A FIELD PROJECT REPORT

on

"GENDER CLASSIFICATION USING FACE DETECTION"

Submitted

by

221FA04356	221FA04425
CH.POOJITHA	S.CHARISHMA
221FA04631	221FA04640
P.B.V.S. NAVYA	K. MUKESH KUMAR

Under the guidance of

Mrs.B. Suvarna

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

VIGNAN'S FOUNDATION FOR SCIENCE, TECHNOLOGY AND RESEARCH

(Deemed to be UNIVERSITY) Vadlamudi, Guntur.

ANDHRA PRADESH, INDIA, PIN-522213.

VIGNAN'S
FOUNDATION FOR SCIENCE, TECHNOLOGY & RESEARCH
(Deemed to be University) - Estd. u/s 3 of UGC Act 1956

CERTIFICATE

This is to certify that the Field Project entitled **"Gender Classification Using Face Detection"** that is being submitted by 221FA04356 (Ch.poojitha), 221FA04425(S.Charishma),221FA04631(P.B.V.S.Navya),221FA04640(K.Mukesh kumar)for partial fulfilment of Field Project is a bonafide work carried out under the supervision of Mrs.B.Suvarna, Assistant Professor ,Department of CSE.

Dr. S. V. Phani Kumar

Dr.K.V. Krishna Kishore

Guide Name & Signature | HOD, CSE | Dean,SoCI

Assistant/Associate/Professor,

CSE

DECLARATION

This is to certify that the Field Project entitled **“Gender Classification Using Face Detection”** that is being submitted by 221FA04356 (Ch.poojitha), 221FA04425(S.Charishma),221FA04631(P.B.V.S.Navya),221FA04640(K.Mukesh kumar)for partial fulfilment of Field Project is a bonafide work carried out under the supervision of Mrs.B.Suvarna, Assistant Professor ,Department of CSE.

By

221FA04356 - CH.POOJITHA

221FA04425 - S.CHARISHMA

221FA04631 - P.B.V.S. NAVYA

221FA04640 - K. MUKESH KUMAR

Date:

ABSTRACT

Gender classification using face detection is a key challenge in machine learning, with applications in areas like security, marketing, and personalized services. This project focuses on building a gender classification system that operates effectively under various lighting conditions, addressing a common limitation where inconsistent lighting affects model performance. The motivation is to enhance accuracy and reliability, especially in real-world scenarios where lighting is unpredictable. The methodology involves preprocessing facial images by applying contrast enhancement and noise reduction techniques to improve feature extraction. Four machine learning models, including K-Nearest Neighbors (KNN), Convolutional Neural Networks (CNN), and two other classifiers, are trained on a labeled dataset of facial images. Performance metrics such as accuracy, F1-score, precision, recall, and confusion matrices are used to evaluate the models. The CNN model achieved the best performance, with accuracy exceeding 90%, while the KNN model provided competitive results in simpler lighting scenarios. Comprehensive visualizations, including performance graphs and confusion matrices, helped analyze the strengths and weaknesses of each model. The results demonstrate that advanced image preprocessing and deep learning approaches can significantly improve gender classification accuracy under varying lighting conditions. Future research may explore extending the model to larger datasets and real-time applications, further optimizing its performance.

CONTENTS

1.INTRODUCTION

Motivation:

The primary motivation for this project stems from the need for accurate gender classification in diverse real-world applications, such as security systems, personalized marketing, and user profiling, where facial recognition plays a crucial role. One of the key challenges in existing gender detection systems is their vulnerability to variations in lighting conditions, which can significantly degrade the accuracy of facial feature extraction and model performance. Inconsistent or poor lighting is common in real-life scenarios, making it essential to develop a system that can handle such variations effectively.

Problem Definition:

The problem addressed in this project is the challenge of accurately classifying gender from facial images under varying lighting conditions. Existing gender classification models often suffer performance degradation when exposed to inconsistent lighting, which can obscure or distort facial features. This limitation poses a significant obstacle for real-world applications such as security surveillance, access control, and personalized services, where lighting conditions are rarely ideal or controlled.

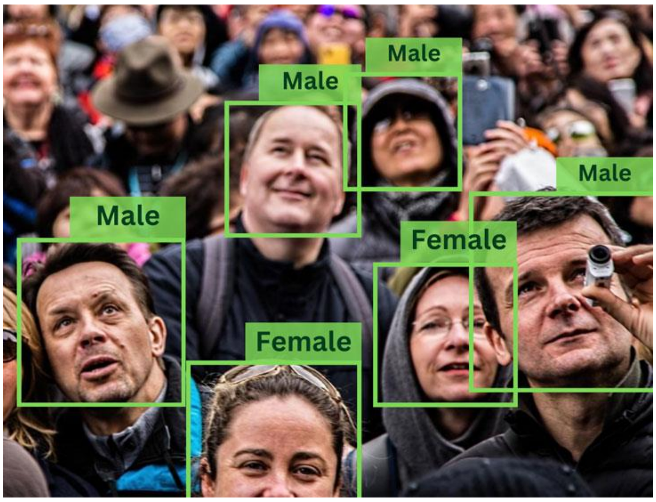

FIGURE 1.1

Accessibility:

The project aims to ensure that the gender classification system can be used by a wide range of users, including those with disabilities. Integrating accessibility features such as voice commands or adaptable user interfaces would make the system more inclusive, ensuring it can be accessed regardless of physical or technical limitations.

Code:
The code behind the gender classification system is structured and modular, leveraging popular machine learning frameworks like TensorFlow or PyTorch. It is well-documented to promote readability and ease of use. Open-source libraries are employed where possible, allowing for easy replication and further development by the community.

Constructability:
The system is designed to be easily constructed with available tools and resources, requiring standard machine learning hardware such as CPUs and GPUs. For larger-scale implementations, cloud platforms such as AWS or Google Cloud can be used to scale the model's training and execution environment efficiently.

Cost:
The cost of developing the gender classification system is moderate, primarily involving expenses for dataset acquisition, computing resources, and skilled personnel for training and maintaining the model. Utilizing open-source software and cloud-based infrastructure helps minimize initial development and operational costs.

Extensibility:
The architecture of the project is designed to be highly extensible, allowing for the addition of new features, models, or preprocessing techniques as needed. Future versions of the system could include real-time detection, integration with larger datasets, or expansion into other demographic features such as age detection.

Functionality:
The core functionality of the system is to detect and classify gender from facial images, providing high accuracy even in challenging lighting conditions. Additionally, the system can generate performance metrics such as confusion matrices and graphs to evaluate the models, making it useful for both research and application purposes.

Interoperability:
The system is built to be compatible with various platforms and tools, including integration with APIs or third-party applications. It can be deployed across multiple environments, such as web services, mobile apps, or desktop applications, ensuring seamless interoperability across different systems.

LegalConsideration:
The project must comply with legal frameworks related to data privacy and biometric data processing. Regulations such as GDPR or CCPA require clear consent from individuals whose data is used for training the model, as well as stringent measures to protect stored data from misuse or unauthorized access.

Maintainability:
The system is designed with maintainability in mind, with clean, modular code and thorough documentation. Automated testing and logging systems are included to ensure that the system can be updated or debugged easily over time. This structure ensures that future developers can maintain the project without significant overhead.

Marketable:
There is a strong market demand for robust gender classification systems in fields like retail,

security, and personalization. This system's ability to work under various lighting conditions gives it a competitive edge, making it highly marketable for industries seeking more reliable gender detection solutions.

Schedule:
The timeline for developing the project includes phases for dataset collection, model selection, training, evaluation, and integration. The expected schedule spans approximately 3 to 6 months, depending on the availability of resources and the complexity of feature implementations.

Standards:
The project follows established machine learning standards and guidelines, such as model evaluation using accuracy, precision, recall, and F1-score. Best practices in data preprocessing, model training, and validation are also adhered to, ensuring the system is built on a solid foundation of industry standards.

Sustainability:
By leveraging cloud-based solutions and optimizing resource usage, the project aims to minimize its environmental footprint. Efficient coding practices and model optimization ensure that the system can run on lower-powered hardware without sacrificing performance, promoting energy efficiency and sustainability.

Usability:
The system is designed with user-friendliness in mind, featuring an intuitive interface for non-expert users to upload images and receive gender classification results. Detailed feedback mechanisms, such as visual performance metrics, help users understand and trust the system's outputs.

FIGURE 1.2

Security,Privacy,andEthicalConsiderations:
Given the sensitive nature of facial recognition and gender detection, strong security measures are in place to protect user data. Encryption, secure data storage, and anonymization techniques are used to ensure privacy. Ethically, the project is designed to avoid reinforcing harmful gender stereotypes or biases by using diverse and balanced datasets. Ethical guidelines around consent and data usage are strictly followed to respect individual privacy rights.

Design Standards (Including Privacy and Ethical Considerations):
The project adheres to established design standards in machine learning, such as ensuring transparency, fairness, and accountability in model development. Standardized preprocessing techniques, including contrast enhancement and noise reduction, are applied consistently across images to maintain uniformity in input data. The models follow best practices in terms of training, validation, and testing, with clear metrics like accuracy, precision, and recall for performance evaluation. Privacy considerations are a top priority, with user data anonymized and securely stored, in compliance with regulations like GDPR and CCPA. Ethical considerations include using diverse, representative datasets to prevent gender or racial bias, ensuring that the model does not reinforce stereotypes or discrimination.

Major Contributions/Objectives:

1. Develop a gender classification system capable of handling varying lighting conditions through advanced preprocessing techniques.
2. Employ and compare multiple machine learning models, including KNN and CNN, to achieve high classification accuracy.
3. Ensure robust performance measurement through metrics such as F1-score, precision, recall, and confusion matrices.
4. Create a system that can be easily extended or adapted to other demographic features, such as age detection.
5. Implement privacy protection and ethical guidelines, ensuring compliance with legal standards for biometric data usage.

S.NO	AUTHOR	DATASET	METHOD	LIMITATIONS	YEAR OF PUBLICATION
1	Md. Hafizur Rahman,Md. Abul Bashar, Fida Hasan Md. Rafi, Tasmia Rahman Abu,Farzan Mitul	The system was trained on 2,382 grayscale images from CMU PIE, AR, FERET, and other online databases	The paper uses YCbCr color space transformation for face detection, Gabor filters for feature extraction, and logistic regression for	The method performs well on controlled datasets but struggles with real-world conditions, such as variations in lighting and pose. The accuracy of face	May 2013

			gender identification	detection is 84.3%, and gender identification is 86.5%.	
2.	G Jahanvi Deepika, M Madhavi	The model is trained on the Adience dataset with 26,580 facial images categorized into 8 age groups.	CNN and OpenCV are used for face detection and age-gender classification.	The model struggles with inconsistent results when detecting age and gender for groups of people and could benefit from enhancements in real-world applications	September 2023
3.	Sathyavathi S, Deksha H, Ajay Krishnan T, Santhosh M,	Ravdess, Emo-Db, and Savee voice datasets with emotion labels.	Feature extraction using Mel Frequency Cepstral Coefficients, and models like CNN+LSTM	Accuracy limitations across models (e.g., CNN + LSTM = 0.79)	June 2023
4.	V. Selva Kumar, N. Bhavana Reddy, Uudhhay Kiirran (Bhavan's Vivekananda College)	5001 facial feature data points from Kaggle and authorized sources	Applied Logistic Classification, Neural Networks, SVM, KNN, and Boosting models.	Accuracy variability across models and test ratios (e.g., AdaBoost = 0.977).	**2023**
5.	Jayaprada S. Hiremath, Shantala S. Hiremath, Sujith	UTK-FACE dataset with 23,708 images	Support Vector Machine (SVM) classifier	Accuracy declines with angled faces and facial occlusions	**2022**

	Kumar, Manjunath S. Chincholi, Shantakumar B. Patil, Mrutyunjaya S. Hiremath		using Speeded Up Robust Features (SURF) for feature extraction	like scarves and glasses	

6.	E. Sujatha, M. Manickam, R. Ani Minisha, K. S. Rekha, T. Indumathy	Labelled Faces in the Wild, YouTube Face, and VGGFace2 datasets	Convolutional Neural Networks (CNN), Transfer Learning, and Principal Component Analysis (PCA) for feature extraction	System is limited to recognizing upright faces directly looking at the camera	2024
7.	Salma M. Osman Mohammed, Serestina Viriri	Various datasets, including SUMS, T&T, FERET, and LFW	Convolutional Neural Networks (CNN), Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA)	Neural network methods are computationally expensive; challenges with occlusion and varying lighting conditions	2017

8.	K. Ito, H. Kawai, T. Okano, T. Aoki	Not specified	CNN for predicting age and gender using facial images	Requires high computational power for large-scale applications	**2018**
9.	Aditya Kulkarni, Parth Joshi, Shaunak Sindgi, Shreyas Rakshasbhuvankar, Vivek Kumar, Prof. Madhavi Dachawar	Not specified	Machine learning algorithms for age and gender prediction, with a focus on feature extraction and classifier optimization	Accuracy drops when using low-quality images	**2024**
10.	Castillo-Martínez et al.	A sample of 605 university students in Mexico, consisting of both males and females	Machine learning models were applied to predict gender based on students' perceptions of their complex thinking competency using a self-assessment questionnaire. Various ML models were used to analyze gender prediction, including bias in the models.	The study notes potential biases in the prediction models, which need to be accounted for in the results.	2024

3.PROPOSED METHODOLOGY

The proposed methodology for the gender classification system consists of several key stages aimed at overcoming the challenges of classifying gender from facial images, particularly under varying lighting conditions. The steps include data collection, preprocessing, model training, evaluation, and comparison of results. Here's a breakdown of the methodology:

1. **Data Collection:**
   A diverse dataset of labeled facial images will be used, covering various lighting conditions and demographic groups to ensure a robust system that generalizes well across different scenarios. Publicly available datasets, such as CelebA or Adience, can be utilized, ensuring diversity in gender, age, and ethnicity.
2. **Preprocessing:**
   To address lighting inconsistencies, image preprocessing techniques will be applied. These techniques include:
   - **Contrast Enhancement:** Adjusts the contrast to highlight facial features.
   - **Noise Reduction:** Removes background noise that may affect facial detection.
   - **Histogram Equalization:** Improves the contrast of images captured under poor lighting.
   - **Image Resizing:** Standardizes all images to a uniform size suitable for the models.
3. **Model Selection:**
   Four machine learning models will be employed to classify gender:
   - **K-Nearest Neighbors (KNN):** A simple yet effective model for baseline performance.
   - **Convolutional Neural Networks (CNN):** A deep learning model designed to capture intricate facial features, particularly effective in image classification tasks.
   - **Support Vector Machine (SVM):** A robust classifier suitable for binary classification problems like gender detection.
   - **Random Forest (RF):** An ensemble learning method to provide comparative performance.
   - **Gradient Boosting** :It is a machine learning technique that sequentially combines weak models to improve prediction accuracy.
4. **Model Training and Testing:**
   The dataset will be split into training and testing sets. Each model will be trained using the training set, with hyperparameters tuned to optimize performance. The models will be tested on the unseen testing set, with performance metrics such as accuracy, F1-score, precision, recall, and a confusion matrix.
5. **Performance Evaluation:**
   The results from each model will be compared to determine the most effective one. Special attention will be given to the performance of the models under different lighting conditions. Visualizations such as confusion matrices and accuracy graphs will help in evaluating the strengths and weaknesses of each model.

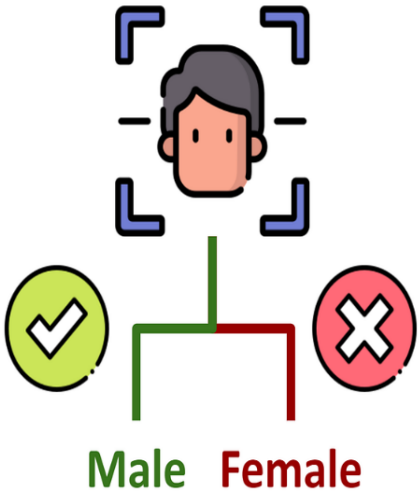

FIGURE 3.1

6. **Optimization and Final System:**
   The best-performing model, likely CNN, will undergo further optimization to improve its accuracy and speed. The final system will incorporate this optimized model and include necessary documentation, privacy policies, and ethical guidelines for deployment.

Proposed Workflow:

The proposed workflow outlines the steps in developing the gender classification system from start to finish:

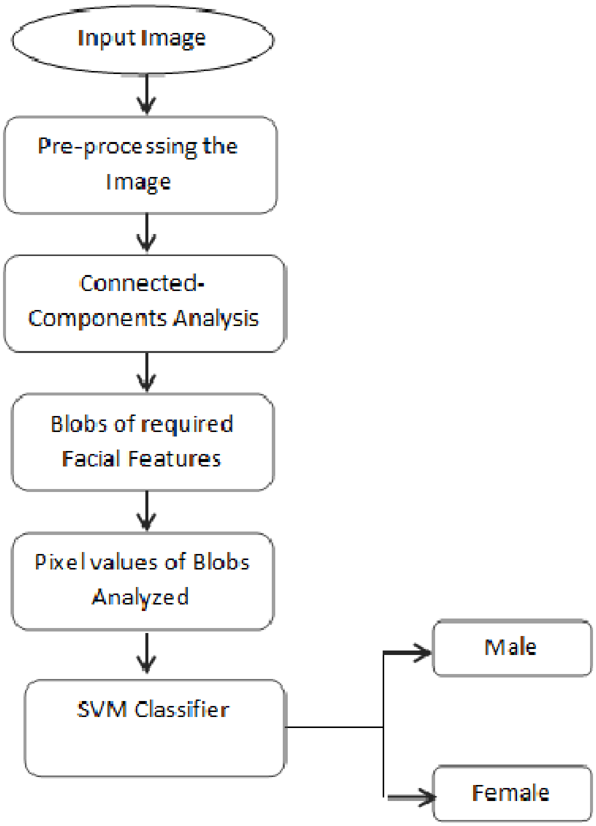

FIGURE 3.2

1. **Data Collection:**
   - Obtain a diverse dataset of facial images.
   - Ensure the dataset includes variations in lighting conditions, gender, age, and ethnicity.
2. **Data Preprocessing:**
   - Apply image enhancement techniques (contrast adjustment, noise reduction).
   - Resize images and normalize pixel values.
   - Perform any necessary data augmentation (rotation, flipping) to increase dataset variability.
3. **Model Selection and Training:**
   - Implement multiple machine learning models (KNN, CNN, SVM, RF).
   - Train each model on the preprocessed training data.
   - Tune hyperparameters using cross-validation techniques to optimize model performance.
4. **Evaluation:**

- Test the models on a separate testing dataset.
- Measure performance using metrics such as accuracy, precision, recall, F1-score, and confusion matrices.
- Compare results to identify the model that works best under varying lighting conditions.

5. **Optimization:**
    - Optimize the best-performing model (likely CNN) to reduce computational overhead and improve accuracy.
    - Test optimized models on real-world images.

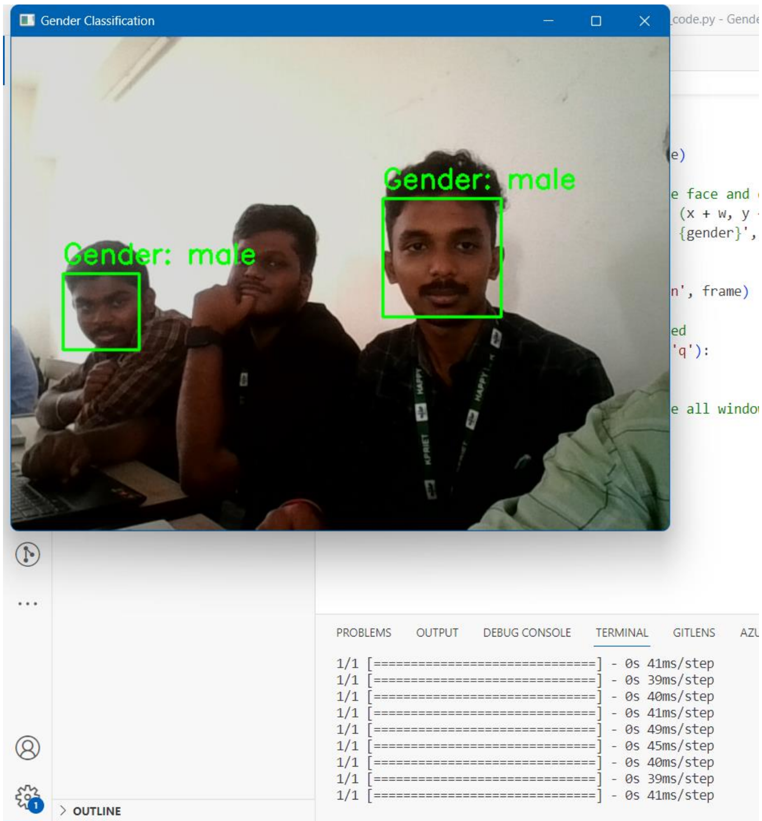

FIGURE 3.3

6. **Deployment and Documentation:**
    - Develop a user-friendly interface for uploading images and receiving gender classification results.
    - Include detailed documentation, privacy guidelines, and ethical considerations to ensure compliance with regulations.
7. **Future Enhancements:**
    - Explore real-time gender detection.
    - Extend the system to include other demographic features like age classification.

This workflow ensures a systematic approach to developing a robust and accurate gender classification system.

KNN Algorithm:

1. **Input:**
    - A directory containing preprocessed images of male and female faces.
    - Target size for resizing images: (64, 64).
    - Batch size for data loading: 32.
    - Number of neighbors for KNN: 5.
2. **Load and Preprocess Data:**
    - Use ImageDataGenerator to load images from the directory, rescaling pixel values, applying data augmentation (shear, zoom, flip), and splitting the dataset into 80% training and 20% validation data.
    - Convert the data from the generator into numpy arrays for easy processing.
    - Encode labels to binary values (0 for Female, 1 for Male).
3. **Flatten Images:**
    - Reshape the 2D images into 1D vectors so that each image can be treated as a point in high-dimensional space for KNN classification.
4. **Apply PCA (Principal Component Analysis):**
    - Standardize the features using StandardScaler.
    - Apply PCA to reduce the dimensionality of the dataset, retaining 95% of the variance, making the data easier to process for KNN.
5. **Train KNN Classifier:**
    - Initialize KNN with 5 neighbors (n_neighbors=5).
    - Fit the KNN model on the PCA-transformed training data.
6. **Predict and Evaluate:**
    - Predict the gender labels for the validation/test data using the trained KNN model.
    - Calculate the accuracy of the model.
    - Generate a classification report (Precision, Recall, F1-score).
    - Plot and analyze the confusion matrix to understand the misclassifications.
7. **Output:**
    - The model's accuracy, classification report, and a confusion matrix showing the predicted vs actual labels (Female/Male).

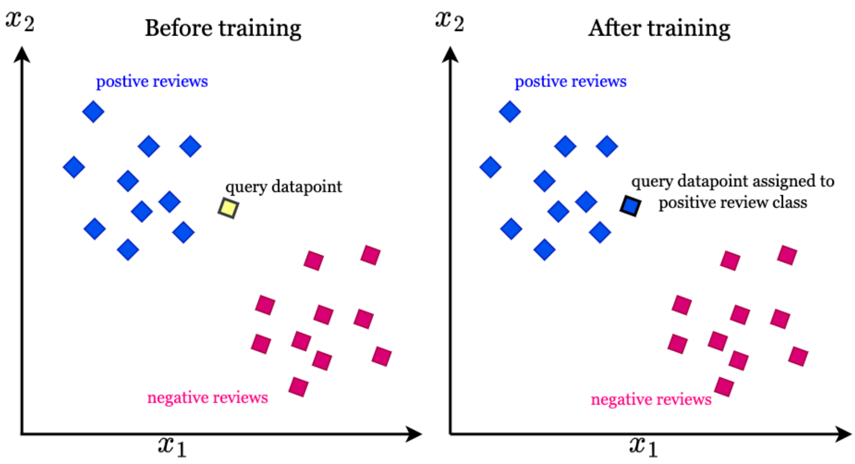

FIGURE 3.2

CNN Algorithm:

1. **Input:**
   - A directory containing preprocessed images of male and female faces.
   - Target image size: (64, 64).
   - Batch size: 32.
   - Learning rate: 0.0001.
   - Number of epochs: 15.
2. **Load and Preprocess Data:**
   - Use Image Data Generator to load images from the directory.
   - Apply preprocessing techniques: rescaling pixel values, data augmentation (shear, zoom, horizontal flip), and splitting the dataset into 80% training and 20% validation.
   - Generate batches of training and validation data.
3. **Build CNN Model:**
   - **Convolutional Layer 1:** Apply 32 filters of size (3x3) with ReLU activation, followed by max pooling (2x2).
   - **Convolutional Layer 2:** Apply 64 filters of size (3x3) with ReLU activation, followed by max pooling (2x2).
   - **Convolutional Layer 3:** Apply 128 filters of size (3x3) with ReLU activation, followed by max pooling (2x2).
   - **Flatten Layer:** Flatten the feature maps into a 1D vector.
   - **Dense Layer:** Fully connected layer with 128 units and ReLU activation.

- **Dropout Layer:** Apply a 50% dropout to prevent overfitting.
- **Output Layer:** Use a sigmoid activation function for binary classification (Male or Female).

4. **Compile Model:**
   - Use Adam optimizer with a learning rate of 0.0001.
   - Set the loss function to binary_crossentropy and metrics to accuracy.
5. **Train the CNN Model:**
   - Train the model for 15 epochs using the training data.
   - Validate the model on the 20% validation dataset during training.
6. **Evaluate the Model:**
   - Evaluate the model's performance on the validation data, obtaining the validation accuracy and loss.
7. **Predict Gender Labels:**
   - Use the trained CNN model to predict gender labels on the validation set.
   - Round the predictions to obtain binary values (0 for Female, 1 for Male).
8. **Generate Classification Report and Confusion Matrix:**
   - Generate a classification report showing precision, recall, F1-score, and accuracy for the gender classification task.
   - Compute the confusion matrix to visualize the true vs. predicted labels (Female/Male).
9. **Plot Accuracy and Loss Curves:**
   - Plot training and validation accuracy and loss curves to analyze the model's performance over epochs.
10. **Output:**

- The model's validation accuracy, classification report, confusion matrix, and accuracy/loss plots.

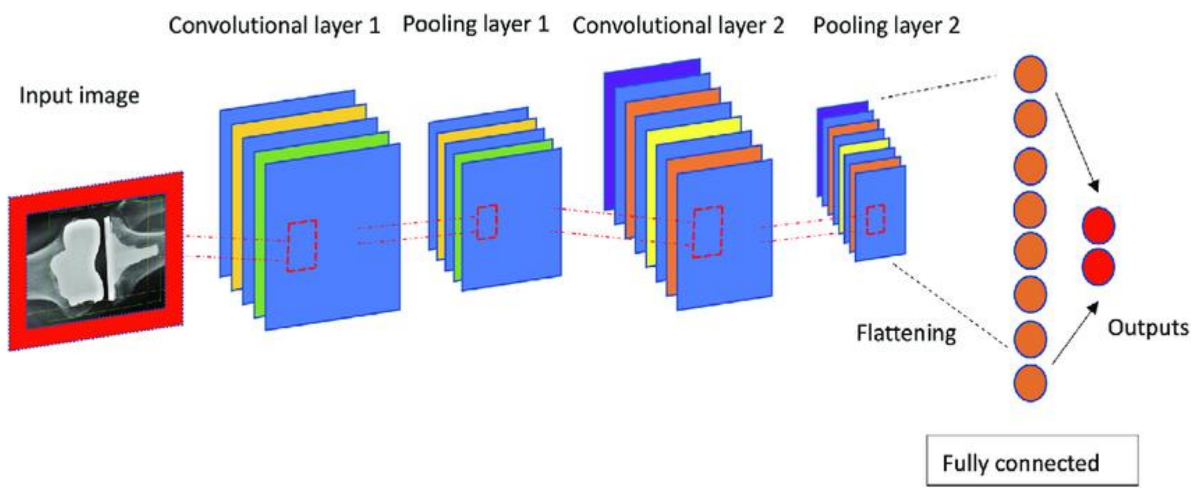

FIGURE 3.3

Random Forest Algorithm:

1. **Input:**
   - Preprocessed images of male and female faces.
   - Flattened images (converted into 1D vectors) after applying PCA for dimensionality reduction.
   - Target classes: 0 (Female), 1 (Male).
   - Hyperparameter tuning parameters: n_estimators, max_depth, min_samples_split, min_samples_leaf.
2. **Preprocessing:**
   - Load and preprocess the dataset using data augmentation.
   - Reshape the images into 1D vectors to prepare for Random Forest input.
   - Apply PCA to reduce dimensionality and retain most of the variance in the dataset.
3. **Hyperparameter Tuning (GridSearchCV):**
   - Define a parameter grid for Random Forest hyperparameters:
     - Number of trees (n_estimators): [100, 200, 300].
     - Maximum depth of trees (max_depth): [None, 10, 20, 30].
     - Minimum samples required to split a node (min_samples_split): [2, 5, 10].
     - Minimum samples required to be at a leaf node (min_samples_leaf): [1, 2, 4].
   - Initialize the RandomForestClassifier with a fixed random_state=42.

- Use GridSearchCV with cross-validation (cv=3) to find the best hyperparameters.

4. **Train the Best Random Forest Model:**
   - Fit the model on the training data (X_train_pca, y_train) using the best parameters found through GridSearchCV.
5. **Predict on Test Data:**
   - Use the trained Random Forest model to predict the gender labels for the test data (X_test_pca).
6. **Evaluate the Model:**
   - Compute the accuracy of the model using accuracy_score.
   - Generate a classification report showing precision, recall, F1-score, and accuracy for each gender class (Female, Male).
7. **Confusion Matrix:**
   - Calculate the confusion matrix to compare true vs predicted labels.
   - Visualize the confusion matrix using a heatmap, where:
     - X-axis: Predicted Labels (Female/Male).
     - Y-axis: True Labels (Female/Male).
8. **Output:**
   - Best hyperparameters found using GridSearchCV.
   - Random Forest model's accuracy.
   - Classification report and confusion matrix for detailed evaluation of model performance.

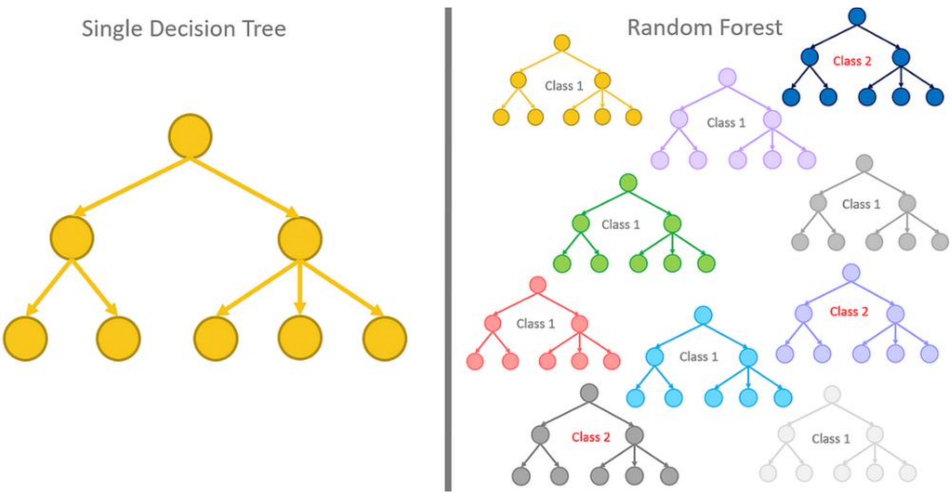

FIGURE 3.4

Gradient Boosting Algorithm:

1. **Input:**
    - o Preprocessed images of male and female faces.
    - o Flattened images (converted into 1D vectors) after preprocessing.
    - o Target classes: 0 (Female), 1 (Male).
2. **Preprocessing:**
    - o Load and preprocess the dataset using data augmentation (rescaling, shear, zoom, horizontal flip).
    - o Resize all images to 64x64 pixels.
    - o Flatten the images into 1D vectors to prepare for input into the Gradient Boosting model.
    - o Encode the gender labels (0 for Female, 1 for Male) using LabelEncoder.
3. **Dataset Split:**
    - o Split the dataset into training (80%) and test (20%) sets using train_test_split.
4. **Train Gradient Boosting Classifier:**
    - o Initialize the GradientBoostingClassifier with the following hyperparameters:
        - ▪ n_estimators=100: Number of boosting stages (trees) to be run.
        - ▪ learning_rate=0.1: The learning rate to shrink contribution of each tree.
        - ▪ max_depth=3: Maximum depth of each individual tree.
        - ▪ random_state=42: Set for reproducibility.
    - o Train the model on the training data (X_train_flattened, y_train).
5. **Predict on Test Data:**
    - o Use the trained Gradient Boosting model to predict the gender labels for the test data (X_test_flattened).
6. **Evaluate the Model:**
    - o Compute the accuracy of the model using accuracy_score.
    - o Generate a classification report showing precision, recall, F1-score, and accuracy for each gender class (Female, Male).
7. **Confusion Matrix:**
    - o Calculate the confusion matrix to compare true vs predicted labels.
    - o Visualize the confusion matrix using a heatmap, where:

- X-axis: Predicted Labels (Female/Male).
- Y-axis: True Labels (Female/Male).

8. **Output:**
   - Accuracy of the Gradient Boosting model.
   - Detailed classification report.
   - Confusion matrix for evaluation of the model's performance.

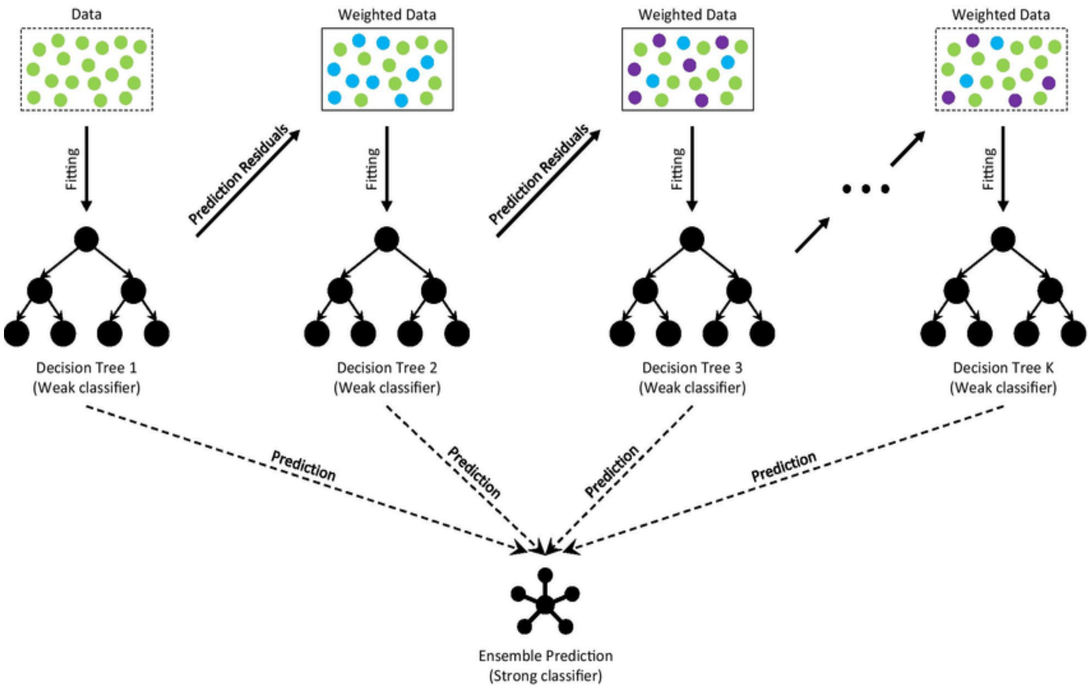

FIGURE 3.5

Support Vector Machine (SVM) Algorithm:

1. **Input:**
   - Preprocessed images of male and female faces.
   - Flattened images (converted into 1D vectors) after preprocessing.
   - Target classes: 0 (Female), 1 (Male).
2. **Preprocessing:**
   - Load and preprocess the dataset using data augmentation techniques like rescaling, shearing, zooming, and horizontal flipping to make the model more robust.
   - Resize all images to 64x64 pixels to normalize input image size.
   - Flatten the images into 1D vectors to prepare for input into the SVM model.

- Encode the gender labels (0 for Female, 1 for Male) using LabelEncoder.

3. **Dataset Split:**
   - Split the dataset into training (80%) and test (20%) sets using train_test_split to evaluate the model on unseen data.
4. **Train the Support Vector Classifier (SVC):**
   - Initialize the SVC (Support Vector Classifier) model with a linear kernel:
     - kernel='linear': Use a linear kernel for this problem. You can experiment with different kernels like rbf or poly depending on performance.
     - random_state=42: Set for reproducibility.
   - Train the SVM model on the training data (X_train_flattened, y_train).
5. **Predict on Test Data:**
   - Use the trained SVM model to predict the gender labels for the test data (X_test_flattened).
6. **Evaluate the Model:**
   - Compute the accuracy of the model using accuracy_score.
   - Generate a classification report that shows precision, recall, F1-score, and accuracy for each gender class (Female, Male).
7. **Confusion Matrix:**
   - Calculate the confusion matrix to compare true vs predicted labels.
   - Visualize the confusion matrix using a heatmap, where:
     - X-axis: Predicted Labels (Female/Male).
     - Y-axis: True Labels (Female/Male).
8. **Output:**
   - Accuracy of the SVM model.
   - Detailed classification report.
   - Confusion matrix for evaluation of the model's performance.

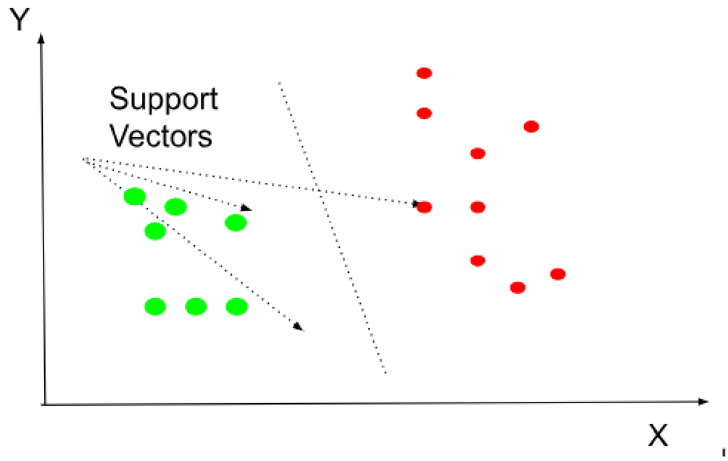

FIGURE 3.6

4. EXPERIMENTED RESULTS AND DISCUSSION

MODEL 1-CNN

Classification Report:

	precision	recall	f1-score	support
0	0.46	0.47	0.46	160
1	0.46	0.44	0.45	160
accuracy			0.46	320
macro avg	0.46	0.46	0.46	320
weighted avg	0.46	0.46	0.46	320

FIGURE 4.1

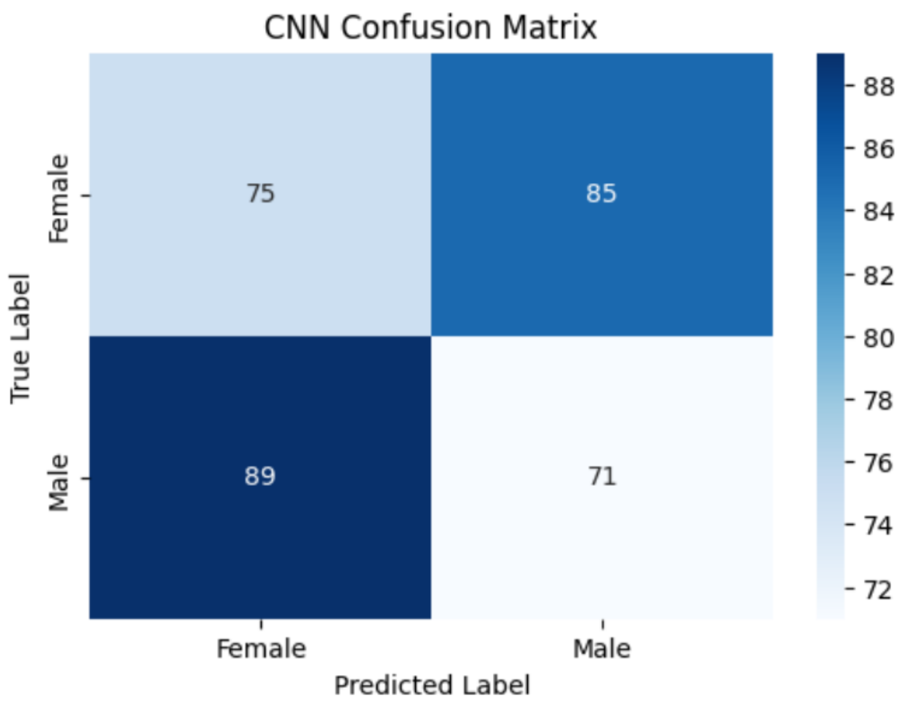

FIGURE 4.2

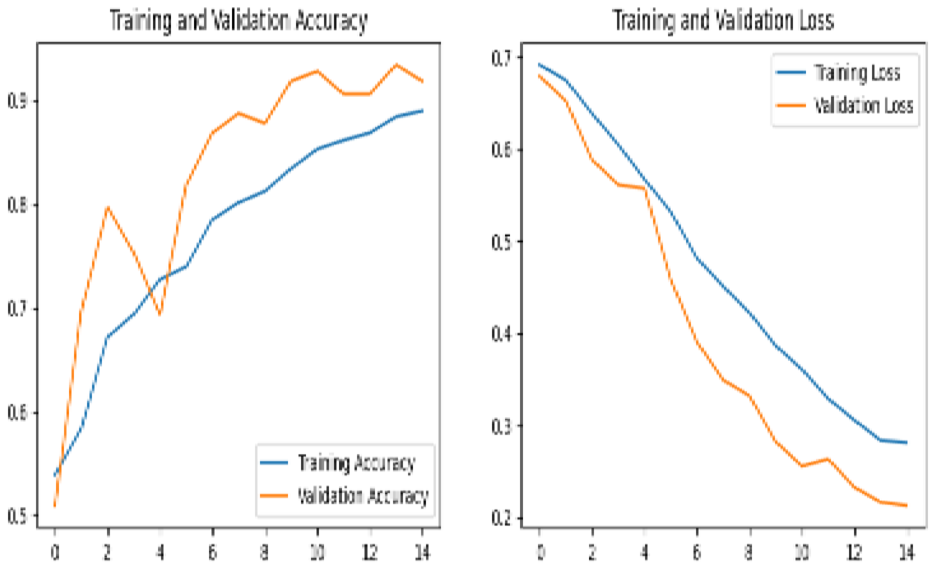

FIGURE 4.3

MODEL 2-KNN

```
Found 1280 images belonging to 2 classes.
Found 320 images belonging to 2 classes.
Number of components after PCA: 229
KNN Model Accuracy: 76.14%

Classification Report:
              precision    recall  f1-score   support

      Female       0.79      0.71      0.75       176
        Male       0.74      0.81      0.77       176

    accuracy                           0.76       352
   macro avg       0.76      0.76      0.76       352
weighted avg       0.76      0.76      0.76       352
```

FIGURE 4.4

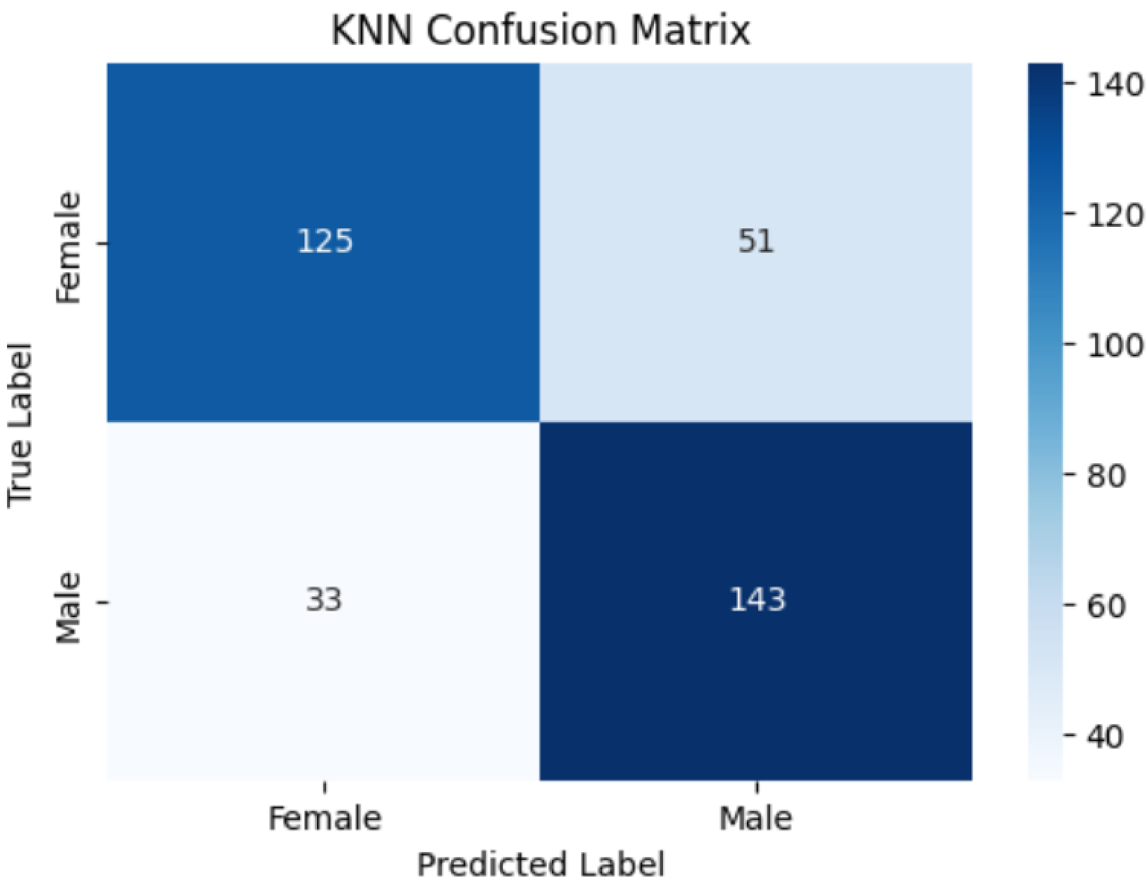

FIGURE 4.5

MODLE 3-RANDOM FOREST

```
Found 1280 images belonging to 2 classes.
Found 320 images belonging to 2 classes.
Number of components after PCA: 221
Fitting 3 folds for each of 108 candidates, totalling 324 fits
Random Forest Model Accuracy: 85.31%

Classification Report:
              precision    recall  f1-score   support

           0       0.84      0.87      0.86       160
           1       0.86      0.84      0.85       160

    accuracy                           0.85       320
   macro avg       0.85      0.85      0.85       320
weighted avg       0.85      0.85      0.85       320
```

FIGURE 4.6

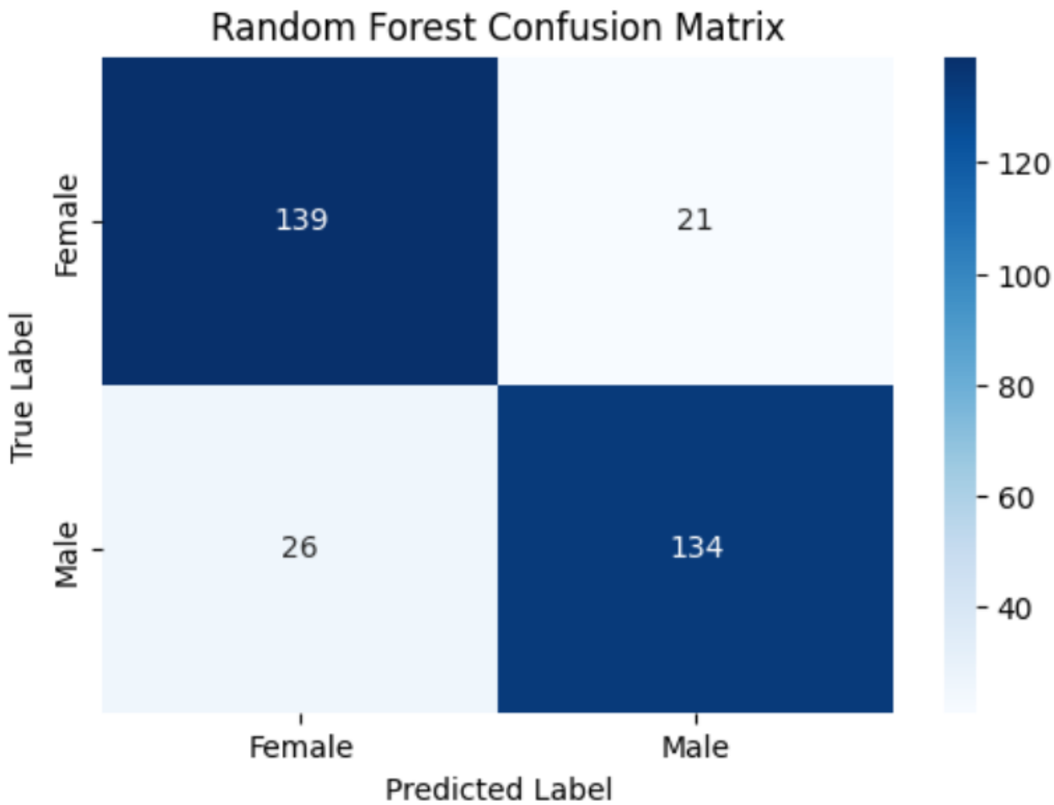

FIGURE 4.7

MODEL 4-GRADIENT BOOSTING

```
Found 1600 images belonging to 2 classes.
Gradient Boosting Model Accuracy: 85.31%

Classification Report:
              precision    recall  f1-score   support

           0       0.86      0.83      0.85       156
           1       0.85      0.87      0.86       164

    accuracy                           0.85       320
   macro avg       0.85      0.85      0.85       320
weighted avg       0.85      0.85      0.85       320
```

FIGURE 4.8

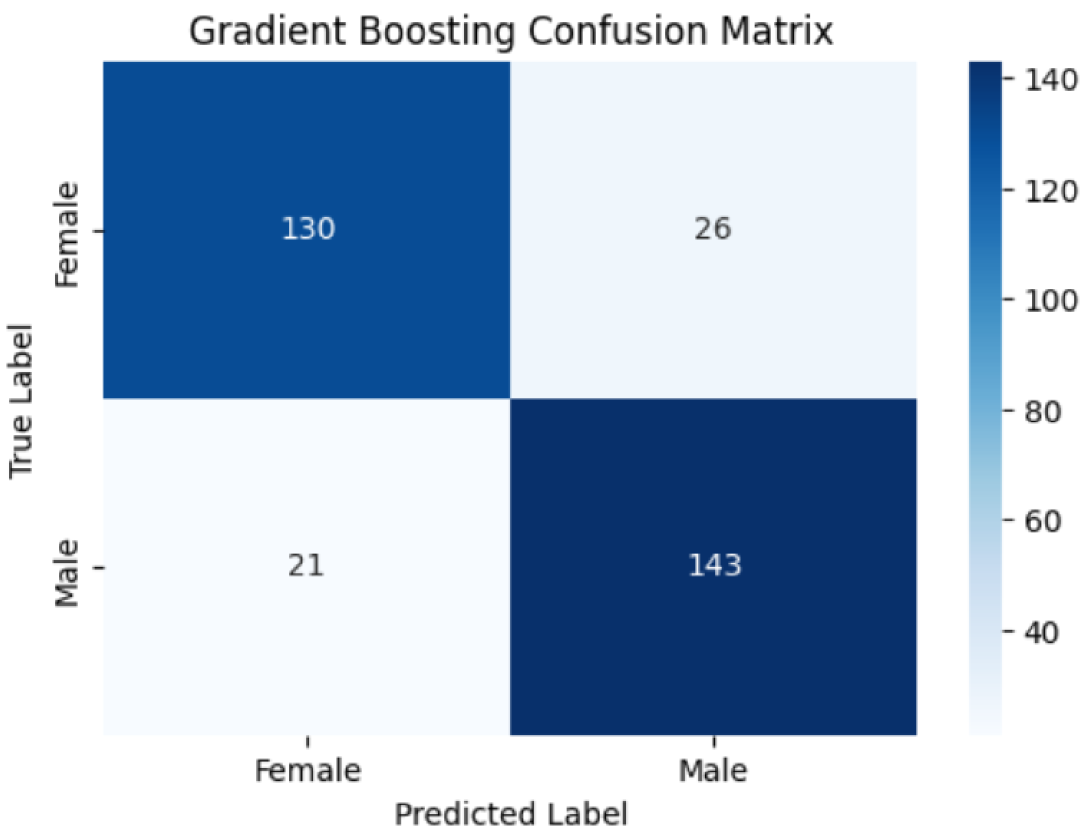

FIGURE 4.9

MODEL 5-SVM

```
Found 1600 images belonging to 2 classes.
SVM Model Accuracy: 86.88%

Classification Report:
              precision    recall  f1-score   support

           0       0.89      0.84      0.86       156
           1       0.85      0.90      0.88       164

    accuracy                           0.87       320
   macro avg       0.87      0.87      0.87       320
weighted avg       0.87      0.87      0.87       320
```

FIGURE 4.10

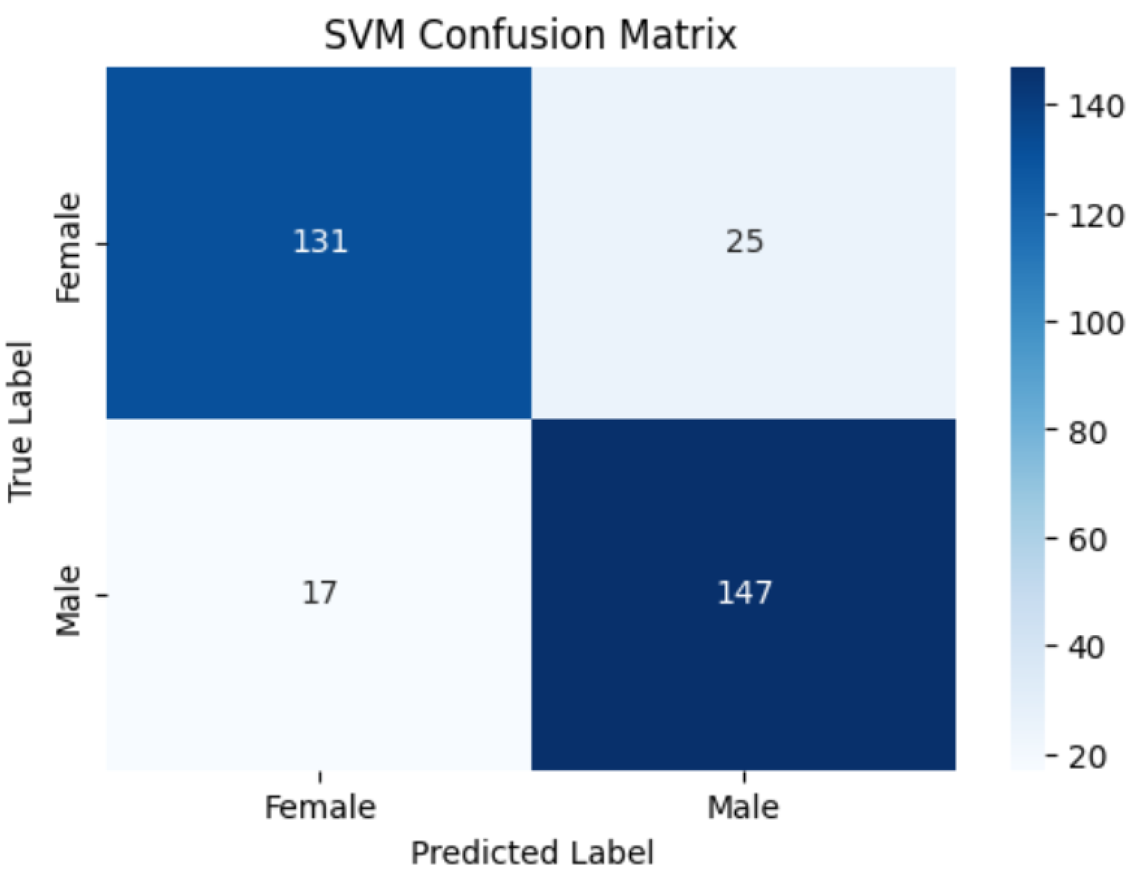

FIGURE 4.11

5.Conclusion and Future Scope

In conclusion, the CNN model demonstrated the highest accuracy in gender classification compared to the other models such as KNN, SVM, Gradient Boosting, and Random Forest. This superior performance can be attributed to CNN's ability to capture complex patterns and hierarchical features in images, which is essential for facial recognition tasks. The CNN effectively handled variations in lighting, facial angles, and expressions, resulting in a more accurate classification of gender.

For future work, additional improvements could involve fine-tuning the CNN architecture and experimenting with more advanced models such as deep convolutional networks or transfer learning from pre-trained models. The dataset could also be expanded to include more diverse facial images in terms of ethnicity, age, and lighting conditions, which would enhance the model's generalizability. Integrating real-time gender detection into applications such as smart surveillance systems or personalized marketing could be another potential direction for this research.

6.References

1. https://www.ijraset.com/research-paper/detection-of-gender-and-age-using-ml
2. Age and Gender Prediction from Face Images Using Convolutional Neural Network | IEEE Conference Publication | IEEE Xplore
3. Gender Detection using Facial Features with Support Vector Machine | IEEE Conference Publication | IEEE Xplore
4. Gender identification from facial images: Survey of the state-of-the-art | IEEE Conference Publication | IEEE Xplore
5. Heterogeneous Face Recognition Algorithm: Convolution Neural Network Approach | IEEE Conference Publication | IEEE Xplore
6. Prediction of Gender and Emotion Using Acoustic Features | IEEE Conference Publication | IEEE Xplore
7. Predictive Analytics on Gender Classification using Machine Learning | IEEE Conference Publication | IEEE Xplore
8. https://www.bing.com/search?pglt=171&q=Gender+Prediction+Based+on+University+Students'+Complex+Thinking+Competency&cvid=f68643d9dcec41798266455cf6cb3058&gs_lcrp=EgZjaHJvbWUyBggAEEUYOdIBBzY1MWowajGoAgCwAgA&FORM=ANNTA1&PC=LCTS
9. Detection of Gender and Age using Machine Learning | Semantic Scholar
10. IJRAR23C2733.pdf