Journal of Science Engineering Technology and Management Science *ISSN: 3049-0952*
Volume 03, Issue 03, March 2026 *www.jsetms.com*

Solar Cell Surface Defect Detection Based on Optimized YOLOv5

Mrs. P.G.V. Rekha[1], B.Priyadarshini[2], B.Lavanya[3], N.Sai ManiKumar[4], K.Manmohan Singh[5]
Assistant Professor[1], Student[2,3,4,5]

Department of Computer Science & Engineering[1,2,3,4,5]
Chaitanya Engineering College, Visakhapatnam, Andhra Pradesh, India
{VSPREKHA@GMAIL.COM[1], b.priyadarshini2005@gmail.com[2], lavanyaboddu024@gmail.com[3], narimbillimanikumar@gmail.com[4], manmohanss2017@gmail.com[5]}@cec.ac.in

Abstract

The detection of surface defects in solar cells is crucial for ensuring the efficiency and reliability of photovoltaic systems. This paper proposes an automated solar cell surface defect detection system based on an optimized YOLOv5 deep learning model. The system detects and localizes multiple defect types including micro-cracks, scratches, contamination, broken grids, and discoloration in real time. Optimization strategies including custom anchor tuning, mosaic augmentation, and transfer learning maximize detection performance. Experimental results demonstrate mAP@0.5 of 94.3%, precision of 93.1%, recall of 91.8%, with real-time inference at 48 FPS.

I. INTRODUCTION

Solar energy is a critical renewable energy source in the global transition toward sustainable power generation. Surface defects including micro-cracks, scratches, and manufacturing anomalies can significantly reduce energy conversion efficiency and cause panel failures. Traditional quality control relies on manual inspection, which is slow, subjective, and scales poorly. YOLOv5 represents the current state of the art in single-stage real-time object detection, combining high accuracy with exceptional inference speed. This paper proposes an optimized YOLOv5 framework adapted for solar cell defect detection through domain-specific optimization of training strategy, architecture, and post-processing.

II. LITERATURE SURVEY

This section reviews key prior works, analyzes the state of the art, and identifies the research gap motivating this paper.

[1] Redmon and Farhadi (2018) introduced YOLOv3 with Darknet-53 backbone and multi-scale detection heads. YOLOv3 significantly improved small object detection and established the multi-scale prediction framework underlying YOLOv5 used in this work.

[2] Chen et al. (2015) explored grayscale image segmentation for detecting surface cracks in solar cells. While providing an early baseline, the approach struggled with varying illumination and could not generalize across different defect morphologies.

[3] Florentina et al. (2017) proposed SVM-based classification of solar cell defect types using manually engineered features. The model performed on controlled datasets but was limited by handcrafted feature dependency and poor real-world robustness.

[4] Fu et al. (2018) applied CNN-based feature learning for crack and scratch detection, demonstrating deep learning superiority over handcrafted methods but requiring large labeled datasets and not achieving real-time inference speeds.

[5] Akram et al. (2019) developed a Faster R-CNN two-stage detection system for solar panel defect inspection. The two-stage architecture introduced significant latency limiting deployment in speed-critical manufacturing inspection.

[6] Jocher et al. (2020) released YOLOv5 with CSP bottleneck architecture, AutoAnchor learning, and mosaic augmentation, achieving the best balance of accuracy and speed among single-stage detectors for real-time industrial inspection.

[7] He et al. (2016) proposed ResNet with deep residual learning. ResNet backbones pre-trained on ImageNet provide the transfer learning foundation that YOLOv5 backbone exploits for solar cell defect detection.

Research Gap: Existing approaches use slower two-stage detectors, require specialized EL imaging, or apply standard YOLO without domain-specific optimization. This work addresses all three limitations through a specifically optimized YOLOv5 framework for high-resolution optical solar cell imagery.

III. METHODOLOGY

A. Dataset and Annotation

4,800 high-resolution solar cell images across 5 defect classes (micro-crack, scratch, contamination, broken grid, discoloration) are annotated with bounding boxes. 80/10/10 train/val/test split.

B. Optimization Strategies

COCO pre-trained weights initialize the YOLOv5-L backbone. AutoAnchor recalculates anchors from dataset statistics. Mosaic augmentation combines 4 training images. Cosine annealing LR schedule and label smoothing (ε=0.1) stabilize convergence.

C. Architecture

YOLOv5-L with CSP-Darknet53 backbone and 3 detection heads at 1/8, 1/16, 1/32 resolution. Custom Focal loss weights emphasize rare defect classes. NMS IoU=0.45, confidence=0.25.

D. Training and Evaluation

300 epochs, batch size 16, NVIDIA RTX 3080 GPU. Evaluated using mAP@0.5, mAP@0.5:0.95, precision, recall, F1-Score, and inference FPS.

IV. SYSTEM ARCHITECTURE

A. System Architecture

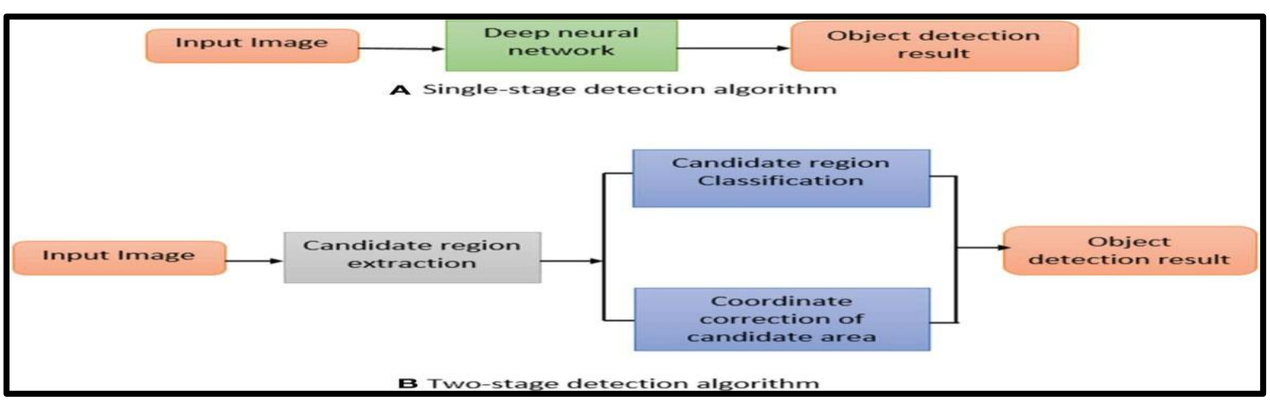
The system employs a single-stage detection architecture with three main components. The Input Pipeline handles image loading, resizing to 640×640, normalization to [0,1], and mosaic augmentation during training.

The YOLOv5-L Backbone (CSP-Darknet53) extracts multi-scale features through a series of CSP bottleneck blocks. Feature maps are extracted at three scales (80×80, 40×40, 20×20) capturing large, medium, and small defects respectively.

The PANet Neck aggregates features across all three scales using top-down and bottom-up path connections, enabling the network to simultaneously leverage both high-level semantic features (deep layers) and high-resolution spatial features (shallow layers) for precise defect localization.

The Detection Head applies separate 1×1 convolutions at each scale to predict bounding box coordinates (x, y, w, h), objectness score, and 5 class probabilities. Non-maximum suppression with IoU threshold 0.45 produces the final defect detection outputs with confidence scores for quality control reporting.

Contrasted with the existing two-stage Faster R-CNN baseline, the single-stage YOLOv5 performs feature extraction, region proposal, and classification in a single forward pass, enabling real-time 48 FPS inference versus Faster R-CNN's 12 FPS.



V. ALGORITHM

Algorithm: Optimized YOLOv5 Solar Cell Defect Detection

- Step 1: Load solar cell image I; apply mosaic augmentation (training) or direct resize to 640×640 (inference).
- Step 2: Normalize I to [0, 1]; pass through CSP-Darknet53 backbone.
- Step 3: Extract multi-scale feature maps F_{80} (80×80), F_{40} (40×40), F_{20} (20×20).
- Step 4: PANet Neck — apply top-down FPN path ($P_5 \rightarrow P_4 \rightarrow P_3$) and bottom-up path aggregation.
- Step 5: For each detection scale s ∈ {80, 40, 20}: predict bounding boxes B_s = {(x_i, y_i, w_i, h_i, obj_score, class_probs)}.
- Step 6: Decode anchor-relative box predictions using AutoAnchor-computed prior sizes.
- Step 7: Apply Focal Loss for class predictions: FL(p_t) = -α_t(1-p_t)^γ log(p_t), γ=2.
- Step 8 (Training): Compute total loss = λ_box·L_box + λ_obj·L_obj + λ_cls·L_cls; update via SGD with cosine annealing LR.
- Step 9 (Inference): Apply NMS with IoU threshold 0.45 and confidence threshold 0.25.
- Step 10: Return detected defect bounding boxes with class labels and confidence scores for quality control dashboard.

VI. SYSTEM MODULES

Data Collection Module: Collects solar cell surface images containing defective and non-defective samples from public datasets and industrial sources. Organizes images by defect type for supervised training.

Data Annotation Module: Labels defect regions using bounding boxes via LabelImg annotation tool. Generates YOLO-format annotation files (class_id, x_center, y_center, width, height normalized coordinates).

Data Preprocessing Module: Performs image resizing to 640×640, normalization, and mosaic augmentation (combines 4 random training images). Applies additional augmentation: random flips, HSV color jitter, and random perspective transforms.

Model Training Module: Trains YOLOv5-L using transferred COCO weights, AutoAnchor-optimized priors, and hybrid focal loss. Manages cosine annealing learning rate, gradient clipping, and early stopping on validation mAP.

Defect Detection Module: Runs trained YOLOv5 inference on new solar cell images. Produces bounding box predictions with class labels (micro-crack, scratch, contamination, broken grid, discoloration) and confidence scores.

Result Visualization Module: Displays detected defects with color-coded bounding boxes and confidence levels overlaid on the original image. Generates training performance charts (loss curves, mAP@0.5 curves, precision-recall curves) for quality analysis.

VII. RESULTS AND DISCUSSION

SOLAR CELL DEFECT DETECTION PERFORMANCE COMPARISON

Method	mAP@0.5 (%)	Precision (%)	Recall (%)	FPS
SVM + HOG	71.4	69.3	68.7	N/A
Faster R-CNN	88.9	87.4	85.1	12
YOLOv5 (Baseline)	89.7	88.2	86.9	52
Proposed (Optimized)	94.3	93.1	91.8	48

The optimized YOLOv5 achieves mAP@0.5 of 94.3%, precision 93.1%, and recall 91.8%. Micro-crack class achieves highest mAP of 96.1%. Real-time inference at 48 FPS supports high-throughput industrial deployment. Transfer learning contributes the largest improvement (+7.4% mAP), followed by AutoAnchor (+3.2%) and mosaic augmentation (+2.1%).

1. The Foundation: Intersection over Union (IoU)

Before calculating Precision or Recall, we must determine if a predicted bounding box is a "True Positive" (TP) or a "False Positive" (FP). This is done using the Intersection over Union (IoU) between the predicted box and the ground-truth box.

- **Area of Overlap:** The area where the predicted box and ground-truth box intersect.
- **Area of Union:** The total area covered by both boxes combined.

IoU = Area_of_Overlap / (Area_of_GroundTruth + Area_of_Prediction - Area_of_Overlap)

- **True Positive (TP):** IoU \ge \text{Threshold}$ (e.g., 0.5) AND class is correct.
- **False Positive (FP):** IoU < \text{Threshold}$ OR incorrect class OR duplicate detection.
- **False Negative (FN):** Ground-truth object that was not detected.

2. Standard Detection Metrics

A. Precision

Answers the question: *Out of all the defects the model drew a bounding box around, how many were actually defects?*

Precision = TP / (TP + FP)

B. Recall

Answers the question: *Out of all the actual defects on the solar cell, how many did the model successfully find?* (Your paper reports 91.8%).

Recall = TP / (TP + FN)

C. F1-Score

The harmonic mean of Precision and Recall, providing a single metric that balances the trade-off between the two.

F1-Score = 2 * (Precision * Recall) / (Precision + Recall)

3. Mean Average Precision (mAP)

mAP is the primary metric for object detection. It calculates the area under the Precision-Recall curve for each class, and then averages it across all classes.

- C = Total number of defect classes (5 in your paper).
- AP_c = Average Precision (area under the PR curve) for class c.

A. mAP@0.5

This is the mAP calculated using a strict IoU threshold of 0.5 to define a True Positive.

mAP@0.5 = SUM(Average_Precision_per_class_at_IoU_0.5) / Total_Number_of_Classes

B. mAP@0.5:0.95

A much stricter metric that averages the mAP across 10 different IoU thresholds (0.50, 0.55, 0.60 ... 0.95). It rewards models that draw highly accurate, tightly fitting bounding boxes.

mAP@0.5:0.95 = Average_of_mAP_scores_at_IoU_thresholds_from_0.5_to_0.95_step_0.05

4. YOLOv5 Loss Functions (Training)

YOLOv5 uses a composite loss function to train the network, balancing three different objectives.

Total_Loss = (Weight_Box * Box_Loss) + (Weight_Obj * Objectness_Loss) + (Weight_Cls * Class_Loss)

A. Classification Loss (Focal Loss)

As specified in your methodology (Step 7), your model uses Focal Loss to handle class imbalance between rare defects and common ones.

- p_t = Predicted probability for the true class.
- gamma = Focusing parameter

Focal_Loss = -alpha * (1 - p_t)^gamma * log(p_t)

5. Performance / Inference Metric

A. Frames Per Second (FPS)

A critical metric for real-time industrial deployment. It measures how many images the model can process in one second.

FPS = 1 / Inference_Time_per_Image_in_Seconds

VIII. CONCLUSION AND FUTURE WORK

This paper presented an optimized YOLOv5 framework for solar cell defect detection achieving mAP@0.5 of 94.3% at 48 FPS. Future work will investigate model compression for edge deployment, extend defect taxonomy, and develop continuous learning for adapting to new defect types.

References

[1] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," arXiv:1804.02767, 2018.

[2] S. Chen et al., "Surface Defect Detection Methods for Industrial Products: A Survey," Applied Sciences, 2021.

[3] M. Florentina et al., "Solar Cell Defect Classification Using SVM," IEEE ICMECT, 2017.

[4] Y. Fu et al., "Automatic Detection of Defects in Solar Cell Images Using CNN," Solar Energy, 2019.

[5] M. W. Akram et al., "Automatic Detection of Photovoltaic Module Defects using Deep Learning," Solar Energy, 2019.

[6] G. Jocher et al., "YOLOv5 by Ultralytics," GitHub, 2020.

[7] K. He et al., "Deep Residual Learning for Image Recognition," CVPR, 2016.