Introduction

The effective utilization of solar energy has become a central focus in the pursuit of sustainable and clean energy sources. Solar panels, equipped with photovoltaic (PV) cells, are at the forefront of this endeavor, as they convert sunlight into electricity, providing a renewable and environmentally friendly energy solution. To ensure the reliability and efficiency of these solar panels, it is essential to monitor the condition of the PV cells they contain.

Traditional methods of visual inspection, primarily reliant on human expertise, are labor-intensive and often fall short when it comes to detecting subtle defects in PV cells. Moreover, certain defects that compromise the integrity of PV cells are not readily visible to the naked eye. On the flip side, visible imperfections may not necessarily lead to a decline in solar panel efficiency. To address these limitations and streamline the inspection process, there is a growing need for comprehensive scanning methods coupled with automated analysis techniques that can rapidly and rigorously assess the health of PV cells.

Electroluminescence (EL) imaging has emerged as a non-destructive and high-resolution technology that can address this challenge. It leverages the application of an electrical current to PV modules, which induces EL emission. This emission can be captured using digital cameras, resulting in images where defective cells appear darker due to the absence of irradiation in disconnected areas. The potential of EL imaging lies not only in capturing these images but also in the ability to automatically analyze them using computer vision methods for defect detection and classification.

The objective of this group project is to design and evaluate computer vision techniques that predict the health of PV cells based on EL images of solar modules. Our work centers around the ELPV dataset, a valuable resource comprising 2,624 EL images extracted from 44 different solar modules. These images have been meticulously normalized to address size and perspective variations, and any distortions introduced by the camera lens have been meticulously removed.

In this report, we will delve into the intricacies of our approach, detailing the methodologies and strategies employed in developing and testing computer vision methods. Our focus will be on classifying cell images based on their probability of defectiveness. Specifically, we aim to create classifiers that categorize an image as representing a fully functional cell, a possibly defective cell, a likely defective cell, or a certainly defective cell.

This report encapsulates our journey, from the inception of the project to the results and insights derived. We will present our methodologies, experimental outcomes, and engage in thoughtful discussions regarding the performance of our developed methods. Our aim is to not only highlight the successes but also acknowledge the limitations and areas for future research that can contribute to advancing the accuracy and efficiency of PV cell defect detection in solar modules.

Literature Review

SVM and CNN Classifications

Among the methods used in the ELPV dataset we noticed a literature called *Automatic Classification of Defective Photovoltaic Module Cells in Electroluminescence Images*. The literature reviews some of the related work about EL imaging and proposes two channels for automatic defect classification: SVM and CNN.

Related work review

In reviewing existing work, the literature has inspired us a lot in two ways. First of all, the literature suggests that the accuracy of EL defect identification is relatively low even with naked eye judgment. This is because finger interruptions in solar wafers, whether or not a solar panel is turned on, can easily be confused with real cracks or stains. This is very important, and suggests that we should adopt appropriate methods in the pre-processing stage to try to enhance the real defections and present to the images.
The second thing is that the literature indicates that CNN model is more accurate than SVM in predicting EL defect rate. This is part of what we will focus on in the training and testing of our method. We will try to find out whether the accuracy remains the same after modifying and combining the methods mentioned in the literature, and why.

SVM Classification

The authors utilize Support Vector Machines (SVM) for classification, a supervised learning approach involving feature extraction.

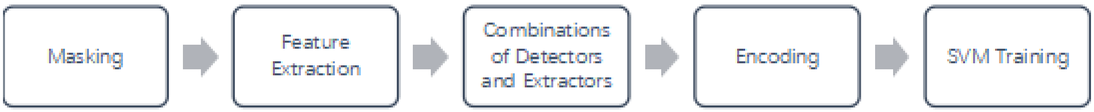


Figure : The process of SVM classification method

In the paper, the authors point out that the masking that divides the foreground background has a minor effect, so we skip this part. For feature extraction, the literature employs different popular combinations of key-point detectors and feature extractors. The literature describes the difficulties encountered by some methods and the performance of feature extraction in images, but does not explicitly compare the advantages and disadvantages of these methods.

Table 1: Investigated keypoint detectors and feature descriptors. SIFT, SURF, and KAZE (in bold) contain both a detector and a descriptor. We explored also combinations of the keypoint detectors of AGAST and KAZE with other feature descriptors. Note, the keypoints provided by SIFT and SURF were not reliable enough and thus not further evaluated.

Method	Keypoint detector	Feature descriptor
AGAST [22]	✓	✗
KAZE [23]	✓	✓
HOG [24]	✗	✓
PHOW [25]	✗	✓
SIFT [26]	(✓)	✓
SURF [27]	(✓)	✓
VGG [28]	✗	✓

Table : (Deitsch et al., 2019)

After extracting the features, the authors employ VLAD for encoding, which provides a more compact representation of the entire image based on clustering. Eventually the authors train SVMs both with a linear and a RBF kernel. In order to optimize for linear classification tasks and large datasets, LIBLINEAR is used for the linear kernel. As for the non-linear RBF kernel, LIBSVM is used.

CNN Classification

The authors considered several strategies for training CNNS, but given the limited amount of data, transfer learning achieved the best results. This method adopts a Convolutional Neural Network (CNN) approach, specifically modifying the VGG-19 architecture pre-trained on IMAGENET.

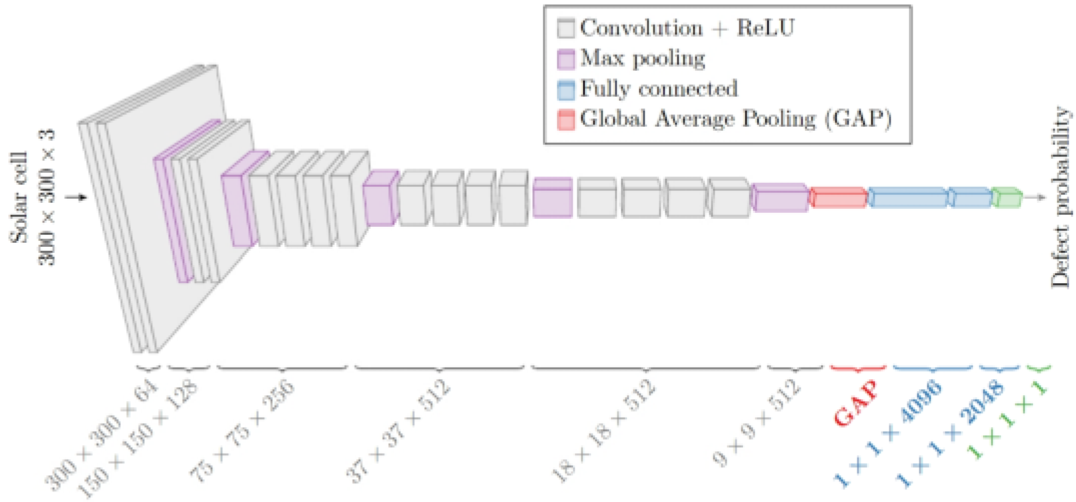


Figure : (Deitsch et al., 2019)

In order to adapt to the task of EL classification, the authors replace the two fully connected layers with the GAP layer and add two new fully connected layers to better handle the samples. On the output layer, the authors create a deep regression network to adapt to multiple classification tasks. Due to the small number of samples, perturbations including translation, rotation, scaling and flipping were added to the training samples, but at the same time, the authenticity of the original image was guaranteed.

Classification Result

In a series of SVM tests, the best performance is the linear SVM method using KAZE/VGG feature extraction. In monocrystalline solar modules, both classifiers behave similarly. In monocrystalline solar modules, the accuracy of CNN classifier is 6% higher than SVM. In general, CNN performs better than SVM. The accuracy of the former is approximately 88.42% and the latter is 82.44%.

Reflection

This literature provides us with two classification methods that are easy to understand and have high classification accuracy. It can be seen from the results that CNN has greater advantages than SVM and is better at handling complex situations. But this does not detract from the value of SVM classifiers. This provides us with a very important idea. Since the two classifiers do not have much overlap in principle and performance, can we combine them to form a better method that can weigh when it should favor SVM's judgment and when it should favor CNN's judgment? In addition, considering that the literature has a certain complexity in the detailed design of the method, we will consider to explore a method to simplify the classifier on the basis of ensuring a certain accuracy.

A pre-processing method for defect detection

In the literature review of *Automatic Classification of Defective Photovoltaic Module Cells in Electroluminescence Images*, we mentioned the inspiration for enhancing image defect features in the preprocessing phase. For this purpose we found a literature called *Defect detection based on extreme edge of defective region histogram*.

Although the purpose of this paper is not to classify ELPV data sets, it points out that existing threshold algorithms such as Otsu are poor at highlighting defective objects for images with multi-peak histogram, and the authors design a new automatic threshold algorithm. For simplicity, we use ***DBHTransform*** for short in representing this algorithm, and we will use this name for all applications of this algorithm in the rest of the report. Specifically, the algorithm consists of the following steps：

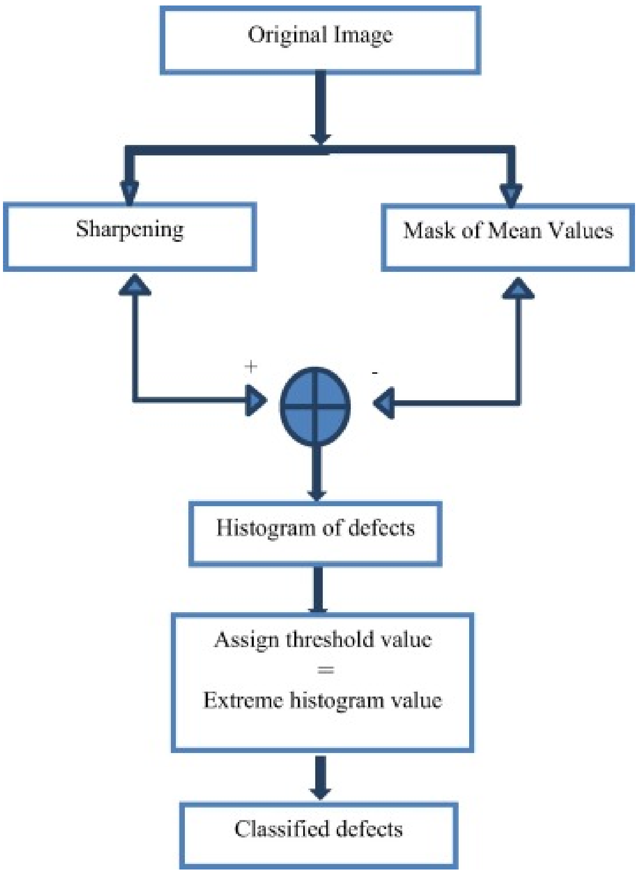


Figure : (Wakaf and Jalab, 2018)

In order to verify the performance of this algorithm on ELPV images, we randomly extract an image from each category to process the steps in the algorithm, and get the results (see appendix *Comparison DBHTransform and Otsu*).
In terms of results, compared with ordinary Otsu algorithm, this algorithm is more sensitive to image details. Although the defect prominence cannot be said to be more obvious, DBHTransform has better continuity for slight scratches. The two are similar for blocky features. Therefore, this algorithm is more excellent on the whole, and provides a choice worth considering for our pre-processing.

Methods

For this project, we envisioned two different ways to address the problem of defect detection and classification in photovoltaic (PV) cells and implemented them in different codes. The following are the selection of the methods and reasons.

Learning Approach:

Unsupervised Learning:

Unsupervised learning is a machine learning paradigm in which algorithms learn from unlabeled data without a specific target variable. In unsupervised learning, the algorithm attempts to discover patterns, structures, or relationships within the data, typically through methods such as clustering or dimensionality reduction.The primary objective of unsupervised learning is to uncover hidden structures or patterns within data for better understanding, analysis, or clustering.

Deep Learning:

Deep learning is a subfield of machine learning that involves using deep neural network models to learn data representations. Deep learning is often part of supervised or reinforcement learning, requiring labeled data for training so that the model can make predictions.The main goal of deep learning is to build a model capable of extracting features from input data and making predictions. It is commonly used for tasks such as image classification, natural language processing, speech recognition, etc.

Data Preprocessing:

Normalization:

We normalize the image pixel values to the range [-1, 1]. Normalization is a common preprocessing step in deep learning, as it helps stabilize training and allows models to converge faster. It is based on the idea that normalizing input data to a standard range can help gradient descent optimization methods converge more quickly and robustly.

DBHTransform:

The initial step in any computer vision task is data preprocessing to enhance the quality and information content of the input images. The DBHTransform class implements several preprocessing techniques:
Gaussian Blur and Sharpening: Gaussian blur helps reduce noise in images, while sharpening enhances edges and details. These techniques are used to improve the image quality and highlight important features.
Thresholding: Thresholding is used to segment the image and separate the foreground (PV cell) from the background.
Histogram Analysis: Histogram analysis helps in identifying appropriate threshold values for image binarization.

Model Architecture:

Support Vector Machine (SVM) Model with HOG Features:

HOG features are well-suited for object detection and classification tasks, including defect detection. They capture the local gradient information in an image, which is crucial for identifying texture and shape characteristics of objects. SVMs are a popular choice for classification tasks because they can handle high-dimensional feature vectors and are effective for binary and multiclass classification.

Convolutional Neural Network (CNN):

CNNs are powerful deep learning models specifically designed for image analysis. They automatically learn hierarchical features from the data, making them well-suited for complex image classification tasks. In this code, a CNN is used to classify PV cells into four categories based on their images.

Transfer Learning:

The script utilizes transfer learning by using the MobileNetV3 architecture as a base model. Transfer learning leverages pre-trained models on large datasets to improve the performance of models on specific tasks. MobileNetV3 is a state-of-the-art architecture for image classification tasks, making it a suitable choice for feature extraction.

Custom Layers:

Custom layers, such as NormalNoise and BinaryMultiplexer, are added to the model. These layers introduce noise during training and enable the model to adaptively combine defect probability predictions, respectively. Custom layers can be used to introduce specific functionalities that are not available in standard pre-trained models.

Loss Functions:

Binary Cross-Entropy Loss:

Binary cross-entropy loss is used for both cell type prediction and defect probability prediction. It is a common loss function for binary classification problems and measures the dissimilarity between predicted and true binary labels.

Mean Squared Error (MSE) Loss:

MSE loss is used for defect probability prediction. It measures the mean squared difference between predicted and true defect probabilities. MSE is commonly used for regression tasks.

Optimization:

RMSprop Optimizer:

RMSprop is used as the optimizer for both the cell type and defect probability prediction tasks. RMSprop is an adaptive learning rate optimization algorithm that helps control the learning rate during training. Adaptive optimizers like RMSprop are suitable for training deep learning models.

Learning Rate Scheduling:

Exponential Decay:

The script implements learning rate scheduling using exponential decay. Exponential decay gradually reduces the learning rate during training, allowing the model to converge more effectively. Learning rate scheduling can improve convergence, especially in deep neural networks.

Model Training and Evaluation:

Early Stopping:

The script uses early stopping with a patience parameter to stop training when the validation performance plateaus. Early stopping prevents overfitting and saves training time.

Confusion Matrix and Evaluation Metrics:

Confusion matrices are used to evaluate the model's performance for both cell type and defect probability predictions. Additionally, evaluation metrics such as accuracy, mean absolute error (MAE), and R^2 score are computed. These metrics provide insights into how well the model performs on different aspects of the task.

Data Visualization:

Matplotlib and Seaborn:

These libraries are used for data visualization, including displaying sample images, learning curves, and confusion matrices. Visualization is essential for understanding the model's behavior and performance.

Logistic Regression for Fusion:

Logistic regression is a simple yet effective method for combining the predictions of multiple models. In this code, it is used to fuse the predictions from both the SVM (HOG features) and CNN (image features) models, potentially improving the overall classification performance.

Experimental Results

In this part, we will analyze the whole test results from different methods. Since we complete two kinds of models which implement our goals in different ways, in this part we will discuss them respectively.

Test Result:

After the test processing, we can find the differences in the performance by using SVM and CNN model in silicon solar panel classification.

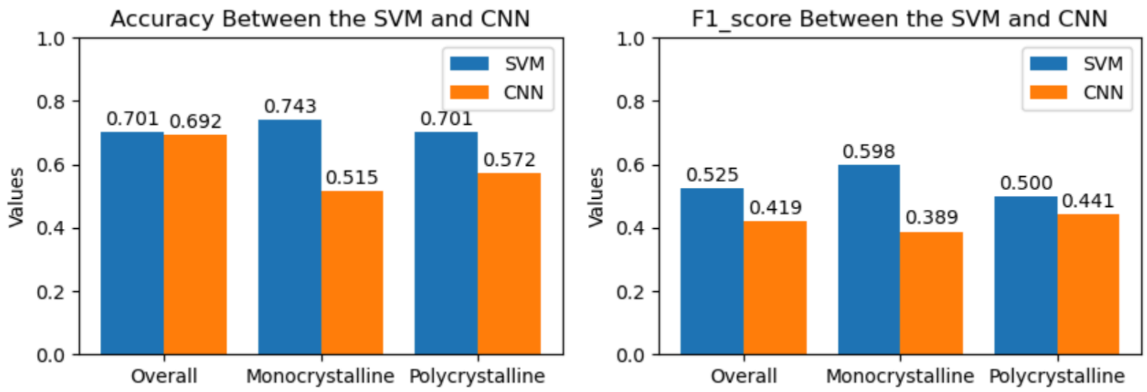
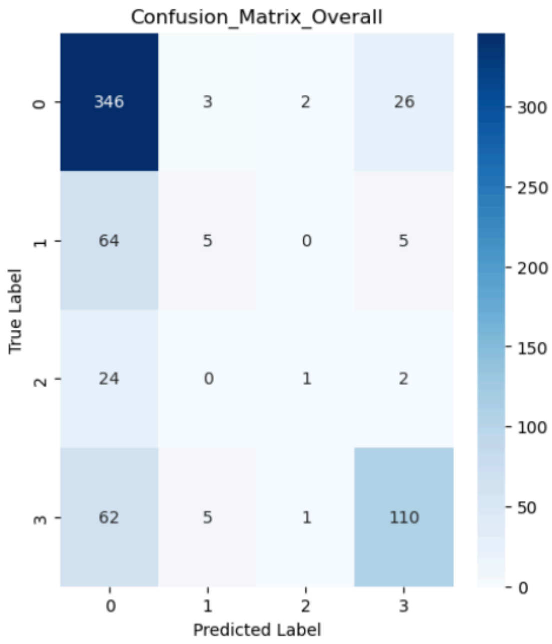


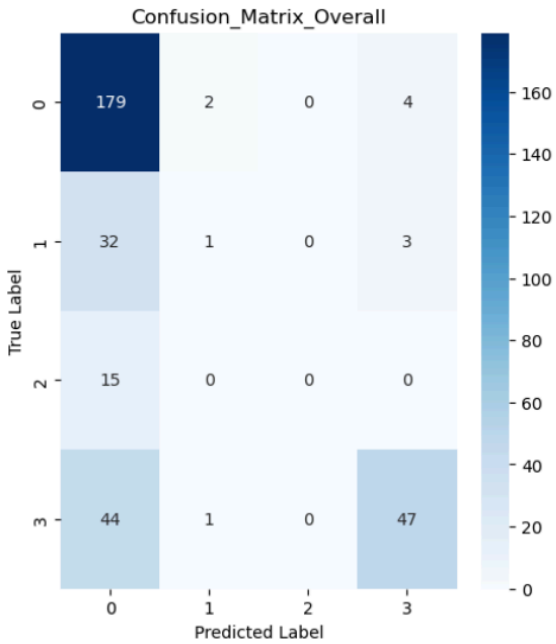
Figure1 comparison between SVM and CNN

From the above graph, we can find the SVM model performs better than CNN because of its superiority in minimizing the regularized hinge loss function as a binary classification model. Even though we add five layers to increase the complexity of the model structure, the SVM still has a better performance according to the higher accuracy and F1 score.

And also we generate the confused matrix, which can help us to deal with the result by finding the True Positive(TP), False Negative(TN), False Positive(FP), False Negative(FN). It can measure the level of model performance.



SVM Confusion Matrix

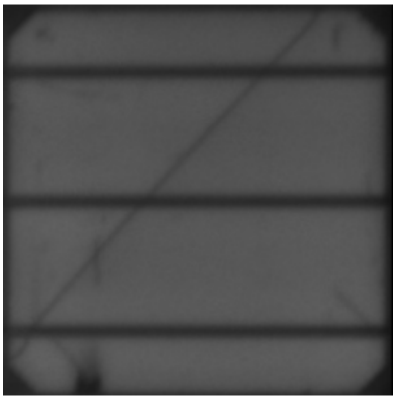


CNN Confusion Matrix

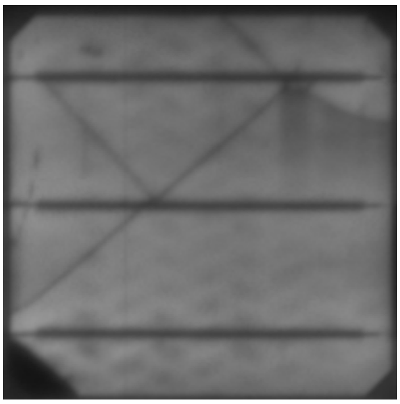
And we can also find this conclusion from the above confusion matrix graphs: the model using the SVM method can seek more correct labels graph than using the CNN method (from the matrix diagonal). What's more, by generating the confusion matrix for the monocrystalline and polycrystalline respectively, the CNN model performs worse as it always recognizes the labels as label 1, the relevant accuracy also decreases significantly.

Classification Result:

Because of the bad data quality in the train set, it contains noise and labels error, the model may learn inaccurate patterns, which affects the bad performance in the new data. For example:



cell0028



cell0431

The cell in the left has the label as fully functional (0% probability of being defective), which causes the error when scanning to the image on the left, it is recognized as fully functional while it should be the possibly defective (33% probability).
The other issues may contain the overfitting or per-processing error, which also can lead to a bad training performance.

Discussion

In this discussion, we will analyze the results and method performance of our project for solar cell defect detection using electroluminescence (EL) images.

Comparison:

Regarding the method used by the creator of the ELPV data set, we can find that in the original ELPV data set, many images and labels have non-corresponding issues, so when we use the SVM model to simulate, the training results are indeed There will be improvement, but relatively the performance is not good enough. The main reason is that there is class imbalance in the ELPV data set. When our SVM model handles imbalanced data, compared with the method used by the creator of the ELPV data set, the performance of our model has declined.

As a result, we used the pre-processing to make features in the image more apparent, and then we used two different models to find a better one to solve this problem.

In conclusion, according to the confused matrix of SVM and CNN models, we can find that the SVM model has a good performance, as the CNN model has the potential overfitting or generalization issues. Hence, by using the SVM model we can obtain a better performance than the method used by the creator of the ELPV data set.

Performance Analysis:

It is easy for us to notice the performance differences in monocrystalline and polycrystalline silicon solar panel classification in the test processing. In the above figure1, it is obvious that the in the SVM model, the monocrystalline has a better performance while the polycrystalline
performance better in the CNN model training environment.Support Vector Machines (SVM) might have several advantages in the context of polycrystalline structure:

1. **Non-linear Decision Boundaries:**
   SVMs can use the kernel functions to achieve non-linear decision boundaries, which can be advantageous for solving complex structures in polycrystalline materials, according to the non-linearly separable in the feature space for the polycrystalline structures.
2. **Noise immunity:**

SVM usually performs better in the face of noise and complex data distribution. While polycrystalline samples has greater shape and texture variations.

As for the CNN model,high feature learning capabilities might cause a good performance in the polycrystalline structures. The CNN model we designed has five convolution layers and five corresponding pooling layers, which can cause the CNN model to extract more specific features from initial data. As a result, the CNNhas more advantages in dealing with the problem of polycrystalline structures.

Future Research Direction:

1. **Advanced Models:**
   Explore more advanced deep learning models, such as convolutional neural networks with pre-training (CNN) and more efficient models, for example, ResNet, Inception, EfficientNet. These models might better capture complex features in the images.

2. **Enhanced Feature Extraction**:
   As for our models, we have already used the pre-processing with Feature Extraction. However, using more advanced feature extraction techniques can combine convolutional neural networks with a lot of complex images processing methods. For instance, specific filters in convolutional layers may better adapt to diverse crystal types.

3. **Transfer Learning:**
   Consider using transfer learning, the pre-trained models trained on large image datasets can speed up the training process while improving the accuracy, especially when we deal with relatively small datasets.

4. **Model Fusion:**
   In the future research direction, Adopt the model fusion approaches, combining predictions from multiple models to obtain a more robust classifier. Here, we can combine the SVM and CNN model together to get a better performance.

Conclusion

In our comparative analysis of silicon solar panel classification using SVM and CNN models, the results indicate the superiority of SVM in minimizing the regularized hinge loss function for binary classification. Despite efforts to enhance the CNN model's complexity with additional layers, SVM consistently demonstrated better performance, as reflected in higher accuracy and F1 scores. Notably, the identified data quality issues within the training set,

including noise, label errors, and inaccurate patterns, underscore the challenges associated with model training on imperfect datasets.
While SVM's success in handling class imbalance and mitigating the impact of data quality issues is evident, the CNN model showed signs of potential overfitting or generalization issues. Future research directions should explore advanced deep learning models like ResNet, Inception, or EfficientNet to better capture intricate features in images. Additionally, incorporating more sophisticated feature extraction techniques and considering transfer learning, particularly with pre-trained models on larger datasets, could address the CNN model's limitations and improve overall classification accuracy.
Moreover, the performance disparities observed between monocrystalline and polycrystalline silicon solar panels suggest nuanced characteristics that influence the choice of model. SVM's strength in handling non-linear decision boundaries aligns well with the varied structures of polycrystalline materials, while CNN's feature learning capabilities offer advantages in dealing with the complexity inherent in polycrystalline structures. A potential avenue for future research involves the fusion of SVM and CNN models to harness the complementary strengths of each, paving the way for a more robust and accurate classifier in solar panel defect detection.

Renferences

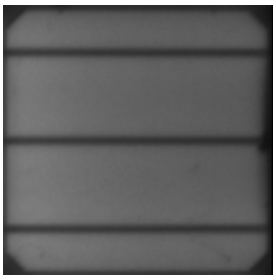
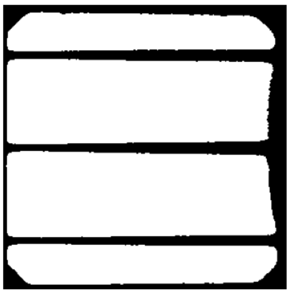
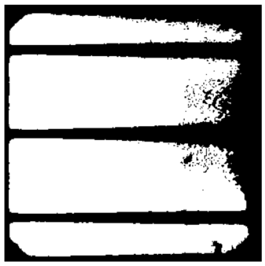
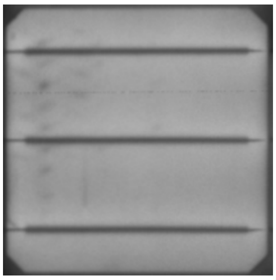
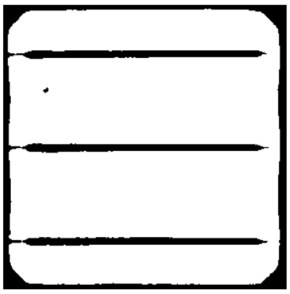
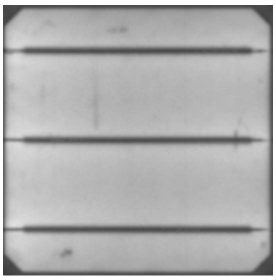
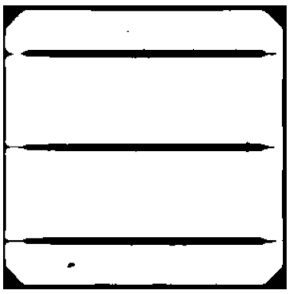
ELPV Dataset. A Benchmark for Visual Identification of Defective Solar Cells in Electro
luminescence Imagery. https://github.com/zae-bayern/elpv-dataset

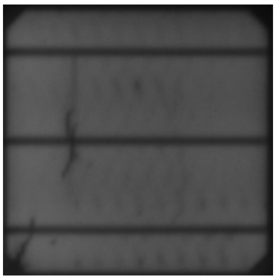
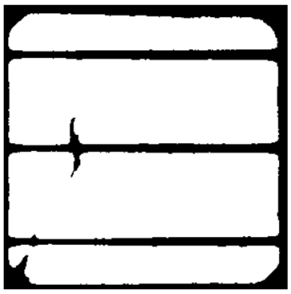
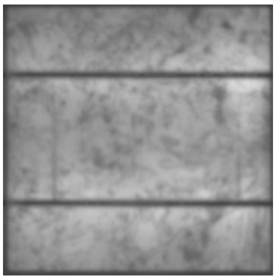
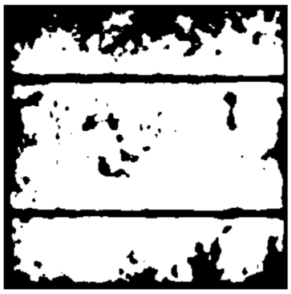
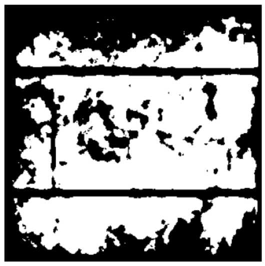
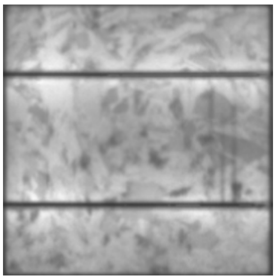
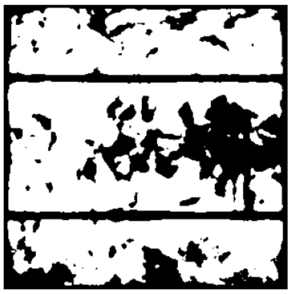
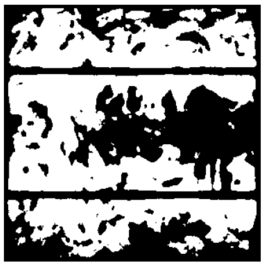
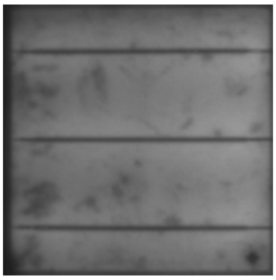
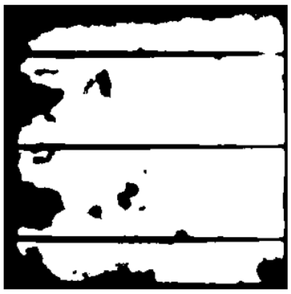
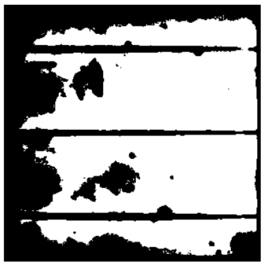
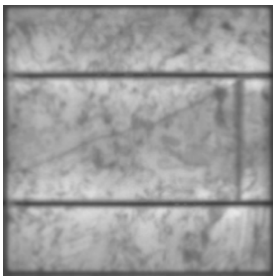
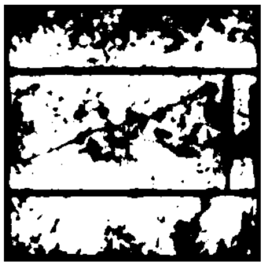
C. Buerhop et al. A Benchmark for Visual Identification of Defective Solar Cells in Electro
luminescence Imagery. European PV Solar Energy Conference and Exhibition (EU PVSEC),
2018. http://dx.doi.org/10.4229/35thEUPVSEC20182018-5CV.3.15

S. Deitsch et al. Automatic Classification of Defective Photovoltaic Module Cells in Electro
luminescence Images. Solar Energy, vol. 185, June 2019, pp. 455-468. https://doi.org/10.1016/j.solener.2019.02.067

Wakaf, Z. and Jalab, H.A. (2018). Defect detection based on extreme edge of defective region histogram. *Journal of King Saud University - Computer and Information Sciences*, 30(1), pp.33–40. doi:https://doi.org/10.1016/j.jksuci.2016.11.001.

Appdendix

Type	Class	Original Image	Otsu	DBHTransformed
mono	0%			
	33%			
	67%			

	100 %			
poly	0%			
	33%			
	67%			
	100 %			

Comparison DBHTransform and Otsu