Gender and Age Detection using Deep Learning

Ms. Vishakha Gawde[1]

Assistant professor,

Department of IT and CS,

Nagindas Khandwala College,

Mumbai,Maharashtra,India

vishakha@nkc.ac.in

Abhishek Mourya[2]

Student, Department of MSc. IT,

Nagindas Khandwala College,

Mumbai,Maharashtra, India

Mouryaa201@gmail.com

Abstract: The demand for automatic gender and age prediction has surged with the proliferation of social media and online platforms. Despite advancements in related fields like facial recognition, the effectiveness of current technologies in real-world scenarios remains limited. This paper explores the application of deep convolutional neural networks (CNNs) to address this challenge. Our proposed method involves a five-step process: facial recognition, background removal, face alignment, application of multiple CNN models, and a voting mechanism to enhance prediction accuracy. We evaluate our approach using the recent Audience-Face benchmark dataset, focusing on gender detection and age estimation. The implementation is carried out using Python.

Keywords: Convolutional neural networks (CNN), Deep learning, Facial recognition, Computer vision, Region of interest (ROI).

1. Introduction

Gender and age are critical attributes that significantly influence social interactions and community dynamics. Accurate gender detection and age estimation are crucial for various intelligent applications, including human-computer interaction, access control, marketing analytics, visual surveillance, and law enforcement, particularly when based on a single facial image. Traditional methods for classifying these attributes rely on facial feature variations, but recent advancements in facial recognition technology, particularly through convolutional neural networks (CNNs), have demonstrated substantial improvements in this area.

This paper proposes a novel approach to gender and age classification using multiple CNNs. The proposed system employs a multi-layered CNN architecture, consisting of three distinct CNN layers interconnected through a majority-voting mechanism for final class prediction.

The layers are structured as follows:

- **CNN Layer 1**: Contains 96 nodes with a kernel size of 7.
- **CNN Layer 2**: Contains 256 nodes with a kernel size of 5.
- **CNN Layer 3**: Contains 384 nodes with a kernel size of 3.

Each CNN layer has a unique depth and architecture, optimizing the feature extraction process for improved classification accuracy. The model processes input images to estimate age and determine gender. For age estimation, we categorize ages into predefined groups: [(0-2), (4-6), (8-12), (15-20), (25-32), (38-43), (48-53), (60-100)]. The gender classification is binary, distinguishing between male and female.

The proposed model was evaluated using the Audience-Face benchmark dataset, which includes unfiltered facial images. We conducted two separate tests: one for age classification and another for gender classification. The results demonstrate the efficacy of our approach in accurately predicting age ranges and gender from facial images.

Automated age and gender estimation offer significant advantages across various sectors. In retail, for example, shopkeepers and business owners can leverage this technology to analyze customer demographics, tailor marketing strategies, and enhance inventory management. This capability can lead to increased business efficiency and improved service delivery, benefiting both enterprises and society.

This research contributes to the advancement of facial attribute classification by integrating multiple CNNs and proposing a robust methodology for age and gender detection. This approach has practical applications in numerous fields and promises to enhance both business operations and societal interactions.

2. Review of Literature

To develop a new CNN structure for age and gender recognition, it is essential to review prior research that highlights the significance of CNN architecture in enhancing recognition performance.

Sr.no	Title	Observation
[1]	Simon & Zisserman (2014): "Very Deep Convolutional Networks for Large-Scale Image Recognition"	Introduces VGG-16 with 13 convolutional layers and 3 fully connected layers, demonstrating that depth between 16-19 layers provides optimal image recognition performance.
[2]	He et al. (2016): "Deep Residual Learning for Image Recognition"	Presents ResNet-152 with 152 convolutional layers, employing residual learning to handle depth complexity, achieving high accuracy in facial image recognition.
[3]	Zagoruyko & Komodakis (2016): "Wide Residual Networks"	Proposes Wide Residual Networks to address latency and inefficiency issues by reducing depth and increasing width of residual blocks, enhancing network efficiency.
[4]	Krizhevsky et al. (2012): "ImageNet Classification with Deep Convolutional Neural Networks"	Introduces AlexNet with 8 convolutional layers and 3 fully connected layers, using Dropout to reduce overfitting and effectively classify over one million images.
[5]	Levi & Hassaner (2015): "Age and Gender Classification using Deep Convolutional Neural Networks"	Develops a CNN with 3 convolutional layers and 2 fully connected layers, applying center cropping and oversampling techniques to improve gender classification performance.

Table 1: Review of literature

3. Methodology

1. **Model and Configuration Files**:

- **Face Detection Weights (.pb)**: The system utilizes pre-trained models with weights stored in a .pb file (Protocol Buffers) for face detection. These weights contain the parameters that the neural network has learned during training, which are crucial for accurately identifying faces in various images or video feeds.
- **Network Configuration (.prototxt)**: The architecture of the network is defined in a .prototxt file. This file describes the structure of the model, including the types and connections of layers. The configuration ensures that the network processes data correctly and performs the intended tasks of detection and classification.
- **Model Parameters (.caffemodel)**: The .caffemodel file holds the pre-trained weights and biases used by the network. These parameters, in combination with the configuration file, allow the network to recognize and classify different features, leading to precise age and gender predictions.

2. **Argument Parsing**:

- **Using argparse for Command-Line Inputs**: The system uses the argparse library to manage command-line inputs, enabling users to easily provide images or video files as input. This feature allows for flexible testing and deployment, giving users the ability to specify the input source directly from the command line.

3. **Protocol and Model Initialization**:

- **Setting Up Protocols, Buffers, and Models**: The initialization phase involves configuring the necessary protocols, buffers, and models for different tasks, including face detection, age estimation, and gender recognition. Protocols ensure smooth data flow, while buffers temporarily store data during processing. The models are then loaded and initialized, preparing the system for efficient operation.

4. **Configuration Initialization**:

- **Defining Age Ranges and Gender Categories**: The system is configured with predefined age ranges (e.g., 0-2, 4-6, 8-12, etc.) and gender categories (typically male and female). These predefined categories help the network interpret the results, mapping the output of the model to specific age ranges and gender labels, making the predictions understandable.

5. **Network Loading**:

- **Loading the Network with readNet()**: The neural network is loaded using the readNet() function, which imports the pre-trained weights and the network configuration from the .pb and .prototxt files. This step is vital for initializing the network, enabling it to perform tasks such as face detection and classification.

6. **Video Stream Capture**:

- **Capturing Video from Webcam**: The system captures video input from a webcam using OpenCV's VideoCapture() function. A padding of 20 pixels is added around the detected face area to ensure that the entire face, including the edges, is properly captured. This padding helps improve the accuracy of the subsequent age and gender classification.
- **Detecting Faces with highlightFace()**: The highlightFace() function utilizes the FaceNet model to detect faces in images or video streams. This function identifies the coordinates of faces, and rectangles are drawn around the detected faces to indicate the areas being analyzed. This visual indication helps users understand which faces the system is processing.

- **Creating a Blob and Classifying Faces**: For each detected face, the system creates a "blob," which is a preprocessed version of the image that can be fed into the neural network. The system then performs a forward pass through the network to classify the face's age and gender. Based on the confidence scores, the system predicts the most likely age range and gender for each detected face.

- **Annotating and Displaying the Results**: After classification, the predicted age and gender are annotated onto the image. The system uses OpenCV's imshow() function to display the final output, showing the annotated image with the detected and classified faces, along with their predicted age and gender. This provides a clear and visual output of the system's analysis.

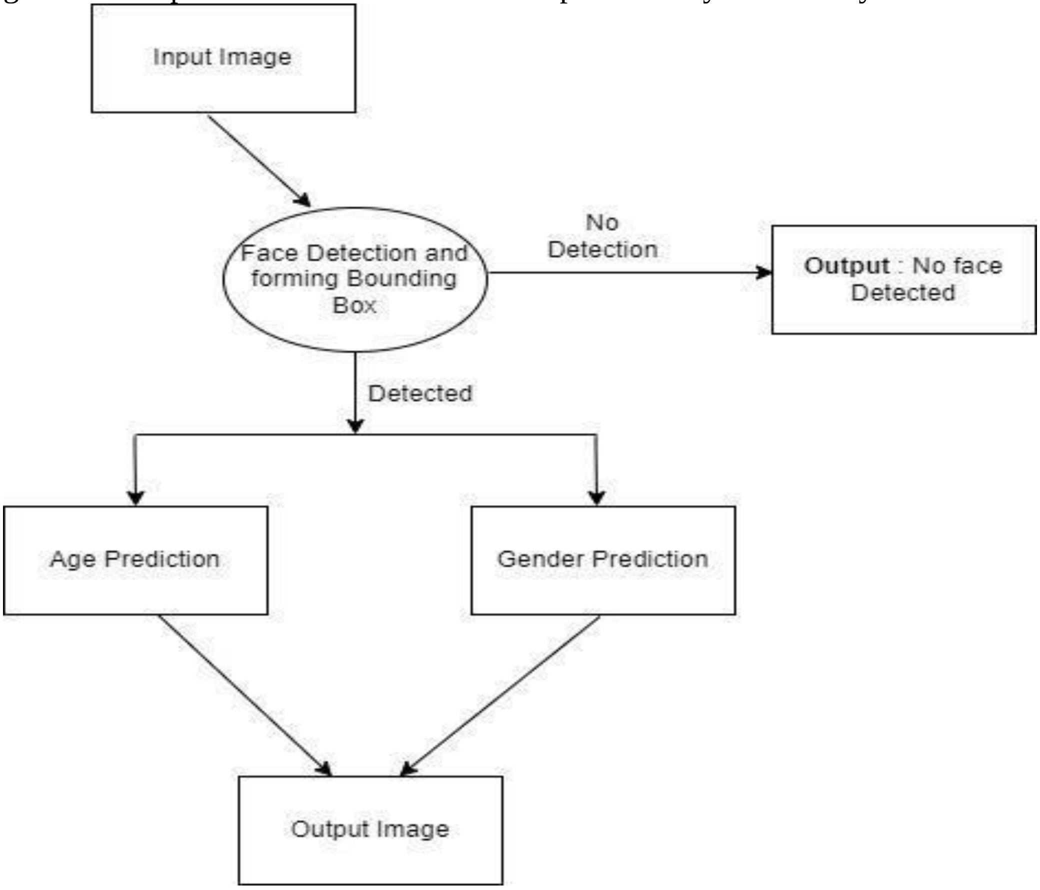

Fig 1: Flow diagram of the model

4. Model with experiment result

In this study, we employed Python-based Deep Learning techniques to identify specific genders and ages from facial images. Deep Learning, a subset of machine learning, leverages Artificial Intelligence to emulate human cognitive functions. It excels at analyzing unstructured data to recognize objects, individuals, speech, and text. The process is divided into four main stages: Input, Face Detection, Face Processing (which includes Gender and Age classification), and Output. We assessed the model's accuracy by comparing the predicted age groups to the actual ones, measuring accuracy both when the prediction exactly matched the true age group and when it was off by one adjacent age group. To boost the model's performance, we expanded the training dataset, which led to enhanced accuracy in the predictions.

```
# Evaluate the model on the validation set
evaluation_results = model.evaluate(X_val, [y_clf_val, y_reg_val], verbose=1)
print("Evaluation Results:", evaluation_results)
try:
    total_loss = evaluation_results[0]
    age_regression_mse = evaluation_results[1]
    gender_classification_accuracy = evaluation_results[2]
    gender_classification_accuracy_percentage = gender_classification_accuracy * 100

    print(f"Gender Classification Accuracy: {gender_classification_accuracy_percentage:.2f}%")
    print(f"Age Regression MSE: {age_regression_mse:.2f}")
except IndexError as e:
    print(f"Error: {e}. Check the indices based on the printed results.")
```

```
149/149 ━━━━━━━━━━━━━━━━━━━━ 2s 14ms/step - a_reg_mse: 119.6301 - g_clf_accuracy: 0.7857 - loss: 120.0945
Evaluation Results: [123.73395538330078, 123.27333068847656, 0.7899177670478821]
Gender Classification Accuracy: 78.99%
Age Regression MSE: 123.27
```

Fig 2: Accuracy of gender classification and MSE of age regression

The code evaluates the performance of a machine learning model on a validation dataset. It begins by calling model.evaluate() with the validation data and labels, which returns a list of metrics, including the total loss, age regression mean squared error (MSE), and gender classification accuracy. The raw evaluation results are printed for inspection.

```
# plotting for Gender Classification Accuracy
plt.plot(history.history['g_clf_accuracy'], label = 'training accuracy')
plt.plot(history.history['val_g_clf_accuracy'], label = 'validation accuracy')
plt.title('Gender classification model Accuracy')
plt.xlabel('epoch')
plt.ylabel('Accuracy for gender classification')
plt.legend()
plt.show()
```

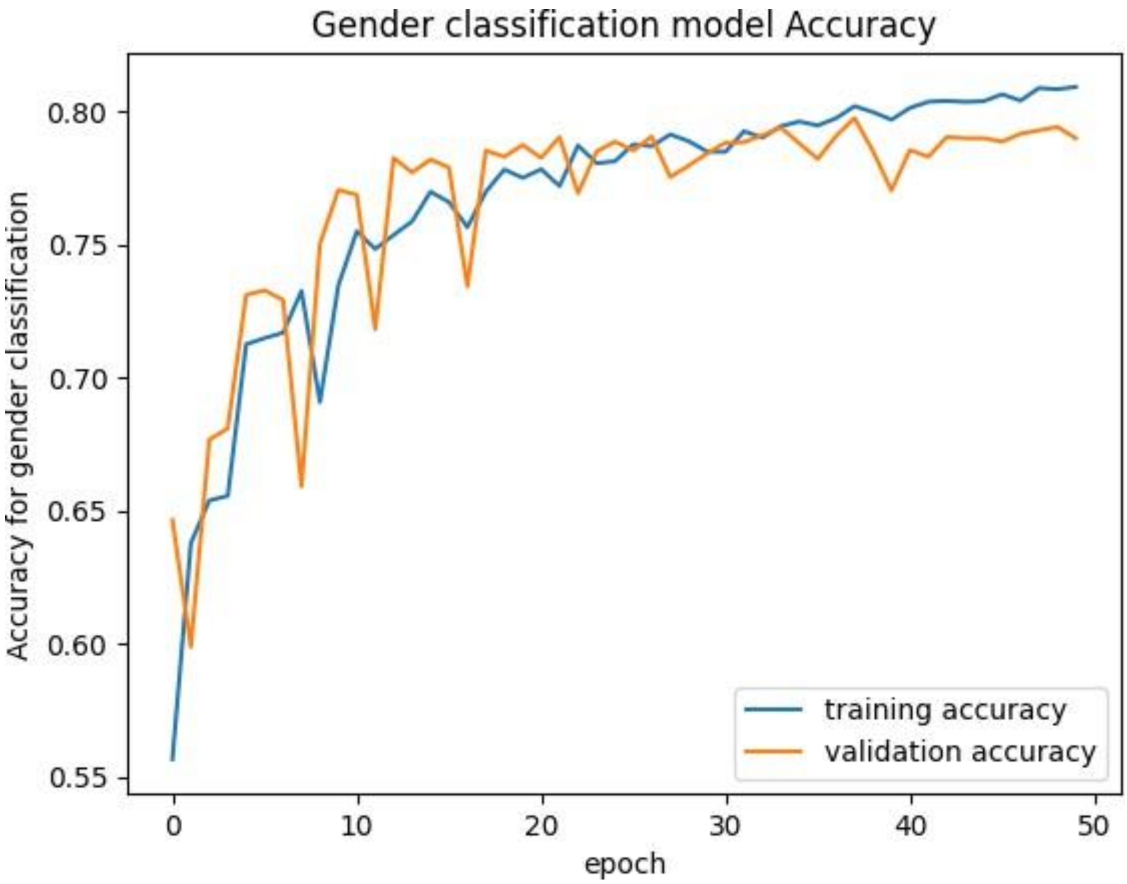

Fig 3: Plotting the training and validation accuracy for the gender classification

1. **Multitask Learning Approach**:

We have used multitask learning approach that involves training a model to handle multiple tasks simultaneously. the model is designed to predict both gender (classification task) and age (regression task) from images. the model has two output layers:

Gender Classification (g_clf): Uses a binary classification loss (binary_crossentropy) and accuracy metric.

Age Regression (a_reg): Uses a regression loss (mse for Mean Squared Error) and MSE metric.

2. **Convolutional Neural Network (CNN) Approach**:

CNNs are a type of deep learning model specifically designed to process data with a grid-like topology, such as images. They use convolutional layers to detect spatial hierarchies in the data. We have applied convolutional filters to extract features from the input images and reducing the dimensionality of the feature maps while retaining important information.

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 09 | Sept - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

The results for age classification and gender detection are summarized below.

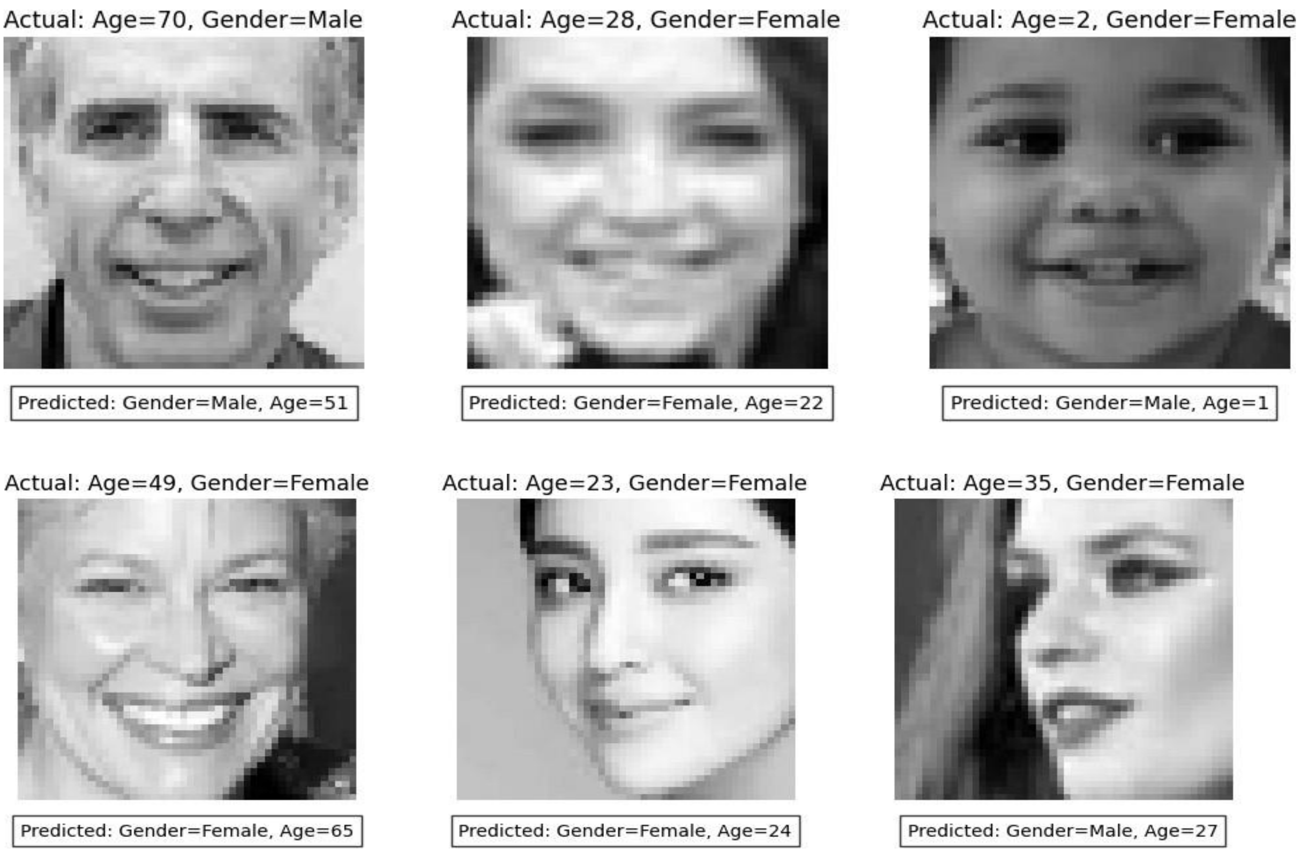

Fig 4: Predicted gender and age of validation images

I have also used some prebuilt models these models are part of a pre-trained deep learning- based model pipeline used for face detection, age prediction, and gender prediction. these models work in a pipeline to detect faces, and once a face is detected, it can be passed to the age and gender networks to predict the person's age range and gender.

```
# Importing Models
face1 = "opencv_face_detector.pbtxt"
face2 = "opencv_face_detector_uint8.pb"
age1 = "age_deploy.prototxt"
age2 = "age_net.caffemodel"
gen1 = "gender_deploy.prototxt"
gen2 = "gender_net.caffemodel"
```

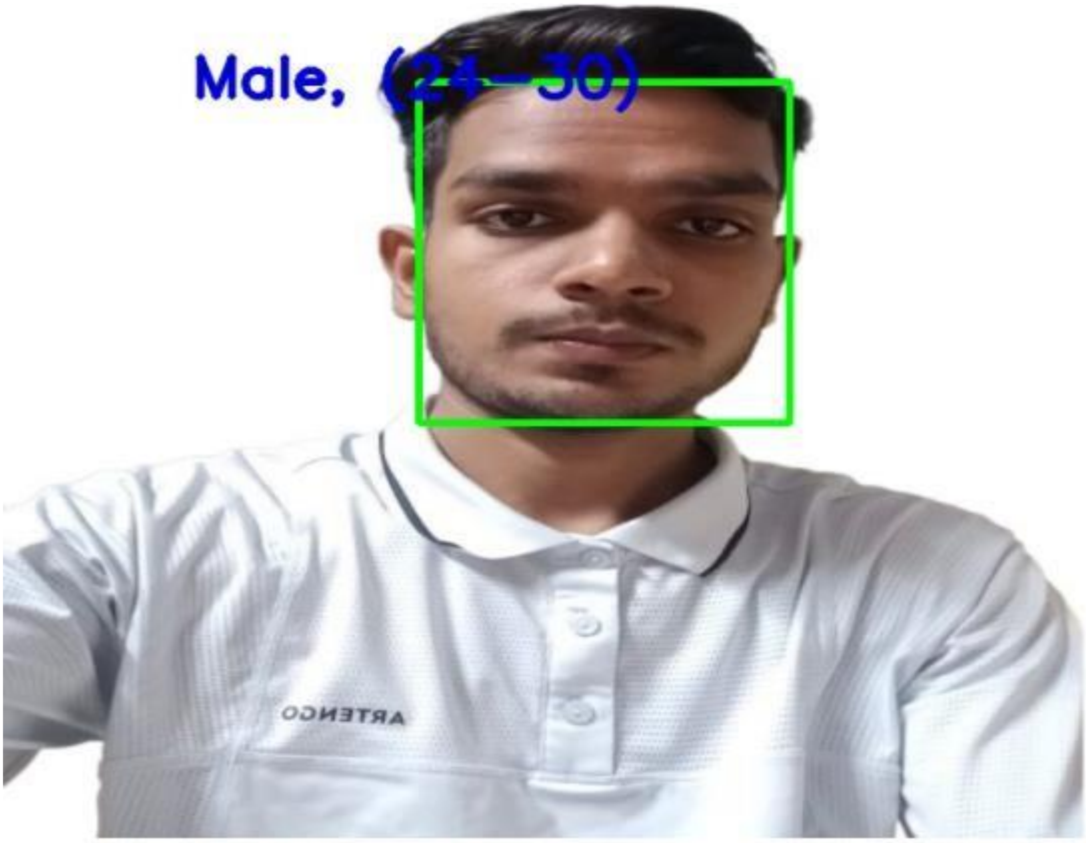

Fig 5: Predicted gender and age

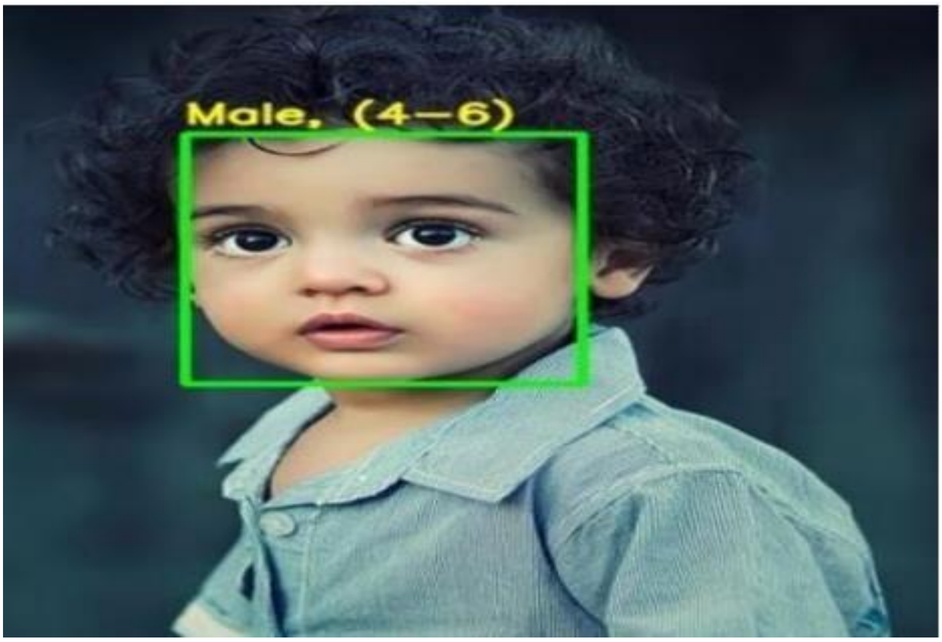

Fig 6: Predicted gender and age

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 09 | Sept - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

5. Conclusion

In this paper, we have introduced a model for gender and age classification that leverages multiple sub-CNNs and other machine learning techniques. Each sub-CNN operates independently, and their outputs are integrated using a voting mechanism. The rationale for employing different sub-CNNs separately, followed by a voting approach, is to capture a diverse set of facial feature representations. This strategy enhances the accuracy of age prediction by providing a more comprehensive analysis of facial attributes. Our experiments demonstrate that utilizing multiple CNN models results in a lower error rate compared to using a single CNN model. The methodology incorporates grouped approaches and computations, with deep learning as the primary component in the model's design.

6. References

1. El-Alfy, E.-S., & Binsaadoon, A. G. (2019). Implementation of fuzzy local binary patterns with optimized parameters for automated gait-based gender identification. *Journal of Ambient Intelligence and Humanized Computing*, 10(7), 2495-2504.

2. Yu, Y., Xiong, Y., Huang, W., & Scott, M. R. (2020). Deformable Siamese attention networks applied to visual object tracking. Presented at the *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6728-6737.

3. Liu, N., Zhang, N., & Han, J. (2020). Utilizing selective self-mutual attention for enhanced RGB-D saliency detection. In the *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 13756-13765.

4. Zhang, K., et al. (2017). Estimating age group and gender in the wild using a deep RoR architecture. *IEEE Access*, 5, 22492-22503.

5. Ranjan, R., Patel, V. M., & Chellappa, R. (2019). HyperFace: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(1), 121-135.

6. Althnian, A., Aloboud, N., Alkharashi, N., Alduwaish, F., Alrshoud, M., & Kurdi, H. (2021). An extensive performance comparison of deep-learned, hand-crafted, and fused features with deep and traditional models for face gender recognition in the wild. *Applied Sciences*, 11(1), 89.

7. Ge, H., Dong, J., & Zhang, L. (2020). Recognition of face attributes based on a one-way inferential correlation between attributes. In the *Proceedings of the International Conference on Multimedia Modeling*, 253-265.

8. Age and Gender Prediction using CNN. (n.d.). Retrieved from GeeksforGeeks: https://www.geeksforgeeks.org/age-and-gender-prediction-using-cnn/