Facial Recognition Project Report (Individual)
Shreya Chinthala
Student Id: 07700005606

▪ Introduction:

Facial recognition technology has made significant strides in recent years, resulting in its extensive utilization across various domains, including surveillance and social media. The purpose of this project was to evaluate distinct methodologies for constructing robust facial recognition systems to explore and assess diverse approaches to guarantee the development of effective solutions for facial recognition.

Our team extensively explored different face recognition methods, aiming to create a system that not only works efficiently but also ensures high accuracy. The project involved closely examining existing methods, carefully preparing the data, using different libraries, trying out various models, and collaborating with others. Adopting this diverse strategy was crucial to make sure we fully understood and optimized each part, playing a key role in building a strong and effective face recognition system.

▪ Key Learnings:

- The KNN model, especially when incorporating both the Cosine Function and Euclidean Distance, proved to be exceptionally effective, achieving perfect accuracy in face recognition. The KNN model demonstrated outstanding performance, specifically when utilizing both the Cosine Function and Euclidean Distance. This exceptional accuracy underscores its efficacy in accurately recognizing faces.

- Random Forest consistently demonstrated high accuracy on both the validation and testing sets, emphasizing its effectiveness in face recognition. Random Forest consistently delivered impressive accuracy levels on both the validation and testing sets, highlighting its reliability and success in accurately recognizing faces.

- While the Bayesian Classifier exhibited solid performance, there is potential for improvement through additional optimization measures to enhance its accuracy. Despite the Bayesian Classifier demonstrating satisfactory performance, there exists an opportunity for enhancement through further optimization efforts aimed at improving its accuracy.

- The SVM model displayed lower accuracy compared to other models, indicating the need for deeper exploration and refinement to better capture underlying data patterns. The SVM model exhibited lower accuracy levels when compared to alternative models, signifying the necessity for a more thorough exploration and refinement process to capture the underlying patterns in the data more effectively.

- Given the exceptional accuracy achieved by the KNN model, particularly when incorporating the Cosine Function and Euclidean Distance, it emerges as the primary choice for face recognition applications. Considering the outstanding accuracy demonstrated by the KNN model, especially when utilizing both the Cosine Function and Euclidean Distance, it stands out as the top choice for face recognition applications.

- Random Forest also presents a practical option, striking a balance between high accuracy and computational efficiency. Random Forest emerges as a practical choice, providing a balance between impressive accuracy and computational efficiency, making it a viable option for face recognition tasks.

- To enhance the accuracy of the Bayesian Classifier and SVM on both the validation and testing sets, further experimentation and parameter tuning are recommended. To improve the accuracy of both the Bayesian Classifier and SVM on both the validation and testing sets, it is advisable to conduct additional experiments and fine-tune the model parameters for better performance.

Task Contributions

Data Manipulation *- Vasavya - 40% ,Shreya- 40%, Aparnaa - 20%*

- Image Resizing
  - The initial step in the face recognition process is to resize the images, ensuring they all conform to a standardized dimension of 224 x 244 pixels.

- Normalizing the Images:
  - Following resizing, the images undergo normalization to maintain uniformity in their representation, facilitating consistent processing within the model.

- Grayscale Conversion

- The subsequent transformation involves converting the images from RGB to grayscale. This simplifies the data and reduces computational overhead, streamlining the face recognition process.

- Cleaning Up
  - A crucial phase is the cleanup process, where images with unclear faces, notably those highly pixelated or featuring individuals wearing hats, are removed. This ensures the dataset maintains clarity and relevance.

- Converting Labels to Uniform Identifiers
  - To enhance consistency, labels are standardized by removing whitespaces and converting them to lowercase. This step is vital, as failing to do so has been correlated with lower accuracy in the model's performance.

- Handling Different Image Extensions
  - The final step involves managing diverse image extensions such as Jpeg, png, and Heic. The team ensures compatibility by converting Heic-format images to Jpeg, ensuring a seamless integration of varied image types within the face recognition system.

Face Detection *- Vasavya - 20% ,Shreya - 40% ,Aparnaa - 40%*

In the face recognition project, the initial step involves face detection, which provides an identifier in the form of a rectangular box bounding the detected face.

- OpenCV
  - The project leverages the OpenCV library for various computer vision tasks, including face detection. OpenCV's functionality is crucial in identifying and extracting facial features within the images.
- Face Recognition
  - The Face Recognition library is utilized as part of the project to enhance the face detection process. This library integrates advanced features that contribute to more accurate face identification, complementing the capabilities of OpenCV.
- Dlib
  - The inclusion of the Dlib library further strengthens the face detection aspect of the project. Dlib excels in robust face detection, providing precise rectangular bounding boxes around detected faces. Additionally, it brings facial landmark detection capabilities, adding an extra layer of detail to the facial analysis.

We have decided to use the Face Recognition library as gave as better results than and OpenCV and Dlib.

Feature Extraction *- Vasavya-40%,Shreya-20%,Aparnaa - 40%*

- In our face recognition project, we used the Region of Interest (ROI),to find the facial features, we use key points that we identify on a person's face in an image. The next step involves looking closely at the face, extracting detailed features, and analyzing various facial characteristics.We utilize OpenCV for this process. OpenCV helps us identify key points on the face and generates standard-size descriptor vectors for each of these points. However, initially, these key points were a bit arbitrary, meaning they were not very useful for what comes next in our processing.

- So, we have focused on Face Landmarks, which are specific points on a person's face. When we use the face landmarks () function, it gives us a coordinate vector of standard size, typically (68x2), where 68 represents the number of landmarks, and each landmark has two coordinates. The order of these features is crucial for our analysis. Our goal is to encode this (68x2) vector into a unique one-dimensional vector. This encoding process is essential because it simplifies the data and makes it easier for our system to learn and recognize faces during training. By transforming the coordinate vector into a one-dimensional form, we enhance the efficiency of our face recognition model.

- In our face recognition project, we employed a CNN-based Encoder-Decoder. This involved training a Convolutional Neural Network (CNN) to encode and decode information from our training set images. Subsequently, multiple models like KNN (K-Nearest Neighbors) and Random Forest were trained using the obtained encodings. The accuracy achieved through this approach ranged between 70-75%. Additionally, we utilized the face recognition library, which comes with a pre-trained model providing a unique (128x1) encoding for each face landmarks vector.

- In simpler terms, we trained a specialized neural network to understand and represent the information in our images. Then, we used these representations to train other models like KNN and Random Forest. Additionally, the face recognition library's pre-trained model transformed the face landmarks into a unique and condensed (128x1) format, making it more efficient than other extractions we worked on for our face recognition project.

Models, Comparison between Models, Data cleaning, refactoring, and Testing - Vasavya-33.33%,Shreya-33.33%,Aparnaa-33.33%

Models and results

- During the initial phase, we undertook an extensive examination of various face recognition techniques. Our exploration delved into Eigenface detection, neural networks, and deep learning methodologies, with the primary objective of comprehending the strengths and limitations inherent in each approach. This involved a detailed analysis to gain a nuanced understanding of the capabilities and constraints associated with Eigenface detection, neural networks, and deep learning methodologies in the context of face recognition.

- Our team tried out different models to see how well they work for recognizing faces. We used models like KNN, Bayesian classifier, Random Forest, and SVM. We carefully adjusted and tested each model to make sure it fit our project needs.

- During this process, we fine-tuned the settings of each model to make them work better and tested how good they are at recognizing faces. This helped us figure out what each model is good at and where it might struggle when it comes to recognizing faces. The goal was not just to find models that work for our project but also to understand which ones are better in certain aspects and where they can be improved.

- We carefully looked at different models to see how well they worked, and we found that their accuracies varied. To choose the best model, we did a detailed analysis considering our face recognition needs.
- In this process, we paid close attention to how each model performed based on how well the accuracy was. This careful examination helped us decide which model was the best fit for the specific requirements and details of our face recognition project.

Models Evaluations are as follows

- **KNN model**

The KNN model, with the inclusion of both the Cosine Function and Euclidean Distance, accurately identified faces in both the validation and testing sets. This indicates the model's effectiveness and efficiency in precisely recognizing faces, establishing its reliability across various situations.

Cosine Function:

Measures vector closeness.

Cosine similarity = (A. B) / |A|. |B|

Distance = 1 - cosine similarity

Results:

Validation Set Accuracy: 100%

Testing Set Accuracy: 100%

- **Bayesian Classifier**

The Bayesian Classifier exhibited strong performance, particularly on the testing set, achieving an accuracy of 93% but not as good as KNN.

Results:

Validation Set Accuracy: 80%

Testing Set Accuracy: 93%

- **Random Forest**

The Random Forest model did exceptionally well, showing high accuracy in both the practice and test sets. Because it performed better than the Bayesian Classifier but not as good as KNN, it suggests that it's a good choice for recognizing faces.

Results:

Validation Set Accuracy: 97.5%

Testing Set Accuracy: 99.21%

- **SVM**

The SVM model showed lower accuracy than other models, suggesting it might struggle to capture the patterns in the data. To make it work better, we may need to investigate and adjust its settings.

Results:

Validation Set Accuracy: 31%

Testing Set Accuracy: 48%

Key Takeaways from other teams

- Significance of Data Preprocessing
  - Data preprocessing assumes a pivotal role in ensuring the overall success of the pipeline. This crucial step involves refining and preparing the data for modeling. Techniques such as scaling and normalization carry weight, exerting a substantial influence on the subsequent performance of machine learning models. Effective preprocessing guarantees that the data is in an appropriate form, thereby contributing to the robustness and efficiency of the entire modeling process.

- Dimensionality Reduction

    - In the realm of feature extraction, formidable tools like Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) come into play to facilitate dimensionality reduction. These techniques effectively enhance model accuracy by concentrating on the most pertinent features while simultaneously reducing the complexity of the dataset. The selection of a specific dimensionality reduction method pivots on the dataset's characteristics and the precise requirements of the modeling task, underscoring the importance of a customized and strategic approach.

- Data Quality Enhancement and Feature Extraction
    - Commitment to improving data quality, employing robust data cleaning processes, and utilizing effective feature extraction methods reflects a deep understanding of the pivotal role of preprocessing in elevating model accuracy.

- Diverse Approaches
    - Exploration of various methods, such as Haar cascades, LDA, and FaceNet, underscores their open-minded approach to experimenting with different techniques. This commitment aims to identify the most effective solution for the face recognition task.

- Overfitting Mitigation
    - Addressed overfitting challenges through strategic measures, including the diversification of training data, incorporation of data augmentation techniques, and a strong emphasis on the importance of hyperparameter tuning. These efforts collectively contribute to managing the common challenge of overfitting.

- Adaptive Learning Rate and Optimization
    - Recognition of the significance of an adaptive learning rate, exemplified using the Adam optimizer, and their acknowledgment of optimization techniques showcase a nuanced understanding of the dynamics involved in model training. This reflects their sophisticated approach to optimizing model training efficiency.

- Selection of Models

    - The landscape of machine learning unfolds a diverse array of models, each harboring unique strengths and suitability for various tasks. Models such as Support Vector Machines (SVM) and Dlib cater to distinct scenarios and datasets. The process of choosing an appropriate model becomes a critical decision point, as it profoundly shapes the outcome of the machine learning process. For example, SVM may exhibit superior performance with smaller datasets, while Dlib could potentially offer heightened accuracy when dealing with larger and more diverse datasets. Hence, a comprehensive grasp of the characteristics of different models and making judicious choices is paramount for achieving optimal results in machine learning endeavors.

- Scalability
    - The adaptability of a model to accommodate larger datasets stands as a pivotal consideration requiring meticulous attention. Models that can sustain their performance without imposing notable increases in computational demands as the volume of data expands are recognized for their efficiency and practicality. This capability ensures that the model remains both effective and manageable, even in the face of substantial data growth, underscoring its versatility and reliability.
- Evaluation Metrics
    - Relying solely on a single metric for assessing model performance may lead to a constrained perspective. It is imperative to expand the evaluation horizon by considering a diverse set of metrics, including accuracy, precision, recall, F1 score, and others. This comprehensive approach ensures a more nuanced and holistic assessment, capturing various facets of the model's performance and effectiveness. By considering a range of metrics, the evaluation becomes more thorough and insightful, providing a well-rounded understanding of the model's capabilities across different dimensions.