Personalized Hairstyle Recommendation System using Face Shape Classification

Rayaan Mohammad Salem[1], Rishabh Singh[1], Kavya Bansal[1], and Narendra VG[1]

[1]Dept of Computer Science and Engineering , Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, Karnataka, 576104, India

ABSTRACT

This project proposes a hairstyle recommendation system utilizing face shape classification, supported by computer vision and machine learning. The system categorizes faces into five primary shapes—heart, oblong, oval, round, and square—by employing image processing techniques such as hair detection, skin masking, Gaussian blur, and Canny edge detection to accurately map the hairline. Using 68 detected facial landmarks on input images, we compute 19 geometric features that effectively capture distinct facial structures.

Our preprocessing pipeline includes histogram equalization, Principal Component Analysis (PCA), and hyperparameter optimization. We evaluate several classifiers, including Support Vector Machine (SVM) with linear, polynomial, and RBF kernels, along with a Random Forest classifier, to identify the most effective model for accurate face shape classification. Once classified, the system overlays selected hairstyles—sourced from online recommendations—directly onto the input image, adjusted based on the detected hairline and facial width to provide a realistic preview. Additionally, K-Nearest Neighbors (KNN) is used to identify the top five similar faces in the dataset, offering style inspiration through matching celebrity faces. Built with Streamlit, the user interface delivers an interactive and accessible platform for showcasing personalized hairstyle recommendations and celebrity comparisons.

Keywords: Hairstyle Recommendation, Face Shape Classification, Image Processing, Facial Landmarks, SVM, Random Forest, KNN, Streamlit.

1. INTRODUCTION

In today's fast-paced digital world, personal style is key to expressing individuality and enhancing self-image, with hairstyling playing a prominent role in grooming. Choosing the right hairstyle, however, can be a challenge for many, often relying on subjective opinions or trial and error, which may result in dissatisfaction. The absence of personalized guidance, particularly in relation to face shape, complicates this process further.

As the demand for personalized beauty solutions rises, technology offers a promising approach to this challenge. This project introduces a Personalized Hairstyle Recommendation System that uses computer vision and machine learning to suggest hairstyles based on a person's face shape. The system classifies faces into five categories—heart, oblong, oval, round, and square—by analyzing facial landmarks and geometric features. By doing so, it offers a tailored, data-driven method for choosing hairstyles that complement an individual's facial structure.

The system not only simplifies the decision-making process but also aligns with the growing trend of virtual try-ons and personalized experiences in the digital age. With an intuitive Streamlit interface, users can seamlessly explore hairstyle recommendations that best suit their face shape, gaining style inspiration by matching their features to celebrity faces with similar characteristics.

This project aims to empower individuals to feel more confident in their appearance, while demonstrating the potential of artificial intelligence in the beauty and fashion industry. Further details on the methodology and implementation of the system are discussed in the following sections.

2. RELATED WORK

This work builds upon previous studies in the field of face shape classification for personalized hairstyle recommendation systems, particularly drawing inspiration from a framework that utilizes Support Vector Machine (SVM) classifiers for face shape classification based on extracted features. In the referenced study, the authors explored a combination of hand-crafted and deep-learned features for improved classification performance, with techniques such as Multiple Kernel Learning (MKL) and hyperparameter optimization. [1], [2], [3]

Our approach follows a similar high-level strategy but differs in several key aspects. Unlike the referenced work, we focus exclusively on hand-crafted features derived from facial landmarks, avoiding deep-learned features like VGG-face. We conducted a comparative study on various classification algorithms, including SVM with different kernels (linear, polynomial, and RBF), as well as a Random Forest classifier. This broader selection of models allows us to assess the strengths and weaknesses of different classifiers in the context of face shape classification, and to determine which performs best on our dataset.

[4], [5], [6]

To further enhance the performance of our models, we incorporated hyperparameter optimization. While the cited paper used Particle Swarm Optimization (PSO), we employed grid search to fine-tune the hyperparameters of our classifiers, which allowed us to systematically explore different parameter settings and identify the most effective combinations for each model.

Another novel contribution of our work is the use of K-Nearest Neighbors (KNN) to identify and display the top similar faces from a database of celebrity images. This feature not only aids in face shape classification but also provides users with personalized style recommendations by matching them to celebrity faces with similar facial structures. This addition offers an interactive and engaging aspect to the system that was not present in the original framework.

In addition to the classification models and KNN, we also introduced preprocessing techniques such as histogram equalization and adaptive histogram equalization, which helps improve image contrast before facial landmark detection. This preprocessing step enhances the clarity of facial features and helps achieve more accurate landmark detection, ultimately improving the performance of the face shape classification task. [7]

In summary, while our methodology is inspired by the previous work in face shape classification, we have expanded upon it by exploring different classifiers, adding KNN for celebrity face matching, and incorporating preprocessing steps to optimize the classification process. These innovations aim to improve the overall accuracy and user experience of our personalized hairstyle recommendation system.

3. DATASET

The dataset consists of five categories, each representing a major face shape—oval, round, square, oblong, and heart—with 100 celebrity images per category, totaling 500 labeled images. Expert-classified for accuracy, it primarily features female faces due to the lack of a comprehensive male dataset. Consequently, the recommendation system is optimized for female faces and hairstyles. The open-source dataset, available on GitHub, is well-organized for face shape classification experiments. Source: GitHub - dsmlr/faceshape.[1]

4. METHODOLOGY

In our project, various image processing and machine learning techniques were employed to enhance facial landmark detection, extract meaningful facial features, and perform classification and recommendation tasks. The methodology consists of sequential steps designed to improve image quality, accurately identify facial landmarks (including the hairline as a 69th landmark), and extract geometric features that define face shape. These features are then used for classification into predefined categories and for finding similar images. The following sections outline the key steps undertaken in this process.

4.1 Preprocessing

This step in our project is supposed to preprocess images to improve their quality for better landmark detection and feature extraction in an image. Here are the techniques used:

- Grayscale Conversion**:** The image uploaded is first converted to grayscale by OpenCV. This approach simplifies computation by only working on one channel (intensity) instead of three (RGB). It is sufficient for facial landmark detection.
- Histogram Equalization**:** Histogram equalization enhances contrast in the grayscale image, improving landmark detection under varying lighting. Adaptive histogram equalization (CLAHE) was tested but showed little improvement, so standard histogram equalization was used.

4.2 Landmark Detection using CV Techniques

The hairline detection method uses several techniques to enhance the accuracy and robustness of the facial landmark system:

Facial Landmark Detection

- **Detector:** Utilizes a pre-trained dlib shape predictor to detect 68 facial landmarks, which represent key facial points such as the eyes, nose, mouth, and jawline.[8]

- **Method:** For each detected face, the model returns a set of landmarks (coordinates) to mark specific facial features.

Hairline Detection and Integration as the 69th Landmark`

- **Detection of forehead region:** The forehead region is identified by the midpoint between the eyebrows (landmarks 19 and 24). The x-coordinate of the midpoint is the average of the x-coordinates of landmarks 19 and 24, and the y-coordinate is 20 pixels above landmark 19. This region helps in localizing the forehead and detecting the hairline in the subsequent steps.

- **Skin Masking:** The image is converted from BGR to HSV to isolate skin color, which aids in identifying regions of interest such as the forehead. The mean skin color of the forehead region is calculated to dynamically determine the skin color range. A skin mask is then created using this dynamically determined skin color range. To improve the mask's quality, Gaussian blur is applied to smooth edges and reduce noise, ensuring better accuracy in the hairline detection process. [9]

- **Canny Edge Detection:** Canny edge detection is applied to the blurred skin mask to find the skin-to-hair boundary. [10]

- **Scanning Upward:** The algorithm scans upward from the forehead region to detect where the skin transitions to hair, marking this as the hairline. If no edge is found, a default position 50 pixels above the forehead is used. [11]

- **Adding the Hairline as the 69th landmark:** After detecting the hairline's y-coordinate, it is added as the 69th landmark i, with the x-coordinate matching the forehead's x-coordinate.

4.3 Feature Extraction

Once the landmarks are identified, we compute a set of geometric features that represent the shape of the face. These features include distances and ratios between key landmarks, as well as angles formed between the chin and other facial points. Out of the 69 landmark points located, only the first 17 points, which outline the face, are used for feature extraction. Additionally, two more points are used: point 58, which corresponds to the bottom of the mouth, and point 69, which corresponds to the hairline. These features are then used to create a feature vector, which serves as a descriptor of the user's face shape. The following features are considered:

- **Face Height to Width Ratio:** The ratio of the height of the face to the width is calculated as

$$F_1 = \frac{p_9 - p_{69}}{p_1 - p_{17}}$$

where p_8 and p_{68} represent the vertical points, and p_1 and p_{17} are the horizontal points of the face.

- **Jaw Width to Face Width Ratio:** The ratio of the distance between both sides of the jaws to the width of the face is calculated as

$$F_2 = \frac{p_5 - p_{13}}{p_1 - p_{17}}$$

where p_5 and p_{13} represent the points on both sides of the jaw, and p_1 and p_{17} are the horizontal points of the face.

- **Chin to Mouth Bottom to Jaw Width Ratio:** The ratio of the distance between the chin and the bottom of the mouth to the distance between both sides of the jaws is calculated as

$$F_3 = \frac{p_9 - p_{58}}{p_5 - p_{13}}$$

where p_9 and p_{58} represent the vertical points (chin and bottom of the mouth), and p_5 and p_{13} are the points on both sides of the jaw.

- **Angles Between Point 9 and Points 1-8:** The angles between point 9 (the chin) and points 1-8 (which represent the points along the left jawline and facial contour) give the features F4 - F11.

- **Angles Between Point 9 and Points 10-17:** The angles between point 9 (the chin) and points 10-17 (which represent the points along the right jawline and facial contour) give the features F12 - F19.

These 19 features will now be made used to make the feature descriptor which will be used for classification into the face shapes.

4.4 Classification

The extracted features were fed into five machine learning classifiers for defect classification:

- **Support Vector Machine (SVM)** with linear kernel
- **Support Vector Machine (SVM)** with polynomial kernel
- **Support Vector Machine (SVM)** with RBF kernel
- **Random Forest (RF)**
- **K-Nearest Neighbors (KNN)**

Before feeding the features into the models, we standardized them using a Standard Scaler, ensuring that each feature contributes equally to the classification process. Following scaling, Principal Component Analysis (PCA) was applied to reduce the feature dimensions, improving computational efficiency while retaining essential information. [12]

To determine the optimal number of components for PCA, we calculated the cumulative explained variance by iterating over all components. This process involved plotting the cumulative explained variance ratio to visualize the point at which additional components contribute minimally to the explained variance, thus guiding our selection of the reduced feature count.

Finally, to optimize each classifier's performance, we performed hyperparameter tuning using Grid Search Cross-Validation. This process identified the most effective configurations for each model, enhancing classification accuracy and ensuring the models could generalize effectively to unseen data.

4.5 Haircut Overlay

The haircut overlay function overlays a hairstyle image onto the original image based on facial landmarks and predefined parameters. This process involves the following key steps:

Head Width Calculation

The head width is calculated using the distance between the two outermost facial landmarks (landmarks 1 and 17), providing an approximation of the head's width. This width is essential for scaling the hairstyle image to match the person's face.

Resizing the Hairstyle Image

Using the head width and the known width of the hairstyle (from the CSV), a scale factor is calculated. This factor ensures that the hairstyle image will match the width of the person's head. The hairstyle image is then resized according to this scale factor.

Overlaying the Hairstyle

The hairstyle is overlaid onto the person's face by calculating the appropriate position based on the hairline landmark (69th landmark) and predefined parameters. The overlay process ensures proper alignment and scaling to fit the person's face, providing a seamless appearance. To achieve the most natural overlay, various interpolation methods were tested, including nearest-neighbor interpolation, bilinear interpolation, and bicubic interpolation. After experimentation, Lanczos interpolation was chosen for its visual quality compared to the others. [11]

4.6 Finding Similar Images Using KNN

The system employs the K-Nearest Neighbors (KNN) algorithm to recommend similar faces based on extracted feature vectors. For each image in the dataset, a feature vector is generated to capture key facial characteristics. When a new query image is provided, its feature vector is compared against the dataset using the Euclidean distance metric to measure similarity. The KNN algorithm then ranks the dataset images based on their proximity to the query image, retrieving the top 'n' nearest neighbors (default: 5). These nearest images are returned as the most similar matches, enabling efficient and accurate face similarity retrieval.

5. EXPERIMENTS AND RESULTS

The image in Fig. 1 represents the 69 points detected in the facial landmark detection stage of the project. For the feature extraction part of the project, we will only be using points 1-17, 58, and 69. These selected landmarks will be used to extract specific facial features that are crucial for the subsequent stages of classification.

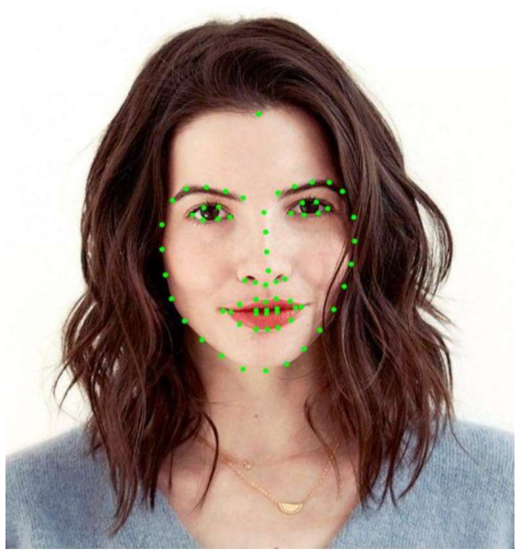

Fig. 1. Facial Landmark Detection (69 Facial Landmarks)

To determine the optimal number of principal components for dimensionality reduction, we applied Principal Component Analysis (PCA) on the feature dataset. The cumulative explained variance plot in Fig. 2

illustrates how much of the total variance is explained by each successive principal component. Based on this analysis, we determined that reducing the dimensions to 12 components captured approximately 99% of the total variance, striking a balance between simplifying the model and retaining significant information in the data.

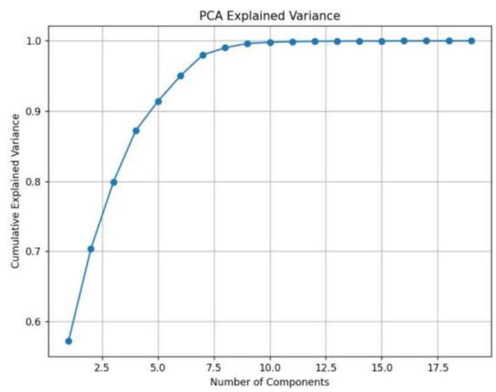

Fig. 2. Cumulative Explained Variance of Principal Components (PCA)

Feature scaling was applied to the dataset to ensure uniformity across different classifiers. The crossvalidation accuracy for each model, providing an average measure of model consistency across multiple data folds, is presented in Table 1. Subsequently, Table 2 shows the classification accuracy on the test set after an 80-20 train-test split, reflecting each model's performance on unseen data.

Table 1. Cross-Validation Accuracy Comparison for Different Models

Model	Cross-Validation Accuracy (%)
SVM with linear kernel	60.1
SVM with RBF kernel	61.4
SVM with polynomial kernel	54.4
Random Forest	60.4
K-Nearest Neighbors (KNN)	46.1

Table 2. Test Set Accuracy Comparison for Different Models

Model	Test Set Accuracy (%)
SVM with linear kernel	61.9
SVM with RBF kernel	62.9
SVM with polynomial kernel	51.5
Random Forest	61.9
K-Nearest Neighbors (KNN)	45.4

The detailed classification reports for each model are shown in Table 3.

Table 3. Classification Report Comparison

Class	SVM Linear			SVM RBF			SVM Poly			Random Forest			KNN		
	Prec	Rec	F1	Prec	Rec	F1	Prec	Rec	F1	Prec	Rec	F1	Prec	Rec	F1
Heart	0.62	0.50	0.56	0.55	0.60	0.57	0.45	0.50	0.48	0.58	0.55	0.56	0.33	0.35	0.34
Oblong	0.65	0.75	0.70	0.65	0.65	0.65	0.73	0.55	0.63	0.70	0.80	0.74	0.52	0.70	0.60
Oval	0.38	0.42	0.40	0.45	0.47	0.46	0.27	0.37	0.31	0.43	0.32	0.36	0.21	0.21	0.21
Round	0.61	0.61	0.61	0.81	0.72	0.76	0.64	0.50	0.56	0.69	0.61	0.65	0.67	0.44	0.53
Square	0.84	0.80	0.82	0.74	0.70	0.72	0.65	0.65	0.65	0.64	0.80	0.71	0.61	0.55	0.58
Average	0.62	0.62	0.62	0.64	0.63	0.63	0.55	0.51	0.53	0.61	0.62	0.61	0.47	0.45	0.45
Accuracy	0.62			0.63			0.52			0.62			0.45		

The classification output for the example image is displayed in the Streamlit front end, where the system predicts the most suitable hairstyle based on the extracted facial features. After processing the input image, the model classifies the user's head shape and recommends a hairstyle accordingly. The following Fig. 3, showcase the classification result for the example image, along with the suggested hairstyle based on the model's prediction.

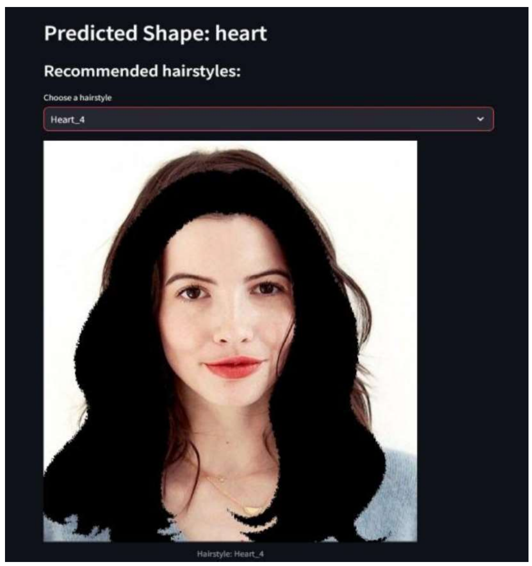

Fig. 3. Classification Output (Recommended Hairstyle 1)

Additionally, the top similar faces from the dataset based on the K-Nearest Neighbors (KNN) algorithm are showcased in the following Fig. 4. This image presents the Streamlit output displaying the most similar faces, which can be used as references for hairstyle recommendations, providing further insights into the facial features that align with the predicted head shape.

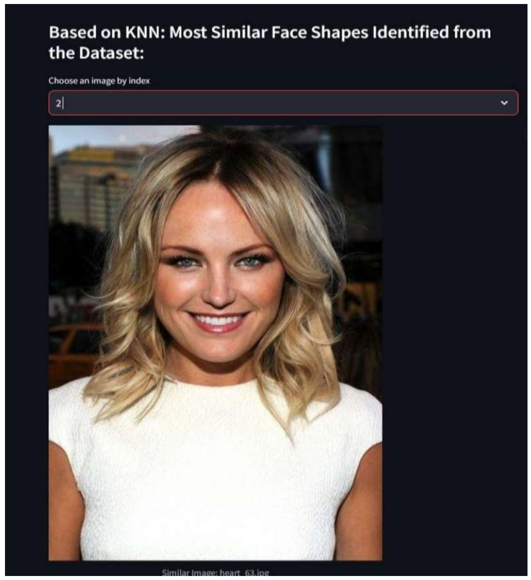

Fig. 4. Top Similar Faces (KNN Output)

6. Discussion

The facial landmark detection approach, using 69 key points, enabled the extraction of crucial features for classification. We selected relevant landmarks (points 1-17, 58, and 69) to capture essential facial structures for accurate head shape classification, aiding hairstyle prediction.

To reduce dimensionality, we applied PCA, retaining 12 components that preserved 99% of the variance, balancing efficiency and performance. Among the tested models, SVM with an RBF kernel achieved the highest test accuracy (62.9%), making it the preferred choice for prediction.

7. Limitations and Future Work

The proposed system effectively recommends and overlays hairstyles but has limitations. It assumes a perfectly aligned frontal face, making misalignments affect accuracy. The model is trained only on female faces, limiting applicability to males; expanding the dataset would improve inclusivity. Additionally, it overlays black hairstyles regardless of the user's hair color, reducing realism. The current study employs only machine learning models for classification, leaving room for improvement in accuracy. Future work should explore ensemble methods and deep learning models, address image orientation adjustments, incorporate diverse datasets, and develop hair color-matching techniques to enhance accuracy and personalization.

8. Conclusion

The Hairstyle Recommendation System developed in this project demonstrates a comprehensive approach to personalized hairstyle suggestions based on face shape classification. The system leverages advanced computer vision techniques, including facial landmark detection, skin color segmentation, and edge detection, to accurately analyze the user's facial features and detect a hairline.

The user interface, built with Streamlit, provides a seamless experience for users to upload their images, view face landmarks, and explore hairstyle recommendations based on their face shape. The combination of facial feature analysis, machine learning, and personalized recommendations offers a novel and practical solution for individuals seeking hairstyle suggestions.

In conclusion, the Hairstyle Recommendation System is an innovative application of computer vision and machine learning that combines facial feature analysis and personalization, ensuring that users receive relevant and suitable hairstyle options based on their unique facial characteristics. Future work can explore the incorporation of more diverse datasets and further refinements in hairline detection for even more accurate and customized recommendations.

REFERENCES

1. K. Pasupa, W. Sunhem, and C. K. Loo, ”A hybrid approach to building face shape classifier for hairstyle recommender system,” Expert Systems with Applications, vol. 120, pp. 439–449, Nov. 2018, doi: 10.1016/j.eswa.2018.11.011.
2. Kulkarni, Raj. (2024). Recommendation System for Hairstyle Based on Face Recognition Using AI and Machine Learning. International Journal of Software Innovation. 12. 1-10. 10.4018/IJSI.309960.
3. H. Weerasinghe and D. Vidanagama, ”Machine Learning Approach for Hairstyle Recommendation,” 2020 5th International Conference on Information Technology Research (ICITR), Moratuwa, Sri Lanka, 2020, pp. 1-4, doi: 10.1109/ICITR51448.2020.9310868.
4. B. Sch¨olkopf and A. J. Smola, Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond, MIT Press, 2002.
5. Breiman, L. Random Forests. Machine Learning 45, 5–32 (2001). https://doi.org/10.1023/A:1010933404324
6. K. Taunk, S. De, S. Verma and A. Swetapadma, ”A Brief Review of Nearest Neighbor Algorithm for Learning and Classification,” 2019 International Conference on Intelligent Computing and Control Systems (ICCS), Madurai, India, 2019, pp. 1255-1260, doi: 10.1109/ICCS45141.2019.9065747.
7. R. Szeliski, Computer Vision: Algorithms and Applications, 2nd ed., Springer, 2022.
8. I. Silva, ”Facial landmarks recognition,” GitHub repository, [Online]. Available: https:
9. //github.com/italojs/facial-landmarks-recognition/blob/master/shape_predictor_ 68_face_landmarks.dat. Accessed: Nov. 9, 2024.
10. D. Chai and K. N. Ngan, ”Face segmentation using skin-color map in videophone applications,” in IEEE Transactions on Circuits and Systems for Video Technology, vol. 9, no. 4, pp. 551-564, June 1999, doi: 10.1109/76.767122.
11. J. Canny, ”A Computational Approach to Edge Detection,” in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-8, no. 6, pp. 679-698, Nov. 1986, doi: 10.1109/TPAMI.1986.4767851.
12. P. Aarabi, ”Automatic Segmentation of Hair in Images,” 2015 IEEE International Symposium on Multimedia (ISM), Miami, FL, USA, 2015, pp. 69-72, doi: 10.1109/ISM.2015.16.
13. J. Shlens, ”A tutorial on principal component analysis,” arXiv, Apr. 2014. [Online]. Available:
14. https://doi.org/10.48550/arXiv.1404.1100. [Accessed: Nov. 9, 2024].

BIOGRAPHIES

Rayaan Mohammad Salem is a third year Computer Science and Engineering student at Manipal Institute of Technology, specializing in AI and Machine Learning. His interests include Machine Learning, Deep Learning, and Computer Vision, with applications in healthcare, AR, security, and personalized recommendations. He is also passionate about NLP and Generative AI, exploring automated text generation, retrieval-augmented generation (RAG), and conversational AI. Rayaan aims to apply advanced AI techniques to real-world challenges in both vision and language-based systems.

Rishabh Singh is a third-year Computer Science and Engineering (AI ML) student at MIT Manipal. With a strong foundation in machine learning, computer vision, and deep learning, he has worked on multiple innovative projects in robotics and circuit design. His expertise includes developing algorithms for robotics competitions and applying AI-driven technologies in agriculture. Passionate about AgriTech, he integrates computer vision and deep learning models to enhance agricultural practices, aiming to bring technological advancements to the field.

Narendra V G received the B.E. degree in CSE from Karnataka University Dharwad, in 1997, the M.Tech. degree from Visvesvaraya Technological University (VTU), Belgaum, in 2001, and the Ph.D. degree from Manipal University, Manipal, India, in 2017. He is currently a Professor with the Department of Computer Science and Engineering, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal. His research interests include image processing, ML, soft computing, and computer vision-based applications for agriculture. He is on the editorial board of some journals. He can be contacted at

email: narendra.vg@manipal.edu