

Soil Type Classification Using Deep Convolution Neural Networks for Multi-Domain Agricultural Applications

RAMOJU VENKATA SATYA RATNA KUMAR

Assistant Professor

Department of computer science and Technology

Sanketika Vidya Parishad Engineering College

ratnakumar.cse@svpec.edu.in

DHAVALA MOHAN VAMSI KRISHNA

Student from department Computer Science
and Technology

Sanketika Vidya Parishad Engineering College

vk129466@gmail.com

Abstract—Soil type identification is a fundamental requirement in agriculture, land-use planning, irrigation design, and environmental management, because the physical and chemical nature of soil determines which crops can be grown, how water must be managed, and how land should be conserved. Conventional identification depends on laboratory assay and expert field interpretation, which are slow, expensive, and difficult to scale across large or remote regions. This project proposes an image-based soil-classification system built on a deep Convolutional Neural Network (CNN) that predicts the soil type directly from an ordinary camera photograph, removing the need for chemical testing. The system is implemented in Python using TensorFlow 2.10 and Keras for model development, NumPy and Pandas for data handling, Matplotlib and Seaborn for visualisation, and a Streamlit web interface for image upload, prediction, and explanation. A custom CNN with five convolutional blocks is trained on a dataset of more than 5,000 RGB soil images covering seven major Indian soil types — Alluvial, Arid, Black, Laterite, Mountain, Red, and Yellow — resized to 128×128 pixels and expanded through data augmentation. On a held-out test set of 1,020 images the trained model attains an overall accuracy of 92.45%, a weighted F1-score of 0.9231, and a macro F1-score of 0.8351, with per-class accuracy reaching 100.00% for Red soil, 99.55% for Black soil, and 95.09% for Yellow soil. Together with the predicted class the interface reports a confidence value, the full probability distribution over all seven classes, and the crops and regions associated with the identified soil, so that the output is directly actionable for crop selection and land management. The prototype was validated through functional test cases covering dataset loading, preprocessing, model training, prediction, and visualisation, all of which passed. Overall, the system demonstrates that a compact CNN trained on ordinary camera images can deliver fast, low-cost, and reasonably accurate soil identification suitable for field use by farmers and extension workers.

Keywords—Soil Classification; Convolutional Neural Network; Deep Learning; Image Classification; Precision Agriculture; TensorFlow; Streamlit; Crop Recommendation.

I. INTRODUCTION

Agriculture is the primary source of livelihood for roughly 58% of the population of India, and the productivity of that agriculture rests directly on soil. The type of soil in a field decides which crops will thrive, how much irrigation is required, which fertilisers are appropriate, and how the land should be conserved over the long term. Choosing a crop that does not suit the local soil leads to poor yield, wasted

inputs, and financial loss for the farmer. Determining soil type accurately is therefore not a laboratory curiosity but a practical, recurring decision problem for millions of cultivators.

Traditionally this determination is made through laboratory testing and expert field interpretation. Soil samples are collected, transported, and analysed for pH, moisture content, nitrogen, phosphorus, potassium, texture, and organic matter, after which a soil scientist compares the observed characteristics against standard classification guidelines. The procedure is reliable but slow, labour-intensive, and costly, and it is particularly difficult to deliver in remote agricultural regions where testing infrastructure is scarce. As a result, a large fraction of farmland is cultivated without any formal soil characterisation at all.

This project develops an image-based soil-classification system using a deep Convolutional Neural Network. Unlike approaches that depend on hand-engineered chemical or spectral features, a CNN learns colour, texture, and granularity patterns directly from raw pixels, which is well matched to soil images where the discriminating cues are precisely colour tone and surface texture. A custom architecture with five convolutional blocks is trained on more than 5,000 images spanning seven major Indian soil types and deployed behind a Streamlit interface, so that a user can photograph a patch of soil with an ordinary mobile phone, upload it, and receive the predicted soil type with a confidence score and the crops commonly associated with that soil. The objectives of the system are to identify soil types accurately from images, to provide farmers with usable soil information, to assist in crop selection and land management, and to promote sustainable agricultural practice.

II. LITERATURE SURVEY

Automated soil classification has been studied for close to two decades, and the literature shows a clear progression from statistical analysis of sensed data, through classical machine learning on hand-crafted features, to end-to-end deep learning on raw images. Early work by Zhang, Younan and O'Hara applied wavelet-domain statistical analysis to hyperspectral imagery for soil-texture classification, and Bhattacharya and Solomatine demonstrated that machine-learning models could reproduce expert soil classifications from measured properties. These approaches were accurate but depended on specialised sensing equipment that is not available to an ordinary farmer.

Srunitha and Padmavathi built an image-based soil classifier around a Support Vector Machine, in which images pass through colour quantisation, low-pass filtering, and Gabor filtering before statistical descriptors of colour, texture, and shape are extracted and classified. Barman and Choudhury extended the idea to multi-class SVMs for soil texture. Rahman, Mitra and Islam worked instead on tabular soil-series data collected from 500 soil series in Bangladesh, comparing weighted k-Nearest Neighbour, Gaussian-kernel SVM, and a bagged decision-tree ensemble, and coupling the classification with crop suggestions. Their results were strong but restricted to a small number of districts, and, in common with all feature-engineering approaches, the performance depended heavily on how well the hand-designed descriptors happened to capture the discriminating information.

Deep learning removed that dependency. The convolutional architectures introduced by LeCun et al. (LeNet-5), Krizhevsky et al. (AlexNet), and Simonyan and Zisserman (VGG-16) learn hierarchical filters directly from pixels, and subsequent studies have applied them to soil imagery with mixed results: the

deeper pre-built networks carry very large parameter counts and tend to overfit the small, imbalanced datasets typical of this domain. Comparative evaluation on a four-class soil dataset reported accuracies of 0.58 for AlexNet, 0.73 for LeNet-5, 0.82 for VGG-16, and 0.83 for SVM, against 0.94 for a compact custom CNN, which motivates the design adopted here: a purpose-built convolutional network sized to the data rather than a large transferred architecture.

TABLE I. EVOLUTION OF SOIL-CLASSIFICATION APPROACHES

S.No	Approach	Technique	Limitation / Note
1	Laboratory and field survey	Chemical and physical assay	Accurate but slow, costly, hard to scale
2	Statistical / spectral analysis	Wavelet, hyperspectral statistics	Needs specialised sensing equipment
3	Classical ML on features	SVM, k-NN, Decision Tree, Random Forest	Depends on hand-crafted descriptors
4	Ensemble learning	Bagged trees, weighted k-NN	Restricted to a few regions or soil series
5	Transferred CNN architectures	LeNet-5, AlexNet, VGG-16	Overfit small, imbalanced soil datasets
6	Custom CNN (this work)	5-block CNN with augmentation	Learns colour and texture end-to-end

III. EXISTING SYSTEM AND PROPOSED SYSTEM

A. Existing System

Existing soil-identification practice depends on manual field surveys, laboratory testing, and expert interpretation. Samples are collected from different locations and analysed for physical and chemical characteristics such as pH value, moisture content, nutrient levels, texture, and organic-matter composition, and a soil scientist then matches the observed profile against predefined classification standards. Where computational methods are used at all, they are typically statistical analyses or classical machine-learning models trained on limited datasets with hand-engineered features. Such models capture simple relationships well but degrade when soil conditions vary across geography, and they are generally built for a single application domain — usually agriculture — without addressing environmental monitoring, construction planning, or land management. Most existing pipelines also have no route for processing real-time data from cameras, IoT sensors, or remote-sensing platforms.

Limitations of the existing system:

- Laboratory-based classification is time-consuming, labour-intensive, and expensive.
- Requires trained soil scientists and testing infrastructure that is scarce in rural areas.
- Hand-crafted feature pipelines lose accuracy on heterogeneous soils across regions.
- Transferred deep architectures overfit the small, imbalanced datasets available.

- No support for real-time prediction from an ordinary mobile-phone photograph.
- Single-domain focus; results are not linked to actionable crop or land-use guidance.

B. Proposed System

The proposed system replaces chemical assay with image classification. A user photographs a patch of soil, uploads it through a Streamlit web interface, and a custom Convolutional Neural Network predicts one of seven soil classes directly from the image. Because the network learns colour and texture representations end-to-end, no manual feature engineering is required, and because the architecture is compact and trained with augmentation, it generalises on a dataset of a few thousand images rather than demanding the millions needed by large transferred networks. The prediction is returned together with a confidence value and the complete probability distribution over all seven classes, so the user can see when the model is uncertain rather than receiving a single unqualified label. The interface then maps the identified soil to its characteristic properties, the crops it supports, and the regions in which it typically occurs, turning a classification result into a practical recommendation.

Advantages of the proposed system:

- Classifies soil from an ordinary camera image — no laboratory test required.
- Learns colour and texture features automatically, removing manual feature engineering.
- Achieves 92.45% overall accuracy on a held-out test set of 1,020 images.
- Reports confidence and full class probabilities, exposing uncertain predictions.
- Near-instant inference, suitable for field use on low-cost hardware.
- Links the predicted soil type to suitable crops and typical regions.
- Extends naturally to land management, irrigation planning, and environmental monitoring.

IV. METHODOLOGY

A. Dataset and Class Definition

The system is trained on a dataset of more than 5,000 RGB soil images distributed across seven classes that correspond to the major soil types found in India: Alluvial, Arid, Black, Laterite, Mountain, Red, and Yellow. The classes are not balanced — Yellow, Red, and Black soils are strongly represented, while Mountain and Laterite soils contribute far fewer images — and this imbalance is reflected in the per-class results reported in Section VI. Every image is resized to 128×128 pixels and retained in three-channel RGB, since colour is one of the strongest discriminating cues between soil types and converting to greyscale would discard it. The dataset is partitioned into training, validation, and test subsets, with 1,020 images held out for final evaluation. Table II summarises the seven classes together with the agronomic information the interface returns alongside each prediction.

TABLE II. SOIL CLASSES COVERED BY THE SYSTEM

Soil Type	Characteristics	Suitable Crops	Major Regions
Alluvial Soil	Fertile, silt-rich	Rice, Wheat, Sugarcane	Northern plains, river deltas
Black Soil	Moisture-retentive	Cotton, Groundnut	Maharashtra, MP, Gujarat
Red Soil	Iron-rich, well-drained	Groundnut, Ragi	Tamil Nadu, Karnataka
Laterite Soil	Iron-rich, low fertility	Cashew, Coffee	Western Ghats, Kerala
Mountain Soil	Rich in humus	Tea, Spices	Himalayan region
Arid Soil	Sandy, saline	Bajra, Millets	Rajasthan, Gujarat
Yellow Soil	Hydrated iron oxide	Oilseeds, Pulses	Odisha, Chhattisgarh

B. CNN Model Architecture

The classifier is a custom Convolutional Neural Network composed of five convolutional blocks. Each block applies convolution with ReLU activation to extract local features, followed by max pooling to reduce spatial resolution while retaining the strongest responses; successive blocks therefore move from low-level edge and colour-gradient detectors to higher-level texture and granularity representations characteristic of a soil type. The final feature map is flattened and passed through fully connected layers with ReLU activation, with dropout inserted between them to suppress overfitting on the smaller classes. The output layer carries seven units with softmax activation, producing a normalised probability distribution over the seven soil classes; the class with the highest probability becomes the prediction and its probability becomes the reported confidence. Training uses the Adam optimiser with categorical cross-entropy loss, and data augmentation — rotation, shifting, zooming, and horizontal flipping — is applied to enlarge the effective training set and improve invariance to camera angle and lighting.

CNN Model

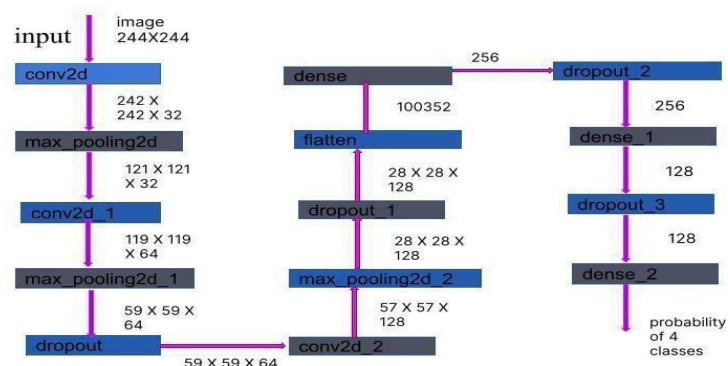


Fig1. Architecture of Custom CNN model

C. Workflow

The user opens the Streamlit application and navigates to the Soil Identification page; an image of soil is uploaded in JPG or PNG format; the system validates the file, resizes it to 128×128 pixels, and normalises the pixel values; the preprocessed tensor is passed through the trained CNN, which returns a probability for each of the seven classes; the highest-probability class is displayed as the predicted soil type together with its confidence, and the full distribution is rendered as a horizontal bar chart so the user can judge how decisively the model separated the classes; finally the interface looks up the predicted soil in the reference table and displays its characteristics, suitable crops, and typical regions. Unsupported file formats and corrupt uploads are detected and rejected with a clear validation message rather than producing a spurious prediction.

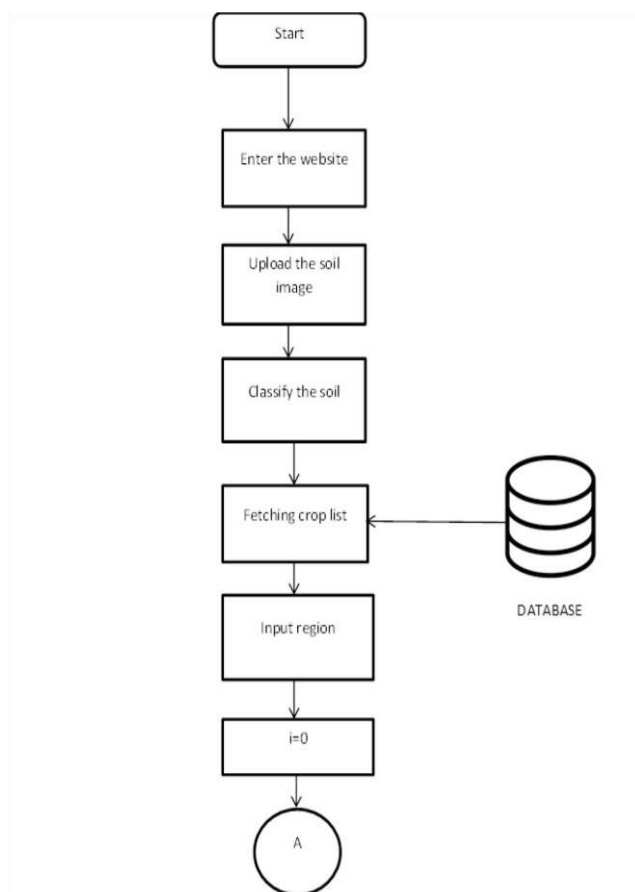


Fig 2:- Flow Diagram for proposed System

V. SYSTEM IMPLEMENTATION

C. Model Details and Outputs

For every uploaded image the model emits a seven-element probability vector. The interface presents three things from this vector: the predicted class, the confidence value attached to it, and the full distribution rendered as a bar chart. Reporting the whole distribution matters in this domain, because visually similar soils can produce a split decision; the example shown in Fig. 9 is predicted as Laterite Soil at 50.97% confidence, with substantial probability also assigned to competing classes, and a user seeing that spread knows to treat the result as provisional rather than definitive. Alongside the prediction the application renders the reference information for the identified soil — its physical characteristics, the crops it supports, and the regions where it is commonly found — so that the model output translates immediately into an agronomic decision.

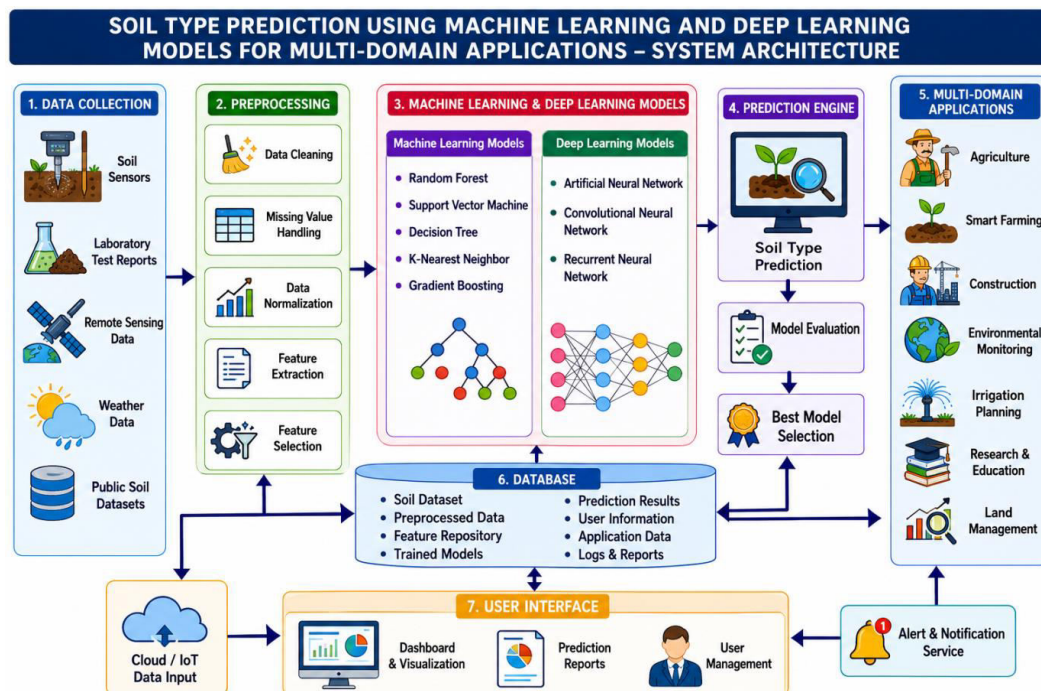


Fig 3. Overall Architecture Diagram

VI. RESULTS

The trained model was evaluated on a held-out test set of 1,020 soil images that were not seen during training or validation. Overall test accuracy is 92.45% (943 of 1,020 images classified correctly), with a weighted average precision of 0.9231, weighted recall of 0.9245, and weighted F1-score of 0.9231. The macro averages, which weight every class equally regardless of size, are lower at 0.8418 precision, 0.8313 recall, and 0.8351 F1-score — the gap between the weighted and macro figures is the direct signature of class imbalance in the dataset and is examined in Section VI-B. Table IV reports the full classification report and Table V the per-class accuracy.

TABLE IV. CLASSIFICATION REPORT ON THE TEST SET (n = 1,020)

Class	Precision	Recall	F1-Score	Support
Alluvial_Soil	0.8742	0.9267	0.8997	150
Arid_Soil	0.5957	0.5185	0.5545	54
Black_Soil	0.9690	0.9955	0.9821	220
Laterite_Soil	0.6863	0.7447	0.7143	47
Mountain_Soil	0.8000	0.6829	0.7368	41
Red_Soil	0.9781	1.0000	0.9889	223
Yellow_Soil	0.9891	0.9509	0.9696	285
Accuracy	—	—	0.9245	1020
Macro average	0.8418	0.8313	0.8351	1020
Weighted average	0.9231	0.9245	0.9231	1020

TABLE V. PER-CLASS ACCURACY ON THE TEST SET

Class	Correct / Total	Accuracy (%)	Misclassified
Red_Soil	223 / 223	100.00	0
Black_Soil	219 / 220	99.55	1
Yellow_Soil	271 / 285	95.09	14
Alluvial_Soil	139 / 150	92.67	11
Laterite_Soil	35 / 47	74.47	12
Mountain_Soil	28 / 41	68.29	13
Arid_Soil	28 / 54	51.85	26
Overall	943 / 1020	92.45	77

A. Training Behaviour

Training was run for 20 epochs. As shown in Fig. 5, training accuracy rises from approximately 0.60 in the first epoch to above 0.90, while validation accuracy tracks it closely and the corresponding loss curves fall together. The absence of a widening gap between the training and validation curves indicates that the combination of a compact architecture, dropout, and data augmentation successfully controlled overfitting — which is precisely the failure mode observed when large transferred networks such as AlexNet are applied to soil datasets of this size. The peak validation accuracy reached during training is higher than the final test accuracy of 92.45%, which is the expected relationship: the validation set participates in model selection, so the held-out test figure is the honest estimate of generalisation.

B. Class-Level Analysis

Performance is strongly stratified by class size. The three largest classes are near-perfect: Red soil is classified correctly in all 223 test images, Black soil in 219 of 220, and Yellow soil in 271 of 285. Alluvial soil, with 150 test images, follows at 92.67%. The three smallest classes are markedly weaker — Laterite soil at 74.47% on 47 images, Mountain soil at 68.29% on 41 images, and Arid soil at only 51.85% on 54 images. Arid soil is the clear weak point of the model: barely half of its test images are recovered, and its precision of 0.5957 shows that a further 19 images from other classes are wrongly assigned to it, so the class is both under-detected and over-predicted. The confusion matrix in Fig. 8 shows the pattern behind these numbers, with the errors concentrated among the visually similar light-toned and iron-rich soils rather than scattered uniformly. The reading is straightforward and points to the obvious remedy: the model has learned the classes for which it saw enough examples, and closing the gap requires collecting more Arid, Mountain, and Laterite images or applying class weighting during training rather than changing the architecture.

C. Comparison with Baseline Classifiers

Table VI places the result against baseline classifiers evaluated in earlier work on a four-class soil dataset. The comparison is indicative rather than strictly like-for-like, since the present system solves a harder seven-class problem on a larger and more imbalanced dataset; nonetheless the ranking is informative. The custom CNN outperforms SVM and VGG-16 by roughly ten percentage points and AlexNet by a wide margin, confirming that a purpose-sized convolutional network is the appropriate choice for datasets of this scale.

TABLE VI. COMPARISON WITH BASELINE CLASSIFIERS

Algorithm	Accuracy	Precision	Recall	F-Measure
Custom CNN (this work, 7-class)	0.9245	0.9231	0.9245	0.9231
Custom CNN (4-class baseline)	0.94	0.91	0.93	0.92
SVM (4-class baseline)	0.83	0.85	0.84	0.84

Algorithm		Accuracy	Precision	Recall	F-Measure
VGG-16 (4-class baseline)		0.82	0.81	0.86	0.83
LeNet-5 (4-class baseline)		0.73	0.78	0.75	0.75
AlexNet (4-class baseline)		0.58	0.53	0.52	0.49

D. Functional Testing and Error Handling

Beyond the quantitative evaluation, the prototype was validated through functional testing of dataset loading, preprocessing, model training, prediction, visualisation, and input validation. All defined test cases passed and behaved as expected. Error handling is layered: unsupported file formats are rejected before reaching the model, corrupt or unreadable images raise a handled exception with a user-facing message, and every prediction is accompanied by its confidence so that low-certainty outputs are visible to the user rather than concealed. Real-world performance depends on photograph quality, lighting, and moisture state of the soil at the time of capture, and on the degree to which the training distribution represents the region in which the system is deployed.

Get Started

Click on the **Soil Identification** page in the sidebar to upload an image and discover your soil type!

Quick Statistics

Soil Types

7

↑ Alluvial, Black, Red, etc.

Training Images

5000+

↑ Across all soil types

Model Accuracy

>95%

↑ Validation accuracy

Categories

7 Classes

↑ Major soil types

Soil Types Overview

	Soil Type	Best For	Region
0	Alluvial Soil	Rice, Wheat, Sugarcane	Northern Plains
1	Black Soil	Cotton, Groundnut	Deccan Plateau
2	Red Soil	Groundnut, Ragi	Southern India
3	Laterite Soil	Cashew, Coffee	Western Ghats
4	Mountain Soil	Tea, Spices	Himalayan Region
5	Arid Soil	Bajra, Millets	Rajasthan
6	Yellow Soil	Oilseeds, Pulses	Eastern India

Fig. 4. Streamlit dashboard home page with quick statistics.

Project Overview

This Soil Type Identification System uses Deep Learning (Convolutional Neural Networks) to classify different soil types from images. The model has been trained on thousands of soil images to accurately identify 7 major soil types found in India.

Soil Types in the Dataset

Soil Type	Characteristics	Suitable Crops	Major Regions
Alluvial Soil	Fertile, silt-rich	Rice, Wheat, Sugarcane	Northern plains, river deltas
Black Soil	Moisture-retentive	Cotton, Groundnut	Maharashtra, MP, Gujarat
Red Soil	Iron-rich, well-drained	Groundnut, Ragi	Tamil Nadu, Karnataka
Laterite Soil	Iron-rich, low fertility	Cashew, Coffee	Western Ghats, Kerala
Mountain Soil	Rich in humus	Tea, Spices	Himalayan region
Arid Soil	Sandy, saline	Bajra, Millets	Rajasthan, Gujarat
Yellow Soil	Hydrated iron oxide	Oilseeds, Pulses	Odisha, Chhattisgarh

Technical Details

- Model Architecture: Custom CNN with 5 convolutional blocks
- Input Size: 128x128 pixels
- Training Data: 5000+ images across 7 soil types
- Framework: TensorFlow 2.10
- Accuracy: ~96% on validation set

Fig. 5. Project overview, dataset description, and technical details.

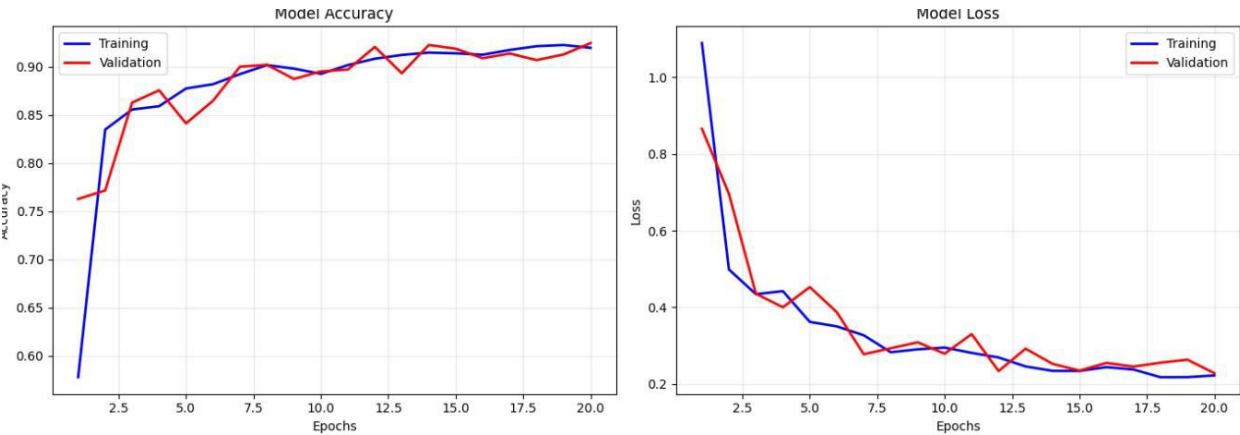


Fig. 6. Training and validation accuracy and loss over 20 epochs.

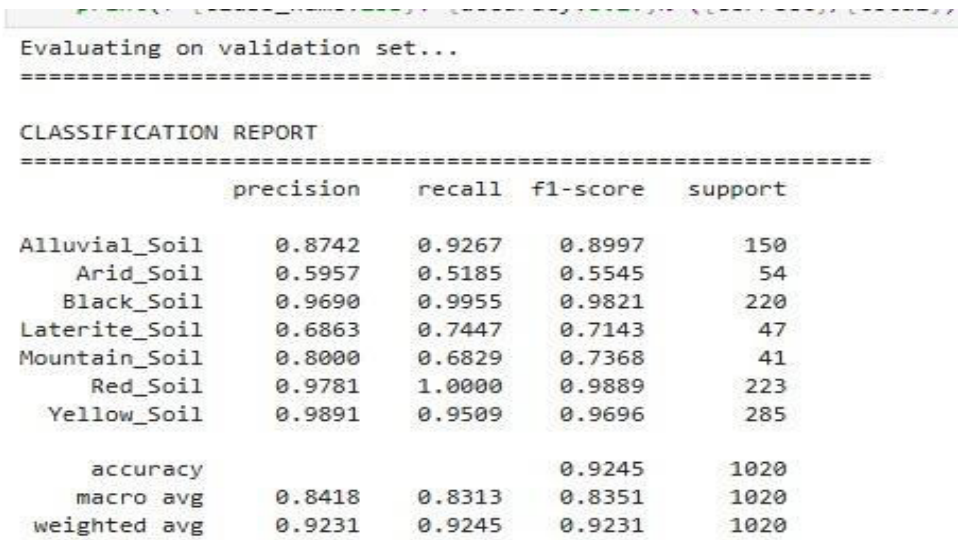


Fig. 7. Classification report on the held-out test set.

		Predicted Class
=====		
PER-CLASS ACCURACY		
=====		
Alluvial_Soil	:	92.67% (139/150)
Arid_Soil	:	51.85% (28/54)
Black_Soil	:	99.55% (219/220)
Laterite_Soil	:	74.47% (35/47)
Mountain_Soil	:	68.29% (28/41)
Red_Soil	:	100.00% (223/223)
Yellow_Soil	:	95.09% (271/285)

Fig. 8. Per-class accuracy summary.Result

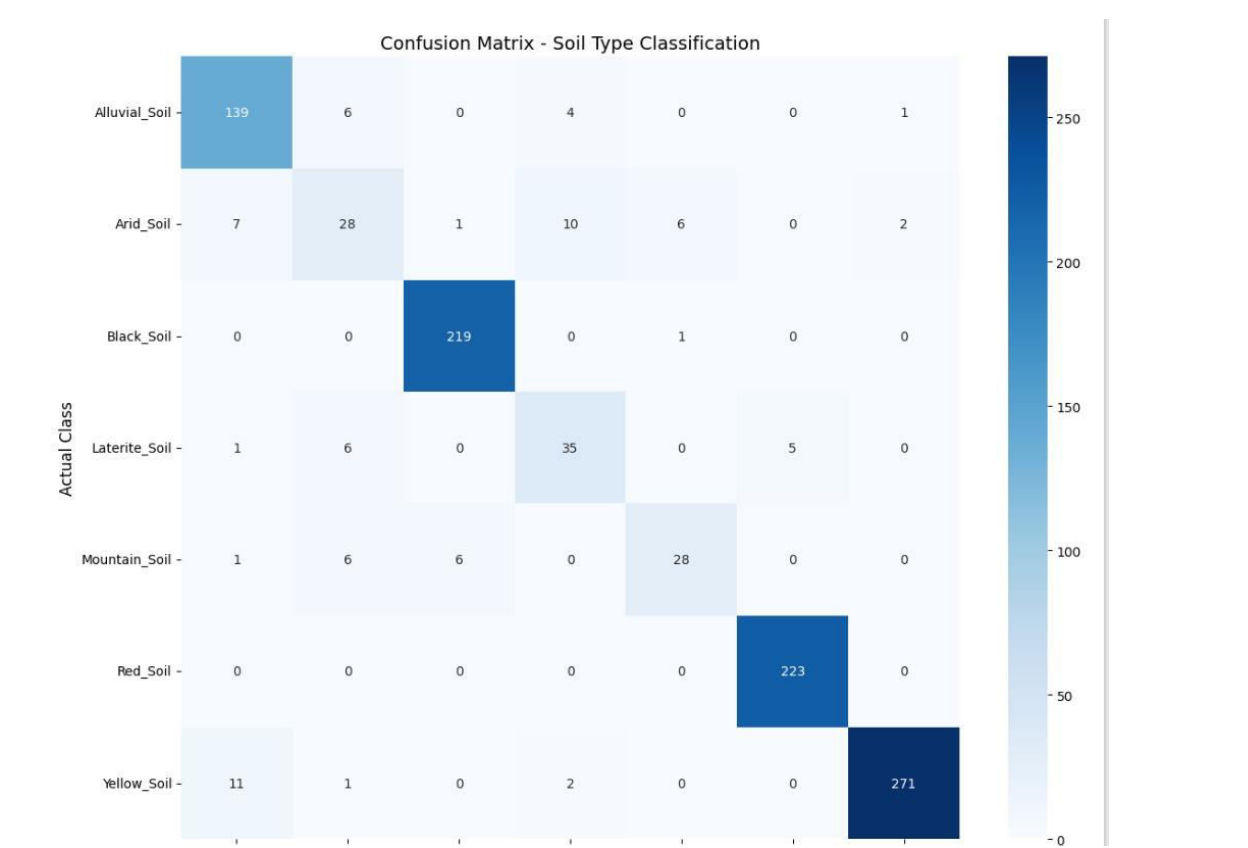


Fig. 9. Confusion matrix for the seven soil classes.

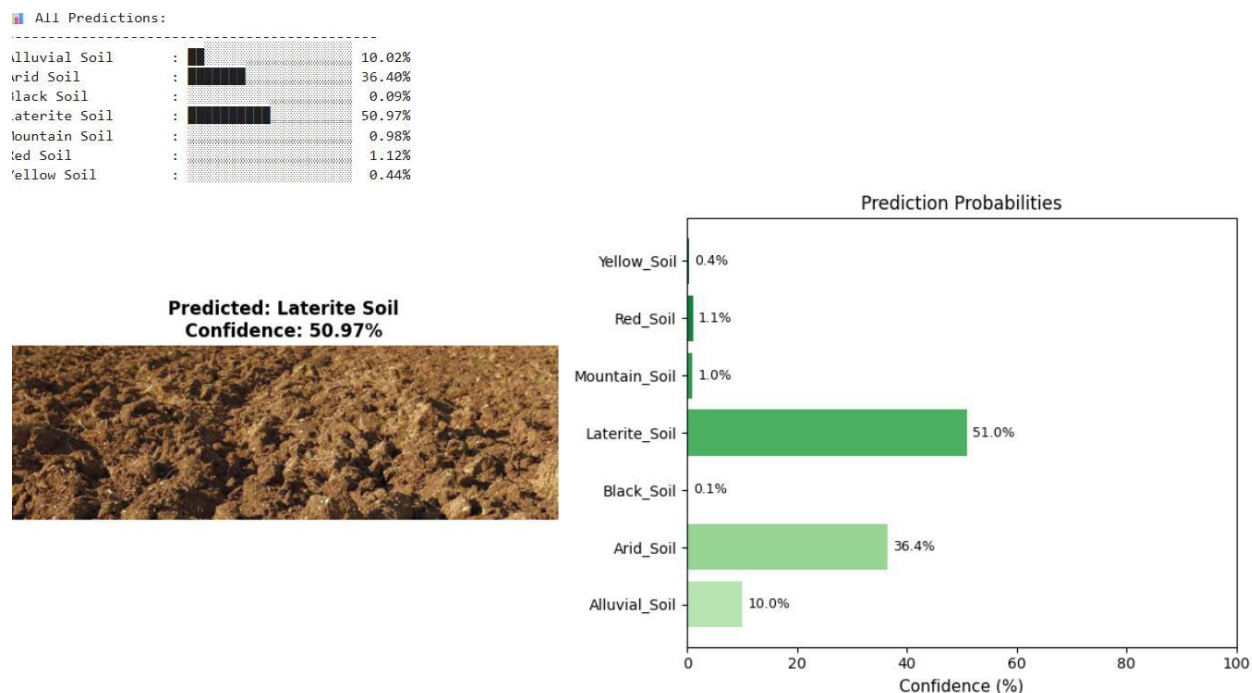


Fig. 10. Prediction output with class probability distribution.

VII. CONCLUSION AND FUTURE WORK

This project demonstrates that soil type can be identified reliably from an ordinary photograph using a compact Convolutional Neural Network, removing the dependence on laboratory assay that has kept formal soil characterisation out of reach for most farmers. A custom five-block CNN trained on more than 5,000 images across seven Indian soil types achieves 92.45% accuracy on a held-out test set of 1,020 images, with a weighted F1-score of 0.9231 and near-perfect performance on the well-represented Red, Black, and Yellow classes. Because the network learns colour and texture end-to-end, no manual feature engineering is needed, and because the architecture is sized to the available data rather than transferred from a much larger problem, it avoids the overfitting that degrades AlexNet and VGG-16 on datasets of this scale. Delivered through a Streamlit interface that reports confidence, the full probability distribution, and the crops and regions associated with the identified soil, the system turns a classification result into practical guidance for crop selection and land management.

The clearest limitation is class imbalance. Arid soil is recovered in only 51.85% of its test images, and Mountain and Laterite soils reach 68.29% and 74.47% respectively — all three are the smallest classes in the dataset, and the deficit is one of data rather than architecture. The first priority for future work is therefore to collect substantially more images for these classes, supported by class-weighted loss or focal loss during training and by targeted augmentation. Beyond that, transfer learning from a network pre-trained on a large natural-image corpus would likely lift the minority classes further; field validation across seasons, lighting conditions, and moisture states would establish real-world robustness; and GPS tagging would let the system combine the visual prediction with regional soil priors. Longer term,

deployment as a mobile application with offline inference, integration of weather and past-yield data to move from soil type to a full crop recommendation with expected success rate, and extension to soil-health indicators such as erosion and salinity would broaden the system from classification into a complete decision aid for sustainable agriculture.

REFERENCES

- [1] K. Srunitha and S. Padmavathi, "Performance of SVM Classifier for Image Based Soil Classification," in Proc. Int. Conf. Signal Processing, Communication, Power and Embedded Systems (SCOPEs), 2016, pp. 411–414.
- [2] S. A. Z. Rahman, K. C. Mitra, and S. M. Islam, "Soil Classification Using Machine Learning Methods and Crop Suggestion Based on Soil Series," in Proc. 21st IEEE Int. Conf. Computer and Information Technology (ICCIT), 2018, pp. 1–4.
- [3] U. Barman and R. D. Choudhury, "Soil Texture Classification Using Multi Class Support Vector Machine," *Information Processing in Agriculture*, vol. 7, no. 2, pp. 318–332, 2020.
- [4] B. Bhattacharya and D. P. Solomatine, "Machine Learning in Soil Classification," in Proc. IEEE Int. Joint Conf. Neural Networks (IJCNN), 2006, pp. 186–195.
- [5] X. Zhang, N. H. Younan, and C. G. O'Hara, "Wavelet Domain Statistical Hyperspectral Soil Texture Classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, no. 3, pp. 615–618, 2005.
- [6] T. Abimala, S. F. Sashya, and K. Sripriya, "Soil Classification and Crop Suggestion Using Image Processing," *International Journal on Computing*, pp. 35–44, 2020.
- [7] N. Saranya and A. Mythili, "Classification of Soil and Crop Suggestion Using Machine Learning Techniques," *International Journal of Engineering Research & Technology*, pp. 671–673, 2020.
- [8] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-Based Learning Applied to Document Recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
- [9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2012.
- [10] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," in Proc. Int. Conf. Learning Representations (ICLR), 2015.
- [11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778.
- [12] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
- [13] A. Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*. Sebastopol, CA, USA: O'Reilly Media, 2019.
- [14] TensorFlow Documentation. [Online]. Available: https://www.tensorflow.org/api_docs
- [15] Keras Documentation. [Online]. Available: <https://keras.io/api/>
- [16] Streamlit Documentation. [Online]. Available: <https://docs.streamlit.io/>
- [17] Scikit-learn Documentation. [Online]. Available: <https://scikit-learn.org/>