Received: 14 June 2021 | Revised: 12 October 2021 | Accepted: 14 November 2021

DOI: 10.1002/pip.3518

RESEARCH ARTICLE

PROGRESS IN PHOTOVOLTAICS WILEY

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] | Ian Marius Peters[1]

[1]Forschungszentrum Jülich GmbH, Helmholtz-Institute Erlangen-Nuremberg for Renewable Energies (HI ERN), Erlangen, Germany

[2]Pattern Recognition Lab, Department Informatik, Universität Erlangen-Nürnberg (FAU), Erlangen, Germany

[3]Institute Materials for Electronics and Energy Technology, Universität Erlangen-Nürnberg (FAU), Erlangen, Germany

Correspondence
Ian Marius Peters, Forschungszentrum Jülich GmbH, Helmholtz-Institute Erlangen-Nuremberg for Renewable Energies (HI ERN), Erlangen, Germany.
Email: i.peters@fz-juelich.de

Funding information
State of Bavaria, Grant/Award Number: 446521a/20/5; German Federal Ministry for Economic Affairs and Energy, Grant/Award Numbers: 0324286, 032429A; IBC SOLAR AG; Allianz Risk Consulting GmbH/Allianz Zentrum für Technik (AZT)

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 labeled 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 labeling 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.

KEYWORDS
anomaly detection, contrastive learning, defect detection, thermography, deep learning, machine learning, nearest neighbor classifier, PV plant inspection

1 | INTRODUCTION

Solar photovoltaics (PV) has emerged as an important renewable energy source with a global installed capacity of 627 GW_p in 2020[1] that is projected to reach 2840 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.[7,10] 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 labeled 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 labeled 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 labeled 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.

2 | 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.

2.1 | 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.

2.2 | Domain adaptation

Domain adaptation addresses the problem of learning transferable representations without the need for large amounts of labeled 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 labeled data of a source domain that generalize to an unlabeled 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.

2.3 | 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 and Chawla.[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.[23,34,64,65] Some works also explored the use of domain adaptation for AD.[49,66-68]

Many AD methods assume an unlabeled training dataset containing mostly normal samples and a few anomalies. If labeled anomalies are available, AD can also be formulated as (semi-)supervised binary classification and achieve state-of-the-art performance.[19-21] Similarly, 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 AD in a target domain which differs from the source domain and does not contain any labeled examples.

2.4 | 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,[4,6,11-13] iterative growth of segmentation masks,[5,8] or template matching.[3] 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[9,14] or classification with an SVM or Random Forest[10] is slightly more robust. Recently, deep learning has shown promising results in overcoming the problems of classical algorithms.[7,10,15,16,69] 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.

3 | DATASET

We use an extended version of the dataset from our previous work.[7] It consists of 4.16 million IR images showing 105 546 PV modules from six different PV plants, which were acquired under clearsky conditions and solar irradiance above 700 Wm^{-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.

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 labeled by an expert either as containing a normal module or a module with one out of the ten typical faults shown in Figure 1. While our method makes only a binary distinction between normal and anomalous modules, fine-grained fault labels are used to evaluate our method.

Tables 1 and 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 sets.

Figure 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. Figure 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.

4 | 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 labeled examples for this PV plant, we have a set of N^S binary labeled 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 Figure 4 overcomes 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.

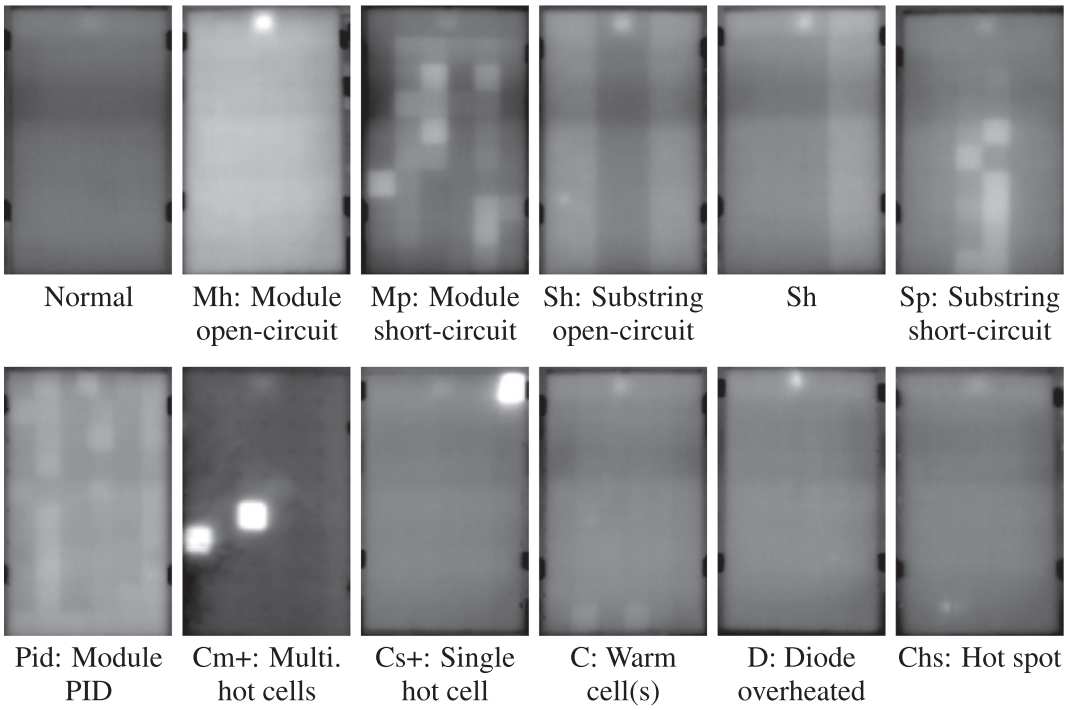

FIGURE 1 Exemplary IR images of a normal and ten different types of anomalous PV modules in our dataset. Temperature ranges from 30°C (black) to 60°C (white). All images except for class Cm+ show plant A. The figure is taken from our previous work[7]

TABLE 1 Numbers of normal and anomalous IR images in our dataset

	Plant					
Class	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

TABLE 2 Numbers of anomalous IR images per underlying fault class

	Plant					
Class	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

4.1 | Supervised contrastive representation learning

As indicated by Figure 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 AD 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:

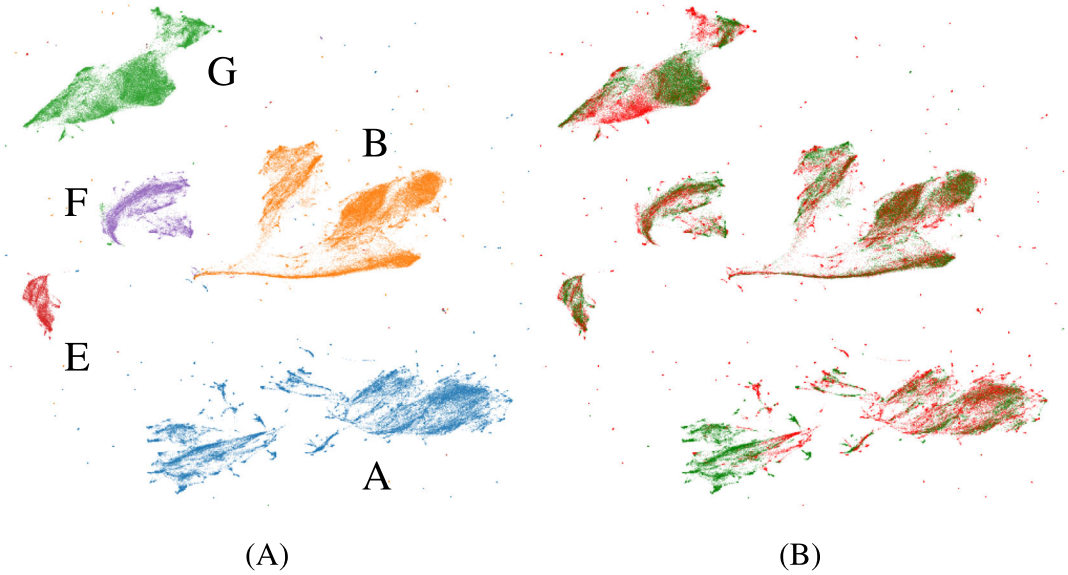

FIGURE 2 Projection of our dataset obtained by UMAP (with 50 neighbors per sample and minimum distance of 0.1). Colors in (A) indicate the PV plant, which reveals the domain shift between different plants. In (B), normal and anomalous samples are colored green and red, respectively. UMAP is applied directly to the flattened images, which are preprocessed as in Section 4.3.2. For better visualization, normal samples are subsampled to match the number of anomalous samples [Colour figure can be viewed at wileyonlinelibrary.com]

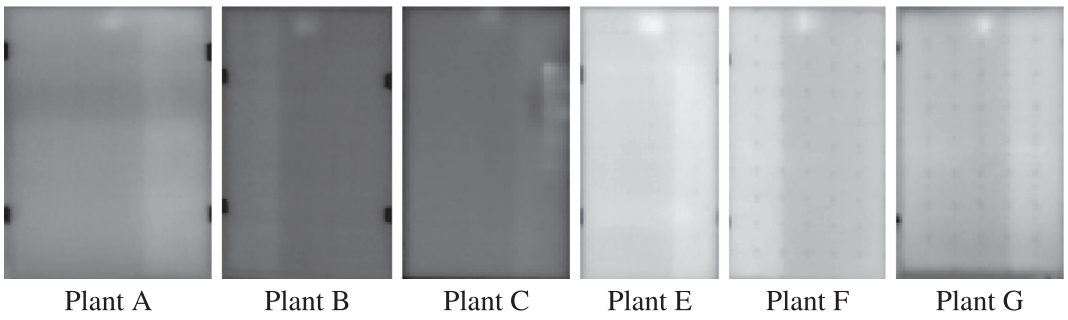

FIGURE 3 IR images differ between the PV plants in our dataset due to different ambient conditions, camera positions, as well as module and cell type. Shown are modules with Sh anomaly. The original aspect ratio is preserved and temperature ranges from 15°C (black) to 50°C (white)

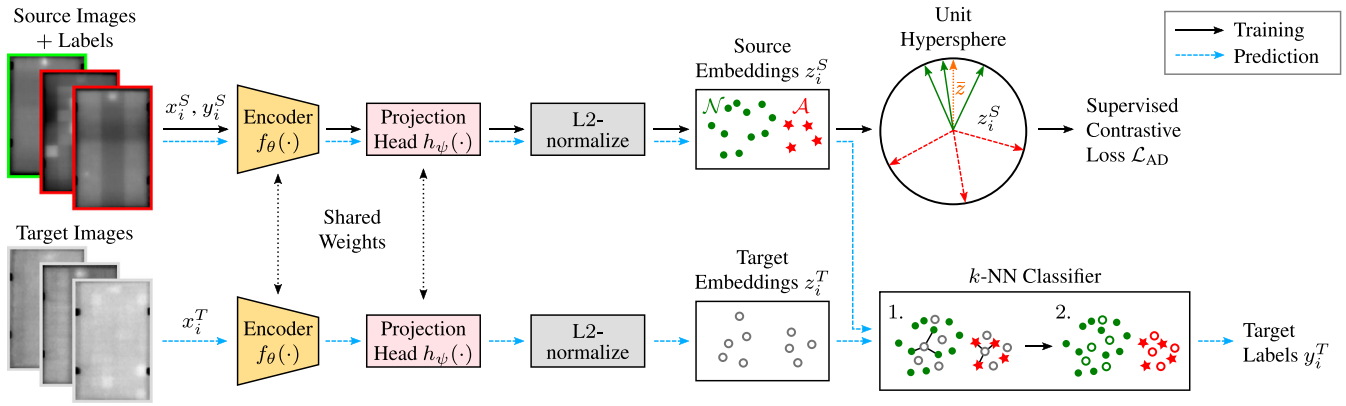

FIGURE 4 Overview of our method for detection of anomalous PV modules in IR images of a *target* PV plant based on labeled samples of a *source* PV plant. Low-dimensional embeddings of both source and target images are extracted by means of contrastive representation learning. A k-NN classifier predicts target labels based on the labels of neighboring source images in the embedding space [Colour figure can be viewed at wileyonlinelibrary.com]

$$v_i^S = h_\psi\left(f_\theta\left(x_i^S\right)\right). \tag{1}$$

Several related works use a projection head to improve representational power of the encoder embeddings.[29,31,73] 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 AD.

After encoding, each embedding vector is normalized to unit L2-norm:

$$z_i^S = v_i^S / \left\| v_i^S \right\|_2. \tag{2}$$

Iterative stochastic gradient descent is performed on embeddings of randomly shuffled batches of N labeled source images $\left\{x_i^S, y_i^S\right\}_{i=1\ldots N}$ to compute suitable network parameters:

$$\left\{\theta^*, \psi^*\right\} = \arg\min_{\theta,\psi} \mathcal{L}_{\mathrm{AD}}\left(z_i^S, y_i^S\right), \tag{3}$$

where $\mathcal{L}_{\mathrm{AD}}$ is a supervised contrastive loss with the following form of a non-parametric softmax classifier[27]:

$$\mathcal{L}_{\mathrm{AD}}\left(z_i^S, y_i^S\right) = -\frac{1}{|\mathcal{N}|}\sum_{i\in\mathcal{N}} \log \frac{\exp\left(z_i^S \cdot \bar{z}^S/\tau\right)}{\sum_{j\in\mathcal{N}\cup\mathcal{A}} \exp\left(z_j^S \cdot \bar{z}^S/\tau\right)}. \tag{4}$$

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.[35] 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.

4.2 | Anomal detection with a k-NN classifier

The AD 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 $\left\{z_i^S\right\}_{i=1\ldots N^S}$ of all source images as in Equations (1) and (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 δ, the target image is predicted to contain an anomalous PV module. Later, in Section 5.4, we will determine optimal settings for the hyperparameters k and δ.

We also tried using temperature-scaled cosine distance $\exp\left(z_j^T \cdot z_i^S/\tau\right)$ 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]

4.3 | Implementation details

4.3.1 | 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 $d = 128$.

4.3.2 | 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 64×64 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.

4.3.3 | Training

We train all models for 110 000 steps using stochastic gradient descent with momentum 0.9 and weight decay 5×10^{-4}.[76,77] The

initial learning rate $\eta_0 = 6 \times 10^{-2}$ is decayed in each step following the Cosine Annealing strategy $\eta = \eta_0/2(1+\cos(p\pi))$ 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.

4.3.4 | 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.

5 | EXPERIMENTS AND RESULTS

In the following, we perform a quantitative analysis of our method and compare it against a binary cross-entropy classifier.

5.1 | Evaluation protocol

As common in AD, we evaluate all our models in terms of the area under the receiver operating characteristic (AUROC) and the average precision score (AP).[18,19,60] AUROC is obtained by plotting the true positive rate $\mathrm{TPR} = \mathrm{TP}/(\mathrm{TP}+\mathrm{FN})$ over the false positive rate $\mathrm{FPR} = \mathrm{FP}/(\mathrm{FP}+\mathrm{TN})$ at various decision thresholds δ 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 is the number of anomalous images classified as normal.

Similarly, the AP is obtained from the precision–recall curve which plots precision $\mathrm{P} = \mathrm{TP}/(\mathrm{TP}+\mathrm{FP})$ over recall $\mathrm{R} = \mathrm{TP}/(\mathrm{TP}+\mathrm{FN})$ at different decision thresholds. The AP summarizes the curve as the weighted mean of precisions achieved at each threshold $\mathrm{AP} = \sum_{i=1}^{n} (\mathrm{R}_i - \mathrm{R}_{i-1})\mathrm{P}_i$.

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 δ. 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* $\mathrm{S} \rightarrow \mathrm{T}$. As mentioned in Section 3, only the data of PV plants A, B, E, and F are 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.

5.2 | 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 labeled 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.

5.3 | Results of the contrastive *k*-NN classifier

We train and evaluate our method on various tasks and report the best target AUROC scores in Figure 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

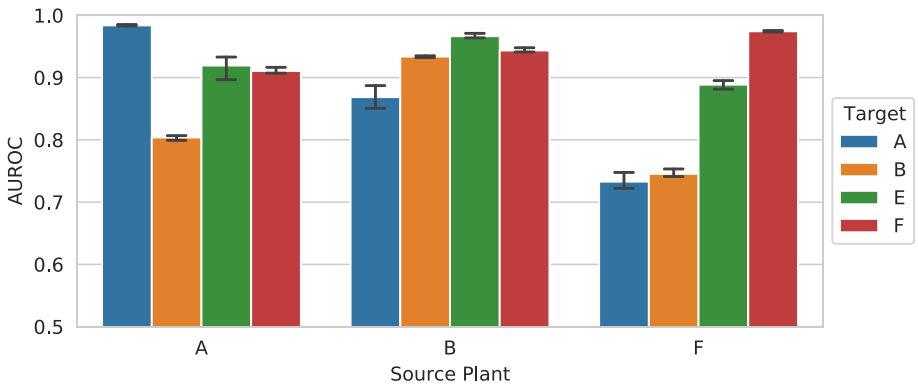

FIGURE 5 Best target AUROC of our method when trained on different source PV plants. When source and target plants are identical, we evaluate on the target test split otherwise on the target train split. Error bars indicate the 95% confidence interval over three runs [Colour figure can be viewed at wileyonlinelibrary.com]

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, plants 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 plant E or F. Possible explanations for this are the larger number of anomalies and the presence of subdomains in plants A and B (see Figure 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.

5.4 | Hyperparameter selection of the *k*-NN classifier

The absence of labeled 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.

Figure 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 δ, which is the fraction of anomalies required in the set of neighbors $\mathcal{N}_k$ to classify a target image as anomalous. Figure 6B shows the geometric mean (G-Mean) of true positive rate and false positive rate for various decision thresholds δ. While the classifier is more sensitive to the choice of δ (as compared to k), it behaves consistent across the tasks,

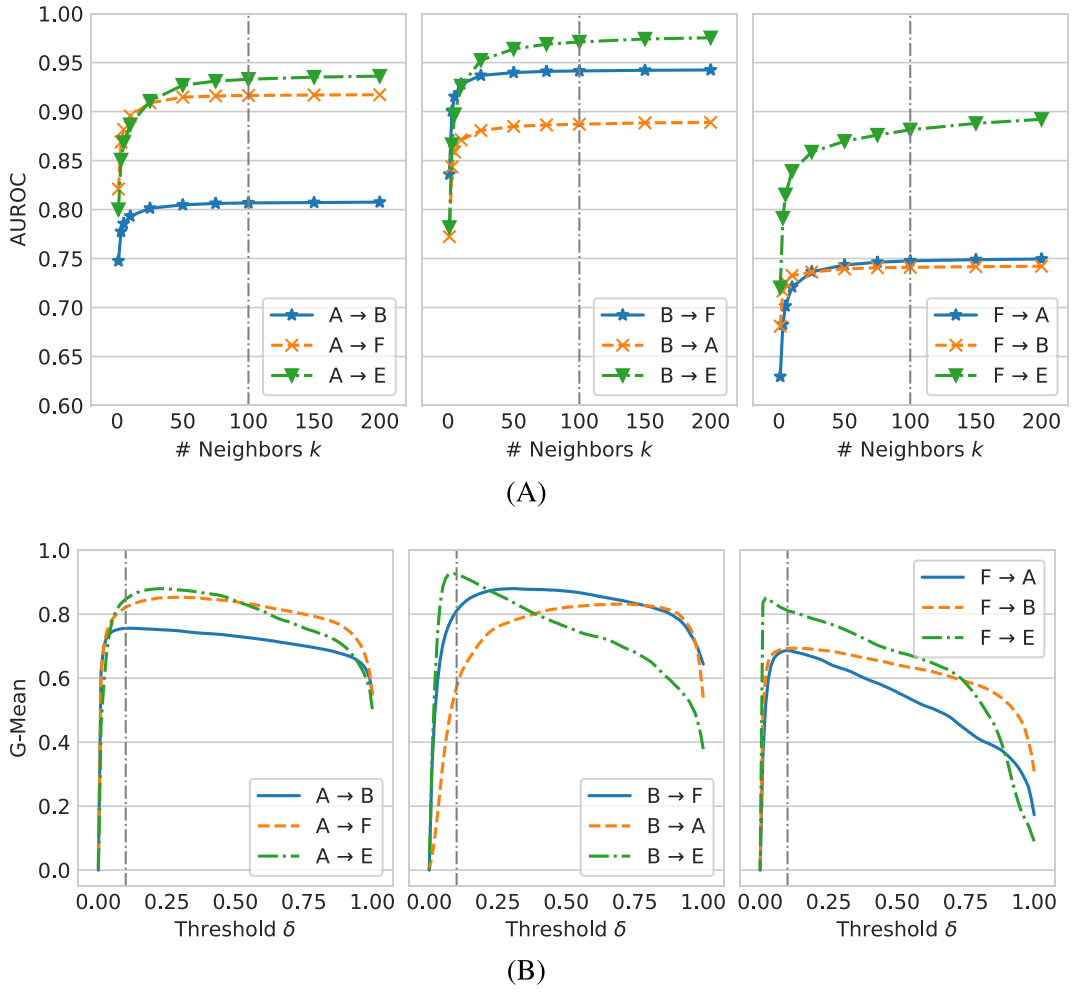

FIGURE 6 Prediction performance of the *k*-NN classifier for different settings of the hyperparameters k and δ. The dashed vertical lines at $k = 100, \delta = 0.1$ represent the trade-offs we use in practice. All classifiers are trained on contrastive embeddings of the model with random seed 1 and best target AUROC on the respective task. (A) AUROC versus numbers k of neighbors. (B) G-Mean versus decision threshold δ [Colour figure can be viewed at wileyonlinelibrary.com]

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.

5.5 | 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 Figure 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 Section 5.3) 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 Figure 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. Table 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.

5.6 | 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 Figure 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.

Figure 8 also highlights a few misclassified normal images. Interestingly, most of these are valid anomalies with false ground truth 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.

5.7 | Embedding visualizations

As another means to interpret our models, we visualize the representations learned by supervised contrastive training in Figure 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

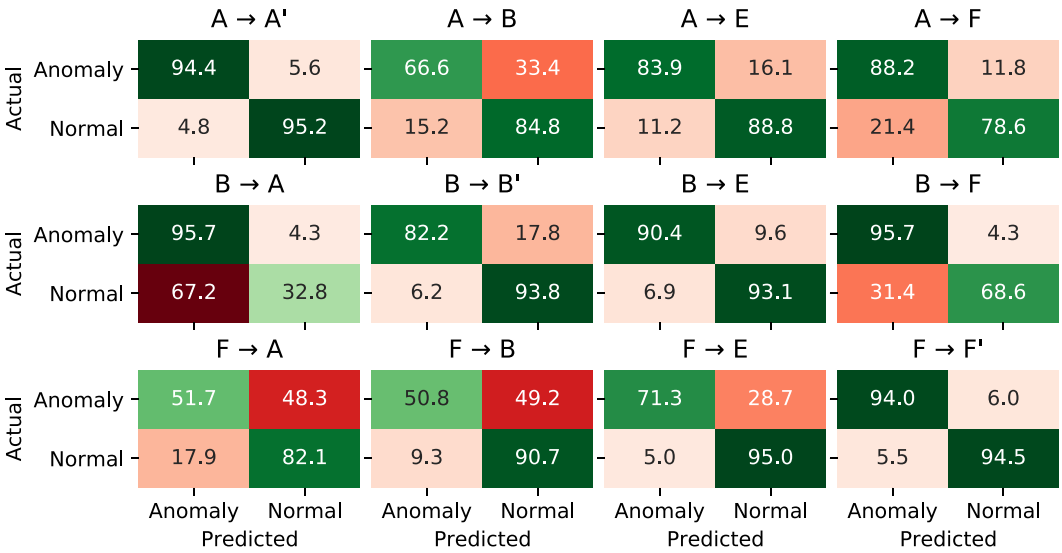

FIGURE 7 Normalized confusion matrices of the k-NN classifier ($k = 100$ and $\delta = 0.1$) on different tasks. We report averages using the model with best target AUROC at each random seed [Colour figure can be viewed at wileyonlinelibrary.com]

TABLE 3 Percentage of target anomalies misclassified as normal by our k-NN classifier ($k = 100$ and $\delta = 0.1$) grouped by fault class

	Actual fault class										
Task	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

Note: 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.

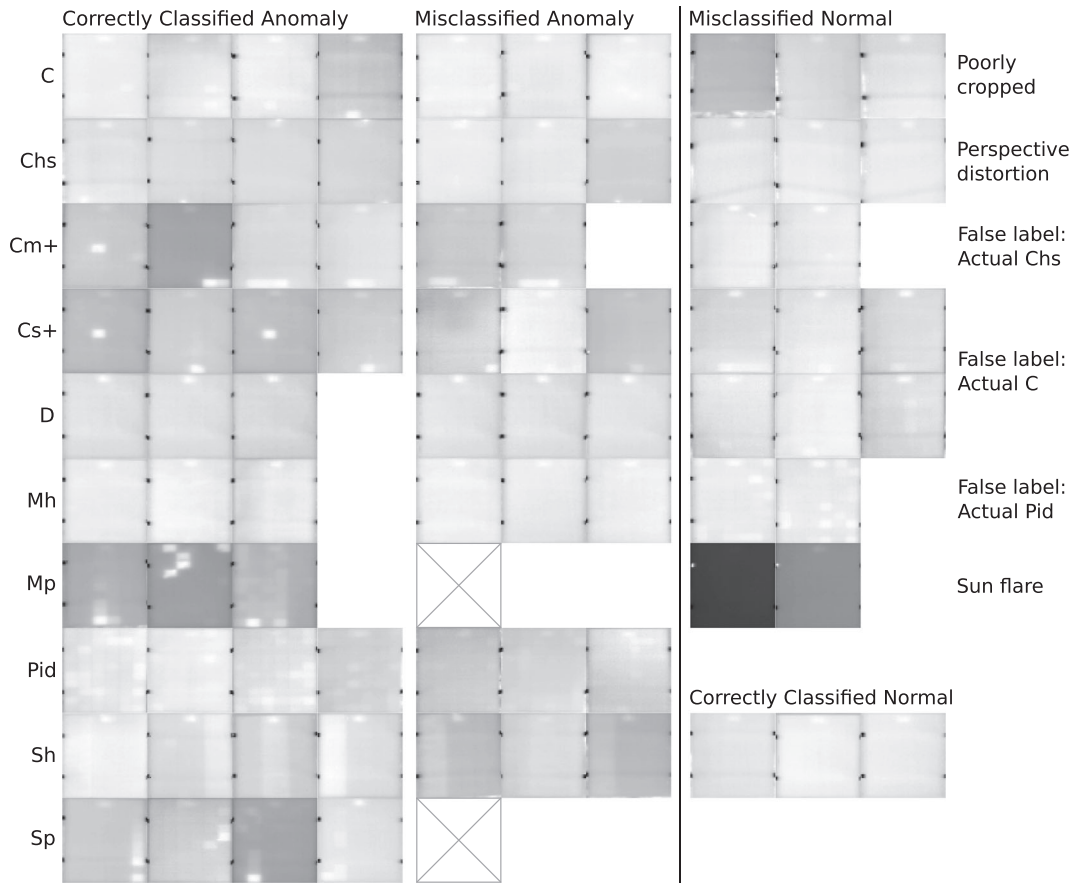

FIGURE 8 Exemplary predictions of our k-NN classifier ($k = 100$ and $\delta = 0.1$) for IR images of plant B by the model trained on plant A. We use the model at seed 1 with best target AUROC and show preprocessed patches

achieved low error rates in Section 5.5, have a larger distance to the normal modules than anomalies with higher error rates (Mh, D, and Chs). The C anomalies often lie somewhere in between, which is in accordance to the slightly higher error rates of around 10%.

5.8 | 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

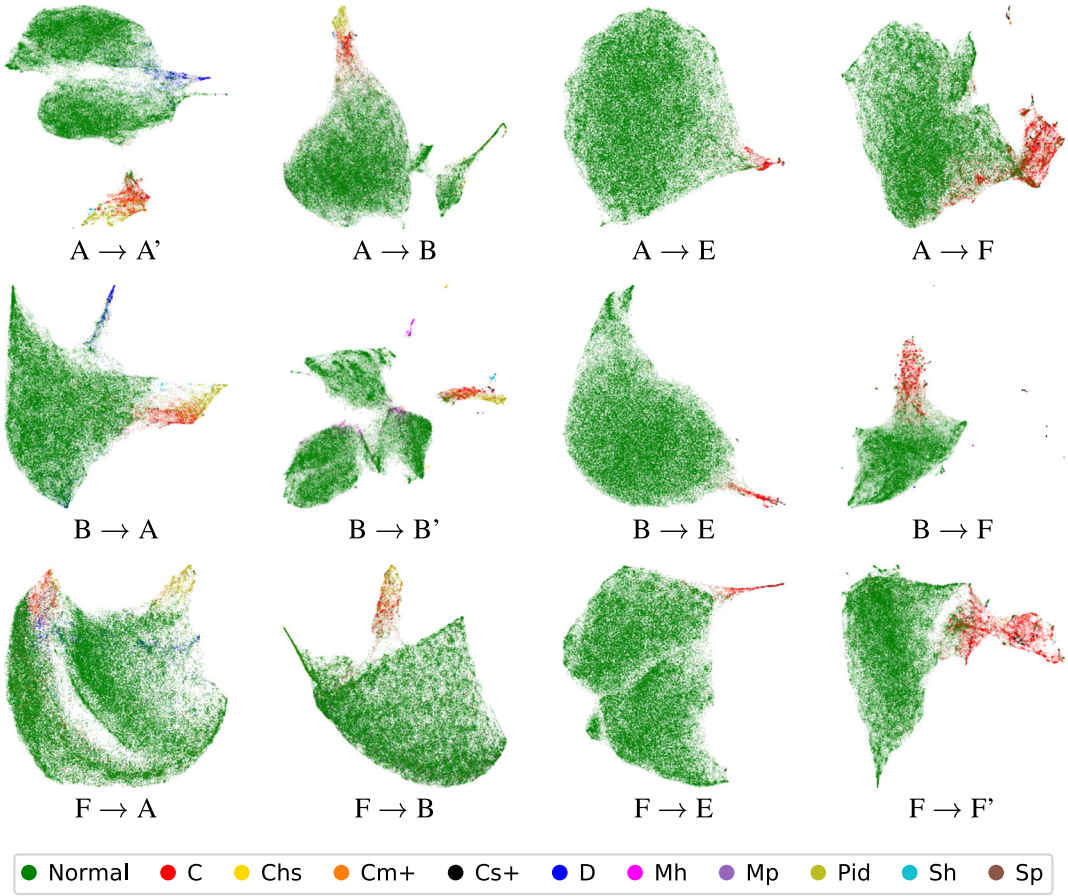

FIGURE 9 UMAP projections (with 50 neighbors per sample and minimum distance of 0.1) of the target datasets embedded by ResNet-34 after supervised contrastive training. Embeddings are obtained behind the ResNet-34 average pooling layer. For each task, the model at seed 1 with best target AUROC is shown [Colour figure can be viewed at wileyonlinelibrary.com]

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

	Variant			Variant	
Task	**Full dataset**	**Leaveout**	**Task**	**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

Note: All values are averages over three training runs.

analyze how well our method deals with such unknown 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 Table 4) do not deviate much from the respective AUROCs of the models trained on all anomaly classes. Similarly, we do not observe any change in model convergence during training, as shown in Figure B1 in Appendix B1. The results indicate that our method can reliably detect unknown anomalies.

5.9 | 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 Section 4.3). While the convolutional backbone is identical to our contrastive model, a softmax-activated fully connected layer with two outputs is used on top of the 2D global average pooling layer. A projection head is not employed.

As shown in Table 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

TABLE 5 Target AUROCs of our contrastive *k*-NN classifier versus a binary classifier trained with cross-entropy loss

			Contrastive AUROC			Cross-entropy AUROC		
Task	**Val 0**	**Val 1**	**@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**

Note: 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.

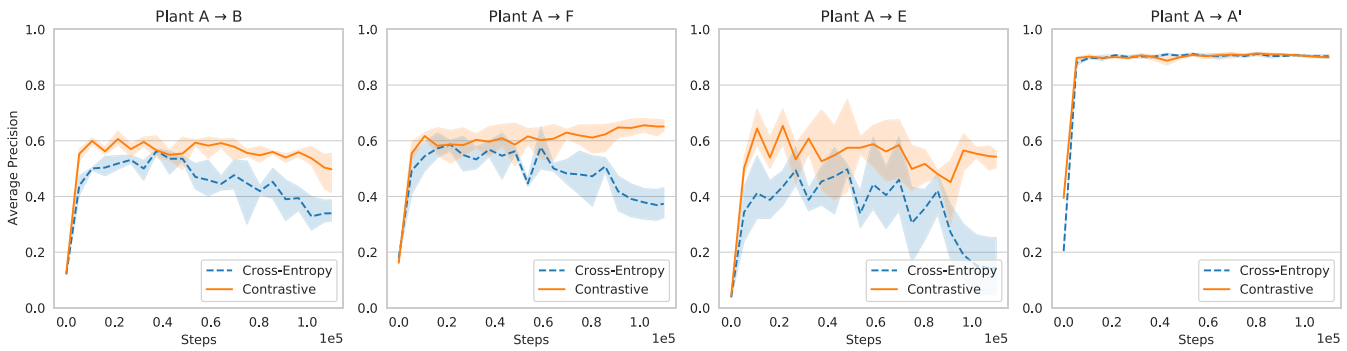

FIGURE 10 Average precision over the course of training of our contrastive *k*-NN classifier (orange line) versus a supervised binary classifier trained on cross-entropy loss (dashed blue line). Plant A is used as source. Shaded regions indicate the 95% confidence interval over three runs [Colour figure can be viewed at wileyonlinelibrary.com]

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, that is, 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 Figure 10. An additional analysis for plants B and F is provided in Appendix A1. Furthermore, we find that due to the large size of our datasets, target AP converges within a single training epoch.

5.10 | 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 Figure 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 Section 5.5), module-level aggregation improves especially upon the detection rate of normal modules but also yields a 1% 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.

5.11 | Exemplary application to labeling 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 labeling such datasets by automatically

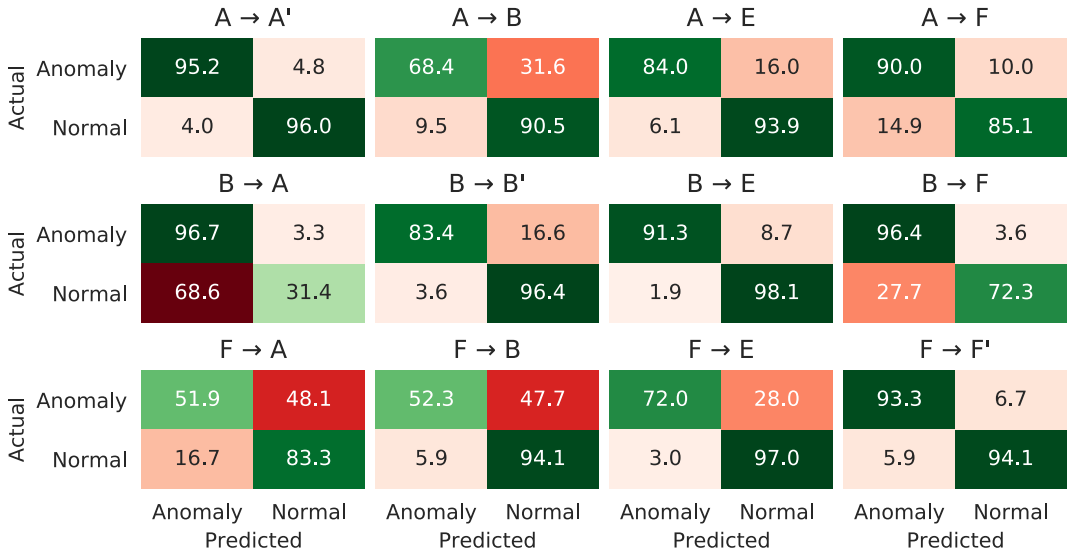

FIGURE 11 Normalized confusion matrices of the k-NN classifier ($k = 100$ and $\delta = 0.1$) for predictions aggregated on module level. We report averages using the model with best target AUROC at each random seed [Colour figure can be viewed at wileyonlinelibrary.com]

rejecting the majority of normal (and thus uninteresting) PV modules. For example, when labeling plant E, one would have to manually sight 14 662 PV modules, of which only 296 are anomalous, that is, actually interesting. Applying our method (trained, e.g., 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.

6 | DISCUSSION AND CONCLUSION

6.1 | 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 labeled 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 ten different types of anomalies, we found that our method most frequently misses anomalies with a small spatial extent in the image, for example, 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 unknown anomalies. Finally, we improved detection accuracy by aggregating predictions of multiple IR images belonging to the same PV module.

6.2 | 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 labor 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 labeled 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 labeled 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 labeling 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.

6.3 | 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 labeled 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 labeled source datasets from different PV plants simultaneously, could be taken into consideration.

ACKNOWLEDGEMENTS

We would like to thank Sanjay Venugopal for valuable discussions about the contrastive learning objective. We gratefully acknowledge the German Federal Ministry for Economic Affairs and Energy (BMWi) and the IBC SOLAR AG for financial funding of the project iPV4.0 (FKZ: 0324286). This work was financially supported by the State of Bavaria via the project PV-Tera (No. 446521a/20/5) and by BMWi via the project COSIMA (FKZ: 032429A). We sincerely thank the Allianz Risk Consulting GmbH/Allianz Zentrum für Technik (AZT) in Munich, Germany, for supporting the project.

DATA AVAILABILITY STATEMENT

Data used in this study is proprietary. Access to data can be arranged via the authors of this study pending consent from the commercial entities owning the data.

ORCID

Lukas Bommes https://orcid.org/0000-0002-1878-0178
Mathis Hoffmann https://orcid.org/0000-0002-1009-3111
Claudia Buerhop-Lutz https://orcid.org/0000-0001-5233-6700
Ian Marius Peters https://orcid.org/0000-0002-1725-0909

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. Addabbo P, Angrisano A, Bernardi ML, et al. UAV system for photovoltaic plant inspection. *Mag IEEE Aerosp Electron Syst Mag*. 2018;33:58-67.
4. Aghaei M, Grimaccia F, Gonano CA, Leva S. Innovative automated control system for PV fields inspection and remote control. *IEEE Trans Ind Electron*. 2015;62:7287-7296.
5. Alsafasfeh M, Abdel-Qader I, Bazuin B, Alsafasfeh Q, Su W. Unsupervised fault detection and analysis for large photovoltaic systems using drones and machine vision. *Energies*. 2018;11:2252.
6. Arenella A, Greco A, Saggese A, Vento M. Real time fault detection in photovoltaic cells by cameras on drones. *International Conference Image Analysis and Recognition (ICIAR)*. Cham: Springer; 2017.
7. Bommes L, Pickel T, Buerhop-Lutz C, Hauch J, Brabec C, Peters IM. Computer vision tool for detection, mapping and fault classification of pv modules in aerial ir videos. *Applications Prog Photovolt Res Appl*. 2021;29(12):1236-1251.
8. Carletti V, Greco A, Saggese A, Vento M. An intelligent flying system for automatic detection of faults in photovoltaic plants. *J Ambient Intell Humanized Comput*. 2019;11:2027-2040.
9. Dotenco S, Dalsass M, Winkler L, Würzner T, Brabec C, Maier A, Gallwitz F. Automatic detection and analysis of photovoltaic modules in aerial infrared imagery. In: IEEE Winter Conference on Applications of Computer Vision (WACV); 2016.
10. Dunderdale C, Brettenny W, Clohessy C, van Dyk EE. Photovoltaic defect classification through thermal infrared imaging using a machine learning approach. *Prog Photovolt Res Appl*. 2020;28:177-188.
11. Francesco G, Sonia L, Alessandro N. 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.
12. Grimaccia F, Leva S, Niccolai A. PV plant digital mapping for modules' defects detection by unmanned aerial vehicles. *IET Renew Power Gener*. 2017;11:1221-1228.
13. Jeong H, Kwon G-R, Lee S-W. Deterioration diagnosis of solar module using thermal and visible image processing. *Energies*. 2020;13:2856.
14. Kim D, Youn J, Kim C. Automatic fault recognition of photovoltaic modules based on statistical analysis of UAV thermography. *Int Arch Photogramm Remote Sens Spat Inf Sci (ISPRS)*. 2017;42:179.
15. Oliveira A, Aghaei M, Rüther R. 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. Pierdicca R, Malinverni E, Piccinini F, Paolanti M, Felicetti A, Zingaretti P. Deep convolutional neural network for automatic detection of damaged photovoltaic cells. *Int Archives Photogramm Remote Sens Spat Inf Sci (ISPRS)*. 2018;XLII-2:893-900.
17. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2016.
18. Bergman L, Cohen N, Hoshen Y. Deep nearest neighbor anomaly detection. arXiv preprint arXiv:2002.10445; 2020.
19. Hendrycks D, Mazeika M, Dietterich T. Deep anomaly detection with outlier exposure. In: International Conference on Learning Representations (ICLR); 2019.
20. Hendrycks D, Mazeika M, Kadavath S, Song D. Using self-supervised learning can improve model robustness and uncertainty. In: Advances in Neural Information Processing Systems (NIPS); 2019.
21. Ruff L, Vandermeulen RA, Franks BJ, Müller K-R, Kloft M. Rethinking assumptions in deep anomaly detection. arXiv preprint arXiv:2006.00339; 2020.
22. Khosla P, Teterwak P, Wang C, et al. Supervised contrastive learning. In: Advances in Neural Information Processing Systems (NIPS); 2020.
23. Winkens J, Bunel R, Roy AG, et al. Contrastive training for improved out-of-distribution detection. arXiv preprint arXiv:2007.05566; 2020.
24. Hadsell R, Chopra S, LeCun Y. Dimensionality reduction by learning an invariant mapping. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR); 2006.
25. Weinberger KQ, Saul LK. Distance metric learning for large margin nearest neighbor classification. *J Mach Learn Res (JMLR)*. 2009;10(9):207-244.
26. Sohn K. Improved deep metric learning with multi-class n-pair loss objective. In: Advances in Neural Information Processing Systems (NIPS); 2016.

27. Le-Khac PH, Healy G, Smeaton AF. Contrastive representation learning: a framework and review. *IEEE Access*. 2020;8:193907-193834.
28. Van den Oord A, Li Y, Vinyals O. Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748; 2018.
29. Chen T, Kornblith S, Norouzi M, Hinton G. A simple framework for contrastive learning of visual representations. arXiv preprint arXiv: 2002.05709; 2020.
30. Wu Z, Xiong Y, Yu SX, Lin D. Unsupervised feature learning via non-parametric instance discrimination. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2018.
31. Chen X, Fan H, Girshick R, He K. Improved baselines with momentum contrastive learning. arXiv preprint arXiv:2003.04297; 2020.
32. He K, Fan H, Wu Y, Xie S, Girshick R. Momentum contrast for unsupervised visual representation learning. In: IEEE/CVF conference on computer vision and pattern recognition (cvpr); 2020.
33. Ye M, Zhang X, Yuen PC, Chang S-F. Unsupervised embedding learning via invariant and spreading instance feature. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2019.
34. Köpüklü O, Zheng J, Xu H, Rigoll G. Driver anomaly detection: a dataset and contrastive learning approach. arXiv preprint arXiv: 2009.14660; 2020.
35. Wei C, Tang Y, Niu C, Hu H, Wang Y, Liang J. Self-supervised representation learning for evolutionary neural architecture search. arXiv preprint arXiv:2011.00186; 2020.
36. Wang M, Deng W. Deep visual domain adaptation: a survey. *Neurocomputing*. 2018;312:135-153.
37. Zhao S, Li B, Reed C, Xu P, Keutzer K. Multi-source domain adaptation in the deep learning era: a systematic survey. arXiv preprint arXiv:2002.12169; 2020.
38. Long M, Cao Y, Wang J, Jordan MI. Learning transferable features with deep adaptation networks. In: International Conference on Machine Learning (PMLR); 2015.
39. Rozantsev A, Salzmann M, Fua P. Beyond sharing weights for deep domain adaptation. *IEEE Trans Pattern Anal Mach Intel*. 2019;41: 801-814.
40. Zhu Y, Zhuang F, Wang D. Aligning domain-specific distribution and classifier for cross-domain classification from multiple sources. In: AAAI Conference on Artificial Intelligence; 2019.
41. Guo H, Pasunuru R, Bansal M. Multi-source domain adaptation for text classification via distancenet-bandits. arXiv preprint arXiv: 2001.04362; 2020.
42. Rakshit S, Banerjee B, Roig G, Chaudhuri S. Unsupervised multi-source domain adaptation driven by deep adversarial ensemble learning. In: Pattern Recognition; 2019.
43. Hoffman J, Mohri M, Zhang N. Algorithms and theory for multiple-source adaptation. In: Bengio S, Wallach H, Larochelle H, Grauman K, Cesa-Bianchi N, Garnett R, eds. *Advances in Neural Information Processing Systems (NIPS)*. Curran Associates Inc.; 2018.
44. Zhuang F, Cheng X, Luo P, Pan SJ, He Q. Supervised representation learning: transfer learning with deep autoencoders. In: International Joint Conference on Artificial Intelligence (IJCAI); 2015.
45. Dai S, Cheng Y, Zhang Y, Gan Z, Liu J, Carin L. Contrastively smoothed class alignment for unsupervised domain adaptation. arXiv preprint arXiv:1909.05288; 2020.
46. Kang G, Jiang L, Yang Y, Hauptmann AG. Contrastive adaptation network for unsupervised domain adaptation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2019.
47. Park C, Lee J, Yoo J, Hur M, Yoon S. Joint contrastive learning for unsupervised domain adaptation. arXiv preprint arXiv:2006.10297; 2020.
48. Zhu Y, Zhuang F, Wang J, et al. Deep subdomain adaptation network for image classification. *IEEE Trans Neural Netw Learn Syst*. 2021;32: 1713-1722.
49. Yang Z, Bozchalooi IS, Darve EF. Anomaly detection with domain adaptation. In: Advances in Neural Information Processing Systems (NIPS); 2019.
50. Pang G, Shen C, Cao L, Hengel AVD. Deep learning for anomaly detection: a review. *ACM Computing Surveys*. 2021;54:1-38.
51. Bulusu S, Kailkhura B, Li B, Varshney PK, Song D. Anomalous example detection in deep learning: a survey. *IEEE Access*. 2020;8: 132330-132347.
52. Chalapathy R, Chawla S. Deep learning for anomaly detection: a survey. arXiv preprint arXiv:1901.03407; 2019.
53. An J, Cho S. Variational autoencoder based anomaly detection using reconstruction probability. In: Special lecture on IE; 2015.
54. Chen J, Sathe S, Aggarwal C, Turaga D. Outlier detection with autoencoder ensembles. In: SIAM International Conference on Data Mining (SDM); 2017.
55. Akcay S, Atapour-Abarghouei A, Breckon TP. GANomaly: semi-supervised anomaly detection via adversarial training. In: Asian Conference on Computer Vision (ACCV); 2019.
56. Zenati H, Romain M, Foo C-S, Lecouat B, Chandrasekhar V. Adversarially learned anomaly detection. In: IEEE International Conference on Data Mining (ICDM); 2018.
57. Ruff L, Vandermeulen R, Goernitz N, et al. Deep one-class classification. In: International Conference on Machine Learning (PMLR); 2018.
58. Ruff L, Vandermeulen RA, GÃűrnitz N, Binder A, Müller E, Müller K-R, Kloft M. Deep semi-supervised anomaly detection. In: International Conference on Learning Representations (ICLR); 2020.
59. Bergman L, Hoshen Y. Classification-based anomaly detection for general data. In: International Conference on Learning Representations (ICLR); 2020.
60. Golan I, El-Yaniv R. Deep anomaly detection using geometric transformations. In: Advances in Neural Information Processing Systems (NIPS); 2018.
61. Wang S, Zeng Y, Liu X, Zhu E, Yin J, Xu C, Kloft M. Effective end-to-end unsupervised outlier detection via inlier priority of discriminative network. In: Advances in Neural Information Processing Systems (NIPS); 2019.
62. Perera P, Patel VM. Learning deep features for one-class classification. *IEEE Trans Image Process*. 2019;28:5450-5463.
63. Yilmaz SF, Kozat SS. Unsupervised anomaly detection via deep metric learning with end-to-end optimization. arXiv preprint arXiv: 2005.05865; 2020.
64. Sohn K, Li C-L, Yoon J, Jin M, Pfister T. Learning and evaluating representations for deep one-class classification. In: International Conference on Learning Representations (ICLR); 2021.
65. Tack J, Mo S, Jeong J, Shin J. CSI: novelty detection via contrastive learning on distributionally shifted instances. *Adv Neural Inf Process Syst (NIPS)*. 2020;33:11839-11852.
66. Chen J, Liu X. Transfer learning with one-class data. *Pattern Recognition Letters*. 2014;37:32-40. https://doi.org/10.1016/j.patrec.2013.07.017
67. Kumagai A, Iwata T, Fujiwara Y. Transfer anomaly detection by inferring latent domain representations. In: Advances in Neural Information Processing Systems (NIPS); 2019.
68. Yamaguchi M, Koizumi Y, Harada N. 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.
69. Mayr M, Hoffmann M, Maier A, Christlein V. Weakly supervised segmentation of cracks on solar cells using normalized Lp norm. arXiv preprint arXiv:2001.11248; 2020.

70. Howard AG, Zhu M, Chen B, et al. Mobilenets: efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv/1704.04861; 2017.
71. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. In: International Conference on Learning Representations (ICLR); 2015.
72. McInnes L, Healy J, Saul N, Großberger L. UMAP: uniform manifold approximation and projection. *J Open Source Softw*. 2018;3(29):861.
73. Grill J-B, Strub F, Altché F, et al. Bootstrap your own latent—a new approach to self-supervised learning. In: Advances in Neural Information Processing Systems (NIPS); 2020.
74. Goldberger J, Roweis S, Hinton G, Salakhutdinov R. Neighbourhood components analysis. In: Advances in Neural Information Processing Systems (NIPS); 2004.
75. Huang G, Liu Z, van der Maaten L, Weinberger KQ. Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2017.
76. Loshchilov I, Hutter F. Decoupled weight decay regularization. In: International Conference on Learning Representations (ICLR); 2019.
77. Sutskever I, Martens J, Dahl G, Hinton G. On the importance of initialization and momentum in deep learning. In: International Conference on Machine Learning (PMLR); 2013.
78. Loshchilov I, Hutter F. SGDR: stochastic gradient descent with warm restarts. In: International Conference on Learning Representations (ICLR); 2017.
79. Saito T, Rehmsmeier M. The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PLoS One*. 2015;10:e0118432.
80. Sun Y, Tzeng E, Darrell T, Efros AA. Unsupervised domain adaptation through self-supervision. arXiv preprint arXiv:1909.11825; 2019.

How to cite this article: Bommes L, Hoffmann M, Buerhop-Lutz C, et al. Anomaly detection in IR images of PV modules using supervised contrastive learning. *Prog Photovolt Res Appl*. 2022;1-18. doi:10.1002/pip.3518

APPENDIX A: Additional comparisons with the cross-entropy classifier

Figure A1 shows additional results for the comparison of our method with a cross-entropy classifier performed in Section 5.9.

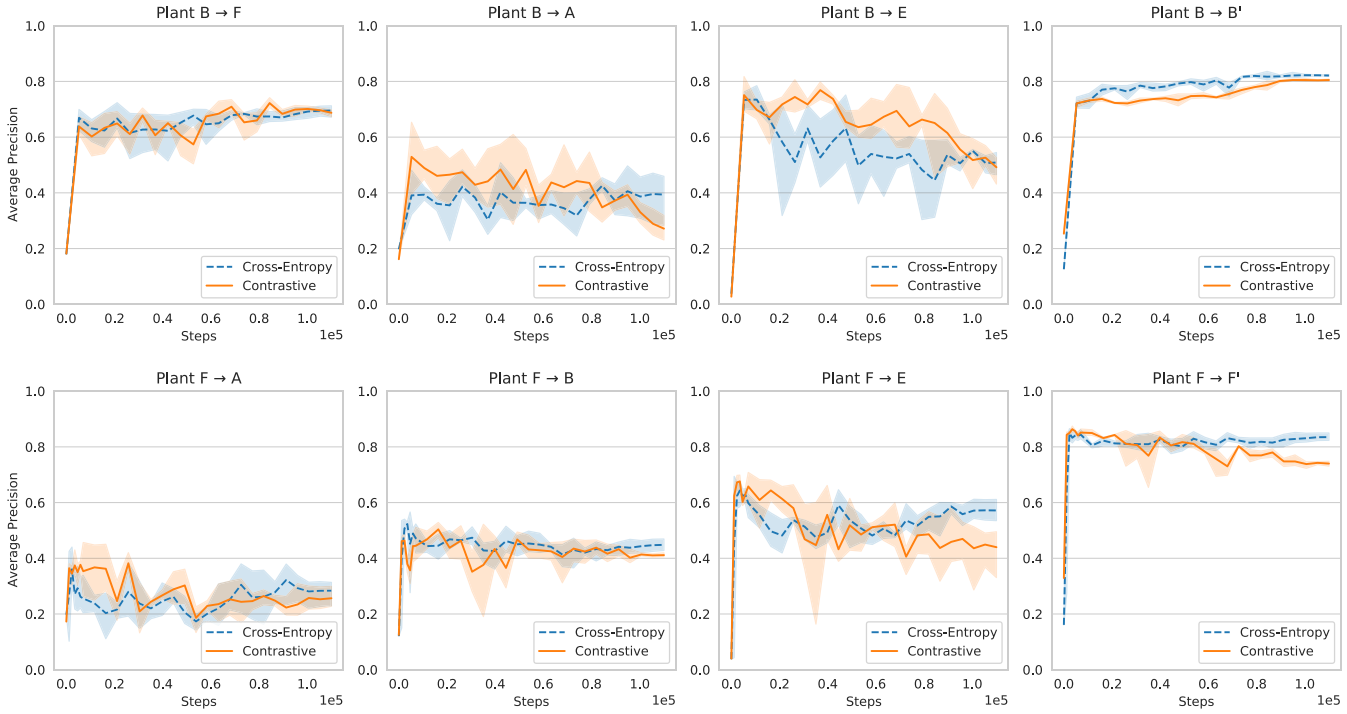

FIGURE A1 Average precision over the course of training of our contrastive *k*-NN classifier (orange line) versus a supervised binary classifier trained on cross-entropy loss (dashed blue line). The top row uses plant B as source, and the bottom row plant F. Shaded regions indicate the 95% confidence interval over three runs [Colour figure can be viewed at wileyonlinelibrary.com]

APPENDIX B: Additional comparisons for training without some anomalies

Figure B1 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 Section 5.8.

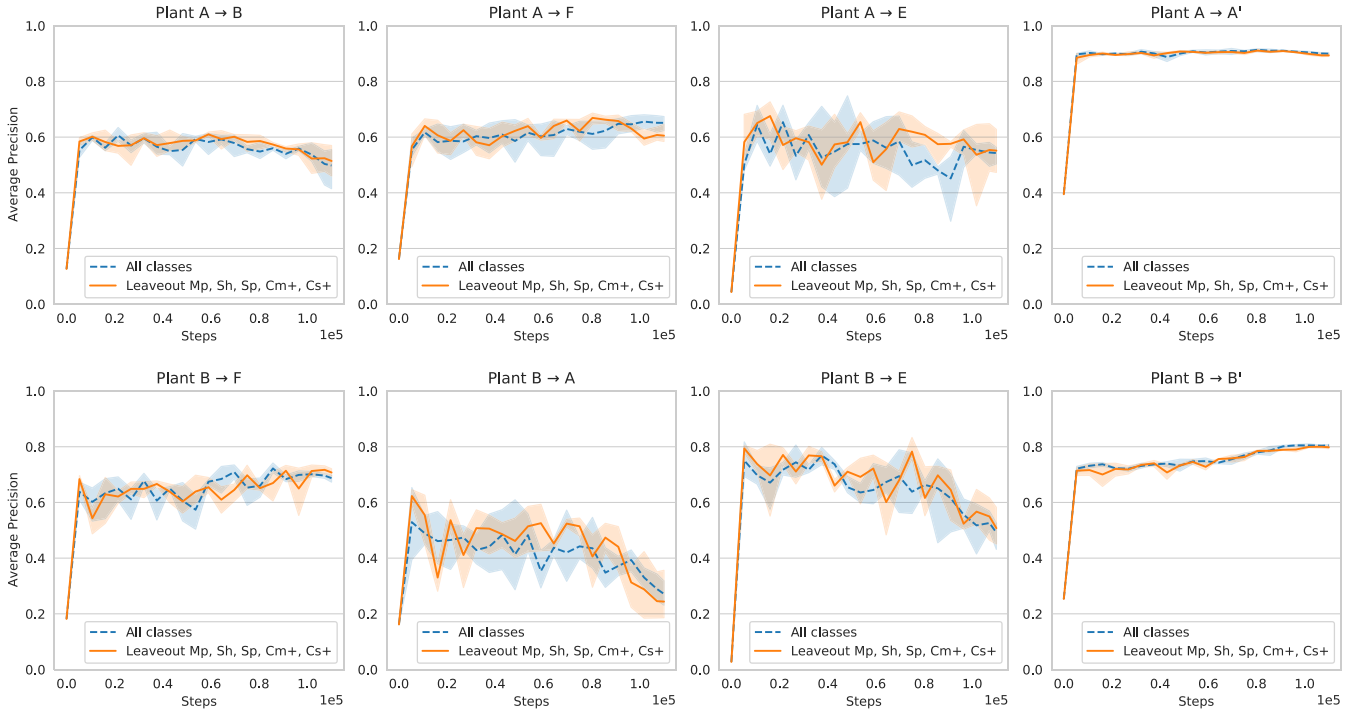

FIGURE B1 Average precision over the course of training of our contrastive *k*-NN classifier trained on datasets without anomaly classes Mp, Sh, Sp, Cm+, and Cs+ (orange line) versus the baseline trained on the full dataset (dashed blue line). Shaded regions indicate the 95% confidence interval over three runs [Colour figure can be viewed at wileyonlinelibrary.com]