IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 6, October 2026
Impact Factor: 8.2

Good and Bad Seeds Detection Using Image Processing and Deep Learning with Automatic Sorting

Mayur Ahire[1], Tejal Deore[2], Payal Thakare[3], Harshali Deore[4], D. N. Pawar[5]
Department of Electronics and Telecommunication[1-5]
MVP's Rajarshi Shahu Maharaj Polytechnic, Gangapur Road, Nashik, Maharashtra, India
mayur748@gmail.com, tejaldeore015@gmail.com,
payalthakare184@gmail.com, deoreharshu105@gmail.com

Abstract: *Seed quality decides how well a crop germinates, how uniform the field is and how much it yields, yet most seed lots are still graded by eye, which is slow, tiring and different from one person to the next. This paper presents the design of a low-cost system that detects good and bad seeds from images and sorts them automatically. A camera inside a lighting box captures each seed, image processing separates the seed from the background, and a lightweight convolutional neural network (CNN) classifies it as good or bad. A Raspberry Pi drives a servo gate that diverts the seed to the correct bin, shows the result on an LCD and publishes the counts over MQTT so that a phone shows the running totals. A decision rule with a confidence band sends doubtful seeds to the reject bin. The paper gives the processing pipeline, a dataset plan, evaluation metrics, a latency budget and a test protocol. No measured results are reported: the figures in the paper are design targets, and the result tables are laid out for the measurements on the prototype, which are planned as the next stage*

Keywords: Seed Quality, Image Processing, Deep Learning, Convolutional Neural Network, Automatic Sorting, Raspberry Pi

I. INTRODUCTION

Good seed is the first input of farming. A good seed is whole, plump, uniform in colour and free of disease; a bad seed is broken, shrivelled, discoloured, mouldy, insect-damaged or immature. Poor seed lowers germination, makes the crop uneven and can carry disease into the field. In most seed shops, farms and small processing units the lot is still checked by hand. Manual sorting is slow, the eyes tire after a short time, and two people seldom agree on every doubtful seed.

Machine vision offers a non-destructive and repeatable alternative. Deep learning has been applied to many agricultural tasks [1] because a network learns its features directly from the images [2]. Before that, classical pipelines described each seed by hand-designed colour, shape and texture features [3] and passed them to a classifier such as a support vector machine (SVM) [4]. Lightweight networks such as MobileNet and MobileNetV2 [5], [6] now run on small, inexpensive boards, so a complete inspection unit can be built at modest cost.

Many studies concentrate on the accuracy of a model on an image data set. Fewer describe the whole machine: how the seed is lit and triggered, how the result drives a gate, how a doubtful seed is treated and how the behaviour will be measured. This paper therefore contributes (a) a layered architecture that joins a lighting box, camera, Raspberry Pi, sorting gate and MQTT-based phone monitoring; (b) a processing pipeline that uses Otsu thresholding [7] and OpenCV [8] for segmentation and a MobileNetV2 classifier [6]; (c) a decision rule with a confidence band that favours rejecting doubtful seeds; and (d) a dataset plan, a set of metrics [10] and a test protocol. The paper presents a design; quantitative results will be reported after the prototype is measured.

Copyright to IJARSCT
www.ijarsct.co.in
DOI: 10.48175/IJARSCT-40374

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 6, October 2026
Impact Factor: 8.2

II. LITERATURE REVIEW

1) Kamilaris and Prenafeta-Boldú [1] survey deep learning in agriculture. They find it used for many tasks, such as plant disease detection, weed identification and fruit counting, and report that it usually gives higher accuracy than the traditional image-processing methods in the studies they reviewed.

2) LeCun et al. [2] explain that deep networks learn layered representations directly from raw data, which removes the need to design features by hand. This is the main reason a CNN is chosen here as the primary classifier.

3) Haralick et al. [3] introduced grey-level co-occurrence texture features, and Cortes and Vapnik [4] introduced the support vector machine. Colour, shape and texture features with an SVM form the classical baseline against which the CNN will be compared on the same test set.

4) Howard et al. [5] and Sandler et al. [6] present MobileNet and MobileNetV2, which use depthwise separable convolutions and inverted residual blocks to reduce computation. This makes them suitable for small embedded boards. Kingma and Ba [9] describe the Adam optimiser used to train the network.

5) Otsu [7] gives an automatic threshold for separating an object from its background, Bradski [8] describes the OpenCV library that implements these operations, and Sokolova and Lapalme [10] analyse performance measures for classification tasks, which guide the choice of metrics in this paper.

III. PROPOSED SYSTEM

The proposed work is the design of a camera-based system that detects good and bad seeds and sorts them into two bins. A seed is placed at the capture point inside a lighting box. An infrared sensor detects it, the camera captures an image under fixed light, and a Raspberry Pi 4 segments the seed, classifies it with the CNN, drives the sorting servo, updates the 16×2 LCD and publishes the counts to an MQTT broker [11]. A mobile app subscribes to the same data.

The distinguishing features are: the lighting and camera distance are fixed so that the images are repeatable; the classifier is compared with an SVM baseline on the same data; a confidence band sends doubtful seeds to the reject bin; the LCD and sorting gate keep working when the network fails; and the phone shows the counts together with a flag when the node is offline.

IV. EXISTING SYSTEM

In the existing practice seeds are sorted by hand. Sieves separate seeds by size and air flow separates them by weight, but neither can see discolouration, mould, cracks or insect damage. Industrial colour sorters can do this, but they are costly and built for large lots, so small farms, seed shops and college laboratories rarely have them. Published camera-based work usually reports the accuracy of a model on a data set and gives less detail on the lighting, the trigger, the actuator, the treatment of doubtful seeds and the measurement plan. The existing approach therefore leaves a gap for a compact, low-cost system that treats imaging, classification and sorting as one designed machine.

V. SYSTEM ARCHITECTURE

The system has four functional layers and one energy subsystem (Fig. 1). The sensing layer holds the camera, an infrared sensor at the capture point and an LDR that checks the light level. The control layer is the Raspberry Pi. The communication layer carries MQTT messages over Wi-Fi. The presentation layer is the LCD, the sorting gate, the buzzer and the mobile app. The energy subsystem (supply, regulator, 5 V bus) feeds all layers.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 6, October 2026
Impact Factor: 8.2

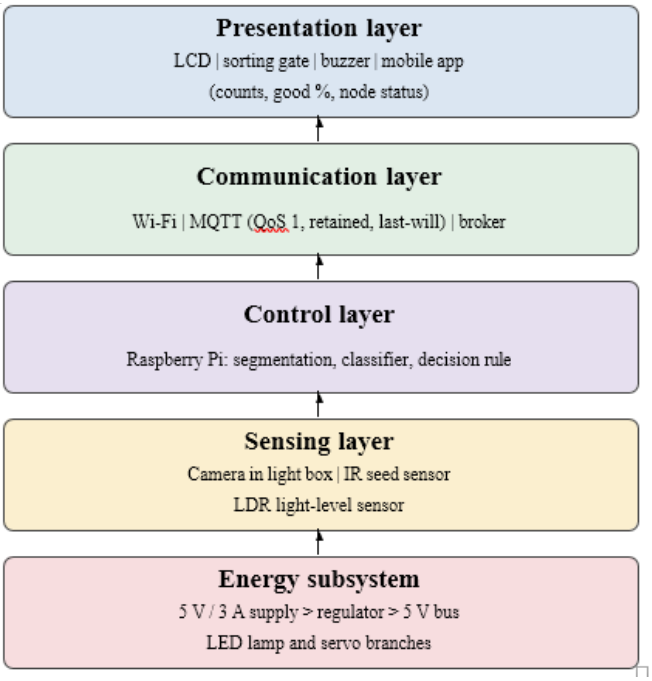


Fig. 1. System Architecture

VI. BLOCK DIAGRAM

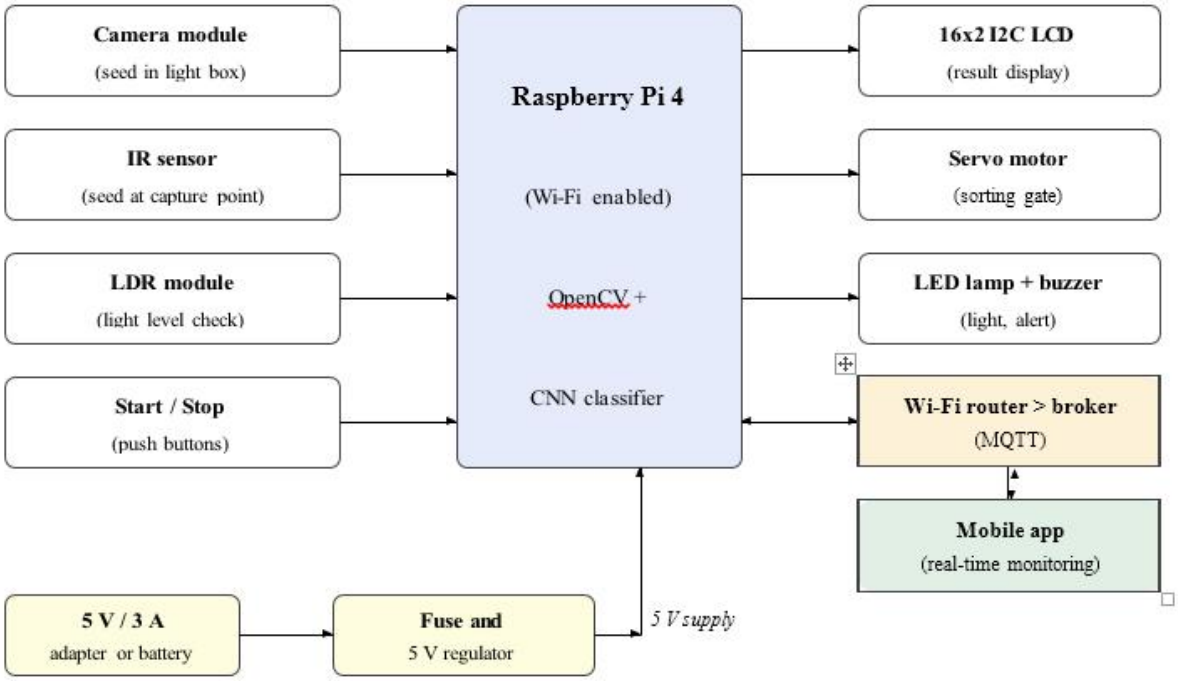


Fig. 2. Block diagram of imaging, classification and sorting flow

Copyright to IJARSCT
www.ijarsct.co.in
DOI: 10.48175/IJARSCT-40374

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 6, October 2026
Impact Factor: 8.2

1) Acquisition: the IR sensor detects a seed at the capture point; the LDR confirms that the light level is in range; the camera captures one image.
2) Preprocessing: the image is converted to grey, smoothed, thresholded with Otsu's method, cleaned by morphological opening, and the seed is cropped and resized to 224×224 pixels.
3) Classification: the CNN returns the probability p that the seed is good.
4) Decision: the seed is sent to the good bin if p is at least 0.70, to the bad bin if p is at most 0.30, and in between it is marked uncertain and sent to the bad bin.
5) Actuation and reporting: the servo gate moves, the LCD and counters are updated, and the MQTT message is published.

VII. WORKING PRINCIPLE

A. Image acquisition and segmentation

Seed images change with lighting, so the capture point sits in a closed box with a diffuse white LED lamp and a matte background whose colour contrasts with the seed. The camera distance is fixed. If the LDR reports a light level outside the calibrated range, the capture is skipped and a warning is shown. After capture, the image is converted to grey, smoothed with a Gaussian filter and thresholded with Otsu's method [7]. A morphological opening removes small specks, the largest contour is taken as the seed, and its bounding box is cropped and resized to 224×224 pixels for the network [6]. These steps use OpenCV [8].

B. Classification model

The primary model is MobileNetV2 [6], pre-trained and fine-tuned: the last layer is replaced by a two-class softmax, and the network is trained with cross-entropy loss and the Adam optimiser [9]. Small rotations, flips and brightness shifts are used to augment the images. The data are split into training, validation and test sets in the ratio 70:15:15, and the split is made by seed lot so that no test seed resembles a training seed too closely. The baseline model describes each seed by colour, shape and texture features [3] and classifies it with an SVM [4]. Both models are scored on the same test set. The network may be converted to TensorFlow Lite to shorten the inference time on the Raspberry Pi.
The data set is planned as in Table I. Good seeds are drawn from one crop and one seed lot at a time. Bad seeds are collected by defect type so that errors can later be analysed type by type.

DATASET PLAN FOR THE PROTOTYPE

Class	Seed condition	Planned images
Good	Whole, plump, uniform colour, no visible damage	≥ 500
Bad: broken	Cracked, chipped or split seed coat	≥ 150
Bad: discoloured	Dark spots, mould or fungal patches	≥ 150
Bad: shrivelled	Wrinkled, undersized or pale (immature) seeds	≥ 100
Bad: insect-damaged	Exit holes or bite marks	≥ 100
Total	Good and bad classes together	≥ 1000

C. Decision rule

Let p be the softmax probability that the seed is good. Thresholds Th = 0.70 and Tl = 0.30 divide the output into three zones: good ($p \geqslant T$), bad ($p \leqslant T$) and uncertain (between the two). An uncertain seed is sent to the bad bin and counted hl separately. The reason is that a bad seed in the good bin can carry disease into the field, whereas a good seed in the bad bin costs little. Both thresholds are design parameters that will be tuned on the validation set, never on the test set.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 6, October 2026
Impact Factor: 8.2

D. Evaluation metrics and latency

The bad seed is taken as the positive class, so a false positive (FP) is a good seed that is rejected and a false negative (FN) is a bad seed that passes into the good bin. With true positives (TP) and true negatives (TN), the metrics are [10]

$$\text{Accuracy} = (TP + TN) / (TP + TN + FP + FN) \quad (1)$$

$$\text{Precision} = TP / (TP + FP) \quad (2)$$

$$\text{Recall} = TP / (TP + FN) \quad (3)$$

$F1 = 2 \cdot \text{Precision} \cdot \text{Recall} / (\text{Precision} + \text{Recall})$ (4) Recall is the most important of these, because it is the share of bad seeds that are caught. The latency budget per seed is 0.10 s for capture, 0.05 s for segmentation, 0.20 s for classification and 0.30 s for the gate movement, a total of 0.65 s;
these are design targets, not measurements.

E. Mobile monitoring

The node publishes a JSON message with the counts of good, bad and uncertain seeds and the running percentage of good seeds, on every sorted seed and as a 30 s heartbeat, with QoS 1 [11]. Messages are retained, so a phone that opens the app immediately sees the last known state, and an MQTT last-will message marks the node offline. The link uses TLS and authenticated access. Seed images are stored only on the Raspberry Pi, for later retraining.

VIII. RESULTS AND DISCUSSION

A. Test protocol

The prototype will be tested with a separate set of at least 200 good and 200 bad seeds that were not used in training. An experienced person labels every seed before the test, and the labels are hidden from the system. Each seed is passed through the machine, and the predicted class is compared with the label. The run is repeated three times and the mean is reported. The counts go into the confusion matrix of Table II, and the metrics of Eqs. (1) to (4) and the timing go into Table III. The entries in the tables are left open because this paper reports no measured values; they will be filled from the prototype log.

TABLE II: CONFUSION MATRIX ON THE TEST SET (NUMBER OF SEEDS)

Actual class	Predicted bad	Predicted good
Bad	TP = —	FN = —
Good	FP = —	TN = —

TABLE III: DESIGN TARGETS AND MEASURED PERFORMANCE

Metric	Design target	Measured
Accuracy	≥ 95 %	—
Precision (bad class)	≥ 95 %	—
Recall (bad class)	≥ 95 %	—
F1-score	≥ 95 %	—
Time per seed	≤ 0.65 s	—
Throughput	≥ 50 seeds/min	—

B. Discussion

Once the tables are filled, the results will be read in four ways. First, the CNN and the SVM baseline will be compared on the same test set, to show whether the learned features justify the extra computation. Second, the errors will be grouped by defect type, because cracks and mould are expected to differ in difficulty. Third, the effect of the confidence band will be shown by the share of seeds sent to the uncertain zone and by the recall with and without it.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 6, October 2026
Impact Factor: 8.2

Fourth, the measured time per seed will be compared with the latency budget of Section VII-D. The targets in Table III are the authors' design goals and may be revised once the first measurements are known.

IX. MODULES

1) Operator (App User):
• Start and stop sorting; view counts of good, bad and uncertain seeds.
• See node online/offline flag and alerts.
2) Acquisition Module:
• Detect the seed, check the light level and capture the image.
3) Processing Module:
• Segment and crop the seed; run the CNN and apply the decision rule.
4) Control Module:
• Drive the servo gate, LCD and buzzer; publish MQTT messages.
5) Administrator:
• Manage broker access and thresholds; retrain the model; view history.

X. ADVANTAGES

1) Non-destructive and repeatable inspection under fixed light.
2) Low-cost, widely available hardware.
3) Doubtful seeds are rejected by an explicit decision rule.
4) Local display and sorting work without the network.
5) Live phone view of the counts and the node status.
6) Evaluation plan with defined metrics makes the design testable.

XI. LIMITATIONS AND FUTURE SCOPE

This paper reports no experimental results. The thresholds, targets and latency figures are design values that must be checked on the prototype. A model trained on one crop and one seed lot may not carry over to another, and will need new images and retraining. The camera sees only the visible surface: internal damage, low vigour and poor germination cannot be detected from one image, so the system complements a germination test and does not replace it. Dust, scratches on the lamp cover and changes in the lamp output can change the images, and a single-seed gate limits the throughput.

Planned evaluation. (i) Classification accuracy, precision, recall and F1 on at least 200 good and 200 bad test seeds, reported with a confusion matrix. (ii) Comparison of the CNN with the SVM baseline on the same test set. (iii) Errors grouped by defect type. (iv) Time per seed, reported as mean, standard deviation and 95th percentile. (v) Recall with and without the confidence band. (vi) Packet loss and reconnection time after a Wi-Fi outage. Future work includes several crops, imaging of both sides of the seed, near-infrared imaging, a conveyor with an air-jet ejector, and prediction of germination.

XII. CONCLUSION

This paper proposed a camera-based system for detecting good and bad seeds in which a lighting box, a MobileNetV2 classifier, a confidence-band decision rule, a servo sorting gate and MQTT phone monitoring are designed together. The pipeline, the dataset plan, the metrics and the test protocol make the design testable, and the decision rule treats a bad seed in the good bin as the costlier error. The results are design targets; the planned measurements on the prototype will show how closely they hold.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 6, October 2026
Impact Factor: 8.2

REFERENCES

[1] A. Kamilaris and F. X. Prenafeta-Boldú, "Deep learning in agriculture: A survey", Computers and Electronics in Agriculture, vol. 147, pp. 70–90, 2018.
[2] Y. LeCun, Y. Bengio and G. Hinton, "Deep learning", Nature, vol. 521, pp. 436–444, 2015.
[3] R. M. Haralick, K. Shanmugam and I. Dinstein, "Textural Features for Image Classification", IEEE Transactions on Systems, Man, and Cybernetics, vol. SMC-3, no. 6, pp. 610–621, 1973.
[4] C. Cortes and V. Vapnik, "Support-vector networks", Machine Learning, vol. 20, no. 3, pp. 273–297, 1995.
[5] A. G. Howard et al., "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications", arXiv:1704.04861, 2017.
[6] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov and L.-C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks", in Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR), 2018, pp. 4510–4520.
[7] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms", IEEE Transactions on Systems, Man, and Cybernetics, vol. 9, no. 1, pp. 62–66, 1979.
[8] G. Bradski, "The OpenCV Library", Dr. Dobb's Journal of Software Tools, 2000.
[9] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization", arXiv:1412.6980, 2014.
[10] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks", Information Processing and Management, vol. 45, no. 4, pp. 427–437, 2009.
[11] A. Banks and R. Gupta (Eds.), "MQTT Version 3.1.1", OASIS Standard, Oct. 2014.