

Automatic Resistor Value Detection Using Computer Vision and Machine Learning

Clark Kneil L. Caoile

College of Engineering and Information Technology, University of Southern Mindanao, Kabacan, Cotabato, 9407, Philippines

Abstract - This study presents a computer vision-based system for automatic resistor value detection using machine learning. A lightweight convolutional neural network (CNN) was trained using approximately 1,200 labeled images of 4-band and 5-band resistors to classify resistor values based on color-band patterns. The system includes a real-time webcam interface that guides the user in positioning the resistor within a defined region of interest before prediction. Experimental results showed a training accuracy of 92.3% and a validation accuracy of 90.1% under controlled conditions. However, real-time performance decreased to approximately 72% under poor lighting and misalignment conditions. These results indicate that whole-image CNN classification is feasible for resistor value detection but remains sensitive to illumination changes and object positioning. To address these limitations, a YOLO–CNN hybrid architecture is proposed as an enhancement, where YOLO-based detection localizes individual resistor bands before CNN-based color classification. This proposed approach is expected to improve robustness, interpretability, and error diagnosis by analyzing individual color bands rather than the whole resistor image. The system has practical applications in electronics education, component verification, servicing, and retail environments. Future work should focus on implementing the YOLO–CNN architecture, expanding the dataset, improving lighting control, and deploying the model on mobile or embedded platforms.

Index Terms — CNN, YOLO, computer vision, resistor color code, machine learning

Manuscript received 9 January 2026; revised 6 April 2026; accepted 26 June 2026. Date of publication: 29 June 2026.

Citation: C. K. L. Caoile, "Automatic Resistor Value Detection Using Computer Vision and Machine Learning," *Journal of Engineering and Technology for Sustainability and Resilience (JETSUR)*, vol. 1, no. 1, pp. 10–16, 2026. <https://doi.org/10.5281/zenodo.20699559>

I. INTRODUCTION

RESISTORS are among the most common passive components used in electronic circuits. They are used for current limiting, voltage division, biasing, signal conditioning, and circuit protection. Through-hole resistors commonly use colored bands printed on the resistor body to represent their resistance value and tolerance. These color markings follow standardized coding systems, such as those defined under IEC 60062 for marking codes of resistors and capacitors [1]. Although the color-code method is widely used, manual decoding can still be difficult for students, technicians, and general users, especially when the resistor is

small, old, discolored, poorly illuminated, or misaligned.

Manual resistor reading depends heavily on accurate visual interpretation of color bands. This becomes challenging when color bands are visually similar, such as brown and red or blue and green. Environmental factors such as lighting variation, shadow, reflection, camera exposure, and background color can distort the perceived color of the bands. Previous resistor-recognition studies using image processing have reported that illumination, orientation, scale, and region-of-interest selection are major causes of error in automatic resistor value detection [2]–[6]. These challenges justify the need for computer vision-based tools that can assist users in reading resistor values more consistently.

Earlier studies on resistor value detection commonly used traditional image processing techniques, including color segmentation, thresholding, region-of-interest extraction, clustering, and classification [2]–[6]. For example, the work in [2] developed a real-time mobile approach for resistor color-code recognition using video processing and HSV-based color recognition. The studies in [3] and [4] proposed resistor-reading methods based on image processing, color-band extraction, and controlled lighting design. Another

study in [5] presented a segmentation and classification approach for resistors in digital images using feature extraction and support vector machines. These studies demonstrate the feasibility of automated resistor recognition but also show that classical image-processing methods remain sensitive to lighting variation, segmentation quality, reflection, and camera positioning.

In recent years, deep learning has improved many visual-recognition tasks because convolutional neural networks can automatically learn spatial features from image data [9]–[12]. CNN-based methods have also been applied to electronic-component classification, showing that deep learning can classify components such as resistors, capacitors, diodes, and other parts with promising accuracy [7], [8]. However, advanced pretrained CNN architectures such as VGG, ResNet, and MobileNet were originally designed for large-scale image classification and may introduce additional computational complexity [11]–[14]. Transfer learning can be useful when the target dataset is limited, but its effectiveness depends on the similarity between the source-domain images and the target-domain images [15]. For resistor color-band recognition, the target features are small, color-based, and highly localized; therefore, the use of a custom lightweight CNN was considered appropriate as a baseline model.

This study develops a lightweight CNN-based resistor value detection system integrated with a real-time webcam interface. The main objective is to establish a simple, interpretable, and real-time-capable baseline system for automatic resistor value classification. Instead of using deeper pretrained CNNs immediately, the study first evaluates whether a custom CNN can learn resistor color-band patterns from a manually labeled dataset. The study also identifies practical limitations such as poor lighting, color similarity, and resistor misalignment. Based on these observed limitations, a YOLO–CNN hybrid architecture is proposed as a future enhancement, where individual resistor bands are detected first before color classification and resistor-value decoding.

II. METHODOLOGY

A. Dataset Preparation

A dataset consisting of approximately 1,200 images of 4-band and 5-band resistors was collected for this study. The images included variations in resistor orientation, lighting conditions, background, and camera positioning to simulate practical usage conditions. Each image was manually labeled based on the standard resistor color-code convention defined in IEC 60062 [1]. Manual labeling was used to ensure that the assigned class corresponded to the actual resistance value decoded from the resistor color bands.

To improve labeling reliability, multiple samples of the same resistor value were cross-checked as shown in Fig. 1. Images that were ambiguous, blurred, poorly captured, or difficult to verify were excluded from the dataset. The final

dataset was organized into class-labeled directories, where each folder represented a corresponding resistor value. This folder-based structure allowed the training pipeline to associate each image with its correct class label.

Before training, all images were resized to 64×64 pixels and normalized to values between 0 and 1. Data augmentation techniques were also applied to improve model generalization, including rotation, horizontal flipping, and brightness adjustment. Data augmentation was used because image-based deep learning models can become sensitive to limited dataset variation, and augmentation helps expose the model to additional variations during training [16].



Fig. 1. Sample dataset images used for training, showing variations in lighting, orientation, and background.

B. CNN Architecture

The CNN model used in this study, as shown in Table I, consisted of two convolutional layers with ReLU activation, each followed by a 2×2 max-pooling layer. The extracted feature maps were flattened and passed to a dense layer with 128 neurons. A softmax output layer was used to classify the input image into its corresponding resistor value class.

A lightweight CNN architecture was selected to match the real-time nature of the system and to reduce computational demand during webcam-based prediction. Unlike deeper pretrained architectures, the custom CNN allowed faster inference, simpler implementation, and easier interpretation of errors during baseline testing. Since the images focused on structured resistor color-band patterns rather than broad object categories, the lightweight CNN was sufficient for establishing an initial performance benchmark before testing more complex architectures in future work.

The model was trained using categorical cross-entropy loss and the Adam optimizer [17]. An 80:20 training-validation split was used, with a batch size of 32. Early stopping was applied to prevent overfitting and to stop

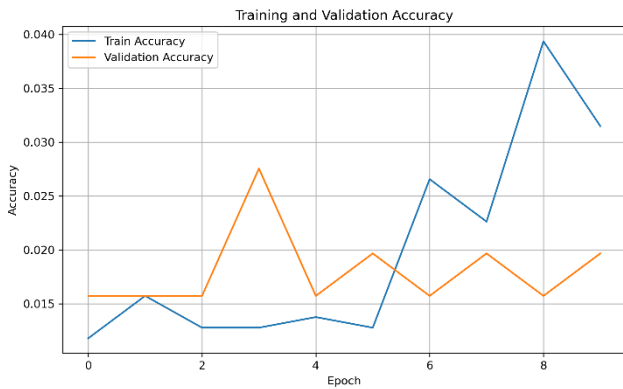


Fig. 4. Training and validation accuracy over epochs.

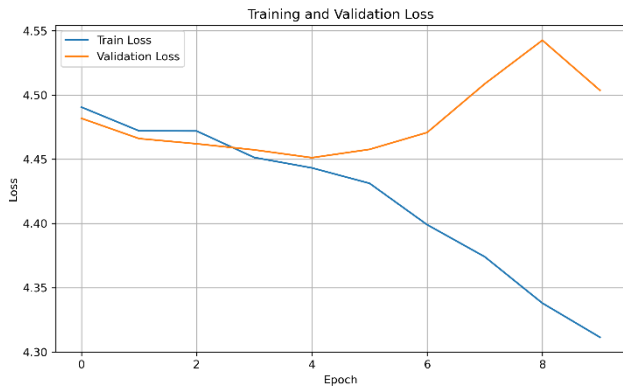


Fig. 5. Training and validation loss over epochs.

Fig. 4 shows that both training and validation accuracy increased during the early epochs and became stable as training progressed. This trend indicates that the CNN was able to learn relevant visual features from the resistor images, such as color distribution, band arrangement, and overall resistor appearance. The close behavior between the training and validation curves suggests that the model did not suffer from severe overfitting during the final training run.

Fig. 5 further supports this observation because both training and validation loss decreased as the number of epochs increased. A decreasing loss curve indicates that the model's prediction error was reduced during training. The stabilization of validation loss also supports the use of early stopping, since additional training after convergence may not provide meaningful improvement and may increase the risk of overfitting. This behavior is consistent with common deep learning training practice, where validation accuracy and validation loss are used to monitor model generalization and determine whether training should be stopped.

However, the difference between the controlled validation accuracy and the lower real-time accuracy under poor lighting shows that good validation performance does not always guarantee strong performance in uncontrolled environments. This result implies that the model learned the dataset conditions well, but it remained sensitive to real-time factors such as illumination changes, shadows, reflections, and resistor misalignment. Similar issues have been reported in image-based resistor recognition and computer vision studies, where lighting quality and object positioning

strongly affect color segmentation and classification performance [2]–[5].

B. Error Analysis

Misclassifications were primarily observed between resistor bands with similar visual characteristics, such as brown versus red and green versus blue. These errors were mainly caused by illumination variation, reflections, shadows, and limited color differentiation in the RGB image representation (Fig. 6). This observation is consistent with earlier resistor color-recognition studies, where color segmentation and classification were affected by lighting and specular reflection [2]–[6].

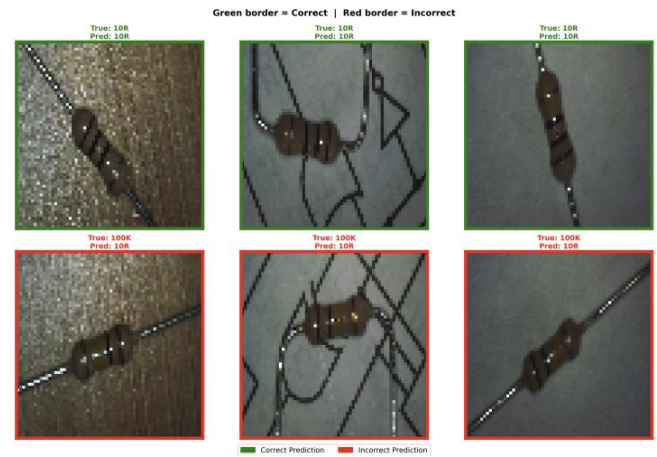


Fig. 6. Sample predictions showing correct and incorrect classifications.

In the present system, these errors indicate that the CNN relied heavily on global visual patterns rather than explicitly identifying and analyzing each resistor band. Since the whole resistor image was treated as one input, the model learned the combined visual appearance of the resistor instead of separately verifying each band color. This can lead to an error when the resistor is rotated, shifted, or captured under poor illumination.

To improve interpretability, the classification outcomes were grouped into broader resistance ranges, as shown in Fig. 7.

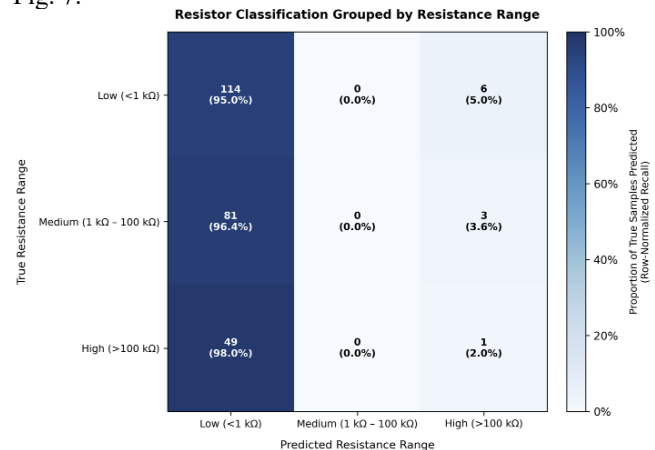


Fig. 7. Grouped confusion matrix for resistor classification across resistance ranges.

The grouped confusion matrix summarizes the classification behavior across low, medium, and high resistance categories. Grouping was used because the original class-level confusion matrix contained many resistance values, making it visually dense and difficult for general readers to interpret. The grouped matrix provides a clearer view of whether the model confuses neighboring resistance ranges.

C. Limitation

The system is sensitive to lighting variation and object misalignment. Since resistor value recognition depends strongly on color interpretation, changes in illumination can distort the RGB values of the resistor bands. Under poor lighting conditions, darker bands may appear similar, resulting in incorrect classification.

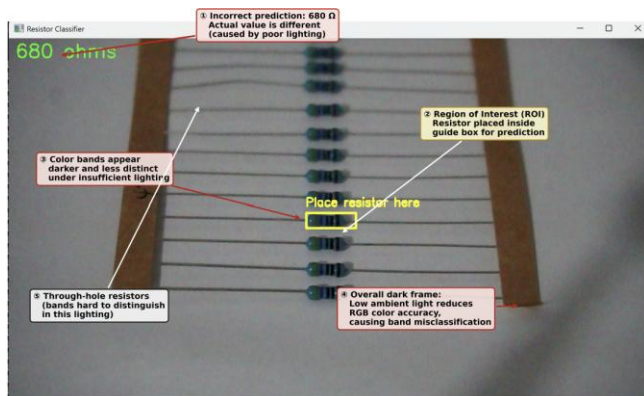


Fig. 8. Example of incorrect prediction under poor lighting conditions.

Fig. 8 illustrates a sample condition where insufficient lighting reduced image brightness and contrast. As a result, the resistor color bands became harder to distinguish, causing the CNN model to generate an incorrect resistor value. This limitation suggests that controlled lighting, automatic exposure adjustment, or color calibration should be considered in future versions.

In addition to lighting, the system is also affected by misalignment and rotation of the resistor within the camera frame. Since the CNN processes the resistor as a whole image, improper positioning changes the spatial relationship between the color bands. This can cause the model to extract features that differ from the patterns learned during training.

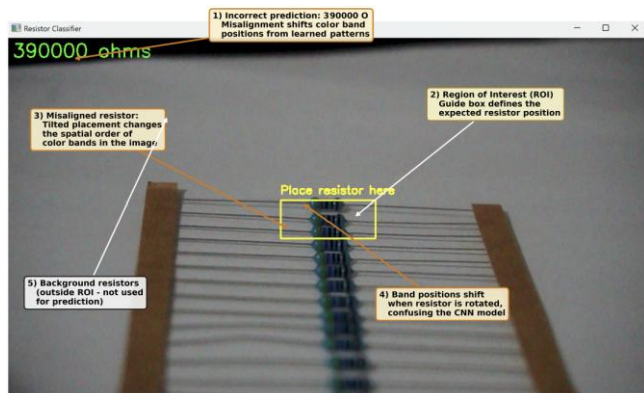


Fig. 9. Example of misclassification due to resistor misalignment.

Fig. 9 shows how tilting or shifting the resistor away from the expected guide-box position can lead to misclassification. This limitation confirms that the current CNN baseline depends on consistent resistor placement during real-time detection. These issues motivate the proposed YOLO-CNN hybrid enhancement.

D. Proposed Enhancement: YOLO-CNN Hybrid Architecture

A YOLO-CNN hybrid offers a way around these limits, sketched in Fig. 10. YOLO-style detectors are a natural fit for real-time localization because they pick out target regions efficiently within a frame, as the original YOLO model [21] and its successors [22], [23] established. The idea is to train YOLO to find each resistor color band, hand the cropped bands to a CNN for color classification, then order the detected colors by their physical position and decode the value with the standard color-code rules.

It was not built for this study, though, because it needs its own object-detection dataset with bounding-box annotations on every band, which in turn means more labeling, detector training, and validation. Since the hybrid was never implemented, no detection or classification figures are reported for it here; that evaluation is left to future work, as noted in the conclusion. The YOLO-CNN architecture therefore stands only as a proposed improvement, not as part of the present results.



Fig. 10. Proposed YOLO-CNN hybrid architecture for future localized resistor band detection and color classification

E. Applications

The system has clear uses across education, electronics servicing, and component handling. In a teaching lab it can confirm resistor values for students mid-activity, heading off the usual slips in manual color-code reading and helping beginners get comfortable identifying parts.

On the repair bench it gives technicians a fast read during troubleshooting, prototyping, and sorting, and in a retail setting it helps sales staff confirm values for customers. Being camera-based, it also has a natural path onto mobile devices or embedded platforms in later versions.

IV. CONCLUSION

This study set out to test whether computer vision and machine learning could automate resistor value detection, and the results suggest that they can. Trained on 1,270 labeled images spanning 89 resistor-value classes, the lightweight CNN reached 92.3% training accuracy and 90.1% validation accuracy under controlled conditions, and it ran inside a real-time webcam interface that returned a predicted value as soon as a resistor was placed in the guide box.

Performance was less consistent once those controlled conditions were removed. Real-time accuracy fell when

lighting was poor or the resistor sat off-center, and most errors traced back to color similarity, shadow, reflection, reduced contrast, and inconsistent placement within the frame. The baseline is therefore dependable for guided or controlled reading but not yet ready for unsupervised use in the field, and that gap is what shapes the work worth pursuing next.

The most direct improvement is to build out the YOLO-based band localization that this paper only proposed. Annotating bounding boxes for individual color bands and training a dedicated detector would let the system read each band on its own terms instead of inferring a value from the resistor as a whole. Illumination-robust color spaces such as HSV or CIELAB are also worth testing in place of raw RGB, and a fixed light source, whether an LED ring or a small enclosed chamber, would cut the shadows and reflections that currently distort color reading.

Stability can also be addressed at the interface level. Averaging a prediction over several consecutive frames before showing a result, together with a confidence threshold that asks the user to reposition or rescan when the model is unsure, would both lower the chance of displaying a wrong value. The training pipeline should likewise log the details needed to reproduce a run, including training time, class count, software versions, random seed, and model size, none of which were fully captured this time.

Comparative evaluation is the other open thread. Benchmarking the baseline against classical image-processing methods, other lightweight CNNs, and transfer-learning models built on VGG, ResNet, or MobileNet would show how much the custom network gives up or gains, with transfer learning becoming more attractive once a larger and more varied dataset is available. Reporting class-wise precision, recall, F1-score, and a full confusion matrix would further pinpoint which resistor values and color-band combinations the model tends to confuse.

Finally, the system should be tried on different cameras, lighting setups, and computing platforms to gauge how far it generalizes beyond the original bench, and a mobile build using TensorFlow Lite or a comparable embedded framework would bring it within reach of the classrooms, repair benches, and field settings where it is most likely to be useful.

REFERENCES

- [1] International Electrotechnical Commission, IEC 60062: Marking Codes for Resistors and Capacitors, IEC, Geneva, Switzerland, 2016.
- [2] M. F. Demir, A. Cankirli, B. Karabatak, A. Yavariabdi, E. Mendi, and H. Kusetogullari, "Real-time resistor color code recognition using image processing in mobile devices," in Proc. 9th IEEE International Conference on Intelligent Systems, 2018, pp. 26–30, doi: 10.1109/IS.2018.8710533.
- [3] Y.-S. Chen and J.-Y. Wang, "Reading resistor based on image processing," in Proc. International Conference on Machine Learning and Cybernetics, 2015, pp. 566–571, doi: 10.1109/ICMLC.2015.7340616.
- [4] Y.-S. Chen and J.-Y. Wang, "Computer vision on color-band resistor and its cost-effective diffuse light source design," Journal of Electronic Imaging, vol. 25, no. 6, p. 061409, 2016, doi: 10.1117/1.JEI.25.6.061409.
- [5] M. Muminovic and E. Sokic, "Automatic segmentation and classification of resistors in digital images," in Proc. XXVII International Conference on Information, Communication and Automation Technologies, 2019, pp. 1–6, doi: 10.1109/ICAT47117.2019.8939034.
- [6] Y. Mitani and Y. Hamamoto, "A method for reading a resistor by image processing techniques," in Knowledge-Based Intelligent Information and Engineering Systems, Lecture Notes in Computer Science, vol. 5177, 2008, pp. 433–439, doi: 10.1007/978-3-540-85563-7_56.
- [7] I. Atik, "Classification of electronic components based on convolutional neural network architecture," Energies, vol. 15, no. 7, p. 2347, 2022, doi: 10.3390/en15072347.
- [8] P. Chand and S. Lal, "Vision-based detection and classification of used electronic parts," Sensors, vol. 22, no. 23, p. 9079, 2022, doi: 10.3390/s22239079.
- [9] 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.
- [10] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in Advances in Neural Information Processing Systems, vol. 25, 2012, pp. 1106–1114.
- [11] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in Proc. International Conference on Learning Representations, 2015.
- [12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proc. IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.
- [13] A. G. Howard et al., "MobileNets: Efficient convolutional neural networks for mobile vision applications," arXiv preprint arXiv:1704.04861, 2017.
- [14] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in Proc. IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 4510–4520.
- [15] S. J. Pan and Q. Yang, "A survey on transfer learning," IEEE Transactions on Knowledge and Data Engineering, vol. 22, no. 10, pp. 1345–1359, 2010, doi: 10.1109/TKDE.2009.191.
- [16] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," Journal of Big Data, vol. 6, no. 60, 2019, doi: 10.1186/s40537-019-0197-0.
- [17] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in Proc. International Conference on Learning Representations, 2015.
- [18] G. Bradski, "The OpenCV library," Dr. Dobb's Journal of Software Tools, vol. 25, no. 11, pp. 120–125, 2000.
- [19] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. Cambridge, MA, USA: MIT Press, 2016.
- [20] L. Prechelt, "Early stopping—but when?" in Neural Networks: Tricks of the Trade, Lecture Notes in Computer Science, vol. 1524, Berlin, Germany: Springer, 1998, pp. 55–69.

- [21] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in Proc. IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 779–788, doi: 10.1109/CVPR.2016.91.
- [22] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," arXiv preprint arXiv:1804.02767, 2018.
- [23] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," arXiv preprint arXiv:2004.10934, 2020.
- [24] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in Advances in Neural Information Processing Systems, 2015, pp. 91–99.