PROJECT ABSTRACT

Solar Panel Defect Detection using CNN

A transfer-learned CNN that classifies cracks and hotspots in solar-cell EL images, with Grad-CAM localization and notebook-computed accuracy, F1 and confusion matrix.

01. Title

Solar Panel Defect Detection using CNN

02. Introduction / Background

A solar farm is thousands of panels, and defects — micro-cracks from handling, hotspots from failing cells, delamination — silently cut output long before anyone notices. Manual inspection of a large installation is slow and inconsistent, and many defects are invisible to the naked eye anyway. Electroluminescence (EL) imaging solves the visibility problem: running a current through the cell makes it glow, and cracks and dead regions show up as dark patterns. This project adds the classification layer on top of that imaging: a ResNet or EfficientNet CNN, pretrained on ImageNet and fine-tuned on public solar-cell EL datasets, classifies each cell image and localizes the defect with Grad-CAM so the inspector sees where to look. The evaluation notebook computes accuracy, F1 and the confusion matrix on the validation split during the build.

03. Problem Statement

Solar installations lose output to defects that manual inspection misses, because panels are numerous and many faults are invisible without EL or thermal imaging. The need is an automated classifier that categorizes cell defects from inspection images, localizes each defect for the inspector, aggregates per-cell results into panel-level summaries, and evaluates itself with a confusion matrix that names the confused defect classes — framed as an inspection aid, never as certified quality-control equipment.

04. Objectives

The objectives are to fine-tune a pretrained ResNet or EfficientNet CNN on public solar-cell EL image datasets; to build an EL enhancement and cell-cropping pipeline; to classify cells into healthy and defect classes (cracks, hotspots, cell damage); to localize defects with Grad-CAM overlays; to aggregate per-cell results into panel-level defect summaries in batch mode; and to evaluate with accuracy, F1 and a confusion matrix on the validation split via the included notebook.

- Fine-tune a pretrained ResNet or EfficientNet on solar-cell EL datasets.
- Build an EL enhancement and cell-cropping pipeline.
- Classify cells into healthy and defect classes (cracks, hotspots, damage).
- Localize defects with Grad-CAM overlays.
- Aggregate per-cell results into panel-level summaries in batch mode.
- Evaluate with accuracy, F1 and confusion matrix on the validation split during the build.

05. Proposed System / Working

EL and inspection images are loaded and contrast-enhanced with OpenCV, then cells are cropped and normalized to the classifier's input size. A fine-tuned ResNet or EfficientNet predicts the defect class with a probability score. Batch mode aggregates per-cell results into a panel-level defect summary. Grad-CAM overlays highlight the regions that drove each defect prediction, giving the inspector a localization map. The demo app serves predictions with overlays for uploaded cell images. The evaluation notebook computes accuracy, F1 and the confusion matrix on the validation split.

- OpenCV contrast enhancement of EL images.
- Cell cropping and normalization.
- Transfer-learned defect classification with probability scores.
- Batch-mode aggregation into panel-level summaries.
- Grad-CAM defect localization overlays.
- Notebook evaluation: accuracy, F1, confusion matrix.

06. Technologies / Components Used

The project uses Python 3.10 and PyTorch for training and inference, with torchvision supplying the pretrained ResNet and EfficientNet backbones. Public solar-cell EL image datasets provide training data; OpenCV handles EL enhancement and defect overlays; NumPy and scikit-learn compute metrics and the confusion matrix; a Jupyter notebook performs the evaluation; and a Flask app serves the demo with a panel-image upload interface.

- Python 3.10, PyTorch (CNN training and inference).
- torchvision (pretrained ResNet and EfficientNet backbones).
- Public solar-cell EL (electroluminescence) image datasets.
- OpenCV (EL image enhancement and defect overlays).
- Jupyter notebook (evaluation with accuracy and F1).
- NumPy, scikit-learn (metrics and confusion matrix).
- Flask demo app with panel-image upload interface.

07. Key Features

Key features include a transfer-learned ResNet/EfficientNet defect classifier, training on public solar-cell EL datasets, defect categories covering cracks, hotspots and cell damage, an EL enhancement pipeline, an evaluation notebook computing accuracy, F1 and the confusion matrix, Grad-CAM defect localization, batch panel-inspection mode, and a demo app for cell images.

- Transfer-learned CNN defect classifier.
- Training on public solar-cell EL image datasets.
- Defect categories: cracks, hotspots, cell damage.
- EL image enhancement pipeline.
- Evaluation notebook: accuracy, F1, confusion matrix.
- Grad-CAM defect localization overlays.
- Batch panel-inspection mode.
- Demo app for uploading cell images.

08. Applications / Use Cases

Panel defect detection applies to solar-farm maintenance triage, manufacturing quality-inspection prototypes, operations-and-maintenance service tooling, and installer training. It also demonstrates transfer learning, EL imaging, and confusion-matrix error analysis as viva-ready deep-learning concepts — always framed as an inspection aid, not a certified quality-control system.

- Solar-farm maintenance triage and inspection aids.
- Manufacturing quality-inspection prototypes.
- Operations-and-maintenance service tooling.
- Installer and technician training material.
- Teaching transfer learning and error analysis.

09. Results & Scope

The expected outcome is a working defect classifier with a design target of ~85%+ validation accuracy, where the actual accuracy, F1 and confusion matrix are computed by the training notebook during the buyer's training run — the report's numbers come from that run, not from a claim. The confusion matrix names the confused defect classes explicitly, which doubles as viva error-analysis material. Scope covers the defect classes labeled in the public EL datasets; documented limits include micro-cracks near the noise floor and imaging setups outside the training data. This is an educational inspection prototype, not a certified quality-control system — it must not be the sole basis for warranty or safety decisions. Future scope includes drone-based panel imaging, instance segmentation for defect sizing, and multi-modal EL plus thermal fusion.

- Design target ~85%+ validation accuracy; actual metrics computed by the notebook during the buyer's training run.
- Confusion matrix naming the confused defect classes.
- Scope: defect classes labeled in the public EL datasets.
- Documented limits: sub-noise-floor micro-cracks, out-of-coverage imaging setups.
- Honest framing: inspection prototype, never certified quality-control equipment.
- Future scope: drone imaging, defect sizing, EL plus thermal fusion.

10. Conclusion

Transfer learning on EL imagery turns solar defect inspection from a manual, eyeball-based task into a measurable classification pipeline, and this project delivers it as a complete auditable experiment: enhancement and cropping, fine-tuning, Grad-CAM localization, batch panel summaries, and a notebook that measures accuracy, F1 and the confusion matrix on the validation split. Built to order, it ships with source code, the evaluation notebook, report, presentation, and viva preparation material.

11. References

The project cites the standard public EL defect-classification study, the ResNet architecture, and the Grad-CAM interpretability method.

- S. Deitsch et al., 'Automatic Classification of Defective Photovoltaic Module Cells in Electroluminescence Images,' Solar Energy, vol. 185, 2019.
- K. He, X. Zhang, S. Ren, and J. Sun, 'Deep Residual Learning for Image Recognition,' Proc. CVPR, 2016.
- R. R. Selvaraju et al., 'Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization,' Proc. ICCV, 2017.