larger anomaly classification dataset with more than 450000 images and perform a much more fine-grained classification of ten thermal anomalies. In addition, we employ majority voting over subsequent video frames to enhance classification accuracy.

### C. Localization of PV Modules in the Plant

To localize PV modules in the PV plant [13, 14, 26] create panorama images of each row, detect modules and assign an ID to each module. This way, module locations are defined relative to other modules. [15] uses the same technique and additionally matches each row panorama to a CAD plan by means of GPS positions. Problematic is the need for an accurate flight path with specified overlap of individual images which makes the UAV operation more complicated. Further, CAD files are not always available and the format can vary for different PV plants.

Several works [36–38] create an orthophoto of the entire PV plant from a higher altitude. This requires nadiral images with a suitable overlap which may not always be feasible in case of nearby power lines, streets or train tracks. Spatial resolution of a high-altitude image is low making fine-grained anomaly classification of PV modules difficult.

Other works [18, 39] use direct georeferencing to estimate the GPS position of each PV module in the image. This requires an expensive Real Time Kinematics system to accurately estimate the UAVs position.

In [40] GPS positions of the video frames containing an anomalous PV module are marked on a map. While this is straightforward it still requires manual localization of the anomalous module within the frame.

Our work uses relative mapping similar to [13, 14, 26]. Instead of creating a panorama, we encode the spatial relationship of PV modules in a graph that is matched with a standardized *plant file* containing module identifiers. This allows for easy integration of other data modalities, such as electrical measurements. The plant file needs to be created only once for each plant which saves time when inspecting the same plant multiple times. We further do not require nadiral images or a specific overlap of adjacent frames and

a standard GPS receiver is sufficient. This reduces cost and allows for a more flexible operation of the UAV.

## III. Video Dataset

For this work we acquire thermographic videos of seven utility-scale PV plants containing a combined 122865 PV modules (ranging from 2850 to 35360 modules per plant). As can be seen in fig. 2 the plants in our dataset cover a variety of row layouts, module sizes, module orientations and module technologies. Plant D comprises of thin-film modules while the others use crystalline silicon modules. In total our dataset contains 8 hours of video footage (231172 frames) with on average 21.8 PV modules per frame. Videos were acquired by a UAV of type DJI Matrice 210 and a DJI Zenmuse XT2 camera which has a resolution of $640 \times 512$ pixels and a frame rate of 8 Hz. Acquisition took place under clearsky conditions and solar irradiance above $700\,\mathrm{W\,m^{-2}}$.
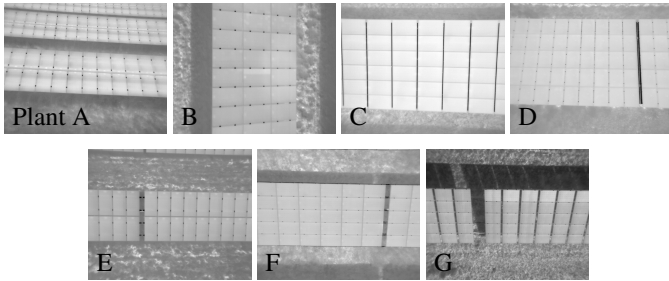


Figure 2: Example video frames of the seven PV plants in our dataset.

## IV. PV Module Extraction

This section introduces our tool for semi-automatic extraction of PV modules from thermographic videos. An overview can be found in fig. 3. First, the tool splits thermographic videos into individual frames and extracts their GPS coordinates. Aided by the GPS coordinates the user manually specifies which frames belong to which row of the PV plant. PV modules are segmented by Mask R-CNN, extracted, rectified and stored to disk. A tracking algorithm associates each PV module in subsequent video frames with a unique *track ID*. This way the extracted patches of each PV module can be grouped together. Finally, track IDs are associated with plant IDs. Plant IDs are specified in a standardized *plant file* and describe the electrical wiring and the location of each module in the plant. We chose a semi-automatic approach to achieve a high degree of flexibility and good generalization to different PV plants.

The rest of this section explains the tool in detail.

### A. Video Acquisition and Preprocessing

Thermographic videos can be captured with any UAV or camera as long as the following requirements are fullfilled:

- Each row of the PV plant is scanned individually.
- The camera moves monotonically along the row, i.e. there is no significant backward movement.
- The current row must be fully visible and always the front-most (bottommost) one in each frame.
- The row must lie approximately horizontal or vertical in each frame.

Our tool is robust to changes of the flight velocity, altitude and camera angle. This allows the operator to manually track rows with varying elevation (e.g. hillsides) and choose the optimal camera angle to reduce sun reflections. Additional rows which may become visible in the background due to low camera angles are filtered out.

After acquisition thermal IR videos are split into individual frames and stored as 16-bit grayscale TIFFs. The GPS position of each frame is extracted and stored in CSV and KML files. They are needed during the manual grouping of frames that follows in the next step. In case the PV rows are vertical we rotate the video frames by 90° to enable equal treatment of both cases in the remaining processing steps.

### B. Grouping of Frames into Rows

For maximum flexibility our tool processes each row of the PV plant independently. To this end, the user has to manually specify which video frames belong to which row of the PV plant. Specifically, he has to provide the plant IDs of the bottom left and top right modules and the index of the first and last frame of each row. A graphical tool (see fig. 4) for browsing frames based on their GPS position simplifies this process. The user can skip parts of the video and rows do not need to be scanned in any particular order. It is also possible to scan rows partially, e.g. when a row contains multiple strings of which only a subset needs to be inspected. Further, single frames can be processed which is useful for short rows.

### C. PV Module Segmentation

To locate PV modules in each video frame we use the Mask R-CNN instance segmentation framework. It outputs an axis-aligned bounding box and a binary segmentation mask for each PV module. We train it to segment only fully visible PV modules. Example outputs are shown in fig. 5.

*1) Dataset:* For fine-tuning of Mask R-CNN we annotate segmentation masks and bounding boxes of 26612 PV modules in 862 video frames of PV plants A, B, C and D. For this we developed a custom annotation tool, however any annotation tool for instance segmentation can be used. We select 60 frames (15 of each PV plant) with a total of 2104 PV modules for validation and the remaining 802 frames for training. For compatibility with Mask R-CNN we convert the 16-bit grayscale frames to Celsius scale, normalize the values to the interval $[0, 255]$, convert to 8-bit, maximize contrast by means of a histogram equalization, convert to RGB and subtract the channel means estimated from the training set. In addition, each frame is padded with zeros to a square of size $640 \times 640$ pixels.

*2) Training:* Starting from MS COCO-pretrained weights [41] we train the segmentation and classification heads of Mask R-CNN for 59 epochs using stochastic gradient descent with a batch size of 2, learning rate 0.001, momentum 0.9 and weight decay 0.0001. Subsequently, all weights are fine-tuned for additional 60 epochs with 1/10th of the previous learning rate. During both training stages frames are augmented by random up-down and left-right flips and (in 50 % of the cases) rotation by a uniform random angle between $-10°$ and $10°$. We additionally rotate images by ±90° in 50 % of the cases to reduce differences between landscape and portrait orientation of modules.

Table 1: Comparison of related works on PV module detection and thermal anomaly detection in aerial IR images of PV plants. F1-scores are taken from the original works and are not directly comparable due to different test datasets and different definitions of the F1-score (pixel-based, bounding box-based, choice of IoU threshold). A unification is out of the scope of this work. F1-scores defined in the same way as in our work are demarked with a [†].

| Work | Test (train) dataset Images / Modules / Plants | Module detection Method | Type | F1/% | Anomaly detection Method | Anomaly classes | | F1/% |
|---|---|---|---|---|---|---|---|---|
| [17] | 20 / 240 / 1 | Region proposal by Maximally Stable Extremal Regions (MSER) + filtering by size | Boxes | n.a. | Segmentation by binary thresholding | 1 | Hot spot | n.a. |
| [16] | 1171 / – / 1 | Edge extraction by Hough transform + postprocessing | Lines | n.a. | Segmentation by binary thresholding | 1 | Hot spot | 59.0 |
| [13–15] | 34 / – / 1 | Segmentation by binary thresholding in HSV-space | Mask | n.a. | Segmentation by binary thresholding with two thresholds + classification heuristics | 3 | Hot spot, hot substring, hot module | 98.8[†] |
| [19] | 37 / 1544 / 2 | Segmentation by binary thresholding with adaptive threshold | Mask | 92.8 | Feature extraction + classification with Grubb's test and Dixon's Q test | 3 | Hot spot, hot substring, hot module | 93.9[†] |
| [24, 25] | 3 / 204 / 1 | Segmentation by binary thresholding + morphological operations | Mask | 95.8 | Feature extraction (mean & std) + comparison with neighbouring modules | 3 | Hot spot, hot substring, hot module | 92.9[†] |
| [18] | 270 / – / 1 | Template matching | Boxes | 83.0 | Template matching | 1 | Hot spot | 75.0[†] |
| [12] | – / 14215 / >1 | Canny edge detection + Hough transform | Lines | 87.0 | Segmentation by water filling algorithm + temporal tracking with majority voting | 1 | Hot spot | 72.0 |
| [21] | test: 20 / – / 3 train: 80 / – / 3 | Rectangle extraction by adaptive thresholding + SVM classifier on texture features | Boxes + Masks | 98.3 | – | – | – | – |
| [20] | test: 77 / – / 3 train: 306 / – / 3 | – | – | – | SIFT feature extraction + Random Forest classifier | 4 | Sh, Sp, Mp, Cs+ (see fig. 11) | 77.2[†] |
| [21] | test: 20 / – / 3 train: 80 / – / 3 | DL instance segmentation (Mask R-CNN) + postprocessing | Boxes + Masks | 98.9 | – | – | – | – |
| [29] | test: – / 14499 / >1 train: – / 36000 / >1 | DL object detection (YOLOv3) | Boxes | 95.0 | – | – | – | – |
| [28] | test: 19 / – / 1 train: 216 / – / 1 | DL semantic segmentation (ResNet-34 + U-Net) | Mask | 97.1 | – | – | – | – |
| [34] | – / – / 1 | – | – | – | Segmentation by VGG-16 based DL model | 3 | Hot spot, hot substring, hot string | n.a. |
| [35] | test: 318 / – / 1 train: 1304 / – / 1 | – | – | – | DL classification (VGG-16) of entire video frame | 1 | Binary | 75.0 |
| [20] | test: 77 / – / 3 train: 306 / – / 3 | – | – | – | DL classification (MobileNet, VGG-16) | 4 | Sh, Sp, Mp, Cs+ (see fig. 11) | 89.5[†] |

*3) Validation Metrics:* We evaluate Mask R-CNN in terms of F1-score and average precision (AP) metric from the MS COCO benchmark [41]. To this end, all pairs of predicted and ground truth module bounding boxes in a validation frame are formed and the intersection over union (IoU) of each pair is computed. Pairs with an IoU larger than a specified threshold are true positives (TP). False positives (FP) are predictions not matched with any ground truth box and false negatives (FN) ground truths without predictions. From this, precision $TP/(TP + FP)$, recall $TP/(TP+FN)$ and F1-score $2TP/(2TP+FP+FN)$ are computed at ten IoU thresholds $\{0.5, 0.55, \ldots, 0.95\}$. AP is the area under the resulting precision recall curve. Finally, F1-score and AP are averaged over all validation frames.

*4) Results:* After fine-tuning Mask R-CNN achieves an AP of 90.01 % and an F1-score of 90.51 %. At IoU threshold 0.5 the AP and F1-score are 99.55 % and 98.92 %, respectively. This very good segmentation accuracy allows us to skip any additional filtering and post-processing of the segmentations. Later, in sec. V-B and V-C we will analyze how Mask R-CNN generalizes to different PV plants and how segmentation errors affect the PV module extraction.

## D. Extraction of Module Patches

This step extracts segmented PV modules from the thermographic frames and stores the resulting patches to disk. Due to perspective distortion and irregular shape of the segmentation masks direct cropping and storing is not possible. Instead, we fit a minimum-perimeter enclosing quadrilateral to each segmentation mask and obtain a homography which maps the quadrilateral to a rectangle. Width and height of this rectangle correspond to the maximum width and height of the quadrilateral. This yields variable-sized patches which retain most of the information of the source frame without wasting storage space. To ensure each pixel within the quadrilateral is valid we restrict it to lie within the frame. If the IoU of a segmentation mask and the fitted quadrilateral is below 0.9 the segmentation mask is most likely incorrect and filtered out.
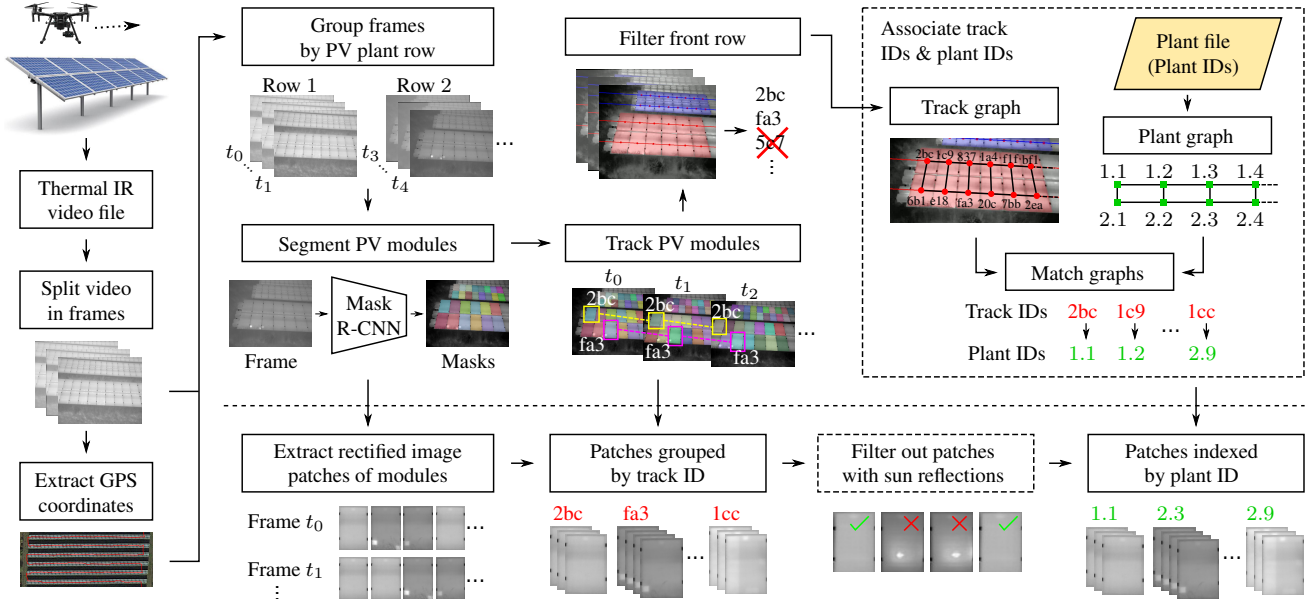
Figure 3: Overview of our tool for semi-automatic extraction of PV modules from thermographic videos.
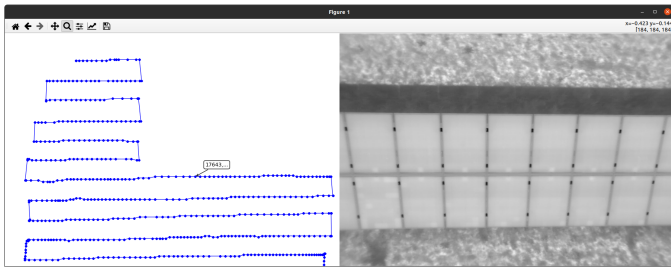


Figure 4: Graphical tool for associating frames with PV plant rows.
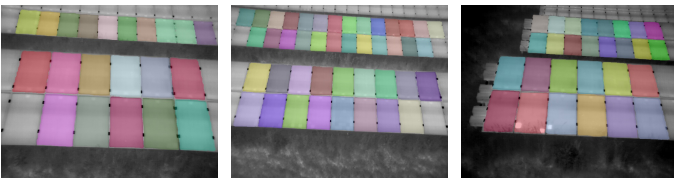


Figure 5: Example results of the PV module segmentation with Mask R-CNN.

### E. PV Module Tracking

Multiple object tracking is performed to associate segmentation masks of the same PV module in subsequent video frames. This enables grouping of the extracted patches by their associated PV module. To this end, mask centers are projected from frame $t - 1$ into frame $t$ using a homography that is estimated by extracting and matching ORB keypoints [42] in both frames. We also tried a Kanade–Lucas–Tomasi tracker but found that it fails due to large motion magnitude whenever the IR camera recalibrates. Each projected mask center is then matched with the nearest segmentation mask center in frame $t$ and its track ID is propagated. If multiple projected mask centers are matched with the same segmentation mask center only the match with the smallest Euclidean distance is considered. The other matches

typically correspond to PV modules that left the frame. Whenever a segmentation mask center in frame $t$ is not matched with any of the projected mask centers, a new unique and random track ID is assigned to it. This usually occurs when a new PV module enters the frame.

### F. Filtering of the Front Row

For low camera angles additional rows of PV modules may be visible in the background of the frame. We develop a filter which discards these background rows and the corresponding patches. It operates independently on each frame and assumes that the currently processed row is the frontmost row (for nadiral videos the bottommost row) in the frame.

The filter iteratively fits a line into the set of segmentation mask centers using RANSAC, removes the inlier mask centers and repeats until no more lines can be fit. Each line must deviate at most ±20° from the horizontal. During iterative fitting outlier lines can occur which intersect the other lines. We remove them by iteratively removing the line which intersects most other lines until no more intersecting lines are present. Given the number $N$ of vertically stacked PV modules in each row we can retrieve the $N$ lines with largest $y$-intercept (the image $y$-axis points downward). The segmentation masks associated with these lines represent the front row and thus are the ones of interest for the further processing steps. Fig. 6 shows some example outputs of the row filter.
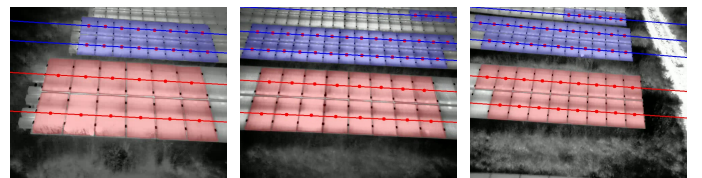


Figure 6: Result of the front row filtering. Segmentation masks in the front row are colored red, all others blue.

## G. Association of Track IDs and Plant IDs

In this step the random track IDs of PV modules are mapped to plant IDs which encode the electrical wiring of the modules and their location in the plant. The algorithm involves three steps: i) track graph creation, ii) plant graph creation and iii) graph matching.

Both track graph and plant graph encode the spatial relation of all PV modules in a single row of the PV plant. Nodes contain the track IDs and plant IDs, respectively. Edges connect IDs of adjacent modules.

*1) Track Graph Creation:* The track graph is built iteratively based on all frames associated with the row. For each new frame previously unseen track IDs are added as nodes to the track graph. However, track IDs of spurious tracks (track ID occurring in less than five successive frames) are ignored. Edges are added whenever the overlap, i.e. the number of shared pixels, of two segmentation masks exceeds a threshold. Prior to that all masks are dilated to ensure sufficient overlaps. For PV plants with gaps between module tables adjacent modules are found by additionally searching along a horizontal line passing through the segmentation mask center. In the end, all but the largest connected component of the track graph are removed. The smaller components correspond to background rows resulting from occasional row filtering failures. Additionally, nodes with degree 1 are removed since they correspond to spurious detections.

*2) Plant Graph Creation:* Plant graphs are created as one-to-one mappings of the rows in the *plant file* which contain plant IDs and correspond directly to the plant layout.

*3) Graph Matching:* The final mapping between plant IDs and track IDs of a row is obtained by finding all isomorphisms of the two graphs and selecting the one compatible with a provided seed match between the track ID and plant ID of the bottom left module in the row. The plant ID of this module is provided by the user in an earlier step. Its track ID is found by searching for the bottom left module in the first or last frame of the row using the multi-line fitting approach from above. Whether the first or last frame is used depends on the scan direction (leftward or rightward) which is estimated from the horizontal motion of the tracked modules. As the track graph can contain imperfections an isomorphism can not always be found and instead a subgraph isomorphism is computed. In the seldom case that this also fails the row can not be processed further.

## H. Filtering Patches with Sun Reflections

For some camera angles sun reflections occur which distort the temperature measurement in the thermographic video and the extracted patches (see fig. 7). Due to the non-stationary nature of the reflection typically only a subset of the patches of a given PV module is affected. We need to filter them out to prevent issues in the downstream anomaly classification.

The filter finds the maximum temperature $(T_i)_{i=1,\ldots,N}$ and its coordinates $(x_i, y_i)$ in all $N$ subsequent patches of a module. Patches in which $T_i$ and $(x_i, y_i)$ deviate significantly from a reference value most likely contain a sun reflection and are filtered out. More specifically, patch $i$ is filtered out if $|T_i - \bar{T}| > 5\,\mathrm{K}$ and $\|(x_i - \bar{x}, y_i - \bar{y})\|_2 > 10\,\mathrm{px}$. The reference values $\bar{T}$ and $(\bar{x}, \bar{y})$ are median values computed from a subsequence of the patches which is obtained as follows. First, the discrete difference

$p_{i+1} - p_i$ of the Euclidean norm $p_i = \|(x_i, y_i)\|_2$ is binarized at a threshold of $10\,\mathrm{px}$. All zero-subsequences of $p_i$ which are longer than $0.3N$ are obtained (the longest is used if none exceeds $0.3N$). Finally, the zero-subsequence with the smallest variance of the maximum temperature $T_i$ is selected for computation of the reference values.

Fig. 7 demonstrates the effectiveness of our filter.
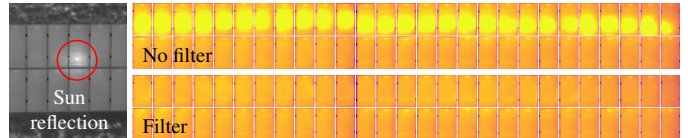


Figure 7: Left: Sun reflection in the thermographic video. Right: Extracted patches of a PV plant row with and without the sun reflection filter.

## V. ANALYSIS OF PV MODULE EXTRACTION

In this section we present the dataset created by our PV module extraction tool and analyze failure cases, processing time and generalization ability.

### A. Extracted Dataset

We run our PV module extraction tool on the seven PV plants in the video dataset and obtain a large-scale dataset with 4.3 million thermographic patches of 107842 PV modules (on average 40.0 patches per module). The tool successfully processed 512 out of the 561 PV plant rows (91.3 %) and extracted 87.8 % of all PV modules from the videos. Tab. 2 shows details of the extracted dataset and success rates. For plants $E$ and $F$ we use the sun reflection filter which removed 52929 and 61923 patches (6.5 % and 22.7 % of the plant total), respectively. The table reports numbers after filtering. Apart from this the same hyper parameters are used for all seven plants indicating a good generalization ability of our extraction tool.

### B. Generalization of the PV Module Segmentation

In this experiment we analyze how well Mask R-CNN generalizes to new PV plants. This is practically relevant as fine-tuning on a new plant is time and cost intensive.

To this end, we create training and validation datasets for PV plants A, B, C and D. Validation uses 25 video frames of each plant, training around 2380 PV modules per plant. Mask R-CNN is trained on all combinations of the training sets and its AP (mean of IoU thresholds $\{0.5, 0.55, \ldots, 0.95\}$) is evaluated on each validation set. Training follows sec. IV-C, however, to speed up the experiment we pretrain and fine-tune for at most 25 epochs each and always select the model with lowest validation loss.

While the results in fig. 8 show an increase in validation AP with more training data, they also indicate that plant C differs significantly from plants A, B and D. This is because PV modules are oriented in landscape in plant C and in portrait in plants A, B and D. We validate this by re-running the experiment without randomly rotating frames by ±90° during training. This leads to a lower AP of 2.1 % to 43.7 % on plant C whenever plant C is not in the training set. Thus, to achieve a high AP Mask R-CNN must be trained on plant C and at least one of the plants A, B or D. At this point we can not fully explain the low sensitivity of AP for plant

Table 2: Numbers of PV modules and patches extracted by our tool from the video dataset.

| Plant | Sector | # Modules | | | | | | # Patches | |
|---|---|---|---|---|---|---|---|---|---|
| | | Total | Extracted | | Failures | | | Extracted | ∅/Module |
| A | S0 | 5280 | 5280 | (100.0 %) | 0 | (0.0 %) | | 205 488 | 38.9 |
| | S1 | 5808 | 5632 | (97.0 %) | 176 | (3.0 %) | | 219 653 | 39.0 |
| | S2 | 3564 | 3300 | (92.6 %) | 264 | (7.4 %) | | 120 100 | 36.4 |
| | S3 | 12 760 | 11 148 | (87.4 %) | 1612 | (12.6 %) | | 430 359 | 38.6 |
| | Total | 27 412 | 25 360 | (92.5 %) | 2052 | (7.5 %) | | 975 600 | 38.5 |
| B | S0 | 9297 | 9020 | (97.0 %) | 277 | (3.0 %) | | 232 973 | 25.8 |
| | S1 | 10 990 | 10 529 | (95.8 %) | 461 | (4.2 %) | | 370 440 | 35.2 |
| | S2 | 11 478 | 10 974 | (95.6 %) | 504 | (4.4 %) | | 364 750 | 33.2 |
| | Total | 31 765 | 30 523 | (96.1 %) | 1242 | (3.9 %) | | 968 163 | 31.7 |
| C | | 2850 | 2850 | (100.0 %) | 0 | (0.0 %) | | 154 476 | 54.2 |
| D | | 3510 | 2115 | (60.3 %) | 1395 | (39.7 %) | | 128 461 | 60.7 |
| E | | 14 688 | 14 679 | (99.9 %) | 9 | (0.1 %) | | 766 901 | 52.2 |
| F | | 7280 | 5015 | (68.9 %) | 2265 | (31.1 %) | | 211 454 | 42.2 |
| G | | 35 360 | 27 300 | (77.2 %) | 8060 | (22.8 %) | | 1 107 711 | 40.6 |
| Total | | 122 865 | 107 842 | (87.8 %) | 15 023 | (12.2 %) | | 4 312 766 | 40.0 |

D to the training data. We assume distinctive visual features of the PV modules, such as clear boundaries, simplify segmentation.

Fig. 8 also reports the mean and standard deviation of all APs when training on one, two, three and four PV plants, respectively. While the standard deviation decreases the mean of the AP increases with more training data. As the AP asymptotically approaches a saturation value the benefit of adding more training data decreases. We found a segmentation model trained on at least three PV plants (of which one is plant C) achieves good results.
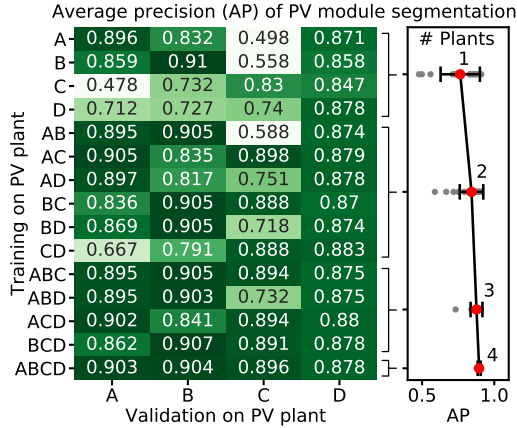


Figure 8: Average precision of the PV module segmentation for all combinations of training sets from PV plants A, B, C and D. The plot on the right shows the mean and standard deviation of the AP when using training data from one, two, three and all four PV plants, respectively.

## C. Failure Cases

Previously, we reported that our tool fails to process 49 out of 561 PV plant rows in our video dataset corresponding to 12.2 % of all PV modules. We identify four common causes: (1) the UAV flight path violates the requirements from sec. IV-A, (2) the PV module segmentation can fail, (3) rows have an irregular layout and (4) the row filtering can fail. Fig. 9 shows examples for each failure and tab. 3 contains the relative frequencies. We report missed rows instead of missed modules because rows contain varying numbers of modules and an error in a single frame usually leads to loss of the entire row.
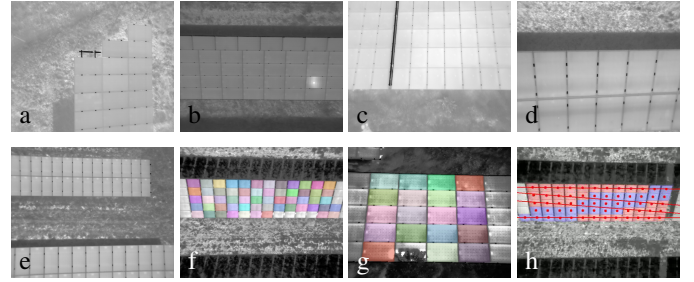


Figure 9: Failure cases of our tool: (a, b) Irregular row layout. (c, d, e) Inadequate UAV trajectory. (f, g) Segmentation error. (h) Row filtering error.

Table 3: Numbers of PV plant rows which our extraction tool failed to process.

| Failure Cause | Plant A | B | C | D | E | F | G | All plants |
|---|---|---|---|---|---|---|---|---|
| UAV trajectory | 9 | 2 | 0 | 4 | 2 | 5 | 0 | 22 |
| Segmentation error | 0 | 1 | 0 | 0 | 0 | 3 | 10 | 14 |
| Irregular row layout | 0 | 4 | 0 | 0 | 0 | 2 | 0 | 6 |
| Row filter error | 0 | 0 | 0 | 0 | 0 | 2 | 4 | 6 |
| Track graph error | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 |
| All failure causes | 9 | 8 | 0 | 4 | 2 | 12 | 14 | 49 |

The majority of rows (22 out of 49) can not be processed due to an inadequate UAV trajectory. This is because some older videos in our dataset were acquired before we established the requirements on the UAV trajectory. Another 14 rows are missed due to false negatives of the PV module segmentation. They occur mostly in plants $F$ and $G$ on which Mask R-CNN is not fine-tuned and which contain PV modules in landscape orientation. In a few cases segmentation also fails due to sun reflections or occlusion of modules by vegetation. Fine-tuning Mask R-CNN on more data can mitigate segmentation failures. Irregular row layouts cause failures in six rows. While our tool can handle missing modules

some failures still occur because Mask R-CNN fills gaps in the grid of modules. Further six rows are missed due to failures of the front row filter. They occur only for plants $F$ and $G$ and are related to the lower module segmentation accuracy. A more robust line-fitting method can solve this issue.

For now we tolerate these failures as our extracted dataset is large enough for downstream tasks.

### D. Timing Analysis

Processing time is a critical factor for scaling our tool to larger PV plants. Fig. 10 reports timings of both manual and automatic steps of our tool. Automatic steps are timed on a workstation with an Intel Core i9-9900K, 64 GB of DDR4 RAM, a 4 TB Seagate IronWolf HDD and a GeForce RTX 2080 Ti running Ubuntu 20.04 LTS. Manual steps comprise of UAV flight, frame grouping and plant file creation. The flight duration is estimated from the number of video frames and the frame rate. This underestimates the true duration slightly as battery changes and row changes of the UAV are not considered. For the manual frame grouping we estimate that the user can configure 30 groups per hour. Due to a lack of accurate measurements fig. 10 omits manual plant file creation. It takes 2 to 8 hours for a 3 MW$_p$ plant (10000 modules) depending on the regularity of its layout.

Timing differences between the plants are due to different video file formats, different plant and row layouts and different UAV flight altitudes and velocities. Track graph creation is faster for plants A, B and C because we can deactivate gap handling. In total, extracting 10000 modules from a 3 MW$_p$ plant takes 8 to 21.7 hours, depending on the plant layout. In here, automatic steps account for 3.8 to 12.1 hours which could be significantly reduced by parallelizing the currently sequential processing of PV plant rows. A further speedup is possible by increasing UAV flight velocity and altitude.
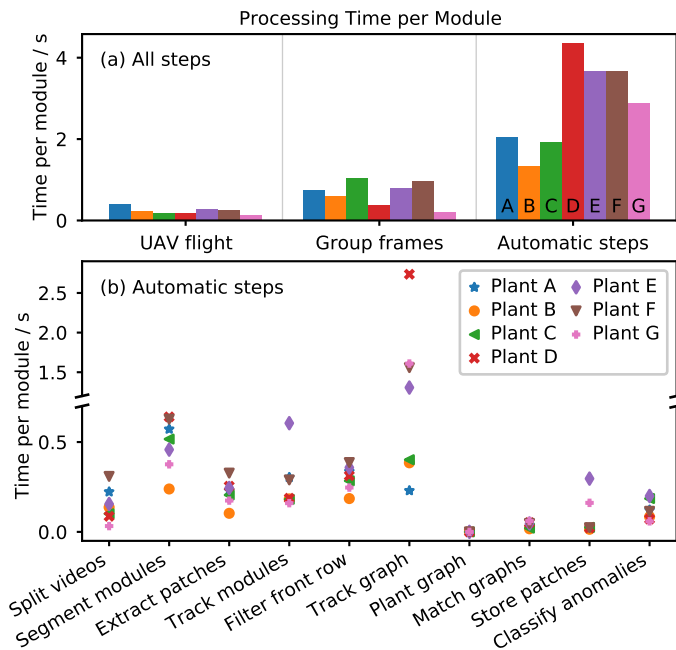


Figure 10: Time needed by our tool to process one PV module. (a) Compares manual and automatic steps. (b) Time distribution of the automatic steps.

## VI. Thermal Anomaly Classification

In this section we use the extracted thermographic patches for supervised classification of thermal anomalies in PV modules. To this end, we label the patches and train a ResNet-50 classifier to predict whether a patch is nominal or exhibits one of ten common anomalies. As our dataset contains on average 40 patches per PV module, we choose the majority class across those patches as the final class label for each module.

### A. Dataset

An expert in our group labels each of the PV modules in our thermographic patch dataset with one out of the ten thermal anomaly classes shown in fig. 11. The class scheme is based on experience and includes relevant module anomalies encountered in previous studies. It is deliberately not optimized for machine learning as the intention is to see how closely the classification of an expert can be reproduced. The structure of our dataset allows to label modules instead of individual patches which speeds up labelling. Note, that we ignore modules of plant D because they are thin-film modules which exhibit different thermal anomalies than the crystalline silicon modules in the other plants. We further exclude all patches with sun reflections from the anomaly dataset and ignore sectors S1 and S2 of plant B to reduce the labelling workload. To reduce class imbalance (only 6.91 % of all modules are anomalous) we balance the numbers of healthy and anomalous modules separately for each plant. Finally, we select 70 % of the PV modules for training, 20 % for testing and 10 % for validation. By splitting the data on module-level we ensure that patches of the same module do not occur in multiple splits. The resulting classification dataset (see tab. 4) contains 453511 patches of 11644 PV modules half of which are anomalous. There are on average 38.95 patches per module which act as different augmented views. Note, that the distribution of anomalies differs significantly between the PV plants.



Figure 11: Example patches for the ten anomaly classes in our dataset. Severity decreases from left to right and top to bottom. Temperature ranges from 30 °C (black) to 60 °C (white). All patches except for class Cm+ are taken from plant A.

Table 4: Class distributions of modules and thermographic patches in our anomaly classification dataset.

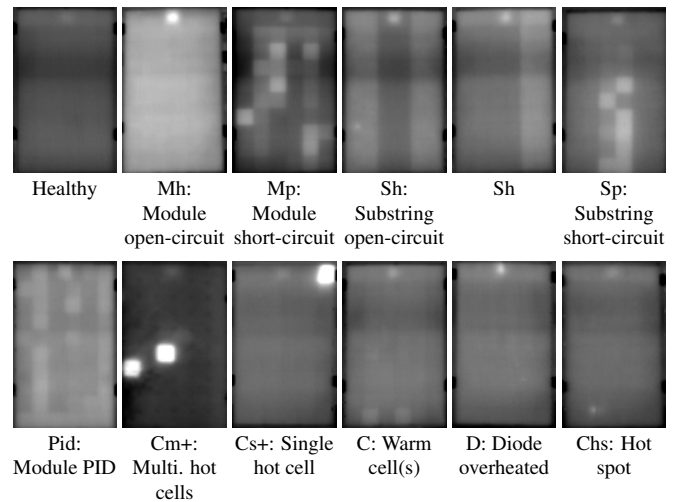| Class | # Modules | | | | | | | # Patches | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Plant A | B | C | E | F | G | All plants | A | B | C | E | F | G | All plants |
| Mh | 5 | 87 | 4 | 0 | 1 | 494 | 591 | 212 | 2636 | 112 | 0 | 38 | 19 968 | 22 966 |
| Mp | 2 | 0 | 2 | 5 | 1 | 1 | 11 | 74 | 0 | 151 | 272 | 62 | 26 | 585 |
| Sh | 61 | 31 | 1 | 1 | 1 | 4 | 99 | 2421 | 804 | 43 | 73 | 13 | 145 | 3499 |
| Sp | 9 | 5 | 0 | 33 | 5 | 37 | 89 | 360 | 118 | 0 | 1802 | 217 | 1573 | 4070 |
| Pid | 980 | 341 | 0 | 0 | 0 | 0 | 1321 | 40 422 | 9143 | 0 | 0 | 0 | 0 | 49 565 |
| Cm+ | 1 | 10 | 0 | 11 | 6 | 0 | 28 | 26 | 243 | 0 | 477 | 352 | 0 | 1098 |
| Cs+ | 12 | 25 | 0 | 11 | 27 | 0 | 75 | 468 | 742 | 0 | 582 | 1348 | 0 | 3140 |
| C | 902 | 184 | 0 | 229 | 570 | 6 | 1891 | 36 955 | 4630 | 0 | 11 618 | 23 539 | 256 | 76 998 |
| D | 608 | 1 | 0 | 0 | 3 | 1024 | 1636 | 24 891 | 26 | 0 | 0 | 197 | 41 210 | 66 324 |
| Chs | 51 | 17 | 0 | 6 | 1 | 6 | 81 | 1957 | 465 | 0 | 350 | 75 | 205 | 3052 |
| Healthy | 2631 | 701 | 7 | 296 | 615 | 1572 | 5822 | 100 725 | 17 960 | 302 | 15 129 | 25 839 | 62 259 | 222 214 |
| All classes | 5262 | 1402 | 14 | 592 | 1230 | 3144 | 11 644 | 208 511 | 36 767 | 608 | 30 303 | 51 680 | 125 642 | 453 511 |

## B. Classifier Training

We initialize ResNet-50 with ImageNet 1.4M pretrained weights and replace the original fully connected (FC) classification layer with a randomly initialized FC layer containing 11 neurons. We fix the base model and train only the FC layer for 10 epochs using Adam optimizer with learning rate 0.001 and batch size 32. Afterwards, we fine-tune all layers starting from layer 101 for another 20 epochs using RMSprop optimizer with learning rate $1e-5$. During training patches are augmented by random left-right and up-down flips. Preprocessing is similar to the one for segmentation (see sec. IV-C), however histogram equalization is skipped and patches are resized to $224 \times 224$ pixels without any padding and without maintaining the aspect ratio. During training we do not address class imbalance explicitly.

## C. Results

*1) Validation Metrics:* The ResNet-50 classifier is evaluated on the test set by means of accuracy and per-class F1-scores averaged over all classes. Both the unweighted average and the average weighted by class support are reported. We further distinguish patch-level and module-level metrics which are obtained before and after majority voting, respectively. For all metrics we report mean and standard deviation over three training runs.

*2) Test Performance:* After fine-tuning ResNet-50 achieves 89.40 % test accuracy on patch-level (see tab. 5). Majority voting improves it to 90.91 %. The results are stable over three independent training runs. Training the classifier only on the first patch of each module instead of all patches reduces test accuracy by 5.4 %. This confirms the benefit of collecting multiple patches per PV module.

Table 5: Test performance of the ResNet-50 classifier on patch- and module-level versus a baseline using only a single patch per PV module.

| | Accuracy | Unweighted F1-score | Weighted F1-score |
|---|---|---|---|
| Single patch | 84.00 ± 0.52 | 58.15 ± 0.64 | 83.38 ± 0.55 |
| Patch-level | 89.40 ± 0.17 | 68.73 ± 1.06 | 89.18 ± 0.15 |
| Module-level | 90.91 ± 0.23 | 70.15 ± 1.98 | 90.68 ± 0.24 |

As can be seen from the per-class metrics in tab. 6 and the confusion matrix in fig. 12 the classifier performs well on most anomaly classes, however is less accurate on classes Mp, Cm+,

Cs+ and Chs. Reason for this is the under-representation of these classes in our dataset leading to poor generalization from training to test set. Other low-resource classes, such as Sh and Sp, are classified more accurately because the underlying visual patterns are less variable and can be learned accurately from a small number of patches. In some cases, the classifier confuses classes C and D with the healthy modules due to high visual similarity of these classes. Similarly, Pid and C are confused. This is because some Pid modules have comparably little overheated cells and some C modules comparably many of them leading to overlap of the two classes. High visual similarity between some classes also makes labelling difficult and may be a source for considerable amount of noise in the ground truth labels.



Figure 12: Module-level confusion matrix of the ResNet-50 classifier on the test set. Values are obtained from the first out of three training runs.

*3) Classifier Visualization:* To understand if the classifier bases its predictions on meaningful features of the patches we compute class activations maps (CAMs). Fig. 13 shows a selection of CAMs. Each CAM visualizes the contribution of a particular image region to the classifier's final prediction. The high correlation between CAMs and temperature anomalies indicates that the classifier draws its confidence mainly from the hot regions in the patch. This is sensible and confirms that the high accuracy of the classifier is based on meaningful image features.

To gain additional insight into the classifier we visualize embeddings of the test set patches in fig. 14. A few large clusters can

Table 6: Per-class module-level metrics of the ResNet-50 classifier on the test set. Shown are mean and standard deviation over three training runs.

| Class | Precision | Recall | F1-score | # Patches |
|---|---|---|---|---|
| Healthy | $95.35 \pm 0.21$ | $96.31 \pm 0.19$ | $95.83 \pm 0.16$ | 1164 |
| Mh | $98.83 \pm 0.42$ | $95.76 \pm 1.38$ | $97.27 \pm 0.90$ | 118 |
| Mp | $66.67 \pm 47.14$ | $33.33 \pm 23.57$ | $44.45 \pm 31.43$ | 2 |
| Sh | $100.00 \pm 0.00$ | $87.72 \pm 2.48$ | $93.44 \pm 1.42$ | 19 |
| Sp | $83.30 \pm 0.76$ | $88.24 \pm 4.81$ | $85.65 \pm 2.67$ | 17 |
| Pid | $86.59 \pm 1.75$ | $83.71 \pm 0.54$ | $85.12 \pm 0.75$ | 264 |
| Cm+ | $33.33 \pm 23.57$ | $13.33 \pm 9.43$ | $19.05 \pm 13.47$ | 5 |
| Cs+ | $57.41 \pm 6.93$ | $28.89 \pm 3.14$ | $38.18 \pm 2.81$ | 15 |
| C | $80.39 \pm 0.26$ | $83.16 \pm 1.75$ | $81.74 \pm 0.97$ | 378 |
| D | $90.06 \pm 0.55$ | $92.35 \pm 0.43$ | $91.19 \pm 0.35$ | 327 |
| Chs | $57.07 \pm 7.04$ | $31.25 \pm 5.10$ | $39.75 \pm 3.42$ | 16 |
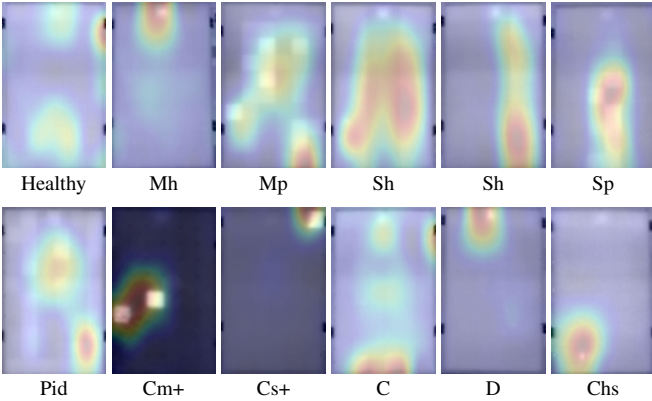


Figure 13: Class activation maps of the ResNet-50 classifier obtained with Grad-CAM++ [43]. The patches correspond to fig. 11.



Figure 14: ResNet-50 embeddings of the test dataset after dimensionality reduction with UMAP [44]. Embeddings are obtained from the last convolutional layer. Colors represent the ground truth class. For better visualization we show only 5 % of all data points.

be observed which correspond to the six PV plants and most of the anomaly classes. For plant A there are two clusters each because modules in the top row are rotated by 180° as compared to those in the bottom row. In addition, several smaller clusters occur which correspond to individual PV modules. Some of them are outliers, others represent classes, such as Cs+ and Sp, which do not form compact clusters due to low sample count and high intra-class variance. The embedding space reflects the classifier's confusion of some classes, e.g. Pid/C and C/D/Healthy, as partial overlap of the respective clusters. Similarly, the low accuracy of some classes, such as Cm+ and Chs, can be explained by the almost complete overlap of the respective clusters with other clusters.

## VII. Discussion and Conclusion

*1) Summary:* In this work, we developed a computer vision tool for semi-automatic processing of UAV thermographic videos. It handles the large amounts of thermographic images acquired during inspection of PV plants, extracts individual PV modules and classifies ten common module anomalies with an accuracy of more than 90 % using a ResNet-50 classifier. It further provides the exact location of defective modules in a plant allowing for targeted repairs. Videos are used instead of single images for faster inspection and increased flexibility of UAV operation. Our tool can be used for automated inspection of PV plants superseding an expensive and time-consuming manual inspection. This can reduce cost of PV plant maintenance, ensures safe operation and maximizes yield.

Furthermore, our tool efficiently creates large-scale thermographic datasets by exploiting redundancy in the video. We use this capability to curate a dataset with 4.3 million thermographic images of 107842 PV modules from seven PV plants. Modules in the dataset are automatically indexed based on their electrical wiring and location in the plant. This unique index and the large size of the dataset enable research on other downstream machine learning tasks, such as power prediction, which are essential for the safe and profitable operation of future PV plants of ever-growing size.

*2) State-of-the-art Improvements:* As compared to many of the related works we use deep learning for PV module detection which improves accuracy and generalization. No hyper parameters had to be adjusted to extract modules from the seven different PV plants. By using a deep convolutional classifier for supervised classification of thermal anomalies we followed a recent trend in the field. However, our dataset is significantly larger and we distinguish ten anomaly classes as opposed to at most four classes in the related works. Distinguishing many anomaly classes is not only of value for research datasets but also for plant operators as it facilitates more detailed cataloguing of anomalies in a plant. This is important because some anomalies can worsen over time eventually causing power losses or outages. Despite the larger number of classes test accuracy of our classifier is on par with the related works. However, we also found that classification accuracy is lower for some under-represented classes in our dataset which confirms the need for very large datasets. This also shows that large-scale datasets are required to detect rare anomalies which affect only a handful out of thousands of modules. Smaller datasets as used in many related works do not sufficiently cover such rare anomalies. To allow for even more accurate and fine-grained classification in future we will expand our dataset and explore other deep learning methods which overcome the issue of low accuracy on under-represented classes.

*3) Future Relevance:* Our work is a first step towards the ultimate goal of automatically characterizing gigawatt-scale PV

plants with millions of modules in a day. It shows a way to organize and process the large amounts of data accrued during inspection. However, to achieve full automation and scale up to gigawatt plants multiple UAVs should be used and UAV operation has to be automated. This leads to a predictable scanning order of plant rows which renders most of the manual steps of our tool unnecessary. Scaling up also requires reducing processing time. Given full automation, the worst case throughput of our tool is 19800 modules per day on a single workstation. To process 3.5 million modules in a $1\,GW_p$ plant in a day requires a 177-fold speedup. This speedup is practically feasible by parallelizing the currently sequential processing of PV plant rows. While this demands for a parallel implementation on a small compute cluster it does not require principle changes to the vision algorithms.

*4) Future Challenges:* Some challenges remain for future works. For example, the detection of string-level anomalies or faults of non-module components, such as inverters. To this end, multimodal datasets (imagery and electrical) as produced by our tool can be used in combination with machine learning. Future work should also consider additional image sources, such as visual and electroluminescence imagery. For wider applicability anomaly classification could be extended to thin-film, bifacial and half-cell modules, and PV module extraction to plants with non-row layouts, as common in floating PV. Furthermore, methods are needed which predict the PV plant's future health state based on historic data. Finally, the dependency of the anomaly classification on ambient conditions should be explored. We have indications for such a dependency but not yet enough data for a systematic analysis.

## VIII. ACKNOWLEDGEMENTS

## REFERENCES

[1] "Renewable energy - BP statistical review of world energy 2020," 2020.

[2] C. Buerhop, D. Schlegel, M. Niess, C. Vodermayer, R. Weißmann, and C. Brabec, "Reliability of IR-imaging of PV-plants under operating conditions," *Solar Energy Materials and Solar Cells*, vol. 107, pp. 154 – 164, 2012.

[3] C. Buerhop, R. Weißmann, H. Scheuerpflug, R. Auer, and C. Brabec, "Quality control of PV-modules in the field using a remote-controlled drone with an infrared camera," in *European Photovoltaic Solar Energy Conference and Exhibition (EUPVSEC)*, 2012.

[4] H. Scheuerpflug and C. Buerhop-Lutz, "Field inspection of PV-modules using aerial, drone-mounted thermography," in *European Photovoltaic Solar Energy Conference and Exhibition (EUPVSEC)*, 2014.

[5] P. B. Quater, F. Grimaccia, S. Leva, M. Mussetta, and M. Aghaei, "Light unmanned aerial vehicles (UAVs) for cooperative inspection of PV plants," *IEEE Journal of Photovoltaics*, vol. 4, no. 4, pp. 1107–1113, 2014.

[6] A. Niccolai, A. Gandelli, F. Grimaccia, R. Zich, and S. Leva, "Overview on photovoltaic inspections procedure by means of unmanned aerial vehicles," in *2019 IEEE Milan PowerTech*, pp. 1–6, 2019.

[7] N. M. Kumar, K. Sudhakar, M. Samykano, and V. Jayaseelan, "On the technologies empowering drones for intelligent monitoring of solar photovoltaic power plants," *Procedia Computer Science*, vol. 133, pp. 585 – 593, 2018. International Conference on Robotics and Smart Manufacturing (ICROSMA 2018).

[8] S. Gallardo-Saavedra, L. Hernández-Callejo, and O. Duque-Perez, "Technological review of the instrumentation used in aerial thermographic inspection of photovoltaic plants," *Renewable and Sustainable Energy Reviews*, vol. 93, pp. 566 – 579, 2018.

[9] F. Bizzarri, S. Nitti, and G. Malgaroli, "The use of drones in the maintenance of photovoltaic fields," *E3S Web of Conferences*, vol. 119, 2019.

[10] S. Gallardo-Saavedra, L. Hernández-Callejo, and O. Duque-Perez, "Image resolution influence in aerial thermographic inspections of photovoltaic plants," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 12, pp. 5678–5686, 2018.

[11] S. Leva, M. Aghaei, and F. Grimaccia, "PV power plant inspection by UAS: Correlation between altitude and detection of defects on PV modules," in *2015 IEEE 15th International Conference on Environment and Electrical Engineering (EEEIC)*, pp. 1921–1926, 2015.

[12] V. Carletti, A. Greco, A. Saggese, and M. Vento, "An intelligent flying system for automatic detection of faults in photovoltaic plants," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, pp. 2027–2040, 2019.

[13] F. Grimaccia, S. Leva, and A. Niccolai, "PV plant digital mapping for modules' defects detection by unmanned aerial vehicles," *IET Renewable Power Generation*, vol. 11, no. 10, pp. 1221–1228, 2017.

[14] G. Francesco, L. Sonia, and N. Alessandro, "A semi-automated method for defect identification in large photovoltaic power plants using unmanned aerial vehicles," in *2018 IEEE Power Energy Society General Meeting (PESGM)*, pp. 1–5, 2018.

[15] A. Niccolai, F. Grimaccia, and S. Leva, "Advanced asset management tools in photovoltaic plant monitoring: UAV-based digital mapping," *Energies*, vol. 12, no. 24, p. 4736, 2019.

[16] A. Arenella, A. Greco, A. Saggese, and M. Vento, "Real time fault detection in photovoltaic cells by cameras on drones," in *International Conference Image Analysis and Recognition (ICIAR)*, pp. 617–625, 2017.

[17] H. Jeong, G.-R. Kwon, and S.-W. Lee, "Deterioration diagnosis of solar module using thermal and visible image processing," *Energies*, vol. 13, no. 11, p. 2856, 2020.

[18] P. Addabbo, A. Angrisano, M. L. Bernardi, G. Gagliarde, A. Mennella, M. Nisi, and S. L. Ullo, "UAV system for photovoltaic plant inspection," *IEEE Aerospace and Electronic Systems Magazine*, vol. 33, no. 8, pp. 58–67, 2018.

[19] S. Dotenco, M. Dalsass, L. Winkler, T. Würzner, C. Brabec, A. Maier, and F. Gallwitz, "Automatic detection and analysis of photovoltaic modules in aerial infrared imagery," in *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1–9, 2016.

[20] C. Dunderdale, W. Brettenny, C. Clohessy, and E. E. van Dyk, "Photovoltaic defect classification through thermal infrared imaging using a machine learning approach," *Progress in Photovoltaics: Research and Applications*, vol. 28, no. 3, pp. 177–188, 2020.

[21] J. J. Vega Díaz, M. Vlaminck, D. Lefkaditis, S. A. Orjuela Vargas, and H. Luong, "Solar panel detection within complex backgrounds using thermal images acquired by UAVs," *Sensors*, vol. 20, no. 21, p. 6219, 2020.

[22] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 2980–2988, 2017.

[23] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

[24] D. Kim, J. Youn, and C. Kim, "Automatic photovoltaic panel area extraction from UAV thermal infrared images," *Journal of the Korean Society of Surveying, Geodesy, Photogrammetry and Cartography*, vol. 34, pp. 559–568, 2016.

[25] D. Kim, J. Youn, and C. Kim, "Automatic fault recognition of photovoltaic modules based on statistical analysis of UAV thermography," *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XLII-2/W6, pp. 179–182, 2017.

[26] M. Aghaei, S. Leva, and F. Grimaccia, "PV power plant inspection by image mosaicing techniques for IR real-time images," in *2016 IEEE 43rd Photovoltaic Specialists Conference (PVSC)*, pp. 3100–3105, 2016.

[27] F. Wu, D. Zhang, X. Li, X. Luo, J. Wang, W. Yan, Z. Chen, and Q. Yang, "Aerial image recognition and matching for inspection of large-scale photovoltaic farms," in *2017 International Smart Cities Conference (ISC2)*, pp. 1–6, 2017.

[28] H. Zhang, X. Hong, S. Zhou, and Q. Wang, "Infrared image segmentation for photovoltaic panels based on Res-UNet," in *Pattern Recognition and Computer Vision*, pp. 611–622, 2019.

[29] A. Greco, C. Pironti, A. Saggese, M. Vento, and V. Vigilante, "A deep learning based approach for detecting panels in photovoltaic plants," in *Proceedings of the 3rd International Conference on Applications of Intelligent Systems*, APPIS 2020, 2020.

[30] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pp. 234–241, 2015.

[31] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 779–788, 2016.

[32] M. Aghaei, F. Grimaccia, C. A. Gonano, and S. Leva, "Innovative automated control system for PV fields inspection and remote control," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 11, pp. 7287–7296, 2015.

[33] M. Alsafasfeh, I. Abdel-Qader, B. Bazuin, Q. Alsafasfeh, and W. Su, "Unsupervised fault detection and analysis for large photovoltaic systems using drones and machine vision," *Energies*, vol. 11, no. 9, p. 2252, 2018.

[34] A. Oliveira, M. Aghaei, and R. Rüther, "Automatic fault detection of photovoltaic arrays by convolutional neural networks during aerial infrared thermography," in *European Photovoltaic Solar Energy Conference and Exhibition (EUPVSEC)*, 2019.

[35] R. Pierdicca, E. Malinverni, F. Piccinini, M. Paolanti, A. Felicetti, and P. Zingaretti, "Deep convolutional neural network for automatic detection of damaged photovoltaic cells," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences (ISPRS)*, vol. XLII-2, pp. 893–900, 2018.

[36] D. H. Lee and J. H. Park, "Developing inspection methodology of solar energy plants by thermal infrared sensor on board unmanned aerial vehicles," *Energies*, vol. 12, no. 15, p. 2928, 2019.

[37] Y. Zefri, A. ElKettani, I. Sebari, and S. A. Lamallam, "Thermal infrared and visual inspection of photovoltaic installations by UAV photogrammetry—application case: Morocco," *Drones*, vol. 2, no. 4, p. 41, 2018.

[38] I. J. Tsanakas, L. Ha, and F. Al Shakarchi, "Advanced inspection of photovoltaic installations by aerial triangulation and terrestrial georeferencing of thermal/visual imagery," *Renewable Energy*, vol. 102 (Part A), p. 224–233, 2016.

[39] M. Nisi, F. Menichetti, V. Bramante, T. Tr, B. Muhammad, and R. Prasad, "EGNSS high accuracy system improving photovoltaic plant maintenance using RPAS integrated with low-cost RTK receiver," in *Global Wireless Summit*, 2016.

[40] C. Henry, S. Poudel, S.-W. Lee, and H. Jeong, "Automatic detection system of deteriorated PV modules using drone with thermal camera," *Applied Sciences*, vol. 10, p. 3802, 2020.

[41] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *European Conference on Computer Vision (ECCV)*, pp. 740–755, 2014.

[42] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: an efficient alternative to SIFT or SURF," in *2011 IEEE International Conference on Computer Vision (ICCV)*, pp. 2564–2571, 2011.

[43] A. Chattopadhay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, "Grad-CAM++: Generalized gradient-based visual explanations for deep convolutional networks," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 839–847, 2018.

[44] L. McInnes, J. Healy, N. Saul, and L. Grossberger, "UMAP: Uniform manifold approximation and projection," *Journal of Open Source Software*, vol. 3, p. 861, 2018.

# Anomaly Detection in IR Images of PV Modules using Supervised Contrastive Learning

Lukas Bommes[1], Mathis Hoffmann[2,3], Claudia Buerhop-Lutz[1], Tobias Pickel[1], Jens Hauch[1], Christoph Brabec[1,3], Andreas Maier[2], and Ian Marius Peters[1]

[1]*Forschungszentrum Jülich GmbH, Helmholtz-Institute Erlangen-Nuremberg for Renewable Energies (HI ERN)*
[2]*Pattern Recognition Lab, Department Informatik, Universität Erlangen-Nürnberg (FAU)*
[3]*Institute Materials for Electronics and Energy Technology, Universität Erlangen-Nürnberg (FAU)*
Correspondence to i.peters@fz-juelich.de

## Abstract

*Increasing deployment of photovoltaic (PV) plants requires methods for automatic detection of faulty PV modules in modalities, such as infrared (IR) images. Recently, deep learning has become popular for this. However, related works typically sample train and test data from the same distribution ignoring the presence of domain shift between data of different PV plants. Instead, we frame fault detection as more realistic unsupervised domain adaptation problem where we train on labelled data of one source PV plant and make predictions on another target plant. We train a ResNet-34 convolutional neural network with a supervised contrastive loss, on top of which we employ a k-nearest neighbor classifier to detect anomalies. Our method achieves a satisfactory area under the receiver operating characteristic (AUROC) of 73.3 % to 96.6 % on nine combinations of four source and target datasets with 2.92 million IR images of which 8.5 % are anomalous. It even outperforms a binary cross-entropy classifier in some cases. With a fixed decision threshold this results in 79.4 % and 77.1 % correctly classified normal and anomalous images, respectively. Most misclassified anomalies are of low severity, such as hot diodes and small hot spots. Our method is insensitive to hyperparameter settings, converges quickly and reliably detects unknown types of anomalies making it well suited for practice. Possible uses are in automatic PV plant inspection systems or to streamline manual labelling of IR datasets by filtering out normal images. Furthermore, our work serves the community with a more realistic view on PV module fault detection using unsupervised domain adaptation to develop more performant methods with favorable generalization capabilities.*

## I. Introduction

Solar photovoltaics (PV) has emerged as an important renewable energy source with a global installed capacity of $627\,\mathrm{GW_p}$ in 2020 [1] that is projected to reach $2840\,\mathrm{GW_p}$ in 2030 [2]. PV modules are prone to defects due to aging, environmental influences or incorrect handling during installation. Defective modules pose safety hazards and reduce power output, yield, and profitability of a PV plant. Thus, regular inspection of PV plants is inevitable. As increasing plant sizes render manual inspection impractical, there is a recent surge in works on automatic inspection tools [3–16], which use computer vision methods to automatically detect defective PV modules in modalities, such as aerial thermographic infrared (IR) images.

The most recent methods frame fault detection as supervised classification and train a deep convolutional neural network with standard cross-entropy loss to classify different types of PV module faults in IR images [3, 14]. These methods achieve a high detection accuracy on the test dataset which is sampled from the same distribution as the training data. However, this setting ignores the fact that data distributions differ between plants, a problem known as domain shift. We find significant domain shift by examining 4.16 million IR images from six different PV plants. Thus, we frame fault detection more realistically as unsupervised domain adaptation. Here, training is performed on labelled IR images of one source PV plant and predictions are made on another target PV plant for which no labels are available. This setting is more realistic as it takes domain shift into account. It is also more practical as training is performed only once, and no subsequent fine-tuning is needed when applying the fault detector to a new PV plant. Another challenge we address is the detection of unknown anomaly types which are present in the target dataset but not in the source dataset. This is generally known as open-set classification.

In this work, we develop a novel PV module anomaly detection method for IR images based on deep learning which addresses the aforementioned challenges. We train a ResNet-34 convolutional neural network [17] with a supervised contrastive loss on labelled IR images of a source plant and use it to extract low-dimensional representations of the images. Based on these representations a $k$-nearest neighbor ($k$-NN) classifier detects anomalies in the target plant. By framing anomaly detection as supervised binary classification we follow a promising recent trend in the field [18–21]. Instead of performing active domain adaptation our method uses contrastive representations which are more informative and less domain-specific than representations learned by the standard cross-entropy loss [22, 23]. This also facilitates generalization beyond the training dataset and thus detection of unknown anomalies.

To summarize, our contributions are as follows:

- We frame PV module fault detection as more realistic unsupervised domain adaptation problem where training is performed on one labelled source plant and anomalies are detected in another target PV plant.
- We introduce a domain-agnostic anomaly detection method based on contrastive representation learning and a binary

$k$-NN classifier which outperforms a binary cross-entropy classifier on some tasks and reliably detects unknown anomalies.

- We validate our method on nine combinations of four source and target datasets containing a total of 2.92 million IR images.

## II. RELATED WORKS

In this section we briefly review related works on contrastive representation learning, domain adaptation, anomaly detection and PV module fault detection in IR images.

### A. Contrastive Representation Learning

Contrastive representation learning is a form of deep metric learning initially proposed by Hadsell et al. [24], which succeeds the older triplet [25] and $N$-pair losses [26]. For a good review see Le-Khac et al. [27]. Contrastive representation learning uses deep neural networks to learn a low-dimensional feature space of high-dimensional data in which semantically similar samples are closer than semantically dissimilar ones. To this end, representations of a set of *positive* samples are attracted and repulsed from the representations of all other (*negative*) samples using for example the InfoNCE [28] or NT-Xent [29] loss. In the conventional self-supervised setting a single sample [28, 30], and optionally perturbed versions of it [29, 31–33], are used as positives. In the supervised setting all samples with the same class label (and optional perturbations) are positives [22, 34, 35]. Self-supervised contrastive representations discriminate individual samples. Supervised contrastive representations on the other hand discriminate classes by learning feature spaces in which samples are clustered based on their class membership. In our work we use contrastive representations because they are more informative than those learned with standard cross-entropy loss which retain only the minimum of information needed to discriminate training samples [22, 23]. This allows to extract discriminative features which are robust against domain shift and generalize to unseen classes.

### B. Domain Adaptation

Domain adaptation addresses the problem of learning transferable representations without the need for large amounts of labelled training data. For a good overview we refer the reader to the surveys by Wang et al. [36] and Zhao et al. [37]. Our problem corresponds to unsupervised domain adaptation where we learn representations on labelled data of a source domain that generalize to an unlabelled target domain. Many domain adaptation methods estimate and minimize the discrepancy between source and target domain by means of loss functions, such as Maximum Mean Discrepancy [38–40], L2- or cosine distance [41, 42], Rényi divergence [43] or KL-divergence [44]. Recently, contrastive losses have been used as well [45–47]. Aligning source and target representations this way improves performance when classifying images [48] or detecting anomalies [49] in the target domain. While our method does not use any domain adaptation loss, it solves the same problem by using more informative and thus less domain-specific contrastive representations.

### C. Anomaly Detection

Anomaly detection (AD) aims at identifying anomalous data samples which deviate from the majority of normal samples. This relates to our dataset which contains mostly normal PV modules and only a small fraction of faulty modules. For a good overview of recent deep learning-based AD methods we refer to the surveys by Pang et al. [50], Bulusu et al. [51] and Chalapathy et al. [52]. Most deep AD methods learn representations of normal data using autoencoders [53, 54], generative adversarial networks [55, 56], one-class losses [57, 58], self-supervised learning [20, 59–61] or metric learning [62, 63] and identify anomalies by a high reconstruction error or a large distance to the normal representations. Recently, (self-)supervised contrastive learning has gained popularity for learning representations for AD [23, 35, 64, 65]. Some works also explored the use of domain adaptation for anomaly detection [49, 66–68].

Many AD methods assume an unlabelled training dataset containing mostly normal samples and a few anomalies. If labelled anomalies are available AD can also be formulated as (semi-)supervised binary classification and achieve state-of-the-art performance [19–21]. Similary, using a supervised $k$-NN classifier on embeddings of a ResNet, which is pretrained on ImageNet with cross-entropy loss, outperforms many other AD methods [18].

Building on this, our work formulates AD as supervised binary classification with a $k$-NN classifier. As opposed to the other works we use contrastive representations and perform anomaly detection in a target domain which differs from the source domain and does not contain any labelled examples.

### D. PV Module Fault Detection

Until recently, PV module faults were detected as hot regions in IR images using classical computer vision algorithms, such as segmentation by intensity thresholding [4–8], iterative growth of segmentation masks [9, 10] or template matching [11]. Downside of these methods is their dependence on heuristics and manual priors, the need for extensive manual tuning and poor generalization to unseen imagery. The extraction of hand-crafted image features and detection of outliers by statistical tests [12, 13] or classification with a SVM or Random Forest [14] is slightly more robust. Recently, deep learning has shown promising results in overcoming the problems of classical algorithms [3, 14–16, 69]. Typically, fault detection is performed as a supervised classification in which deep convolutional networks, such as ResNet, MobileNet [70] or VGG [71], are trained with standard cross-entropy loss to distinguish a predefined set of fault classes. To the best of our knowledge, related works in the field have neither addressed the problem of domain shift nor the detection of unknown anomaly classes.

## III. DATASET

We use an extended version of the dataset from our previous work [3]. It consists of 4.16 million IR images showing 105546 PV modules from six different PV plants, which were acquired under clearsky conditions and solar irradiance above $700\,\mathrm{W\,m^{-2}}$. Note, that we name the PV plants A to G in accordance to our previous work. We omit plant D as it contains thin-film modules instead of crystalline silicon modules like the other plants.

Figure 1 images:

Normal | Mh: Module open-circuit | Mp: Module short-circuit | Sh: Substring open-circuit | Sh | Sp: Substring short-circuit

Pid: Module PID | Cm+: Multi. hot cells | Cs+: Single hot cell | C: Warm cell(s) | D: Diode overheated | Chs: Hot spot
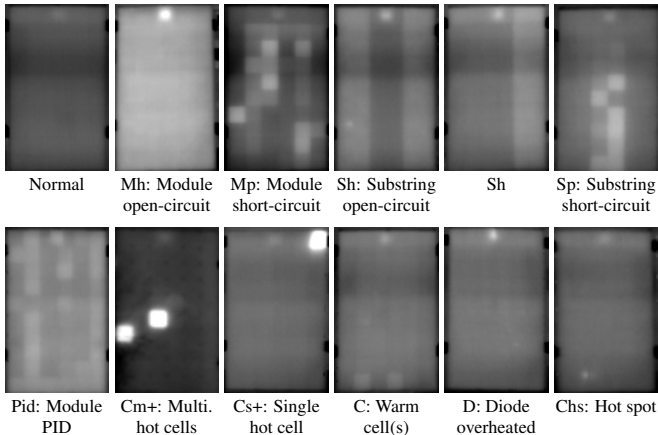
Figure 1: Exemplary IR images of a normal and ten different types of anomalous PV modules in our dataset. Temperature ranges from 30 °C (black) to 60 °C (white). All images except for class Cm+ show plant A. The figure is taken from our previous work [3].

Table 1: Numbers of normal and anomalous IR images in our dataset.

| Class | Plant | | | | | |
|---|---|---|---|---|---|---|
| | A | B | C | E | F | G |
| Normal | 864 394 | 869 957 | 135 342 | 751 261 | 185 613 | 1 043 216 |
| Anomalous | 107 786 | 98 206 | 306 | 15 174 | 25 841 | 63 383 |
| Normal (%) | 88.91 | 89.86 | 99.77 | 98.02 | 87.8 | 94.27 |
| Anomalous (%) | 11.09 | 10.14 | 0.23 | 1.98 | 12.2 | 5.73 |

Images are cropped from IR videos of a drone-mounted DJI Zenmuse XT2 camera and rectified to remove perspective distortion. Due to redundancies in the video, there are on average 39.4 images of each PV module which serve as multiple augmented views. Each image is labelled by an expert either as containing a normal module or a module with one out of the ten typical faults shown in fig. 1. While our method makes only a binary distinction between normal and anomalous modules, fine-grained fault labels are used to evaluate our method.

Tab. 1 and tab. 2 show the distribution of anomaly classes in our dataset. To ensure a realistic setting, we do not balance the numbers of normal and anomalous images. For our experiments we use only data of plants A, B, E and F as plant C contains very few anomalies and ground truth labels of plant G were not obtained by an expert. Each dataset is split each into 70 % train and 30 % test data. Here, we ensure that images of the same PV module do not occur in both train and test set.

Fig. 2 shows UMAP embeddings [72] of our dataset. Here, images form distinctive clusters or *domains* depending on the PV plant they originate from. This domain shift has various reasons, such as differences in ambient conditions, camera position, as well as module and cell type. For most plants, we additionally observe sub-domains which correspond to different rows of vertically stacked modules. Fig. 3 shows an exemplary patch for each plant clearly revealing differences. We also found that different module orientations in the images lead to domain shift. To account for this we rotate all images so that module junction boxes are always at the top edge.

Table 2: Numbers of anomalous IR images per underlying fault class.

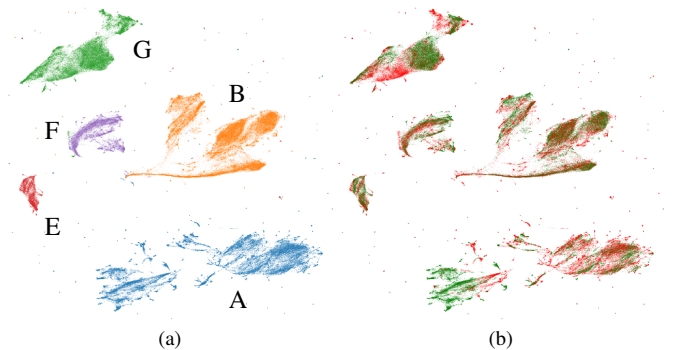| Class | Plant | | | | | |
|---|---|---|---|---|---|---|
| | A | B | C | E | F | G |
| Mh | 212 | 33 129 | 112 | 0 | 38 | 19 968 |
| Mp | 74 | 185 | 151 | 272 | 62 | 26 |
| Sh | 2421 | 2594 | 43 | 73 | 13 | 145 |
| Sp | 360 | 328 | 0 | 1802 | 217 | 1573 |
| Pid | 40 422 | 23 174 | 0 | 0 | 0 | 0 |
| Cm+ | 26 | 388 | 0 | 477 | 352 | 0 |
| Cs+ | 468 | 1651 | 0 | 582 | 1348 | 0 |
| C | 36 955 | 28 174 | 0 | 11 618 | 23 539 | 256 |
| D | 24 891 | 66 | 0 | 0 | 197 | 41 210 |
| Chs | 1957 | 8517 | 0 | 350 | 75 | 205 |

Figure 2 images:

(a)     (b)

Figure 2: Projection of our dataset obtained by UMAP (with 50 neighbors per sample and minimum distance of 0.1). Colors in (a) indicate the PV plant, which reveals the domain shift between different plants. In (b) normal and anomalous samples are colored green and red, respectively. UMAP is applied directly to the flattened images, which are preprocessed as in sec. IV-C.2. For better visualization, normal samples are subsampled to match the number of anomalous samples.

## IV. METHOD

The aim of our method is to predict binary labels $\left\{\hat{y}_i^T\right\}_{i=1\ldots N^T}$ for $N^T$ IR images $\left\{x_i^T\right\}_{i=1\ldots N^T}$ of a *target* PV plant, depending on whether a normal or an anomalous PV module is shown. While we have no labelled examples for this PV plant, we have a set of $N^S$ binary labelled images $\left\{\left(x_i^S, y_i^S\right)\right\}_{i=1\ldots N^S}$ of at least one other *source* PV plant. Typically, there is a domain shift between source and target images and the distribution of anomaly classes between source and target can differ significantly. The target data can even contain unknown anomalies, which are not present in the source data. Our method shown in fig. 4 overcomes

Figure 3 images:

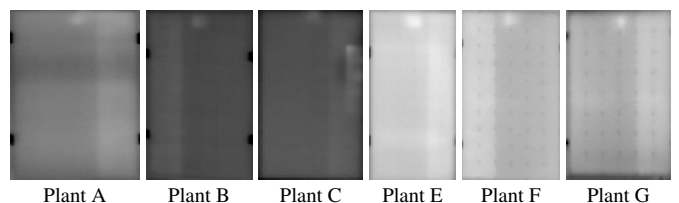Plant A | Plant B | Plant C | Plant E | Plant F | Plant G

Figure 3: IR images differ between the PV plants in our dataset due to different ambient conditions, camera positions, as well as module and cell type. Shown are modules with Sh anomaly. The original aspect ratio is preserved and temperature ranges from 15 °C (black) to 50 °C (white).

these challenges by i) learning informative and domain-agnostic representations with a supervised contrastive loss and ii) detecting unknown anomalies on top of the representations with a $k$-NN classifier.

## A. Supervised Contrastive Representation Learning

As indicated by fig. 2b we observe that IR images form clusters depending on the PV plant they originate from. However, they do not form clusters of normal and anomalous images. We employ representation learning to compute a low-dimensional embedding of the IR images which forms distinctive clusters of normal and anomalous images and reduces clustering by plants. Extraction of low-dimensional embeddings from the high-dimensional IR images is also needed to make anomaly detection computationally tractable. Instead of using hand-crafted features, we employ deep neural networks and a supervised contrastive loss to learn a suitable embedding end-to-end. Specifically, we use a convolutional encoder $f_\theta(\cdot)$ and a fully connected projection head $h_\psi(\cdot)$ to extract a $d$-dimensional embedding vector $v_i^S \in \mathbb{R}^d$ from each source image $x_i^S$

$$v_i^S = h_\psi\big(f_\theta\big(x_i^S\big)\big). \tag{1}$$

Several related works use a projection head to improve representational power of the encoder embeddings [29, 33, 73]. We follow this architecture choice. Note, however that the effect on the encoder embeddings is less relevant in our case as we use the embeddings after the projection head instead for anomaly detection.

After encoding, each embedding vector is normalized to unit L2-norm

$$z_i^S = v_i^S / \big\|v_i^S\big\|_2. \tag{2}$$

Iterative stochastic gradient descent is performed on embeddings of randomly shuffled batches of $N$ labelled source images $\big\{x_i^S, y_i^S\big\}_{i=1\ldots N}$ to compute suitable network parameters

$$\{\theta^*, \psi^*\} = \arg\min_{\theta, \psi} \mathcal{L}_{\text{AD}}\big(z_i^S, y_i^S\big) \tag{3}$$

where $\mathcal{L}_{\text{AD}}$ is a supervised contrastive loss with the following form of a non-parametric softmax classifier [27]

$$\mathcal{L}_{\text{AD}}\big(z_i^S, y_i^S\big) = -\frac{1}{|\mathcal{N}|} \sum_{i \in \mathcal{N}} \log \frac{\exp\big(z_i^S \cdot \bar{z}^S / \tau\big)}{\sum_{j \in \mathcal{N} \cup \mathcal{A}} \exp\big(z_j^S \cdot \bar{z}^S / \tau\big)}. \tag{4}$$

Here, the $\cdot$ symbol denotes the dot product of two vectors and $\tau \in \mathbb{R}^+$ is a scalar temperature hyperparameter as used by Wu et al. [30] and He et al. [32]. We set $\tau = 0.1$ for all experiments. Further, $\mathcal{N}$ and $\mathcal{A}$ denote the indices of all normal and anomalous embeddings in the current batch and $\bar{z}^S \in \mathbb{R}^d$ is the mean vector of all normal embeddings

$$\bar{z}^S = \frac{1}{|\mathcal{N}|} \sum_{i \in \mathcal{N}} z_i^S. \tag{5}$$

This loss is based on the normalized temperature scaled cross-entropy loss [22, 29] and the central contrastive loss [34]. Intuitively, it pulls all normal samples in the batch towards the normal mean vector and pushes the anomalies away. While this causes formation of a single cluster of normal IR images in embedding

space, anomalies can potentially form multiple clusters depending on the underlying anomaly class. Note, that pulling each normal sample towards the normal mean embedding has the same effect as pulling all pairs of normal embeddings towards each other. We use the first variant as it is easier to implement.

## B. Anomaly Detection with a $k$-NN Classifier

The anomaly detection stage predicts for each target image $x_j^T$ whether it shows a normal or an anomalous PV module using a $k$-NN classifier on top of the learned representations. First, the trained base encoder and projection head are used to compute the embeddings $\big\{z_i^S\big\}_{i=1\ldots N^S}$ of all source images as in eq. 1 and eq. 2. This needs to be done only once, as the embeddings are persisted in memory. Similarly, the target embedding $z_j^T$ is computed. Now, the $k$ source embeddings nearest to the target embedding in terms of Euclidean distance are obtained. We denote them as $\mathcal{N}_k$. As all embeddings have unit L2-norm using Euclidean distance is equivalent to using cosine distance. The final prediction $\hat{y}_j^T$ for the target image is made by aggregating the labels of the images in $\mathcal{N}_k$. If the fraction of anomalies in $\mathcal{N}_k$ exceeds the specified threshold $\delta$, the target image is predicted to contain an anomalous PV module. Later, in sec. V-D we will determine optimal settings for the hyperparameters $k$ and $\delta$.

We also tried using temperature-scaled cosine distance $\exp\big(z_j^T \cdot z_i^S / \tau\big)$ and distance-weighted label aggregation for prediction as in Wu et al. [30]. However, we did not observe a large impact on the predictions.

While in theory it is computationally expensive to compare each target embedding with all source embeddings, we do not observe this to be a bottleneck in practice for our dataset sizes. A possible workaround for significantly larger datasets is to perform $k$-means clustering on the source embeddings, and to use only the cluster centroids for distance computations [18, 74].

## C. Implementation Details

*1) Network Architecture:* We employ a randomly initialized ResNet-34 without the final classification layer as convolutional encoder $f_\theta(\cdot)$. We add a 2D global average pooling layer [17, 75] as final layer which outputs a 512-dimensional vector for each input image in the batch. The projection head $h_\psi(\cdot)$ is implemented by two randomly initialized fully-connected layers with 512 and 128 outputs, respectively, where the first layer is followed by a ReLU activation. Thus, the dimensionality of embeddings after the projection head is $d = 128$.

*2) Image Preprocessing:* Prior to feature extraction each 16-bit grayscale IR image is converted to Celsius scale, normalized to the interval $[0, 255]$ using the minimum and maximum temperature value in the image, converted to 8-bit and resized to $64 \times 64$ pixels. Each image is standardized by subtracting the dataset mean and dividing by the dataset standard deviation. To account for the domain shift, we compute a separate mean and standard deviation for each PV plant. As ResNet expects an RGB image as input, we finally stack three copies of the grayscale image along the channel-direction.

*3) Training:* We train all models for 110000 steps using stochastic gradient descent with momentum 0.9 and weight decay $5 \times 10^{-4}$ [76, 77]. The initial learning rate $\eta_0 = 6 \times 10^{-2}$ is decayed in each step following the Cosine Annealing strategy
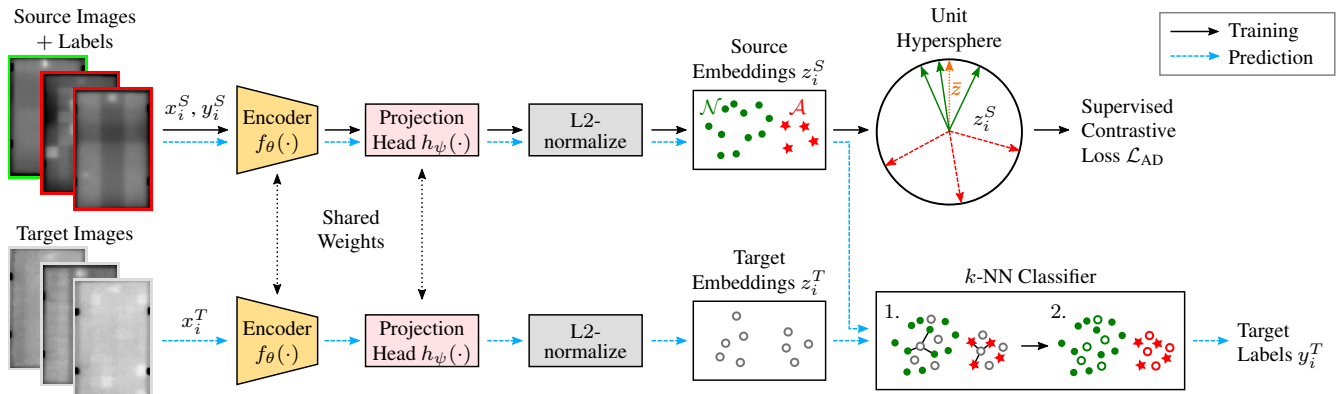
Figure 4: Overview of our method for detection of anomalous PV modules in IR images of a *target* PV plant based on labelled samples of a *source* PV plant. Low dimensional embeddings of both source and target images are extracted by means of contrastive representation learning. A $k$-NN classifier predicts target labels based on the labels of neighboring source images in the embedding space.

$\eta = \eta_0/2\,(1 + \cos{(p\pi)})$ where $p \in [0, 1]$ is the training progress [78]. We train with 16-bit precision and batch size 128 which is the maximum trainable on our hardware. We believe larger batch sizes can benefit contrastive representation learning as reported in similar works [22, 29, 32]. During training, we augment both source and target images independently from another by random up-down and left-right flips and random rotation by multiples of $90°$. All images in a batch are augmented identically.

*4) Hardware and Software:* All models are trained on a desktop workstation with an Intel i9-9900K, 64 GB RAM and a GeForce RTX 2080 Ti running Ubuntu 20.04 LTS, Python 3.6.9, PyTorch 1.7.1 and PyTorch Lightning 1.1.5.

## V. EXPERIMENTS & RESULTS

In the following, we perform a quantitative analysis of our method and compare it against a binary cross-entropy classifier.

### A. Evaluation Protocol

As common in anomaly detection, we evaluate all our models in terms of the area under the receiver operating characteristic (AUROC) and the average precision score (AP) [18, 21, 59]. AUROC is obtained by plotting the true positive rate $\text{TPR} = \text{TP}/(\text{TP}+\text{FN})$ over the false positive rate $\text{FPR} = \text{FP}/(\text{FP} + \text{TN})$ at various decision thresholds $\delta$ and integrating the resulting curve. Here, TP and TN denote the numbers of correctly classified anomalous and normal images, FP is the number of normal images misclassified as anomalous and FN the number of anomalous images classified as normal.

Similarly, the AP is obtained from the precision-recall curve which plots precision $\text{P} = \text{TP}/(\text{TP} + \text{FP})$ over recall $\text{R} = \text{TP}/(\text{TP} + \text{FN})$) at different decision thresholds. The AP summarizes the curve as the weighted mean of precisions achieved at each threshold $\text{AP} = \sum_{i=1}^{n}(\text{R}_i - \text{R}_{i-1})\,\text{P}_i$.

While AUROC takes both the normal class and the anomalous class into account, AP puts more emphasis on the anomalies [79]. Both AUROC and AP do not depend on a specific decision threshold $\delta$. Instead, they measure classification performance over the entire spectrum of threshold values. This makes them more informative than other metrics, such as classification accuracy or F1-score, which are computed at a single threshold value.

Because of this, AUROC and AP enable a fair comparison of different methods, which can depend differently on the decision threshold.

In the following, each model is trained on a source dataset S (train split) and evaluated on a target dataset T (train split), which we refer to as *task* S → T. As mentioned in sec. III, only the data of PV plants A, B, E and F is used. When we train and evaluate on the same PV plant, we use the source test split for evaluation and refer to it as A', B', E' or F'. We train each model three times with different random seeds and report the mean of AUROC and AP.

### B. Model Selection

In the following experiments, we compute AUROC and AP after each training epoch and report the best values obtained. In practice this is not feasible as target labels are unknown. Thus, we use labelled data of a second PV plant as validation dataset and report the target AUROC (AP) for the epoch at which the highest validation AUROC (AP) is achieved.

Sun et al. [80] proposed to use the cosine distance between the mean source and target embeddings for model selection. However, in our experiments this did not correlate well to the target metrics.

### C. Results of the Contrastive $k$-NN Classifier

We train and evaluate our method on various tasks and report the best target AUROC scores in fig. 5. All scores are above $70\,\%$ and thus well above the $50\,\%$ of a random guess. When training and evaluating on the same plant AUROC scores are generally higher, as there is no domain shift between train and test data. The results suggest that the choice of source plant has a considerable effect on the achievable target AUROC. For example, plant B is a better source plant than A and plant A is better than F. Plant F is most likely the worst source plant because its dataset is $4.6$ times smaller than that of plants A and B. However, plant A and B are similar in sample count and distribution of anomaly classes. Hence, it is interesting that plant B is a better source plant. This indicates that other effects, like image quality and module/cell types are important factors as well.

We further find that AUROC is generally lower when using plant A or B as target as opposed to plants E or F. Possible
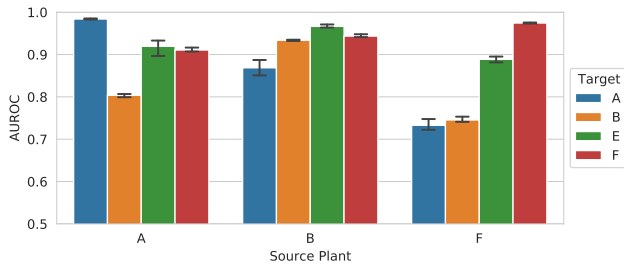
Figure 5: Best target AUROC of our method when trained on different source PV plants. When source and target plant are identical we evaluate on the target test split otherwise on the target train split. Error bars indicate the 95 % confidence interval over three runs.



(a) AUROC versus numbers $k$ of neighbors.



(b) G-Mean versus decision threshold $\delta$.

Figure 6: Prediction performance of the $k$-NN classifier for different settings of the hyperparameters $k$ and $\delta$. The dashed vertical lines at $k = 100, \delta = 0.1$ represent the trade-offs we use in practice. All classifiers are trained on contrastive embeddings of the model with random seed 1 and best target AUROC on the respective task.

explanations for this are the larger number of anomalies and the presence of sub-domains in plants A and B (see fig. 2) which make the accurate prediction of anomalies harder.

Note, that we do not report results for training on plant E as the contrastive loss did not converge. This is due to the lower fraction of anomalies in plant E, resulting in batches with only very few anomalous images. A larger batch size or special sampling strategy could have solved this issue.

### D. Hyperparameter Selection of the $k$-NN Classifier

The absence of labelled target images renders hyperparameter tuning of the $k$-NN classifier on the target plant impossible. Thus, for practical applications, it is important that the $k$-NN classifier is insensitive to the choice of hyperparameters.
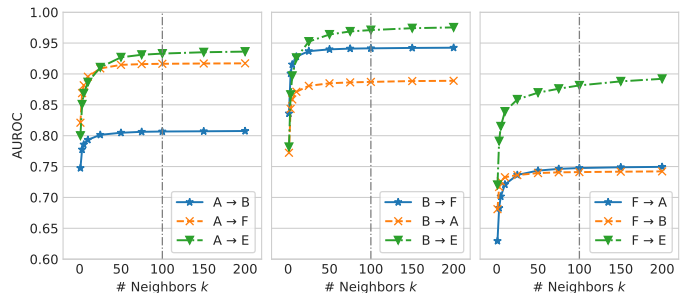
Fig. 6a shows the $k$-NN classifier AUROC for different numbers of neighbors. For all tasks the $k$-NN classifier is insensitive to the choice of $k$ once it exceeds 25. For some tasks the AUROC is still slightly increasing at $k = 200$. However, as runtime also increases, we choose $k = 100$ as trade-off in our experiments.

Another important hyperparameter is the decision threshold $\delta$, which is the fraction of anomalies required in the set of neighbors $\mathcal{N}_k$ to classify a target image as anomalous. Fig. 6b shows the geometric mean (G-Mean) of true positive rate and false positive rate for various decision thresholds $\delta$. While the classifier is more sensitive to the choice of $\delta$ (as compared to $k$), it behaves consistent across the tasks, taking on a high value for small thresholds. We choose $\delta = 0.1$ in practice to account for the imbalance between normal and anomalous images.
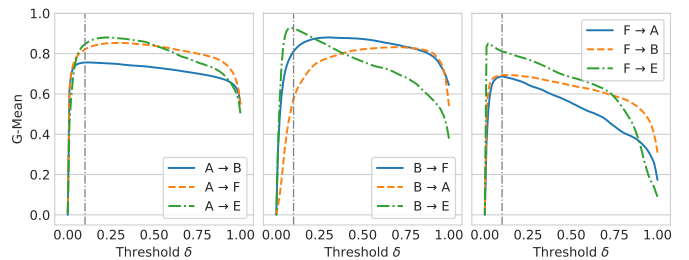
### E. Which Faults are Misclassified?

Using the hyperparameter settings from above, we make predictions with our contrastive $k$-NN classifier and show the resulting confusion matrices in fig. 7. Averaged over all tasks, the fractions of correctly classified normal and anomalous images are 79.4 % and 77.1 %, respectively. Furthermore, the fraction of anomalies misclassified as normal is only 22.9 % on average. Higher misclassification rates for the model trained on plant F suggest (in line with the results from sec. V-C) that plant F is a poor choice for training. The less critical fraction of normal images misclassified as anomalous is 20.6 % on average. An outlier in this metric is task B → A, which would require a higher decision threshold as can be seen in fig. 6b.

For the purpose of analysis we have access to fine-grained target labels. Thus, we can analyze, which specific anomaly

classes are misclassified, allowing us to identify potential systematic errors. Tab. 3 reports our findings. With a few exceptions, error rates are below 15 % for faults Mp, Sh, Sp, Pid, Cm+, Cs+ and C when training on plants A and B.

For homogeneously overheated modules (Mh), we observe a high error rate. This is caused by the image-wise normalization applied during preprocessing and may be addressed in future works. High error rates also occur for D and Chs faults due to their small spatial extent in the image. This is a typical problem of convolutional neural networks. However, as D and Chs faults are not critical, we can accept the higher error rates. Interestingly, the model trained on plant F correctly identifies many Pid modules, despite the lack of Pid training examples in F. The model most likely transfers knowledge from the visually similar Mp class. This fails for the visually more unique Sh anomaly, of which plant F contains only 13 examples.

### F. Visualization of Misclassified IR Images

To build an intuition for the quantitative results of our method, we make predictions on IR images and visualize both correct and false predictions in fig. 8.

As shown by the examples, the high misclassification rates for the Mh, D and Chs anomalies can be explained by their high visual similarity to the normal images. Similarly, we find that primarily those anomalous images are misclassified that exhibit lower local temperature differences and are visually more similar to the normal images. This is a good indicator for the smoothness of the learned contrastive representations and thus the robustness of our approach.

Fig. 8 also hightlights a few misclassified normal images. Interestingly, most of these are valid anomalies with false ground truth

Figure 7: Normalized confusion matrices of the $k$-NN classifier ($k = 100$ and $\delta = 0.1$) on different tasks. We report averages using the model with best target AUROC at each random seed.

Table 3: Percentage of target anomalies misclassified as normal by our $k$-NN classifier ($k = 100$ and $\delta = 0.1$) grouped by fault class. Fault criticality decreases from left to right. Error rates below 15 % are green, between 15 % and 50 % orange and above 50 % red. We report averages over three runs using the model with best target AUROC at each run.

| Task | Actual Fault Class | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | Mh | Mp | Sh | Sp | Pid | Cm+ | Cs+ | C | D | Chs | All |
| A → A' | 7.8 | 0.0 | 0.0 | 11.2 | 0.3 | – | 0.0 | 3.4 | 16.6 | 28.4 | 5.6 |
| A → B | 76.6 | 0.3 | 1.3 | 0.0 | 1.5 | 3.6 | 2.7 | 13.5 | 55.8 | 33.0 | 33.4 |
| A → E | – | 0.0 | 2.7 | 0.0 | – | 0.0 | 0.1 | 18.6 | – | 72.0 | 16.1 |
| A → F | 0.0 | 16.1 | 0.0 | 2.1 | – | 1.2 | 4.9 | 11.9 | 55.3 | – | 11.8 |
| B → A | 0.0 | 0.0 | 0.1 | 0.0 | 0.7 | 0.0 | 0.0 | 3.8 | 10.8 | 9.4 | 4.3 |
| B → B' | 37.0 | 0.0 | 3.9 | 0.0 | 1.1 | 35.0 | 0.3 | 11.2 | 97.4 | 21.7 | 17.8 |
| B → E | – | 0.0 | 8.7 | 0.1 | – | 0.0 | 3.9 | 10.3 | – | 51.9 | 9.6 |
| B → F | 0.0 | 0.0 | 0.0 | 0.0 | – | 0.0 | 0.2 | 3.9 | 66.0 | – | 4.3 |
| F → A | 30.4 | 0.0 | 60.5 | 0.0 | 27.0 | 0.0 | 1.3 | 62.1 | 62.1 | 59.8 | 48.3 |
| F → B | 91.2 | 0.0 | 71.8 | 1.0 | 8.6 | 3.6 | 1.0 | 26.6 | 30.8 | 75.8 | 49.2 |
| F → E | – | 7.0 | 68.9 | 2.3 | – | 22.3 | 2.3 | 32.6 | – | 84.2 | 28.7 |
| F → F' | – | – | – | 0.0 | – | 0.7 | 0.3 | 6.7 | – | 0.0 | 6.0 |



Figure 8: Exemplary predictions of our $k$-NN classifier ($k = 100$ and $\delta = 0.1$) for IR images of plant B by the model trained on plant A. We use the model at seed 1 with best target AUROC and show preprocessed patches.

labels. There are also cases of poorly cropped images, images with strong perspective distortion or images with sun reflections. Our method correctly identifies them as anomalies despite never having been trained on such examples.

*G. Embedding Visualizations*

As another means to interpret our models, we visualize the representations learned by supervised contrastive training in fig. 9. Here, for most tasks, the representations clearly separate normal and anomalous images, which explains the overall high AUROC and AP scores achieved. Exceptions are tasks F → A and F → B, where many anomalies lie within the normal cluster resulting in a low recall. We can also see that the anomaly classes Mp, Sh, Sp, Pid, Cm+ and Cs+, which achieved low error rates in sec. V-E, have a larger distance to the normal modules than anomalies with higher error rates (Mh, D, Chs). The C anomalies often lie somewhere in between, which is in accordance to the slightly higher error rates of around 10 %.

*H. Detection of Unknown Anomalies*

One goal of our method is the ability to reliably detect anomalies in the target dataset, which are not contained in the source dataset. To analyze how well our method deals with such un-
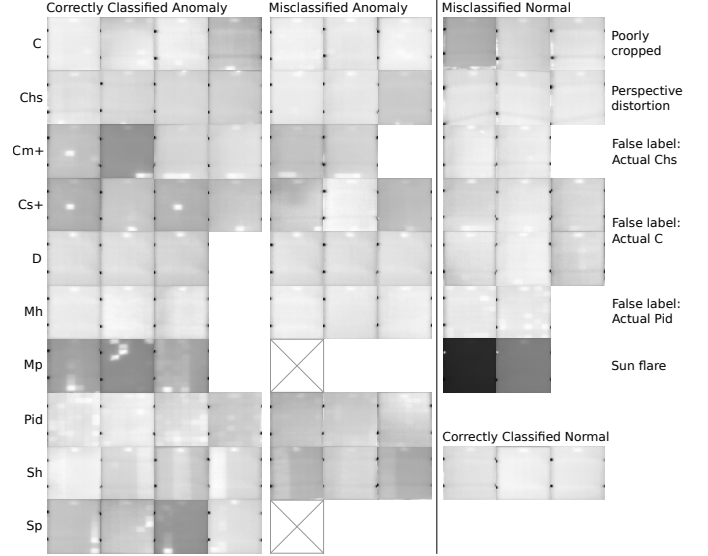


Figure 9: UMAP projections (with 50 neighbors per sample and minimum distance of 0.1) of the target datasets embedded by ResNet-34 after supervised contrastive training. Embeddings are obtained behind the ResNet-34 average pooling layer. For each task the model at seed 1 with best target AUROC is shown.

known anomalies, we remove all anomalies of classes Mp, Sh, Sp, Cm+ and Cs+ from the source datasets of plants A and B, retrain our models and evaluate on the full target datasets containing all anomaly classes. We chose precisely these classes, as they make up only 3.1 % and 5.2 % of all anomalies in datasets A and B. This leaves dataset sizes nearly unchanged, providing us with a more comparable result. For most tasks, the resulting target AUROCs (see tab. 4) do not deviate much from the respective AUROCs of the models trained on all anomaly classes. Similarly, we do

Table 4: Target AUROCs of our contrastive $k$-NN classifier trained on datasets where anomaly classes Mp, Sh, Sp, Cm+ and Cs+ are left out versus the baseline trained on the full dataset. All values are averages over three training runs.

| Task | Variant | | Task | Variant | |
|------|---------|----------|------|---------|----------|
| | Full dataset | Leaveout | | Full dataset | Leaveout |
| A → A' | 98.39 | 98.29 | B → A | 86.86 | 83.43 |
| A → B | 80.38 | 80.60 | B → B' | 93.38 | 93.38 |
| A → E | 91.93 | 92.26 | B → E | 96.64 | 96.67 |
| A → F | 91.06 | 91.63 | B → F | 94.35 | 93.92 |

Table 5: Target AUROCs of our contrastive $k$-NN classifier versus a binary classifier trained with cross-entropy loss. We report the best values achieved and values for models selected via two different validation datasets (Val 0 and Val 1). All values are averages over three training runs. Values of the better method are in bold.

| Task | Val 0 | Val 1 | Contrastive AUROC | | | Cross-Entropy AUROC | | |
|------|-------|-------|--------|--------|------|--------|--------|------|
| | | | @Val 0 | @Val 1 | Best | @Val 0 | @Val 1 | Best |
| A → A' | – | – | – | – | 98.39 | – | – | **98.64** |
| A → B | F | E | **78.83** | **79.62** | **80.38** | 74.68 | 77.76 | 78.19 |
| A → E | F | B | **90.85** | **90.02** | **91.93** | 81.45 | 86.49 | 88.71 |
| A → F | B | E | **88.82** | **90.49** | **91.06** | 85.18 | 86.10 | 89.26 |
| B → A | F | E | **77.01** | **80.25** | **86.86** | 68.35 | 69.95 | 76.81 |
| B → B' | – | – | – | – | 93.38 | – | – | **95.45** |
| B → E | F | A | **95.20** | **93.20** | **96.64** | 92.54 | 88.32 | **96.66** |
| B → F | A | E | 88.13 | 92.42 | **94.35** | **89.94** | **92.51** | 93.85 |
| F → A | B | E | **66.14** | **61.58** | 73.29 | 63.46 | 55.49 | **75.22** |
| F → B | A | E | 69.22 | 72.68 | 74.54 | **74.91** | **76.83** | **77.52** |
| F → E | A | B | 84.06 | 86.42 | 88.86 | **88.02** | **88.65** | **90.64** |
| F → F' | – | – | – | – | 97.44 | – | – | **97.54** |

not observe any change in model convergence during training, as shown in fig. 13 in appendix A-B. The results indicate that our method can reliably detect unknown anomalies.

*I. Comparison with Cross-Entropy Classifier*

We compare our method with a deep convolutional binary classifier based on ResNet-34, which is trained with standard cross-entropy loss using the same data preprocessing, data augmentation and training settings as our method (see sec. IV-C). While the convolutional backbone is identical to our contrastive model, a softmax-activated fully connected layer with 2 outputs is used on top of the 2D global average pooling layer. A projection head is not employed.

As shown in tab. 5, our method outperforms the cross-entropy classifier in terms of target AUROC in many cases which is in accordance to the literature [22]. Only on tasks F → B and F → E our method falls behind. This could be due to the smaller dataset size and thus smaller absolute number of anomalies in plant F. It indicates that our method is more sensitive to the dataset size, i.e., is less accurate on small datasets but profits more from larger dataset sizes than the cross-entropy classifier.

The same result is reflected in the AP, which is exemplary shown for plant A over the course of the training in fig. 10. An additional analysis for plants B and F is provided in Appendix A-A. Furthermore, we find that due to the large size of our datasets target AP converges within a single training epoch.

*J. Module-level Aggregation of Predictions*

As there are on average $39.4$ IR images of each PV module we can aggregate predictions of those images to obtain a final prediction for the module. Specifically, we predict a module as anomalous if at least one half of the corresponding images are predicted anomalous. As indicated by the resulting confusion matrices in fig. 11, on average $82.9\%$ of all normal and $78.1\%$ of all anomalous modules are correctly classified. On average, $17.1\%$ of the normal modules are misclassified as anomalous and $21.9\%$ of the anomalous modules are misclassified as normal. As compared to the image-level predictions (see sec. V-E) module-level aggregation improves especially upon the detection rate of normal modules, but also yields a one percent higher detection rate for anomalies. These results suggest that the hierarchical structure of our dataset is beneficial for the accurate detection of anomalous PV modules.

*K. Exemplary Application to Labelling of IR Datasets*

For the development of future fault classification methods large IR image datasets are needed. Our method can drastically reduce the time and effort needed for labelling such datasets by automatically rejecting the majority of normal (and thus uninteresting) PV modules. For example, when labelling plant E, one would have to manually sight $14662$ PV modules, of which only $296$ are anomalous, i.e., actually interesting. Applying our method (trained for example on plant B) could automatically reject $98.1\%$ of the normal modules, leaving only $273$ normal modules for manual sighting. The cost for this improvement is the loss of $26$ anomalous modules, which are misclassified as normal. In total, $543$ modules are left for manual sighting. Assuming an expert takes three seconds to label one module, this reduces the time needed to label plant E from $12.2$ hours to only $27$ minutes. Adjusting the decision threshold during module-level aggregation allows to trade off lost anomalies and time savings.

## VI. DISCUSSION AND CONCLUSION

*1) Summary:* In this work, we proposed a novel method for the detection of PV module faults in IR images using supervised contrastive learning. Instead of sampling train and test data from the same PV plant, we performed training with labelled IR images of one source plant and made predictions on another target plant. We identified domain shift between source and target data as a problem in this setting and addressed it by learning transferable representations with a supervised contrastive loss. A $k$-NN classifier was used on top of these representations to detect unknown anomalies in the target plant. Experiments on nine different combinations of four source and target datasets showed the effectiveness of our method, which achieved an AUROC of $73.3\%$ to $96.6\%$ and even outperformed a binary cross-entropy classifier in some cases. We further found that our method converges quickly and is relatively insensitive to hyperparameter settings, making it well suited for practical applications. Using fault labels for 10 different types of anomalies, we found that our method most frequently misses anomalies with a small spatial extent in the image, e.g. overheated bypass diodes or small hot spots. Most striking, our method showed no significant drop in AUROC after removing five of the ten anomaly classes from the training datasets, proving its ability to reliably detect
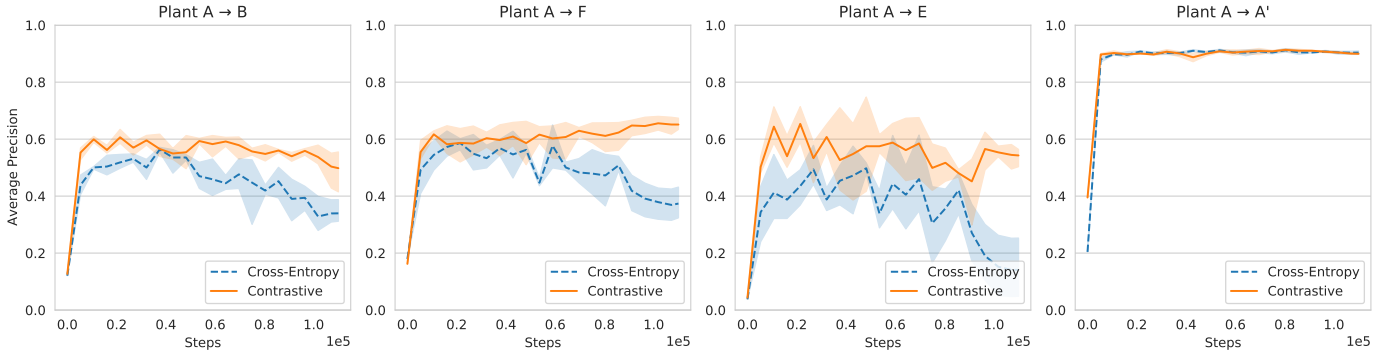
Figure 10: Average precision over the course of training of our contrastive $k$-NN classifier (orange line) versus a supervised binary classifier trained on cross-entropy loss (dashed blue line). Plant A is used as source. Shaded regions indicate the 95 % confidence interval over three runs.



Figure 11: Normalized confusion matrices of the $k$-NN classifier ($k = 100$ and $\delta = 0.1$) for predictions aggregated on module-level. We report averages using the model with best target AUROC at each random seed.

unknown anomalies. Finally, we improved detection accuracy by aggregating predictions of multiple IR images belonging to the same PV module.

*2) Practical Relevance:* Increasing PV deployments and aging PV plants require regular inspections to ensure a safe operation and maximum power output, yield and profitability of a plant. The large size of most PV plants and potentially high labour cost renders a manual inspection economically infeasible and raises the need for fully automatic plant inspection. Our method is highly relevant for such inspection systems, as it automatically identifies anomalous PV modules in a large number of IR images. This enables targeted repairs and restoration of the original performance of a PV plant. Apart from the inspection of existing plants, automatic inspection is further useful for the commissioning of new plants.

One problem of existing fault detection methods is that they do not explicitly consider domain shift between different PV plants. This means a fault detector must be fine-tuned on labelled training images of each new PV plant that is inspected. This is not only labor-intensive, but also time-consuming, as training a neural network takes several hours. Opposed to that, our method explicitly handles domain shift. This way, it needs to be trained only once on a labelled dataset and generalizes afterwards to new PV plants without further fine-tuning. This is of major importance for realizing economically viable plant inspection systems that

work for many different PV plants without the need for a time-consuming and costly setup phase.

Apart from automatic plant inspection, our method can also aid the manual labelling of IR datasets. This facilitates creation of large-scale datasets, which are needed for the development of the next generation of automatic fault detection algorithms.

*3) Future Works:* We presented a PV module fault detection method, which overcomes domain shift between different PV plants and generalizes beyond the training dataset without the need for huge amounts of labelled training data. While this is an important milestone, further measures could improve domain adaption and increase detection accuracy on new PV plants. For example, future works could explore active domain adaptation techniques, such as Maximum Mean Discrepancy. In addition, multi-domain adaptation, which uses multiple labelled source datasets from different PV plants simultaneously, could be taken into consideration.

## VII. ACKNOWLEDGEMENTS

## References

[1] REN21, "Renewables 2020 global status report," 2020.

[2] IRENA, "Future of solar photovoltaic: Deployment, investment, technology, grid integration and socio-economic aspects," *International Renewable Energy Agency, Abu Dhabi*, 2019.

[3] L. Bommes, T. Pickel, C. Buerhop-Lutz, J. Hauch, C. Brabec, and I. M. Peters, "Computer vision tool for detection, mapping and fault classification of pv modules in aerial ir videos," *Progress in Photovoltaics: Research and Applications*, 2021.

[4] M. Aghaei, F. Grimaccia, C. A. Gonano, and S. Leva, "Innovative automated control system for PV fields inspection and remote control," *IEEE Transactions on Industrial Electronics*, 2015.

[5] A. Arenella, A. Greco, A. Saggese, and M. Vento, "Real time fault detection in photovoltaic cells by cameras on drones," in *International Conference Image Analysis and Recognition (ICIAR)*, 2017.

[6] H. Jeong, G.-R. Kwon, and S.-W. Lee, "Deterioration diagnosis of solar module using thermal and visible image processing," *Energies*, 2020.

[7] F. Grimaccia, S. Leva, and A. Niccolai, "PV plant digital mapping for modules' defects detection by unmanned aerial vehicles," *IET Renewable Power Generation*, 2017.

[8] G. Francesco, L. Sonia, and N. Alessandro, "A semi-automated method for defect identification in large photovoltaic power plants using unmanned aerial vehicles," in *IEEE Power Energy Society General Meeting (PESGM)*, 2018.

[9] M. Alsafasfeh, I. Abdel-Qader, B. Bazuin, Q. Alsafasfeh, and W. Su, "Unsupervised fault detection and analysis for large photovoltaic systems using drones and machine vision," *Energies*, 2018.

[10] V. Carletti, A. Greco, A. Saggese, and M. Vento, "An intelligent flying system for automatic detection of faults in photovoltaic plants," *Journal of Ambient Intelligence and Humanized Computing*, 2019.

[11] P. Addabbo, A. Angrisano, M. L. Bernardi, G. Gagliarde, A. Mennella, M. Nisi, and S. L. Ullo, "UAV system for photovoltaic plant inspection," *IEEE Aerospace and Electronic Systems Magazine*, 2018.

[12] S. Dotenco, M. Dalsass, L. Winkler, T. Würzner, C. Brabec, A. Maier, and F. Gallwitz, "Automatic detection and analysis of photovoltaic modules in aerial infrared imagery," in *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2016.

[13] D. Kim, J. Youn, and C. Kim, "Automatic fault recognition of photovoltaic modules based on statistical analysis of UAV thermography," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences (ISPRS)*, 2017.

[14] C. Dunderdale, W. Brettenny, C. Clohessy, and E. E. van Dyk, "Photovoltaic defect classification through thermal infrared imaging using a machine learning approach," *Progress in Photovoltaics: Research and Applications*, 2020.

[15] A. Oliveira, M. Aghaei, and R. Rüther, "Automatic fault detection of photovoltaic arrays by convolutional neural networks during aerial infrared thermography," in *European Photovoltaic Solar Energy Conference and Exhibition (EUPVSEC)*, 2019.

[16] R. Pierdicca, E. Malinverni, F. Piccinini, M. Paolanti, A. Felicetti, and P. Zingaretti, "Deep convolutional neural network for automatic detection of damaged photovoltaic cells," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences (ISPRS)*, 2018.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[18] L. Bergman, N. Cohen, and Y. Hoshen, "Deep nearest neighbor anomaly detection," *arXiv preprint arXiv:2002.10445*, 2020.

[19] L. Ruff, R. A. Vandermeulen, B. J. Franks, K.-R. Müller, and M. Kloft, "Rethinking assumptions in deep anomaly detection," *arXiv preprint arXiv:2006.00339*, 2020.

[20] D. Hendrycks, M. Mazeika, S. Kadavath, and D. Song, "Using self-supervised learning can improve model robustness and uncertainty," in *Advances in Neural Information Processing Systems (NIPS)*, 2019.

[21] D. Hendrycks, M. Mazeika, and T. Dietterich, "Deep anomaly detection with outlier exposure," in *International Conference on Learning Representations (ICLR)*, 2019.

[22] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, "Supervised contrastive learning," in *Advances in Neural Information Processing Systems (NIPS)*, 2020.

[23] J. Winkens, R. Bunel, A. G. Roy, R. Stanforth, V. Natarajan, J. R. Ledsam, P. MacWilliams, P. Kohli, A. Karthikesalingam, S. Kohl, T. Cemgil, S. M. A. Eslami, and O. Ronneberger, "Contrastive training for improved out-of-distribution detection," *arXiv preprint arXiv:2007.05566*, 2020.

[24] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006.

[25] K. Q. Weinberger and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," *Journal of Machine Learning Research (JMLR)*, 2009.

[26] K. Sohn, "Improved deep metric learning with multi-class n-pair loss objective," in *Advances in Neural Information Processing Systems (NIPS)*, 2016.

[27] P. H. Le-Khac, G. Healy, and A. F. Smeaton, "Contrastive representation learning: A framework and review," *IEEE Access*, 2020.

[28] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[29] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," *arXiv preprint arXiv:2002.05709*, 2020.

[30] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, "Unsupervised feature learning via non-parametric instance discrimination," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[31] M. Ye, X. Zhang, P. C. Yuen, and S.-F. Chang, "Unsupervised embedding learning via invariant and spreading instance feature," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[32] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[33] X. Chen, H. Fan, R. Girshick, and K. He, "Improved baselines with momentum contrastive learning," *arXiv preprint arXiv:2003.04297*, 2020.

[34] C. Wei, Y. Tang, C. Niu, H. Hu, Y. Wang, and J. Liang, "Self-supervised representation learning for evolutionary neural architecture search," *arXiv preprint arXiv:2011.00186*, 2020.

[35] O. Köpüklü, J. Zheng, H. Xu, and G. Rigoll, "Driver anomaly detection: A dataset and contrastive learning approach," *arXiv preprint arXiv:2009.14660*, 2020.

[36] M. Wang and W. Deng, "Deep visual domain adaptation: A survey," *Neurocomputing*, 2018.

[37] S. Zhao, B. Li, C. Reed, P. Xu, and K. Keutzer, "Multi-source domain adaptation in the deep learning era: A systematic survey," *arXiv preprint arXiv:2002.12169*, 2020.

[38] M. Long, Y. Cao, J. Wang, and M. I. Jordan, "Learning transferable features with deep adaptation networks," in *International Conference on Machine Learning (PMLR)*, 2015.

[39] A. Rozantsev, M. Salzmann, and P. Fua, "Beyond sharing weights for deep domain adaptation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[40] Y. Zhu, F. Zhuang, and D. Wang, "Aligning domain-specific distribution and classifier for cross-domain classification from multiple sources," *AAAI Conference on Artificial Intelligence*, 2019.

[41] H. Guo, R. Pasunuru, and M. Bansal, "Multi-source domain adaptation for text classification via distancenet-bandits," *arXiv preprint arXiv:2001.04362*, 2020.

[42] S. Rakshit, B. Banerjee, G. Roig, and S. Chaudhuri, "Unsupervised multi-source domain adaptation driven by deep adversarial ensemble learning," in *Pattern Recognition*, 2019.

[43] J. Hoffman, M. Mohri, and N. Zhang, "Algorithms and theory for multiple-source adaptation," in *Advances in Neural Information Processing Systems (NIPS)* (S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, eds.), 2018.

[44] F. Zhuang, X. Cheng, P. Luo, S. J. Pan, and Q. He, "Supervised representation learning: Transfer learning with deep autoencoders," in *International Joint Conference on Artificial Intelligence (IJCAI)*, 2015.

[45] G. Kang, L. Jiang, Y. Yang, and A. G. Hauptmann, "Contrastive adaptation network for unsupervised domain adaptation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[46] S. Dai, Y. Cheng, Y. Zhang, Z. Gan, J. Liu, and L. Carin, "Contrastively smoothed class alignment for unsupervised domain adaptation," *arXiv preprint arXiv:1909.05288*, 2020.

[47] C. Park, J. Lee, J. Yoo, M. Hur, and S. Yoon, "Joint contrastive learning for unsupervised domain adaptation," *arXiv preprint arXiv:2006.10297*, 2020.

[48] Y. Zhu, F. Zhuang, J. Wang, G. Ke, J. Chen, J. Bian, H. Xiong, and Q. He, "Deep subdomain adaptation network for image classification," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[49] Z. Yang, I. S. Bozchalooi, and E. F. Darve, "Anomaly detection with domain adaptation," in *Advances in Neural Information Processing Systems (NIPS)*, 2019.

[50] G. Pang, C. Shen, L. Cao, and A. V. D. Hengel, "Deep learning for anomaly detection: A review," *ACM Computing Surveys*, 2021.

[51] S. Bulusu, B. Kailkhura, B. Li, P. K. Varshney, and D. Song, "Anomalous example detection in deep learning: A survey," *IEEE Access*, 2020.

[52] R. Chalapathy and S. Chawla, "Deep learning for anomaly detection: A survey," *arXiv preprint arXiv:1901.03407*, 2019.

[53] J. Chen, S. Sathe, C. Aggarwal, and D. Turaga, "Outlier detection with autoencoder ensembles," in *SIAM International Conference on Data Mining (SDM)*, 2017.

[54] J. An and S. Cho, "Variational autoencoder based anomaly detection using reconstruction probability," in *Special Lecture on IE*, 2015.

[55] S. Akcay, A. Atapour-Abarghouei, and T. P. Breckon, "GANomaly: Semi-supervised anomaly detection via adversarial training," in *Asian Conference on Computer Vision (ACCV)*, 2019.

[56] H. Zenati, M. Romain, C.-S. Foo, B. Lecouat, and V. Chandrasekhar, "Adversarially learned anomaly detection," in *IEEE International Conference on Data Mining (ICDM)*, 2018.

[57] L. Ruff, R. A. Vandermeulen, N. Görnitz, A. Binder, E. Müller, K.-R. Müller, and M. Kloft, "Deep semi-supervised anomaly detection," in *International Conference on Learning Representations (ICLR)*, 2020.

[58] L. Ruff, R. Vandermeulen, N. Goernitz, L. Deecke, S. A. Siddiqui, A. Binder, E. Müller, and M. Kloft, "Deep one-class classification," in *International Conference on Machine Learning (PMLR)*, 2018.

[59] I. Golan and R. El-Yaniv, "Deep anomaly detection using geometric transformations," in *Advances in Neural Information Processing Systems (NIPS)*, 2018.

[60] S. Wang, Y. Zeng, X. Liu, E. Zhu, J. Yin, C. Xu, and M. Kloft, "Effective end-to-end unsupervised outlier detection via inlier priority of discriminative network," in *Advances in Neural Information Processing Systems (NIPS)*, 2019.

[61] L. Bergman and Y. Hoshen, "Classification-based anomaly detection for general data," in *International Conference on Learning Representations (ICLR)*, 2020.

[62] P. Perera and V. M. Patel, "Learning deep features for one-class classification," *IEEE Transactions on Image Processing*, 2019.

[63] S. F. Yilmaz and S. S. Kozat, "Unsupervised anomaly detection via deep metric learning with end-to-end optimization," *arXiv preprint arXiv:2005.05865*, 2020.

[64] K. Sohn, C.-L. Li, J. Yoon, M. Jin, and T. Pfister, "Learning and evaluating representations for deep one-class classification," in *International Conference on Learning Representations (ICLR)*, 2021.

[65] J. Tack, S. Mo, J. Jeong, and J. Shin, "CSI: Novelty detection via contrastive learning on distributionally shifted instances," in *Advances in Neural Information Processing Systems (NIPS)*, 2020.

[66] J. Chen and X. Liu, "Transfer learning with one-class data," *Pattern Recognition Letters*, 2014.

[67] M. Yamaguchi, Y. Koizumi, and N. Harada, "AdaFlow: Domain-adaptive density estimator with application to anomaly detection and unpaired cross-domain translation," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019.

[68] A. Kumagai, T. Iwata, and Y. Fujiwara, "Transfer anomaly detection by inferring latent domain representations," in *Advances in Neural Information Processing Systems (NIPS)*, 2019.

[69] M. Mayr, M. Hoffmann, A. Maier, and V. Christlein, "Weakly supervised segmentation of cracks on solar cells using normalized $L_p$ norm," *arXiv preprint arXiv:2001.11248*, 2020.

[70] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv/1704.04861*, 2017.

[71] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations (ICLR)*, 2015.

[72] L. McInnes, J. Healy, N. Saul, and L. Großberger, "UMAP: Uniform manifold approximation and projection," *Journal of Open Source Software*, 2018.

[73] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. Richemond, E. Buchatskaya, C. Doersch, B. Avila Pires, Z. Guo, M. Gheshlaghi Azar, B. Piot, k. kavukcuoglu, R. Munos, and M. Valko, "Bootstrap your own latent - A new approach to self-supervised learning," in *Advances in Neural Information Processing Systems (NIPS)*, 2020.

[74] J. Goldberger, S. Roweis, G. Hinton, and R. Salakhutdinov, "Neighbourhood components analysis," in *Advances in Neural Information Processing Systems (NIPS)*, 2004.

[75] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[76] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the importance of initialization and momentum in deep learning," in *International Conference on Machine Learning (PMLR)*, 2013.

[77] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *International Conference on Learning Representations (ICLR)*, 2019.

[78] I. Loshchilov and F. Hutter, "SGDR: stochastic gradient descent with warm restarts," in *International Conference on Learning Representations (ICLR)*, 2017.

[79] T. Saito and M. Rehmsmeier, "The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets," *PLoS One*, 2015.

[80] Y. Sun, E. Tzeng, T. Darrell, and A. A. Efros, "Unsupervised domain adaptation through self-supervision," *arXiv preprint arXiv:1909.11825*, 2019.

# A.  APPENDIX

## A.  *Additional Comparisons with the Cross-Entropy Classifier*

Fig. 12 shows additional results for the comparison of our method with a cross-entropy classifier performed in sec. V-I.
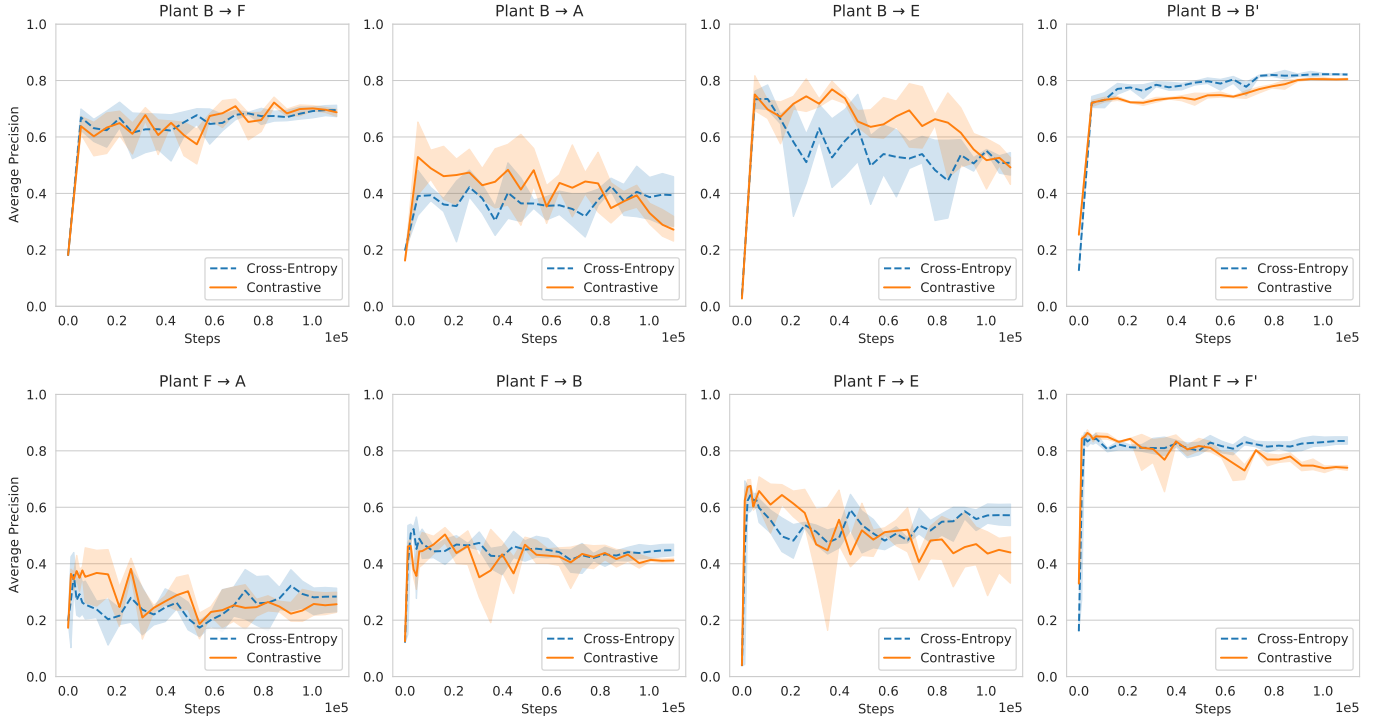


Figure 12: Average precision over the course of training of our contrastive $k$-NN classifier (orange line) versus a supervised binary classifier trained on cross-entropy loss (dashed blue line). The top row uses plant B as source, the bottom row plant F. Shaded regions indicate the 95 % confidence interval over three runs.

## B. Additional Comparisons for Training without some Anomalies

Fig. 13 shows additional results for our method trained on reduced source datasets without Mp, Sh, Sp, Cm+ and Cs+ anomalies. It extends the results presented in sec. V-H.
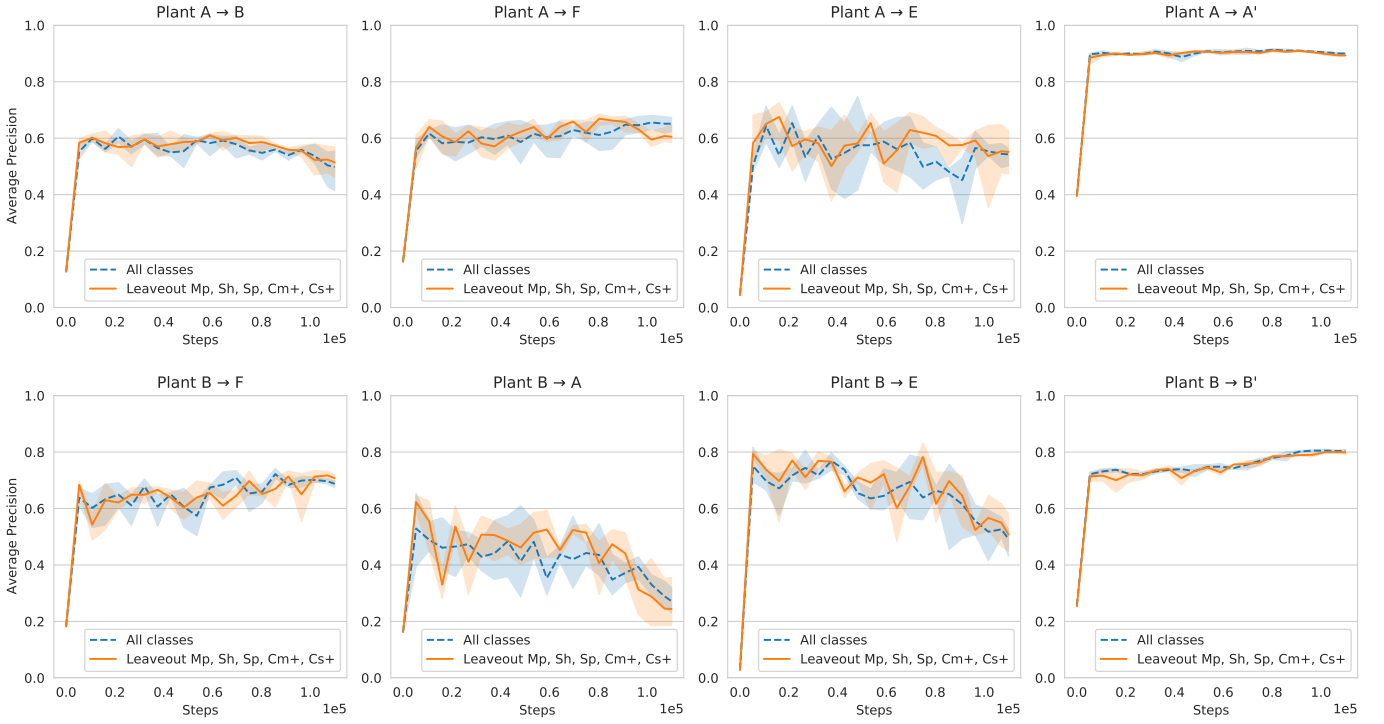


Figure 13: Average precision over the course of training of our contrastive $k$-NN classifier trained on datasets without anomaly classes Mp, Sh, Sp, Cm+ and Cs+ (orange line) versus the baseline trained on the full dataset (dashed blue line). Shaded regions indicate the 95 % confidence interval over three runs.

# Georeferencing of Photovoltaic Modules from Aerial Infrared Videos using Structure-from-Motion

Lukas Bommes[1], Claudia Buerhop-Lutz[1], Tobias Pickel[1], Jens Hauch[1], Christoph Brabec[1,2], and Ian Marius Peters[1]

[1]*Forschungszentrum Jülich GmbH, Helmholtz-Institute Erlangen-Nuremberg for Renewable Energies (HI ERN)*
[2]*Institute Materials for Electronics and Energy Technology, Universität Erlangen-Nürnberg (FAU)*
Correspondence to i.peters@fz-juelich.de

## Abstract

*To identify abnormal photovoltaic (PV) modules in large-scale PV plants economically, drone-mounted infrared (IR) cameras and automated video processing algorithms are frequently used. While most related works focus on the detection of abnormal modules, little has been done to automatically localize those modules within the plant. In this work, we use incremental structure-from-motion to automatically obtain geocoordinates of all PV modules in a plant based on visual cues and the measured GPS trajectory of the drone. In addition, we extract multiple IR images of each PV module. Using our method, we successfully map 99.3% of the 35084 modules in four large-scale and one rooftop plant and extract over 2.2 million module images. As compared to our previous work, extraction misses 18 times less modules (one in 140 modules as compared to one in eight). Furthermore, two or three plant rows can be processed simultaneously, increasing module throughput and reducing flight duration by a factor of 2.1 and 3.7, respectively. Comparison with an accurate orthophoto of one of the large-scale plants yields a root mean square error of the estimated module geocoordinates of 5.87 m and a relative error within each plant row of 0.22 m to 0.82 m. Finally, we use the module geocoordinates and extracted IR images to visualize distributions of module temperatures and anomaly predictions of a deep learning classifier on a map. While the temperature distribution helps to identify disconnected strings, we also find that its detection accuracy for module anomalies reaches, or even exceeds, that of a deep learning classifier for seven out of ten common anomaly types. The software is published at https://github.com/LukasBommes/PV-Hawk.*

## I. Introduction

The large amount of global installed solar photovoltaics (PV) and expected future growth require automatic image analysis for adequate quality control. As PV modules may develop defects due to environmental influences, aging or incorrect handling, PV plants need to be inspected regularly to ensure safe operation and maximum yield. Due to the large size of most PV plants, inspection is only economic if highly automated [1]. Thus, recent years have seen a surge in automated PV plant inspection systems, such as the ones by Zefri et al. [2], Pierddicca et al. [3], Henry et al. [4], and Carletti et al. [5]. These systems rely on drones equipped with a thermal infrared (IR) camera, that enables detection of abnormal PV modules based on their thermal signature [6]. The large amounts of acquired IR images are automatically processed by computer vision algorithms, which typically detect PV modules in the images, predict module anomalies, and localize each module in the PV plant.

In this work, we focus on the localization of PV modules in large-scale plants. Localization is a crucial task as it enables targeted repairs of abnormal modules. However, it is also notoriously difficult to identify the correct module among millions of identically looking and densely packed modules from a highly repetitive video with only a limited viewport. Previous works attempted to solve this problem by stitching adjacent video frames of a PV plant row into a panorama image [7]–[9]. This approach was successful, yet only works well for short video sequences. And, as also shown in our previous work [10], panorama stitching requires manual selection of the video frames for each row and provides the module location only relative to other modules. Niccolai et al. [11] also use panorama stitching and additionally match each row panorama to a CAD plan. While this yields absolute module locations, it requires a CAD plan, which is not always available and, even if it is available, is by no means standardized across different PV plants.

Other works explore direct georeferencing of PV modules in each image based on the measured GPS position and altitude of the drone [12], [13]. Georeferencing requires a centimeter-accurate Realtime Kinematics GPS (RTK-GPS) and is prone to GPS measurement errors as no additional visual cues are considered. Further, georeferencing is limited to nadiral images, which may contain sun reflections and exhibit sub-optimal contrast compared to images taken under the optimal viewing angle. Being limited to nadiral images also makes drone operation more difficult.

Another method for module localization is the creation of an orthophoto from a few high-altitude images [14]–[16]. Orthophotos allow visualizing the temperature distribution of the entire PV plant. One issue with this approach is that it is not always possible to take images from high altitudes, e.g. if there are nearby streets. Furthermore, a low spatial resolution and possible visual artefacts impede accurate detection of abnormal modules based on the orthophoto alone.

This work presents a new method for PV plant inspection based on aerial IR videos. As opposed to the related works, our method is fully automated, provides the absolute geocoordinates of each PV module instead of a relative location, works on long video sequences of large-scale plants, requires no CAD plan, works with both standard GPS and RTK-GPS and is not limited to nadiral videos. Furthermore, videos can be acquired from low flight

altitudes and multiple high resolution images of each module are obtained, which are important for downstream analysis.

Our method builds on our previous work [10], but features a more general approach for PV module localization based on structure-from-motion (SfM) [17], [18] to obtain absolute geo-coordinates of the PV modules in a plant. In addition, PV module images are extracted from each video frame and tracked over subsequent frames. Based on the extracted images, module anomalies can be detected with a deep learning classifier [19] and visualized on a map. This enables quick assessment of the health state of the entire PV plant and helps performing targeted repairs. Similarly, module temperatures can be mapped across the PV plant. Temperature mapping allows detecting abnormal modules by comparison with neighbouring modules. This approach can replace more complex deep learning classifiers for detecting abnormal modules, as we will show. As opposed to the related works, our method relies on both visual cues and measured GPS trajectory for georeferencing. This improves robustness to GPS measurement errors and allows to use standard GPS instead of RTK-GPS. We further use videos instead of individual images. Video analysis speeds up data acquisition and works not only with automatic waypoint flights, but also with manual flights performed ad-hoc for small and irregular plants. Videos also yield larger amounts of data as each PV module is captured in multiple video frames, which is beneficial for training machine learning algorithms on the extracted data. Compared to our previous work, requirements on the flight trajectory are less stringent, and, in principle, plants with non-row layouts, such as rooftop plants, can be processed. Having a single tool for different types of PV plants is more cost-effective and requires less maintenance than multiple plant-specific solutions. We also show that for regular plants, multiple rows can be scanned simultaneously, which significantly increases throughput.

## II. METHOD

This section introduces our method for fully automatic extraction and georeferencing of PV modules from aerial IR videos. For an overview see fig. 1. After acquisition with the drone, IR videos of a PV plant are split into individual frames and the GPS trajectory of the drone is extracted and interpolated. Following Bommes et al. [10], PV modules are segmented by Mask R-CNN [20], tracked over subsequent frames, extracted and stored to disk. To georeference PV modules, a subset of keyframes is selected based on travelled GPS distance and visual overlap. Subsequently, a georeferenced 3D reconstruction of the PV plant is obtained by incremental SfM alongside the 6-DOF camera pose of each keyframe. This requires calibrated camera parameters, which are obtained beforehand. The known keyframe poses are then used to triangulate observed PV modules into the 3D reconstruction, yielding the desired module geocoordinates.

### A. Camera Model and Calibration

Several steps of our pipeline use a calibrated pinhole camera model to project 3D scene points into image coordinates and to triangulate image points into a 3D reconstruction of the scene. Lens distortion is modelled by a Brown-Conrady radial distortion model [21] with five distortion coefficients.

Calibration is performed once for each camera using OpenCV's [22] calibration method with around 150 images of a chessboard calibration target (see fig. 2). The target consists of foil patches applied to a polymer panel, providing sufficient contrast in the IR image due to different emissivities. We obtain best results when capturing calibration images outside on a cloudy day.

### B. Drone Flight and Video Acquisition

Our method is intended to be used with IR videos acquired by a drone, which scans one or multiple rows of a PV plant at an altitude of 10 m to 30 m and at a velocity that ensures blur-free images. Acquisition should take place under clearsky conditions and solar irradiance above $700 \, \mathrm{W \, m^{-2}}$. Similar to our previous method [10], both nadiral and non-nadiral videos can be processed and the camera angle and flight velocity may be varied during the flight. For accurate georeferencing of the SfM reconstruction the drone needs to travel a sufficient distance in at least two orthogonal directions. Furthermore, the flight altitude should be kept approximately constant in case standard GPS is used and no accurate altitude measurement is available. These requirements are much less restrictive than those of our previous work, resulting in higher flexibility and robustness. It is, for example, no problem, if the drone moves non-monotonically along a plant row, or if the same row is scanned multiple times. Furthermore, situations, in which the scanned row is cropped at the top or bottom of the frame, or in which other rows become visible in the camera viewport, can be handled.

For compatibility with the remaining processing steps, we split the acquired IR videos into individual 16-bit grayscale images, convert each image to Celsius scale, normalize to the interval $[0, 255]$ using its minimum and maximum temperature value, convert to 8-bit, and finally, perform histogram equalization.

### C. Segmentation, Tracking and Extraction of PV modules

These steps correspond to our previous work [10] and are therefore described only briefly. A Mask R-CNN instance segmentation model, which is trained on a photovoltaic-specific dataset, obtains a binary segmentation mask for each PV module in each video frame. After fitting a quadrilateral to each mask, the underlying image region is extracted, warped to a rectangular region by a homography and stored as a 16-bit radiometric image file. A tracking algorithm associates masks of the same PV module over subsequent frames and assigns a unique tracking ID to each module. The tracking ID is then used to group the extracted image patches of each module.

### D. Preprocessing of the GPS Trajectory

The drone records its latitude and longitude in WGS-84 coordinates at a rate of 1 Hz. As we do not use RTK-GPS, the measured altitude is unreliable and we assume it as unknown in the subsequent steps. To match the rates of GPS measurements to the higher frame rate of the camera, we perform piecewise linear interpolation of the GPS trajectory and sample a GPS position for each frame. Prior to this, we transform the trajectory from WGS-84 coordinates to local tangent plane (LTP) coordinates [23]. LTP coordinates are Cartesian with their origin at or near the inspected site. This enables accurate interpolation and enhances numerical stability in the subsequent SfM procedure.
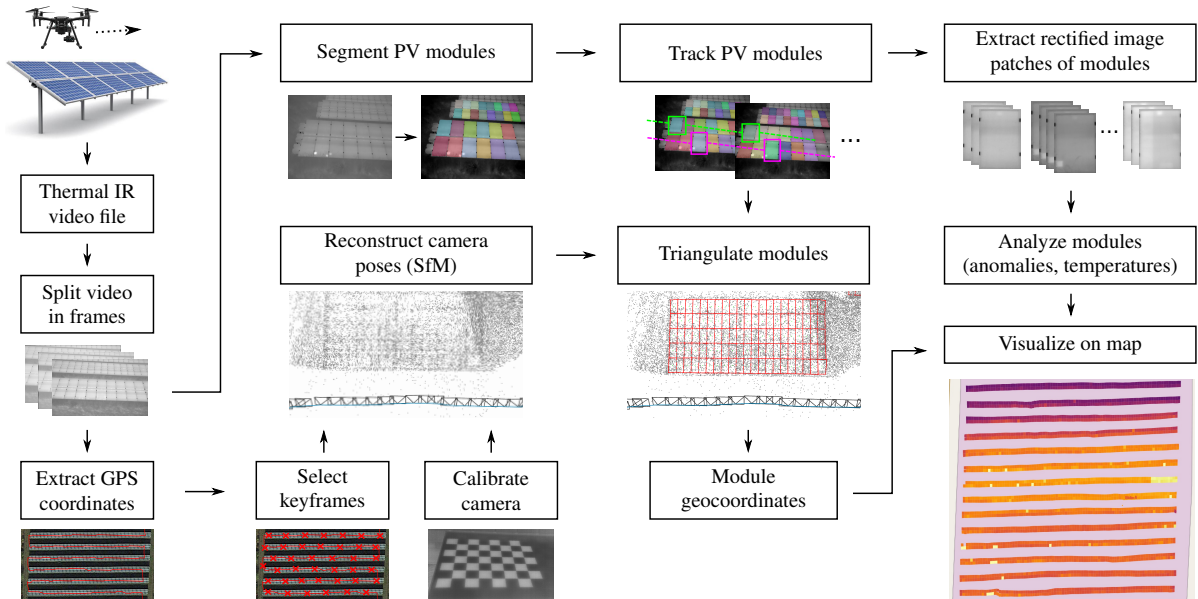
Figure 1: Overview of our method for automatic extraction and georeferencing of PV modules from aerial IR videos.
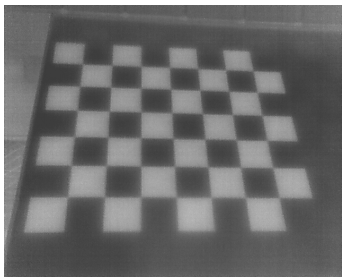


Figure 2: Examplary IR image of the camera calibration target.

### E. Selection of Frames for Reconstruction

In this step we select a subset of partially overlapping video frames for the subsequent SfM procedure, which we call keyframes. Subsampling the frames keeps the computational cost of the SfM procedure, which is quadratic in the number of frames, within an acceptable range. It also decouples SfM from the video frame rate, simplifying the use of different cameras. SfM further benefits from the larger parallax between any two keyframes, ensuring more accurate triangulation of scene points.

We select a frame as a keyframe if i) its distance to the previous keyframe along the GPS trajectory exceeds 0.75 m, or if ii) its intersection over union (IoU) with the previous keyframe is smaller than 85 %. To obtain the IoU, ORB features [24] of the frame and the previous keyframe are extracted and matched. A homography is estimated from the matches, which projects the bounding rectangle of the frame onto that of the previous keyframe. The IoU is then the intersection area of both rectangles divided by their total area.

One advantage of capturing videos over individual images, is the ability to adjust the overlap between images after the data is already captured.

### F. Reconstruction of Camera Poses with SfM

In this step the 6-DOF camera pose of each keyframe is reconstructed using OpenSfM, an incremental SfM library [25]. Inputs are the calibrated camera parameters and the selected keyframes with their GPS positions in LTP coordinates. Due to unavailability of reliable measurements we set the GPS altitude to zero and fix the dilution of precision (DOP) to 0.1 m. Outputs are the rotation and translation of each keyframe in a LTP coordinate system and a 3D point cloud of reconstructed scene points, which is not further needed. An example is shown in fig. 4a. In the following, we explain briefly how the SfM library works.

*1) Feature detection and matching:* The SfM library first finds HAHOG features [26], i.e. characteristic points, in each keyframe. Overlapping frames are then found by matching these features between pairs of frames. To limit the search space matches are computed only for frame pairs which are at most 15 m apart.

*2) Initialization of the reconstruction:* One frame pair with sufficient parallax is selected for initialization of the reconstruction. The pose of the first frame is set as world coordinate origin. The pose of the second frame relative to the first frame is estimated with the five-point algorithm [27] or, in case of a planar scene, by decomposing a homography [28]. An initial set of 3D scene points is triangulated from the matched feature points in both frames.

*3) Iterative reconstruction:* Starting from the initial frame pair the other keyframes are added incrementally to the reconstruction. In each iteration the frame with most matches to any of the reconstructed frames is selected. Its pose is estimated from observed 3D scene points in the reconstruction and their corresponding 2D projections in the frame by solving the perspective-n-point problem [29]. Subsequently, new scene points are triangulated from feature points shared between the newly added frame and other frames in the reconstruction. Afterwards, the entire reconstruction is rigidly transformed, so that camera positions best

align with their measured GPS positions. In regular intervals bundle adjustment optimizes all reconstructed camera poses and scene points simultaneously by minimizing the reprojection error of the scene points in all frames. Here, camera positions are kept close to their measured GPS positions. Additionally, camera parameters are refined. We use these refined parameters in all subsequent steps.

*4) Post-processing of the reconstruction:* Under some circumstances the reconstruction of a long video sequence can fail partially. This results in multiple partial reconstructions each with a different LTP coordinate origin. To register all partial reconstructions in a common LTP coordinate frame, we transform each partial reconstruction to WGS-84 coordinates using the reconstruction-specific LTP origin. We then transform the reconstruction back to LTP coordinates, this time using the common LTP origin. This common origin is arbitrarily set to the origin of the first partial reconstruction.

### G. Obtaining Geocoordinates of PV Modules

Once the keyframe poses are reconstructed, we triangulate the corner/center points of segmented PV modules into the reconstruction, yielding corresponding LTP geocoordinates. Examples of this are shown in fig. 4b and 4c. Due to inaccuracies in the module segmentation a robust triangulation procedure and subsequent refinement of the obtained LTP coordinates are required.

*1) Triangulation of PV modules:* For each tracked module, we obtain pixel-coordinates of the four corner points and the center point in all keyframes, in which the module is visible. Modules observed in less than two keyframes are skipped as they are likely spurious detections and triangulation is impossible. The five module points are then undistorted with the calibrated Brown-Conrady model, and triangulated from all possible pairs of keyframes, in which they are observed. The so triangulated points are only retained if the following two conditions are met: i) The angle between the two viewing rays is larger than $1°$ for all five points, and ii) none of the reprojection errors of the five points exceeds a threshold of $5$ pixels. As there are typically several pairs of keyframes observing the same module, we get a noisy set of triangulated modules (see fig. 3a). We fuse them robustly by computing the median of corresponding points (see fig. 3b).

*2) Merging of duplicate detections:* A PV module may be lost during tracking and reappear a few frames later with a different tracking ID, resulting in multiple overlapping triangulations of the module in the reconstruction (see fig. 3b). This step identifies and fuses such duplicates. To this end, for each keyframe, all triangulated modules are projected back into the frame. Two or more modules are identified as overlapping if the mean Euclidean distance between the corresponding four corner points and the center point is smaller than $20$ pixels. To merge overlapping modules, module points are re-triangulated according to the procedure above, this time using all keyframes of the overlapping modules. The result is shown in fig. 3c.

*3) Refinement of triangulated modules:* The triangulation is further refined by moving nearby PV module corners closer together, yielding a smoother result (see fig. 3d). To this end, we build a graph containing all triangulated module points as vertices $\mathcal{P}$ and edges between those points that are close to another. Points are considered close if their Euclidean distance is
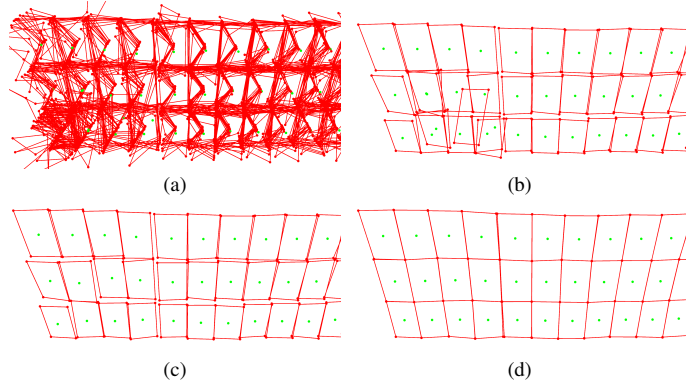


Figure 3: Steps of PV module triangulation: a) initial triangulation from all keyframe pairs, b) after computing median points, c) after merging duplicates, and d) after iterative refinement.

at most 1 m in the reconstruction and 20 pixels in projected image coordinates. Given this graph we use the g2o graph optimization framework [30] to obtain refined module points $\mathcal{P}^*$ by optimizing the following objective

$$\mathcal{P}^* = \arg\min_{\mathcal{P}} \sum_{\langle i,j \rangle \in \mathcal{C}} \rho_h \left( \mathbf{e}_{ij}^\mathsf{T} \mathbf{\Omega}_{ij} \mathbf{e}_{ij} \right). \tag{1}$$

Here, $\mathcal{C}$ is the set of pairs of indices for which an edge exist, $\mathbf{e}_{ij} = \mathbf{P}_i - \mathbf{P}_j$ is the difference between two points, and $\mathbf{\Omega}_{ij}$ is the information matrix, which we set to the identity matrix. The robust Huber cost function $\rho_h$ reduces the impact of outliers.

We do not apply any further refinements, such as aligning surface normals of modules, or enforcing a rectangular shape, to retain maximum flexibility of our method with respect to the layout of PV modules.

### H. Final Dataset Structure

After triangulation, LTP coordinates of PV module corners and center points are transformed back to WGS-84 coordinates and stored in a GeoJSON file together together with the module tracking ID. Similarly, the extracted IR patches of each module are stored as image files in a directory named after the tracking ID. This dataset structure allows for analysis of the extracted image patches and visualization of results on a map.

## III. EXPERIMENTS & RESULTS

In this section, we apply our method to five different PV plants. We quantify the module extraction success rate together with the georeferencing error, and validate the tools' ability to process multiple plant rows in parallel. We further map predicted module anomalies and module temperatures and investigate, to what extent the temperature distribution can replace a deep learning-based classifier for the detection of abnormal modules.

### A. Video Dataset

To validate our method, we acquire IR videos of five PV plants in Germany with a combined 35084 PV modules using a drone of type DJI Matrice 210. Tab. 1 contains details of the PV plants, drone flights and weather conditions during data acquisition. Plants A to D are large-scale open-space plants with regular row-based layouts. Plant E is a less regular arrangement
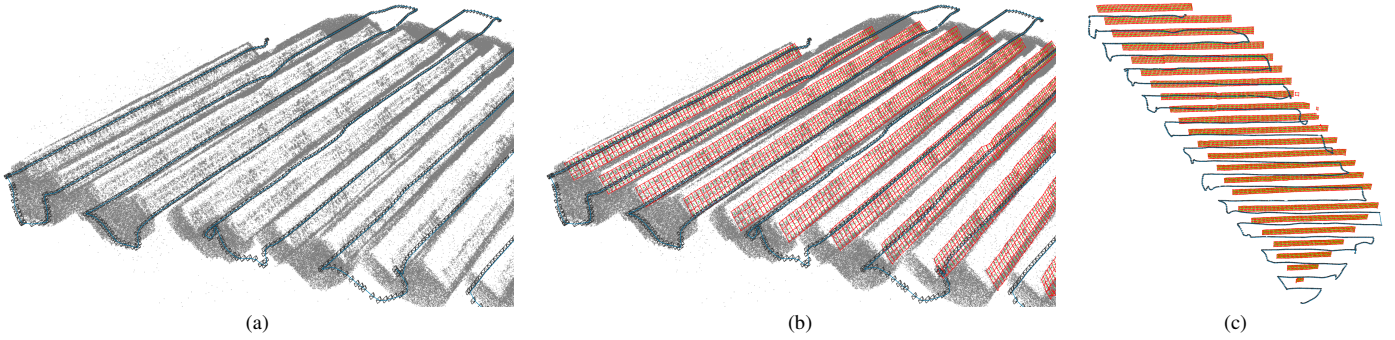
Figure 4: Reconstruction results of the SfM procedure: a) reconstructed feature points (grey) and camera poses (blue line and black camera frustrums), b) with triangulated PV modules, c) top-down view on the triangulated modules of an entire PV plant.
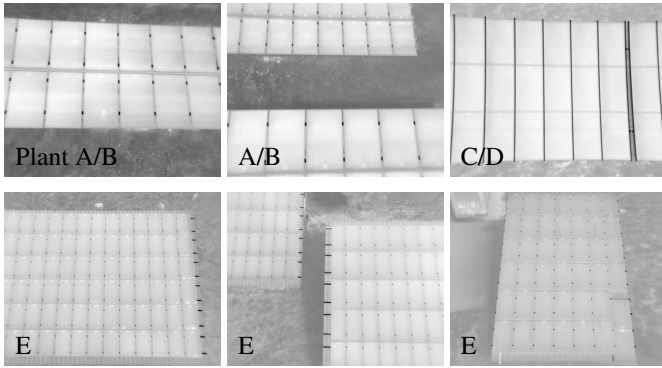


Figure 5: Exemplary IR video frames of the PV plants in our study.



Figure 6: Map with estimated geocoordinates of PV modules in plant B.

of PV arrays mounted on several rooftops. All plants consist of 60-cell crystalline silicon modules. Videos of plants A, B and E are recorded by a DJI Zenmuse XT2 thermal camera with $640 \times 512$ pixels resolution, 8 Hz frame rate and 13 mm focal length. For plants C and D we use another variant of the DJI Zenmuse XT2 with 30 Hz frame rate and 19 mm focal length. Fig. 5 shows exemplary video frames from our dataset. Note, that we could not use the dataset from our previous work due to an exchange of our camera and unavailability of calibrated camera parameters.

### B. Module Extraction Success Rate

We apply our method on the five PV plants in our video dataset and extract over 2.2 million IR images of the 35084 PV modules of all plants (on average 64.1 images per module). Additionally, geocoordinates are obtained for each module as exemplary shown for plant B in fig. 6 (for the other plants see appendix A-A). As detailed in tab. 2, 99.3 % of all modules are successfully extracted and georeferenced. As compared to the 87.8 % success rate of our previous work, we now miss only one in 140 modules instead of one in eight. This 18-fold improvement of the extraction success rate is mostly due to the higher robustness of our new method to errors in the data acquisition process, such as cropping of the scanned row, double acquisition of the same row, or small loops in the drone trajectory. Such an error caused our previous method to loose all modules in an entire plant row. Opposed to that, our new method can handle many of those acquisition errors, and fails at most locally for a few modules. The almost perfect success rate of our new method is important in practice, as every missed module
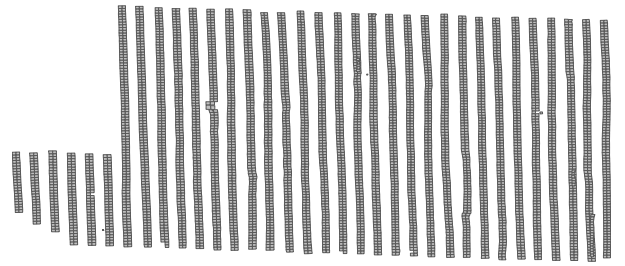
is a missed opportunity to increase yield and profitability of the plant. Furthermore, safety critical anomalies (e.g. fire hazards) could be overlooked.

Tab. 2 also contains a detailed breakdown of the failure modes of the 234 modules missed by our method. In total, 13 modules exhibit substantial distortions, and 40 modules are missing in the reconstructions, because they are not covered by sufficiently many video frames to be accurately triangulated. Another 181 modules appear multiple times in the reconstruction because the merging procedure (sec. II-G.2) failed. This happens for modules appearing in video frames, which are temporally far apart. As these frames have a large relative pose error (due to the use of standard GPS) the triangulated modules do not align well and can not be merged correctly. This can most likely be mitigated by using RTK-GPS. Finally, there are 24 false positive modules corresponding to other objects, which are mistaken as PV modules by the Mask R-CNN segmentation model.

### C. Georeferencing Accuracy

In this section we quantify the accuracy of the georeferenced PV module locations in terms of the root mean square error (RMSE) between estimated LTP geocoordinates $(\hat{e}, \hat{n})$ and ground truth geocoordinates $(e, n)$ of $N$ selected PV modules

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (\hat{e}_i - e_i)^2 + \frac{1}{N} \sum_{i=1}^{N} (\hat{n}_i - n_i)^2}. \quad (2)$$

Here, $e$ and $n$ are the east and north positions in the LTP coordinate system. The altitude coordinate is omitted as for mapping only the horizontal error is of interest. Because point

Table 1: Details of PV plants, drone flights and weather conditions in our study. Start and end time are in UTC+2:00. Peak velocity is the 99.9 % quantile of all velocities estimated from position and time delta of subsequent video frames. Weather data is from Deutscher Wetterdienst [31]. We report mean and standard deviation of measurements taken at the nearest weather station every 10 minutes during the flight.

| Plant Details | | | Flight Details | | | | | Weather Conditions | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ID | # Modules | Type | Start time | End time | # Frames | Distance | Peak velocity | Air temp. | Global radiation | Wind speed | Wind dir. |
| A | 13 640 | open-space | 10:28:48 | 12:40:14 | 42 272 | 7612 m | 4.1 m s$^{-1}$ | 25.9 ± 0.5 °C | 39.7 ± 1.8  J cm$^{-2}$ | 2.8 ± 0.4 m s$^{-1}$ | WSW |
| B | 5280 | open-space | 13:37:59 | 14:14:30 | 13 715 | 2929 m | 4.1 m s$^{-1}$ | 26.8 ± 0.4 °C | 30.3 ± 6.1  J cm$^{-2}$ | 3.6 ± 0.4 m s$^{-1}$ | SW |
| C | 6210 | open-space | 12:16:21 | 12:39:05 | 34 593 | 2468 m | 6.6 m s$^{-1}$ | 22.3 ± 0.3 °C | 46.2 ± 10.6 J cm$^{-2}$ | 5.7 ± 0.6 m s$^{-1}$ | WNW |
| D | 8460 | open-space | 11:01:00 | 11:33:34 | 50 348 | 3479 m | 7.2 m s$^{-1}$ | 23.4 ± 0.2 °C | 57.4 ± 1.5  J cm$^{-2}$ | 2.0 ± 0.8 m s$^{-1}$ | W |
| E | 1494 | rooftops | 11:30:24 | 11:54:02 | 4527 | 485 m | 4.2 m s$^{-1}$ | 19.0 ± 0.3 °C | 42.8 ± 3.0  J cm$^{-2}$ | 2.6 ± 0.3 m s$^{-1}$ | SE |

Table 2: Numbers of PV modules and module image patches extracted from the plants in our dataset. Failures are missing (MM), duplicate (DP), distorted (DS) and false positive (FP) modules.

| Plant | # Modules | | # Patches | | # Failures | | | |
|---|---|---|---|---|---|---|---|---|
| | Total | Extracted | Extracted | ⌀/Module | MM | DP | DS | FP |
| A | 13 640 | 13 463 | 398 221 | 29.2 | 18 | 152 | 7 | 13 |
| B | 5280 | 5246 | 140 120 | 26.6 | 6 | 28 | 0 | 3 |
| C | 6210 | 6200 | 635 437 | 102.3 | 4 | 1 | 5 | 2 |
| D | 8460 | 8453 | 936 867 | 110.8 | 7 | 0 | 0 | 4 |
| E | 1494 | 1488 | 138 008 | 90.8 | 5 | 0 | 1 | 2 |
| Total | 35 084 | 34 850 | 2 248 653 | 64.1 | 40 | 181 | 13 | 24 |



Figure 7: RMSE between ground truth and estimated horizontal geo-coordinates of PV modules in plant A. The RMSE is computed for selected points (black dots) and linearly interpolated on a $4000 \times 800$ grid (heatmap).

Table 3: Results for simultaneous scanning of one, two and three rows.

| | One Row | Two Rows | Three Rows |
|---|---|---|---|
| Flight distance | 1307 m | 681 m | 461 m |
| Flight duration | 707 s | 338 s | 189 s |
| Average module resolution | 141 px × 99 px | 73 px × 50 px | 46 px × 33 px |
| Module throughput | 3.36 s$^{-1}$ | 7.03 s$^{-1}$ | 12.57 s$^{-1}$ |

correspondences have to be found manually, we select only every 11th module in every second row of the plant and consider only the top-left module corners. We further limit the accuracy analysis to plant A, as it is the largest plant in our dataset and the only one, for which a ground truth is available. Ground truth positions are obtained from an orthophoto of the plant. This is possible, as this orthophoto exhibits a small RMSE of less than 2 cm, facilitated by the use of RTK-GPS, ground control points, high-resolution visual imagery and a higher flight altitude.

The RMSE for the entire plant is 5.87 m. This is close to the expected 4.9 m accuracy of GPS under open sky conditions [32]. However, the RMSE is not constant for the entire plant, but instead smoothly increases from 0.42 m in the east to 9.39 m in the west. This also becomes evident in fig. 7, which shows the spatial interpolation of the RMSE over the entire plant. This error drift in the SfM reconstruction is most likely caused by the low accuracy and unknown DOP of the measured GPS trajectory of the drone. As the SfM reconstruction consist of seven partial reconstructions and we do not use ground control points, another possible cause is misalignment of the partial reconstructions.

To analyze the distortion of each individual row, we remove the trend in the RMSE distribution. To this end, we align each row with the respective ground truth positions prior to computing the RMSE for the row. The resulting per-row RSME values range from 0.22 m to 0.82 m, indicating low distortion of individual rows. Due to this, accurate localization of PV modules within the plant is possible, despite the large absolute RMSE of 5.87 m.

### D. Simultaneous Processing of Multiple Rows

One important advantage of our new method is the ability to process multiple PV plant rows simultaneously. We validate this experimentally by acquiring IR videos of the first 12 rows of plant A. We perform three flights, scanning one, two and three rows at a
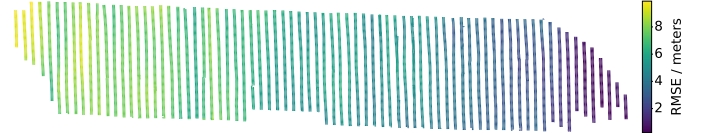
time. Fig. 8 shows exemplary video frames of each flight as well as the reconstructions of modules and flight trajectories produced by our method.

As reported in tab. 3, scanning two and three rows simultaneously speeds up the flight duration by a factor of 2.1 and 3.7, respectively. Module throughput increases accordingly from 3.36 s$^{-1}$ to 7.03 s$^{-1}$ and 12.57 s$^{-1}$. This means, scanning all 2376 modules of the 12 selected rows takes only 338 or 189 seconds when scanning two or three rows at a time. Additionally, flight distance decreases by a factor of 1.9 and 2.8. This has the benefit of increasing the range of the drone before a battery change is needed. The cost for the improvement in throughput is a two- or threefold reduction in the resolution of extracted module images. Furthermore, we found the manual flight is slightly more complicated when scanning three rows at a time instead of one or two, because it is easier to miscount the rows when shifting over to the next row triplet. However, this is not a limitation when flying autonomously.

This experiment confirms the ability of our method to significantly increase throughput simply by scanning more than one plant row at a time. This is highly relevant in practice, as it significantly reduces duration and cost of the inspection. It is also an improvement over our previous method [10], which could process only one row at a time.

### E. Mapping Module Anomalies

In this section we apply a deep learning-based binary classifier to the extracted IR image patches of each PV module in plant A,
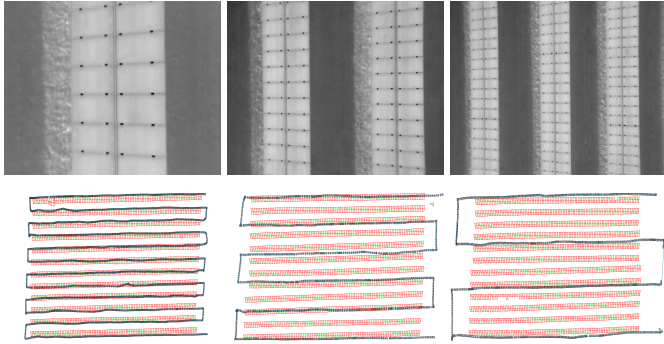
Figure 8: Top row: Exemplary video frames for scanning one, two and three PV plant rows simultaneously. Bottom row: Resulting reconstructions of modules and flight trajectory.

which predicts whether the module is abnormal or not. [1] We then use the estimated module geocoordinates to visualize the distribution of abnormal modules on a map (see fig. 9). Since there are multiple images for each module, we can plot the fraction of images, in which a module is predicted as abnormal. We call this the *anomaly ratio*. As opposed to a simple binary prediction, the anomaly ratio is an approximate indicator for the severity of a module anomaly. This is, because for severe, i.e. clearly visually expressed, anomalies the classifier is more confident, reaching a larger consensus of its predictions over all images of a module.

The so obtained anomaly map enables not only targeted repairs of severely abnormal modules, but also facilitates the identification of fundamental problems of the plant. For the analyzed plant, we find for example, that anomalies occur much more frequently in the bottom row, where modules are closer to the ground, rather than in the top row. A possible explanation for this is the intrusion of moisture into the PV modules near the ground. Being aware of such an issue allows the operator to monitor affected modules more thoroughly and to take action to prevent further damage to the plant.

*F. Mapping Module Temperatures*

Apart from module anomalies, we visualize the spatial distribution of module temperatures in plant A (see fig. 10). Temperatures are obtained from the extracted IR image patches of each module and plotted on a map using the module geocoordinates. For each module the maximum, minimum, mean or median temperature over the module area can be computed. Prior to this, we cut away a few pixels (5 % of the image width) from the image borders to ignore module frames and mounting brackets. To obtain a final temperature value for each module, we take the mean over the values estimated for each of the image patches of the module. As opposed to using a single representative image patch or the maximum over all patches, the mean is more robust to artifacts, which may be present in some of the module images. Of both mean and maximum temperature distributions, we find the maximum temperatures (see fig. 10b) more informative as they are sensitive to the local hot spots typically occurring in abnormal modules.

---

[1]We use the ResNet-34 convolutional neural network (CNN) classifier from Bommes et al. [19], which is trained with a supervised cross-entropy loss on labelled IR module patches of plant B in the dataset of the original work.
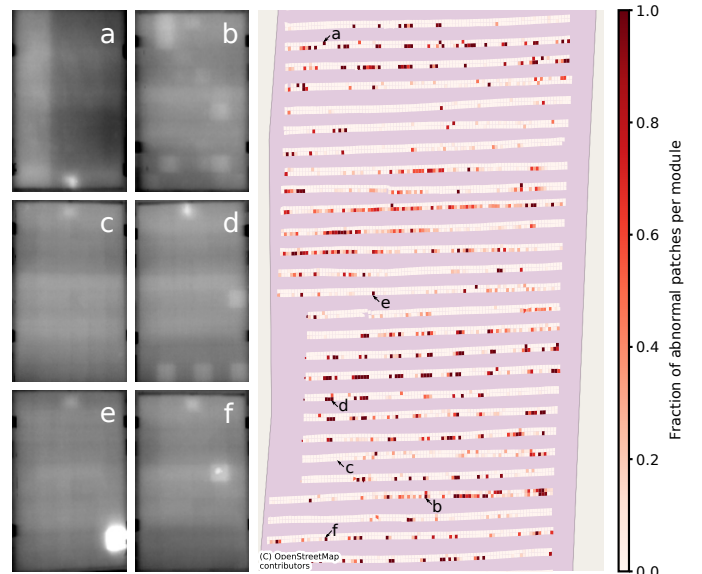


Figure 9: Map of predicted module anomalies in a section of plant A. On the left, IR images of the modules highlighted on the map are shown. Their temperature range is 30 °C (black) to 50 °C (white).

However, both mean and maximum module temperatures reveal the global temperature distribution of the plant, which is not constant, but exhibits a low-frequency pattern with temperature differences of up to 15 K. As module images are acquired over a duration of 132 minutes, possible explanations for this pattern are slow changes in the solar irradiance [33], cloud cover [34], [35], air temperature, wind speed, and camera temperature [36]. The temperature distribution is also affected by local differences in the radiative and convective heat transfer, and by the number of neighbouring modules, leading to cooler modules at the edges of each plant row [37]. Direct use of this temperature distribution for anomaly detection is not possible, as there is no common threshold value, which separates normal from abnormal modules. To account for this, we compute local temperature differences between neighbouring modules. Specifically, we subtract the median of the maximum module temperatures of all neighbouring modules within a radius of 7 m from each module. Fig. 10c shows the resulting relative maximum module temperatures. These relative temperatures are independent of the changes in environmental conditions during the flight, and consequently facilitate detection of abnormal modules by selecting a suitable temperature threshold.

Apart from locally overheated modules, the temperature distribution allows to identify string anomalies. For example in plant A, there is an inactive string (in the middle of the 19th row counted from the bottom), which is clearly visible in the temperature map (see fig. 10). Being able to identify such anomalies is important, as an entire inactive string causes large yield and power losses.

*G. Anomaly Detection with Module Temperatures*

In this section we analyze, whether the relative maximum module temperatures (see fig. 10c) alone are sufficient to accurately identify abnormal modules, and whether they can replace the more complex deep learning-based anomaly classifier from
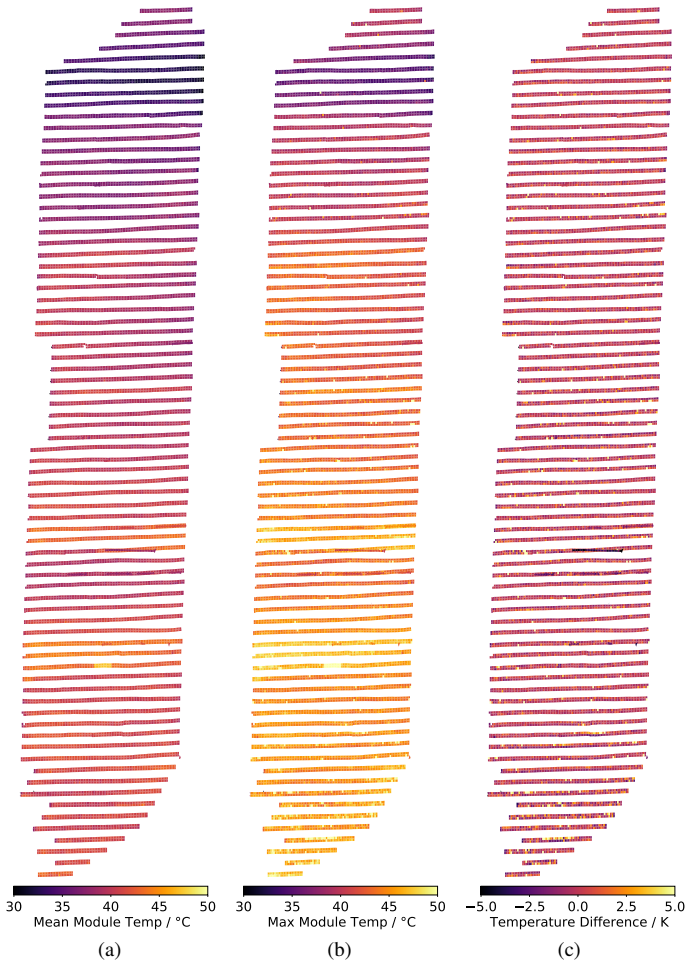
Figure 10: Map of plant A showing the distribution of mean (a) and maximum (b) module temperatures. Each value is the average over all images showing a module. For higher contrast temperatures are clipped below 30 °C and above 50 °C. In (c) local differences of the maximum module temperature are emphasized by subtracting the median temperatures of neighbouring modules within a radius of 7 m.

sec. III-E. To this end, we manually label all 13463 modules of plant A as healthy or as abnormal with one out of the ten anomaly classes shown in fig. 11. For each module a binary anomaly prediction is obtained by comparing its relative maximum module temperature to a specified threshold value. Similarly, for the deep learning classifier we compare the anomaly ratio (see fig. 9) of the module to a threshold value.

As common in the anomaly detection literature [38]–[40], we quantify the anomaly detection performance as the area under the receiver operating characteristic (AUROC). This metric is independent of a specific threshold value, and therefore, enables a fair comparison of both classifiers, which depend differently on their threshold values. AUROC is defined as the area under the true positive rate $TPR = TP/(TP + FN)$ plotted against the false positive rate $FPR = FP/(FP + TN)$ at different threshold values. Here, TP and TN are the numbers of correctly classified abnormal and healthy modules, and FP is the number of healthy modules falsely classified as abnormal and FN the number of abnormal modules falsely classified as healthy.
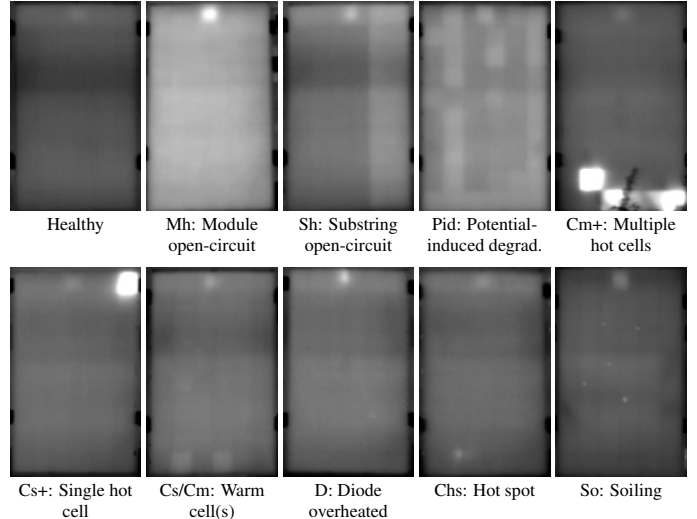


Figure 11: Exemplary IR images of the module anomaly classes in our analysis. Temperature ranges from 30 °C (black) to 60 °C (white). The figure is adopted from our previous work [10].

Tab. 4 reports the resulting AUROC scores for each anomaly class and an overall AUROC score, which considers all anomaly classes. The results indicate that both module temperature distribution and deep learning classifier perform equally and nearly perfect on severe anomalies (Sh, Cs+, Cm+), while they complement each other on the less severe anomaly classes. The deep learning classifier performs better for Pid, Cs and Cm anomalies, which are characterized by low temperature gradients, and are therefore not as accurately identifiable by the temperature distribution. On the contrary, the temperature distribution performs better for D, So, Chs anomalies, which have large temperature gradients and a small spatial extent. The small spatial extent makes detection of these anomalies difficult for a convolutional neural network. Both classifiers perform poorly on homogeneously overheated modules (Mh) because their predictions are based on temperature differences within the image (deep learning classifier) or within the local neighbourhood of modules (temperature distribution). However, using absolute instead of the relative maximum module temperatures allows to accurately identify Mh anomalies (see sec. III-F).

Summing up, the module temperature distribution can supersede a complex deep learning-based anomaly classifier for the detection of seven out of ten common module anomalies in a PV plant. This is beneficial for practical applications because of the simplicity, higher speed and better interpretability of the temperature distribution. Furthermore, no training is required, which saves the effort of creating a labelled training dataset and circumvents the issue of having to generalize from the training to the test dataset.

## IV. CONCLUSION

In this work, we developed a method for the automatic extraction and georeferencing of PV modules from aerial IR videos, which can be used for fully automatic PV plant inspection. One possible future improvement of our method is the use of centimeter-accurate RTK-GPS instead of standard GPS, which

Table 4: AUROC scores for the detection of module anomalies in plant A by the module temperature distribution versus a deep learning classifier. Scores of the better classifier and scores above 99 % are in bold.

| Anomaly | # Modules | AUROC / % | |
| --- | --- | --- | --- |
| | | Temp. Distribution | Deep Learning Clf. |
| Mh | 22 | **59.04** | 52.73 |
| Sh | 32 | **99.78** | **99.95** |
| Pid | 149 | 76.01 | **95.92** |
| Cm+ | 11 | **100.00** | **99.61** |
| Cs+ | 30 | **99.81** | **99.90** |
| Cm | 420 | 60.71 | **86.16** |
| Cs | 294 | 61.65 | **78.84** |
| D | 294 | **99.31** | 62.06 |
| Chs | 23 | **89.64** | 81.53 |
| So | 136 | **79.71** | 61.29 |
| Overall | 1411 | 74.31 | **78.04** |

could reduce the RMSE of module geocoordinates and stabilize the SfM procedure. Similarly, accuracy and stability of the SfM procedure could be improved by using visual videos instead of IR videos, as visual videos provide a higher resolution, wider viewing angle, color information and exhibit lower variation of image intensities [41]. However, this requires accurate temporal synchronization and spatial registration of the visual and IR stream, which is a challenging task. Another future direction is the correlation of the obtained temperature distribution with electrical data, such as power and yield, which could provide additional insights into the health state of a PV plant. Finally, our method could be extended for augmented reality applications by rendering a more immersive 3D model of the plant with overlaid textures, module images and interactive reports for each module.

## V. ACKNOWLEDGEMENTS

## REFERENCES

[1] F. Bizzarri, S. Nitti, and G. Malgaroli, "The use of drones in the maintenance of photovoltaic fields," *E3S Web of Conferences*, vol. 119, 2019. DOI: 10.1051/e3sconf/201911900021.

[2] Y. Zefri, I. Sebari, H. Hajji, and G. Aniba, "Developing a deep learning-based layer-3 solution for thermal infrared large-scale photovoltaic module inspection from orthorectified big UAV imagery data," *International Journal of Applied Earth Observation and Geoinformation*, vol. 106, no. 5, p. 102 652, 2022. DOI: 10.1016/j.jag.2021.102652.

[3] R. Pierdicca, M. Paolanti, A. Felicetti, F. Piccinini, and P. Zingaretti, "Automatic faults detection of photovoltaic farms: solAir, a deep learning-based system for thermal images," *Energies*, vol. 13, p. 6496, 2020. DOI: 10.3390/en13246496.

[4] C. Henry, S. Poudel, S.-W. Lee, and H. Jeong, "Automatic detection system of deteriorated PV modules using drone with thermal camera," *Applied Sciences*, vol. 10, p. 3802, 2020. DOI: 10.3390/app10113802.

[5] V. Carletti, A. Greco, A. Saggese, and M. Vento, "An intelligent flying system for automatic detection of faults in photovoltaic plants," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, pp. 2027–2040, 2019. DOI: 10.1007/s12652-019-01212-6.

[6] S. Gallardo-Saavedra, L. Hernández-Callejo, and O. Duque-Perez, "Technological review of the instrumentation used in aerial thermographic inspection of photovoltaic plants," *Renewable and Sustainable Energy Reviews*, vol. 93, pp. 566–579, 2018. DOI: 10.1016/j.rser.2018.05.027.

[7] F. Grimaccia, S. Leva, and A. Niccolai, "PV plant digital mapping for modules' defects detection by unmanned aerial vehicles," *IET Renewable Power Generation*, vol. 11, no. 10, pp. 1221–1228, 2017. DOI: 10.1049/iet-rpg.2016.1041.

[8] G. Francesco, L. Sonia, and N. Alessandro, "A semi-automated method for defect identification in large photovoltaic power plants using unmanned aerial vehicles," in *IEEE Power Energy Society General Meeting*, Portland, OR, USA, 2018, pp. 1–5.

[9] M. Aghaei, S. Leva, and F. Grimaccia, "PV power plant inspection by image mosaicing techniques for IR real-time images," in *IEEE 43rd Photovoltaic Specialists Conference*, Portland, OR, USA, 2016, pp. 3100–3105.

[10] L. Bommes, T. Pickel, C. Buerhop-Lutz, J. Hauch, C. Brabec, and I. M. Peters, "Computer vision tool for detection, mapping, and fault classification of photovoltaics modules in aerial IR videos," *Progress in Photovoltaics: Research and Applications*, vol. 29, no. 12, pp. 1236–1251, 2021. DOI: 10.1002/pip.3448.

[11] A. Niccolai, F. Grimaccia, and S. Leva, "Advanced asset management tools in photovoltaic plant monitoring: UAV-based digital mapping," *Energies*, vol. 12, no. 24, p. 4736, 2019. DOI: 10.3390/en12244736.

[12] P. Addabbo, A. Angrisano, M. L. Bernardi, G. Gagliarde, A. Mennella, M. Nisi, and S. L. Ullo, "UAV system for photovoltaic plant inspection," *IEEE Aerospace and Electronic Systems Magazine*, vol. 33, no. 8, pp. 58–67, 2018. DOI: 10.1109/MAES.2018.170145.

[13] M. Nisi, F. Menichetti, V. Bramante, T. Tr, B. Muhammad, and R. Prasad, "EGNSS high accuracy system improving photovoltaic plant maintenance using RPAS integrated with low-cost RTK receiver," in *Global Wireless Summit*, Aarhus, Denmark, 2016.

[14] D. H. Lee and J. H. Park, "Developing inspection methodology of solar energy plants by thermal infrared sensor on board unmanned aerial vehicles," *Energies*, vol. 12, no. 15, p. 2928, 2019. DOI: 10.3390/en12152928.

[15] Y. Zefri, A. ElKettani, I. Sebari, and S. A. Lamallam, "Thermal infrared and visual inspection of photovoltaic installations by UAV photogrammetry—Application case: Morocco," *Drones*, vol. 2, no. 4, p. 41, 2018. DOI: 10.3390/drones2040041.

[16] I. ( Tsanakas, L. Ha, and F. Al Shakarchi, "Advanced inspection of photovoltaic installations by aerial triangulation and terrestrial georeferencing of thermal/visual imagery," *Renewable Energy*, vol. 102 (Part A), pp. 224–233, 2016. DOI: 10.1016/j.renene.2016.10.046.

[17] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, 2016, pp. 4104–4113.

[18] S. Agarwal, N. Snavely, I. Simon, S. M. Seitz, and R. Szeliski, "Building rome in a day," in *IEEE International Conference on Computer Vision*, Kyoto, Japan, 2009, pp. 72–79.

[19] L. Bommes, M. Hoffmann, C. Buerhop-Lutz, T. Pickel, J. Hauch, C. Brabec, A. Maier, and I. M. Peters, "Anomaly detection in infrared images of photovoltaic modules using supervised contrastive learning," *Progress in Photovoltaics: Research and Applications*, 2022. DOI: 10.1002/pip.3518.

[20] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *IEEE International Conference on Computer Vision*, Venice, Italy, 2017, pp. 2980–2988.

[21] D. C. Brown, "Decentering distortion of lenses," *Photogrammetric Engineering*, vol. 32, no. 3, pp. 444–462, 1966.

[22] G. Bradski, "The OpenCV library," *Dr. Dobb's Journal of Software Tools*, vol. 120, pp. 122–125, 2000.

[23] W. Torge and J. Müller, *Geodesy*. De Gruyter, 2012. DOI: 10.1515/9783110250008.

[24] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *International Conference on Computer Vision*, Barcelona, Spain, 2011, pp. 2564–2571.

[25] Mapillary, *Opensfm*, https://github.com/mapillary/OpenSfM, 2021.

[26] K. Mikolajczyk and C. Schmid, "An affine invariant interest point detector," in *European Conference on Computer Vision*, Copenhagen, Denmark, 2002, pp. 128–142.

[27] D. Nister, "An efficient solution to the five-point relative pose problem," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 6, pp. 756–770, 2004. DOI: `10.1109/TPAMI.2004.17`.

[28] O. D. Faugeras and F. Lustman, "Motion and structure from motion in a piecewise planar environment," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 02, no. 03, pp. 485–508, 1988. DOI: `10.1142/S0218001488000285`.

[29] V. Lepetit, F. Moreno-Noguer, and P. Fua, "EPnP: An accurate O(n) solution to the PnP problem," *International Journal of Computer Vision*, vol. 81, no. 2, pp. 1573–1405, 2008. DOI: `10.1007/s11263-008-0152-6`.

[30] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "G2o: A general framework for graph optimization," in *IEEE International Conference on Robotics and Automation*, Shanghai, China, 2011, pp. 3607–3613.

[31] DWD Climate Data Center (CDC), *Recent 10-minute station observations of air temperature, global radiation, wind speed and wind direction for germany, quality control not completed yet, version recent, last accessed: 08/10/2021*.

[32] F. van Diggelen and P. K. Enge, "The world's first GPS MOOC and worldwide laboratory using smartphones," in *28th International Technical Meeting of the Satellite Division of The Institute of Navigation*, Tampa, FL, USA, 2015, pp. 361–369.

[33] P. Rajendran and H. Smith, "Modelling of solar irradiance and daylight duration for solar-powered UAV sizing," *Energy Exploration & Exploitation*, vol. 34, no. 2, pp. 235–243, 2016. DOI: `10.1177/0144598716629874`.

[34] R. Nomura, T. Harigai, Y. Suda, and H. Takikawa, "Second by second prediction of solar power generation based on cloud shadow behavior estimation near a power station," *AIP Conference Proceedings*, vol. 1807, no. 1, p. 020 024, 2017. DOI: `10.1063/1.4974806`.

[35] K. Lappalainen and S. Valkealahti, "Analysis of shading periods caused by moving clouds," *Solar Energy*, vol. 135, pp. 188–196, 2016. DOI: `10.1016/j.solener.2016.05.050`.

[36] Q. Wan, B. Brede, M. Smigaj, and L. Kooistra, "Factors influencing temperature measurements from miniaturized thermal infrared (TIR) cameras: A laboratory-based approach," *Sensors*, vol. 21, no. 24, 2021. DOI: `10.3390/s21248466`.

[37] J. Denz, C. Buerhop, C. Camus, I. Kruse, T. Pickel, B. Doll, J. Hauch, and C. Brabec, "Quantitative assessment of the power loss of silicon PV modules by IR thermography and its practical application in the field," in *37th European Photovoltaic Solar Energy Conference and Exhibition*, Lisbon, Spain, 2020, pp. 1542–1547.

[38] I. Golan and R. El-Yaniv, "Deep anomaly detection using geometric transformations," in *Advances in Neural Information Processing Systems (NIPS)*, Montréal, Canada, 2018, pp. 9781–9791.

[39] L. Bergman, N. Cohen, and Y. Hoshen, "Deep nearest neighbor anomaly detection," *arXiv preprint arXiv:2002.10445*, 2020. arXiv: `2002.10445`. [Online]. Available: `http://arxiv.org/abs/2002.10445`.

[40] D. Hendrycks, M. Mazeika, and T. Dietterich, "Deep anomaly detection with outlier exposure," in *International Conference on Learning Representations (ICLR)*, New Orleans, LA, USA, 2019.

[41] S.-S. Lin, "Review: Extending visible band computer vision techniques to infrared band images," *Technical Reports (CIS)*, 2001.

# A. APPENDIX

(b) Plant C

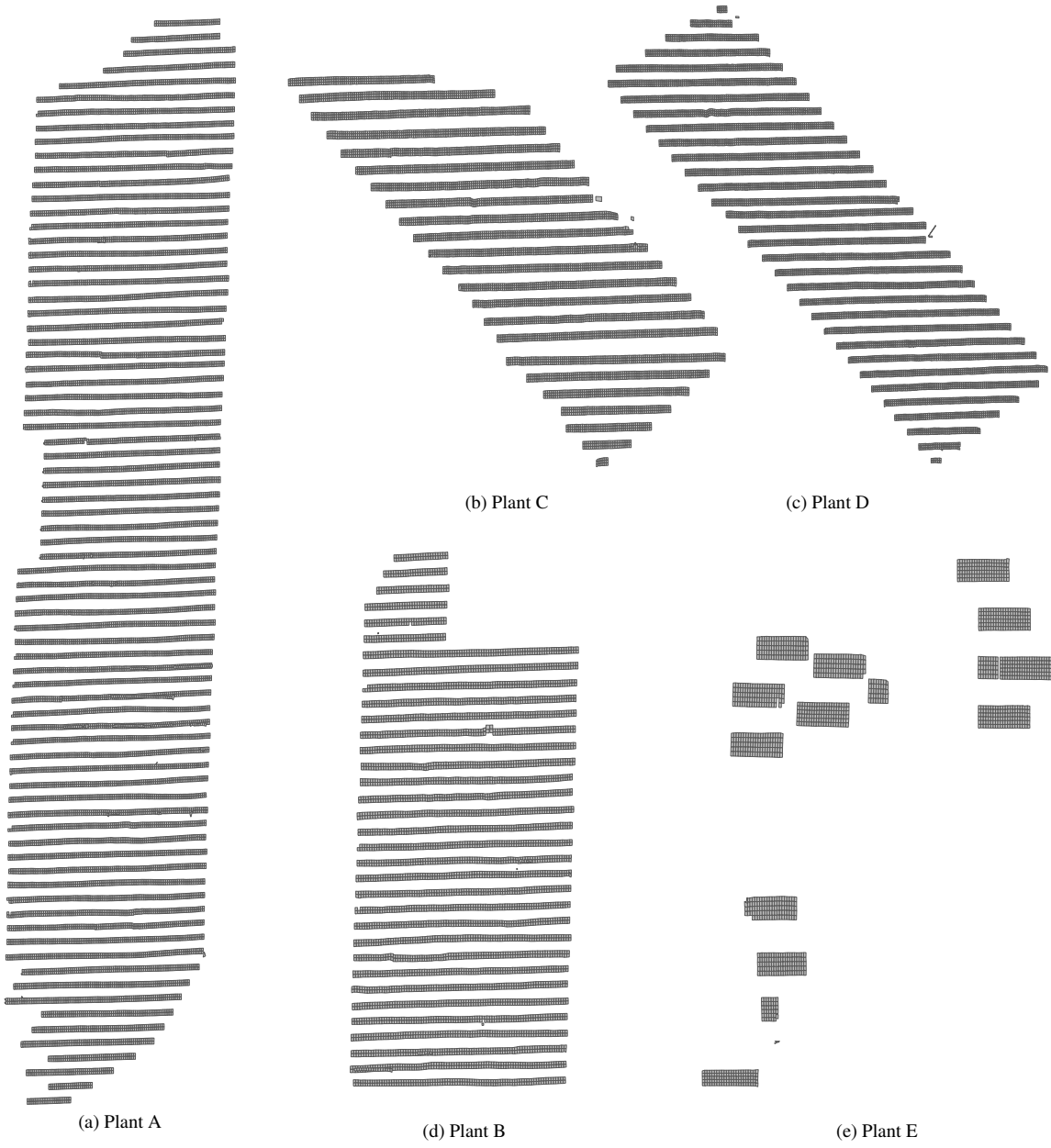(c) Plant D

(a) Plant A

(d) Plant B

(e) Plant E

Figure 12: Georeferencing results for all PV plants in our dataset.