Final Assignment

Facial Recognition

Gang Wang

Yunjie Xiong

Zhenyu Zhang

EAI 6080 - Advanced Analytical Utilization

Course Number (CRN): 21914

Instructor: Dr. Mimoza Dimodugno

October 23, 2024

Introduction

Facial recognition is one of the most popular technologies nowadays because it is important to almost every part in our life. It covers our life from security problems like search for thief to use it as Entertainment tools.

The technology we have chosen for this project is CNN, the reason is that CNN is one of the greatest models to deal with image recognition.

In our project, we do this project just for Entertainment tool. In detail, what we are going to predict is age, gender and race, thus, when our project receives a facial picture, it can identify these characters. It is a useful tool in social communication as it shortens the distance between strangers.

In the data preprocessing part, let have a look at what the picture in our dataset looks like.

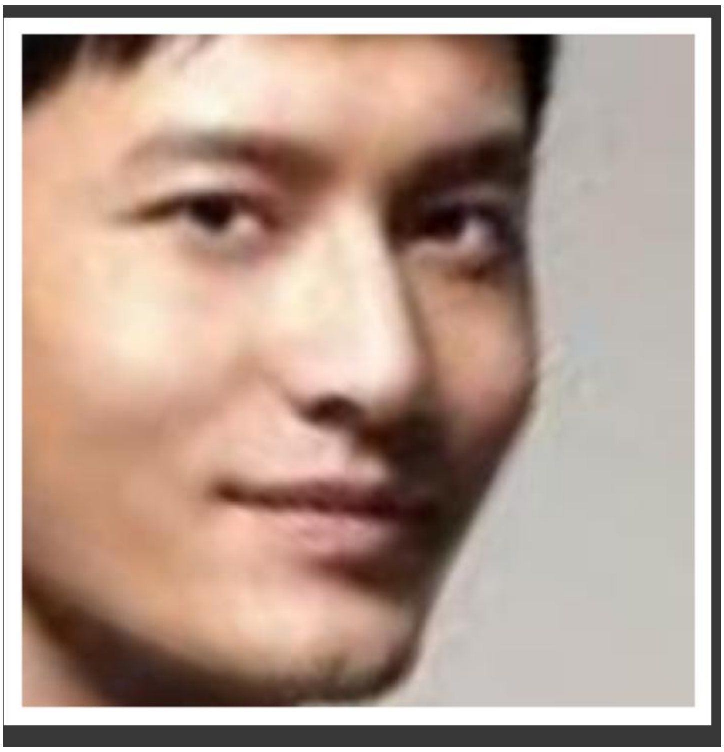

Figure 1.1

The next thing we did is translate the name of files to the labels for each picture, and then build a DataFrame for them. The following is the figure of the head of our basic df.

	image	age	gender	race
0	26_0_1_20170116221037571.jpg.chip.jpg	26	0	1
1	7_0_0_20161219201514284.jpg.chip.jpg	7	0	0
2	6_0_4_20161221200807209.jpg.chip.jpg	6	0	4
3	7_0_0_20170110215327391.jpg.chip.jpg	7	0	0
4	26_0_1_20170116220224657.jpg.chip.jpg	26	0	1

Figure 1.2

Also, with df.shape, we can see that there are 23705 pictures in our dataset. There is no doubt that it is big enough for CNN to use. Now, let go deeper into age, gender, and race which are the three characters we are going to predict. And let's first see unique value for each character to see whether there are many values that need to be clean.

```
Unique ages: ['1' '10' '100' '101' '103' '105' '11' '110' '111' '115' '116' '12' '13'
 '14' '15' '16' '17' '18' '19' '2' '20' '21' '22' '23' '24' '25' '26' '27'
 '28' '29' '3' '30' '31' '32' '33' '34' '35' '36' '37' '38' '39' '4' '40'
 '41' '42' '43' '44' '45' '46' '47' '48' '49' '5' '50' '51' '52' '53' '54'
 '55' '56' '57' '58' '59' '6' '60' '61' '62' '63' '64' '65' '66' '67' '68'
 '69' '7' '70' '71' '72' '73' '74' '75' '76' '77' '78' '79' '8' '80' '81'
 '82' '83' '84' '85' '86' '87' '88' '89' '9' '90' '91' '92' '93' '95' '96'
 '99']
Unique genders: ['0' '1']
Unique races: ['0' '2' '3' '1' '4' '20170116174525125.jpg.chip.jpg'
 '20170109142408075.jpg.chip.jpg' '20170109150557335.jpg.chip.jpg']
```

Figure 1.3

As there are some image paths mixed into the races, we need to clean them.

Next, let's check the value count of each character to see that whether this dataset is balance or not.

```
gender
0    12391
1    11314
Name: count, dtype: int64
```

Figure 1.4.1

```
race
0    10078
1     4526
3     3975
2     3434
4     1692
Name: count, dtype: int64
```

Figure 1.4.2

```
age
26     2197
1      1123
28      918
35      880
24      859
       ...
115       3
101       2
91        2
111       1
103       1
Name: count, Length: 104, dtype: int64
```

Figure 1.4.3

With the figures above, we can see that there is no bias problem in gender, and the race has no small bias problem which we might need to consider. To solve this problem, we have chosen to cut the dataset to make it balance. And the condition we choose is cut the picture in age group that has picture over 500 until it is 500. And the shape of the picture also reduce to 19385, so we can see that we have lost almost 4000 picture.

The next step is to do data cleaning and augmentation. The first thing we used is OpenCV in cuda, we use it to do alignment and tailoring, and then increase the light of picture. In the part of alignment and tailoring, we use the distance between two eyes as the condition to cut the picture. Then, we use the clahe and gamma to increase the light of picture, the clahe use Histogram Equalization to increase the tiles of picture. And I set the value of Gamma to 1.2 which is increases the light a little.

After increasing the quality of data, I also do data augmentation, that is rotate the picture, increasing or decreasing the size of the picture. This can increase the size of our data set to prevent the problem of overfitting. And the final step of preprocessing is changing the picture to numbers that use the RGB of each picture and translate into matrix, and finally storage these matrix in NumPy series. Now, the data is enough for traditional models and CNN model.

EDA

First of all, in the old dataset, we can see that the age group 1 and 26 are extremely high which we need to cut some of them to make the dataset balanced. The Histogram and pie chart below also prove this problem.

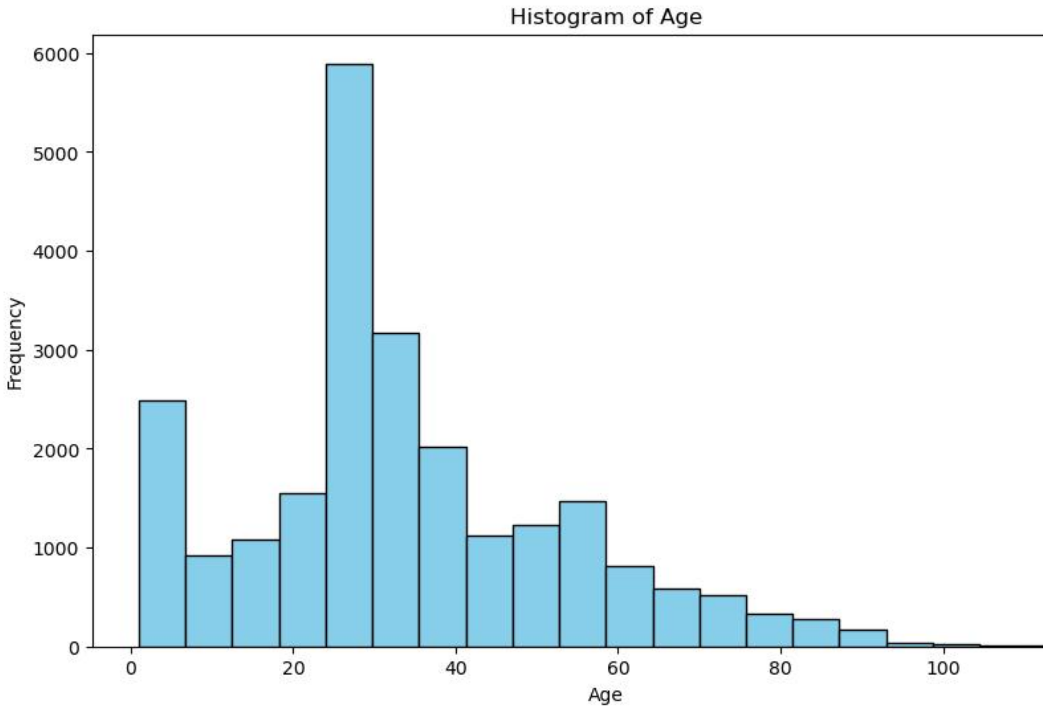
Histogram of Age
Frequency
Age
0
1000
2000
3000
4000
5000
6000
0
20
40
60
80
100

Figure 1.5.1

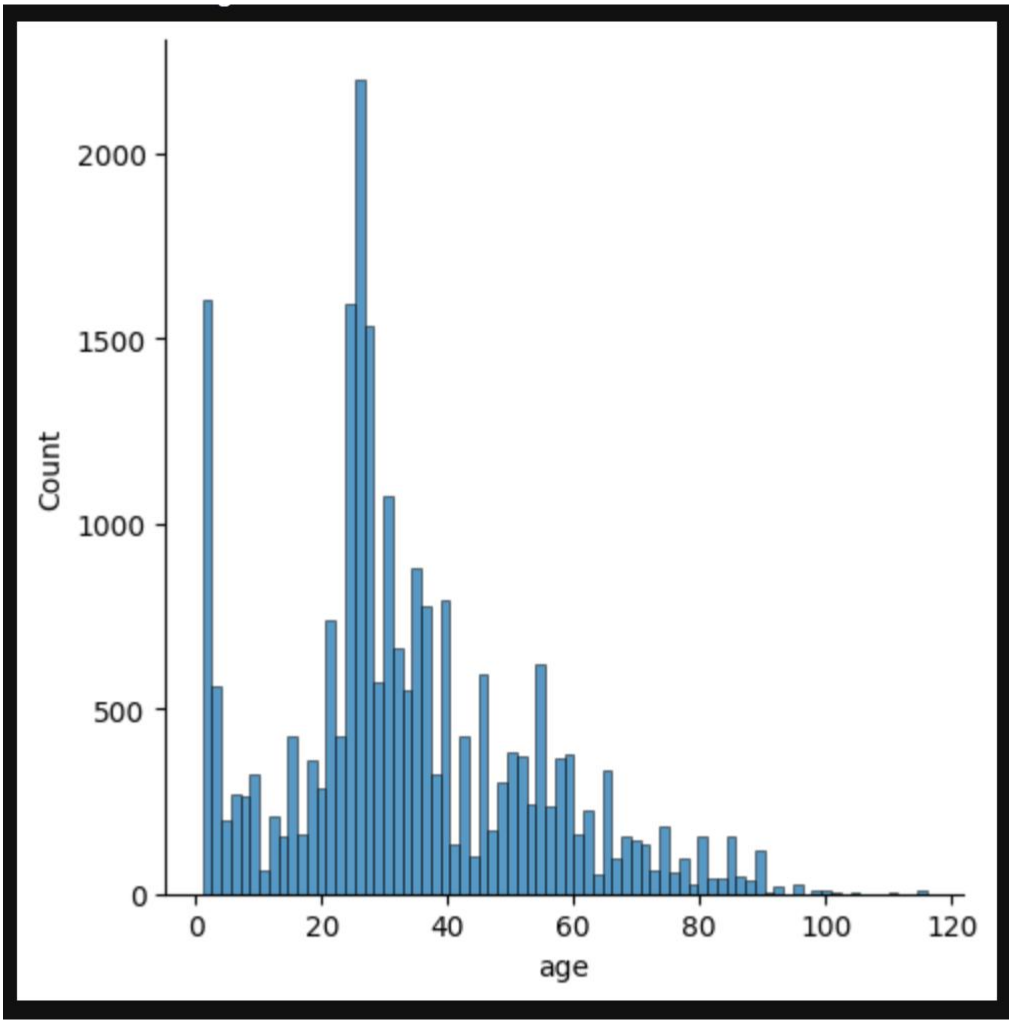
Count
age
0
500
1000
1500
2000
0
20
40
60
80
100
120

Figure 1.5.2

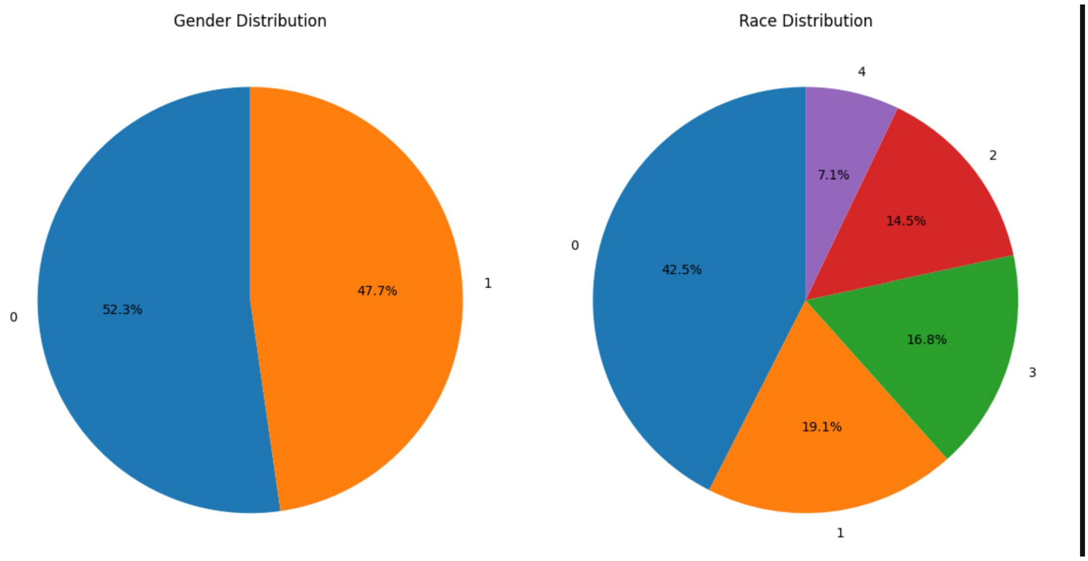

Figure 1.5.3

And follow is the Histogram of age after cutting the picture we don’t want which is new dataset.

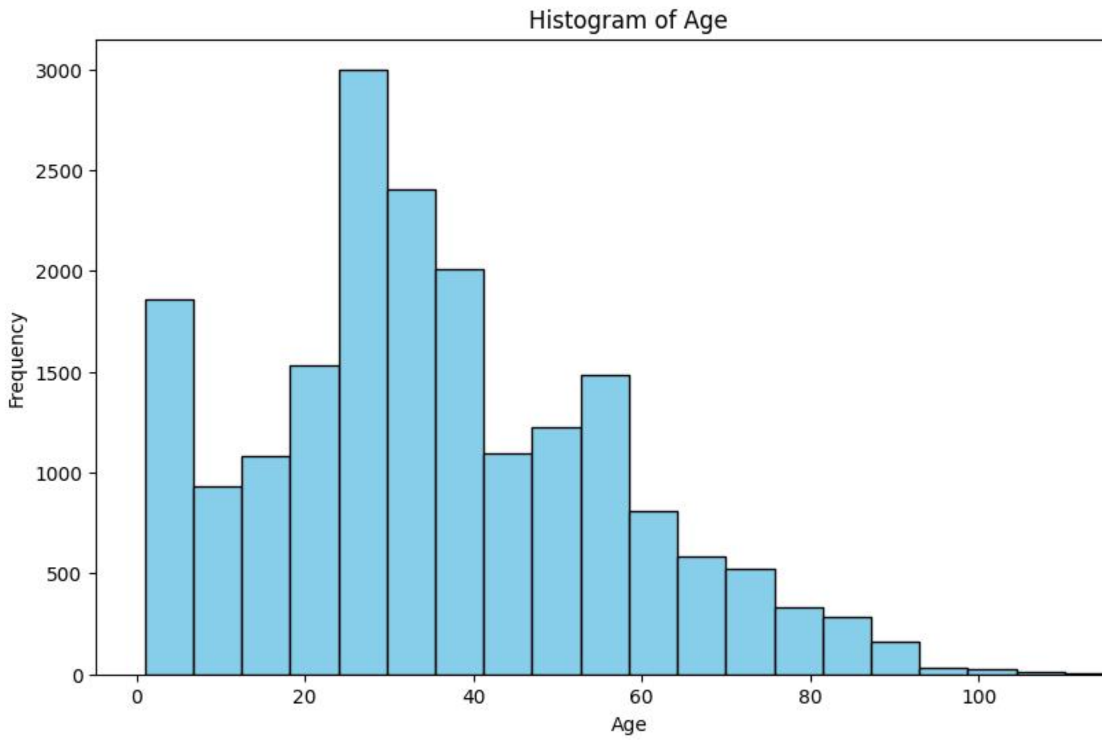

Figure 1.6.1

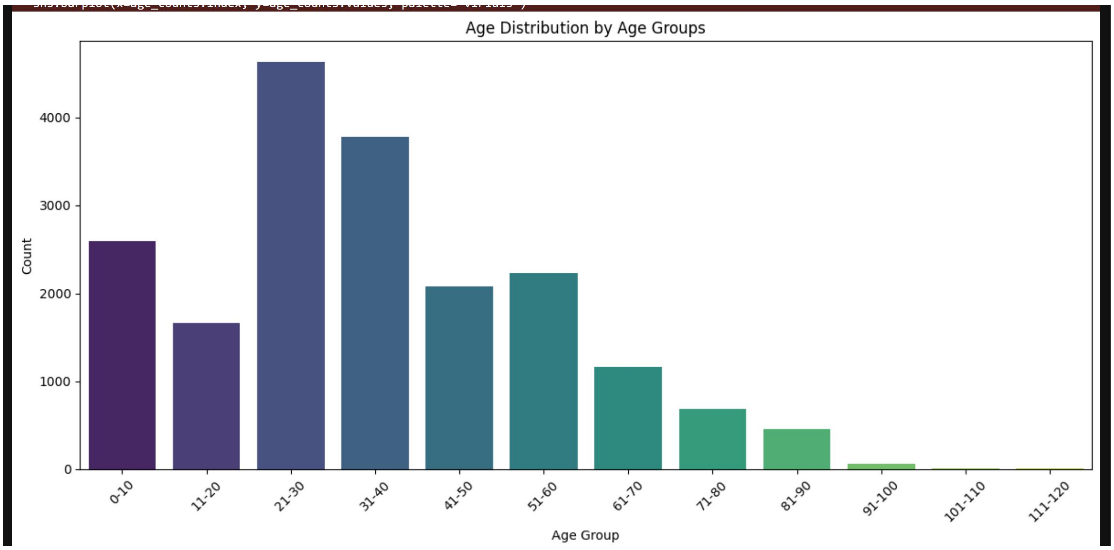

Figure 1.6.2

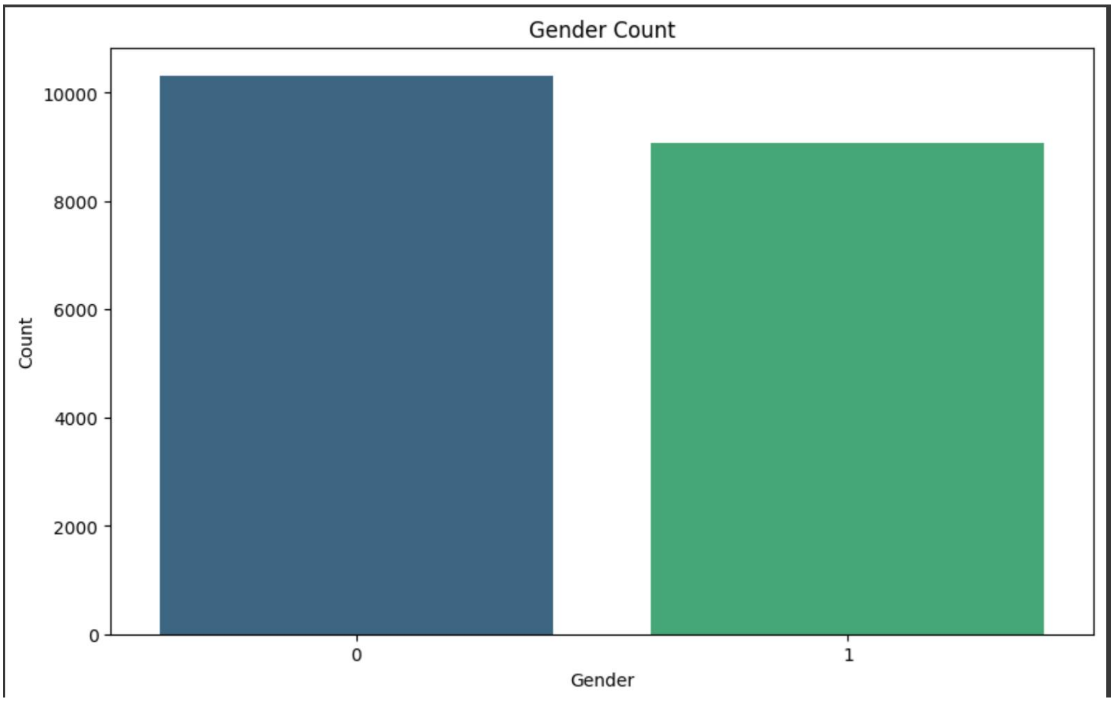

Figure 1.6.3

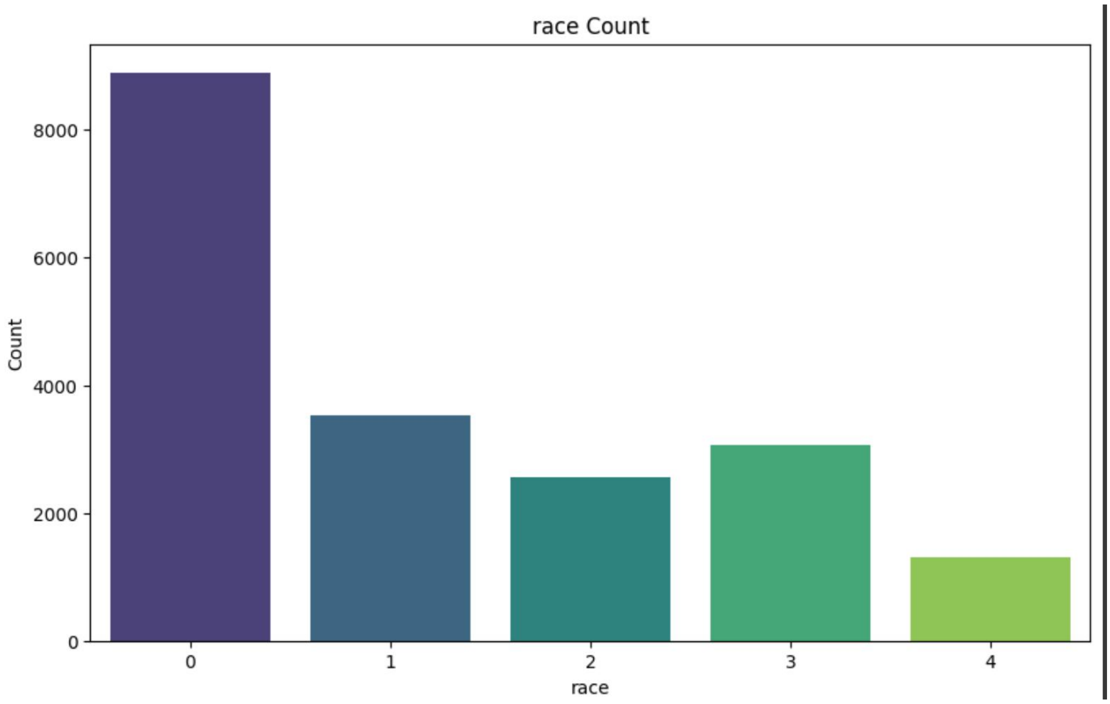

Figure 1.6.4

We can clearly see that the problem of unbalanced data problem show in old dataset has been solved.

Body

In body, we build four models for our dataset, that are SVM model, Random Forest, un-Optimized CNN model and Optimized CNN model.

Traditional Model

First, let's talk about Two Classic Non-linear Models: Discuss the effects and results of classic non-linear models .

1. Model Introduction

Support Vector Machine (SVM)

The SVM is a supervised learning model commonly used for classification tasks. It works by finding a hyperplane that maximally separates different classes of data. In this code, SVM is applied to two classification tasks: age prediction and race prediction. The Radial Basis Function (RBF) kernel is used, which enhances the model's ability to handle nonlinear classification. The gamma='scale' parameter automatically adjusts the weight of the RBF kernel, and C=1 controls the penalty for misclassification, balancing the trade-off between accuracy and generalization.

Random Forest

Random Forest is an ensemble learning method that builds multiple decision trees and aggregates their results for classification. This model reduces overfitting by selecting random subsets of both data samples and features. In this code, Random Forest is applied to the same age and race classification tasks. It uses 100 trees (n_estimators=100), which usually provides a stable classification performance.

2. Feature Extraction and Data Splitting

Before training the SVM and Random Forest models, the Histogram of Oriented Gradients (HOG) algorithm is used to extract features from images. Each image is resized to 64x64 pixels, and the HOG descriptor captures the image's features for classification. The extracted features are stored in X, with the corresponding labels being y_age (age labels) and y_race (race labels).

The dataset is split into training and testing sets, with 80% used for training and 20% for testing. This ensures consistency and randomness in both sets.

3. Model Training and Prediction

SVM Model:

- **Age Prediction:** The SVM model is trained on the training set for age classification and then used to predict on the test set. Results are displayed using classification_report and accuracy_score.

- **Race Prediction:** Similarly, the SVM model is applied to the race classification task, following the same process of training and testing.

Random Forest Model:

- **Age Prediction:** The Random Forest model uses 100 decision trees to train on the age classification task and predict on the test set.
- **Race Prediction:** Likewise, the Random Forest model is used for race classification, following the same process of training and prediction.

4. Results Analysis

SVM Model Results:

- The classification report for age prediction shows precision, recall, and F1 scores for each class. The overall accuracy is calculated using accuracy_score.
- The race prediction results are similarly reported with a classification report and accuracy.

Random Forest Model Results:

- The results for both age and race classification are displayed using classification reports and accuracy scores.

Before data cleaning, both models don’t perform well, here is their accuracy:

```
Training SVM model for age prediction...
Training SVM model for race prediction...
SVM Model Results for Age Prediction:
              precision    recall  f1-score   support

           1       0.45      0.92      0.61       212
           2       0.38      0.03      0.05       103
           3       0.50      0.02      0.03        58
           4       0.33      0.04      0.06        57
           5       0.00      0.00      0.00        37
           6       0.00      0.00      0.00        23
           7       0.00      0.00      0.00        25
           8       0.16      0.09      0.12        53
           9       0.00      0.00      0.00        25
          10       0.00      0.00      0.00        30
          11       0.00      0.00      0.00        13
          12       0.00      0.00      0.00        24
          13       0.00      0.00      0.00        13
          14       0.00      0.00      0.00        33
          15       0.00      0.00      0.00        35
          16       0.00      0.00      0.00        41
          17       0.00      0.00      0.00        35
          18       0.00      0.00      0.00        46
          19       0.00      0.00      0.00        13
          20       0.00      0.00      0.00        45
...
   macro avg       0.53      0.51      0.51      4741
weighted avg       0.61      0.66      0.62      4741

Accuracy: 0.657878084792238
```

```
Training Random Forest model for age prediction...
Training Random Forest model for race prediction...
Random Forest Model Results for Age Prediction:
              precision    recall  f1-score   support

           1       0.30      0.86      0.45       212
           2       0.17      0.04      0.06       103
           3       0.00      0.00      0.00        58
           4       1.00      0.02      0.03        57
           5       0.00      0.00      0.00        37
           6       0.00      0.00      0.00        23
           7       0.00      0.00      0.00        25
           8       0.25      0.02      0.04        53
           9       0.00      0.00      0.00        25
          10       0.00      0.00      0.00        30
          11       0.00      0.00      0.00        13
          12       0.00      0.00      0.00        24
          13       0.00      0.00      0.00        13
          14       0.00      0.00      0.00        33
          15       0.00      0.00      0.00        35
          16       0.00      0.00      0.00        41
          17       0.00      0.00      0.00        35
          18       0.00      0.00      0.00        46
          19       0.00      0.00      0.00        13
          20       0.00      0.00      0.00        45
...
   macro avg       0.47      0.31      0.28      4741
weighted avg       0.52      0.50      0.41      4741

Accuracy: 0.5036912043872601
```

The accuracy of the SVM is 65.79%, and the Random Forest is 50.37%.

After data cleaning, the performance is much better:

```
Training SVM model for age prediction...
Training SVM model for race prediction...
SVM Model Results for Age Prediction:
              precision    recall  f1-score   support

           1       0.51      0.93      0.66       215
          10       0.33      0.03      0.06        30
         100       0.00      0.00      0.00         3
         101       0.00      0.00      0.00         1
         105       0.00      0.00      0.00         1
          11       0.00      0.00      0.00        14
         110       0.00      0.00      0.00         4
         116       0.00      0.00      0.00         2
          12       0.00      0.00      0.00        27
          13       0.00      0.00      0.00        16
          14       0.00      0.00      0.00        28
          15       0.00      0.00      0.00        37
          16       0.00      0.00      0.00        53
          17       0.00      0.00      0.00        24
          18       0.00      0.00      0.00        55
          19       0.00      0.00      0.00        23
           2       0.15      0.05      0.08        99
          20       0.00      0.00      0.00        56
          21       0.00      0.00      0.00        67
          22       0.00      0.00      0.00        74
...
   macro avg       0.79      0.58      0.58      4742
weighted avg       0.74      0.71      0.68      4742

Accuracy: 0.714888232813159
```

```
Training Random Forest model for age prediction...
Training Random Forest model for race prediction...
Random Forest Model Results for Age Prediction:
              precision    recall  f1-score   support

           1       0.39      0.92      0.55       230
          10       0.33      0.03      0.05        35
         100       0.00      0.00      0.00         4
         101       0.00      0.00      0.00         1
         103       0.00      0.00      0.00         1
         105       0.00      0.00      0.00         2
          11       0.00      0.00      0.00         6
         110       0.00      0.00      0.00         2
         116       0.00      0.00      0.00         1
          12       0.00      0.00      0.00        24
          13       0.00      0.00      0.00        12
          14       1.00      0.02      0.05        41
          15       0.20      0.03      0.06        31
          16       0.00      0.00      0.00        45
          17       0.00      0.00      0.00        31
          18       0.00      0.00      0.00        31
          19       0.00      0.00      0.00        21
           2       0.09      0.04      0.05        77
          20       0.00      0.00      0.00        51
          21       0.11      0.02      0.03        59
...
   macro avg       0.69      0.41      0.43      4742
weighted avg       0.65      0.58      0.53      4742

Accuracy: 0.5822437789962042
```

Figure 2.1

Figure 2.2

The accuracy of the SVM is 71.49%, and the Random Forest is 58.22%.

5. Model Comparison

- **SVM Model:** SVM performs well when handling high-dimensional data, especially for nonlinear classification problems. In age and race classification, SVM may have an edge in terms of accuracy and generalization due to its effective hyperplane selection.

- **Random Forest Model:** Random Forest excels at handling noisy data and preventing overfitting. By combining the results of multiple decision trees, Random Forest often provides stable prediction results, particularly when dealing with large datasets.

In conclusion, the SVM model demonstrates its effectiveness in this task by successfully handling the classification challenges, particularly in distinguishing between different age groups and racial categories.

CNN Model

Now, let's talk about the CNN model which is the main part of our model. To start with, let's talk about the un-Optimize CNN model. Initial CNN Model: Present the outcomes of the initial CNN model. And the next is Optimized CNN Model and Result Prediction: Describe the integrated model using a voting mechanism and its outcomes.

Part 1 CNN model construction:

The CNN model consists of multiple convolution layers and dense layers. The output predicts three different tasks: gender (binary classification), age (regression), and race (multiclass classification).

• Model training:

The model was trained with 10 epoches using the Adam optimizer and the loss of different outputs was monitored.

Significance:

This section describes how to create a CNN model to process image data for tasks such as gender, age, and race predictions. It emphasizes proper preprocessing and data processing, which are critical to model performance.

Part 2: Model optimization

• Increase model complexity:

The CNN architecture gets deeper by adding an additional convolutional layer with more filters. This is to improve the feature extraction capability of the network.

• Hyperparameter tuning:

The ReduceLROnPlateau callback is used to dynamically adjust the learning rate, which helps the model converge better by reducing the learning rate when performance stagnates.

• Assessment:

For more eras (50 in this case), the model was trained again, and performance was improved by monitoring the loss and accuracy of various metrics such as gender, age, and race predictions.

Significance:

This section focuses on the process of optimizing the CNN model, including increasing the capacity of the model and fine-tuning the hyperparameters. The goal is to improve the model's ability to generalize and make more accurate predictions.

Part three: comprehensive model and evaluation

• Build and assemble models:

Create a different CNN architecture for each task (gender, age, race). The final solution involved training three independent models (CNN, Random Forest, SVM) and comparing their performance on each prediction task.

• Class weights for unbalanced data:

The use of class weights in training to account for class imbalances ensures that the model does not favor more frequent classes.

• Evaluation indicators:

Assessments included accuracy scores for classification reports and gender and race predictions, as well as mean absolute error (MAE) for age predictions.

Significance:

This section explores the combination of different models to handle individual tasks and to solve the problem of data imbalance through the use of class weights. It shows how different architectures (CNN, Random Forest, SVM) can be integrated and compared to select the best performing model for each task.

Model optimization process:

Model optimization includes:

1. Enhanced model architecture: By adding more layers and filters, the network becomes more capable of learning complex features.

2. Dynamic learning rate adjustment: Using callbacks such as ReduceLROnPlateau helps to avoid overshooting the minimum during training, resulting in better convergence.

3. Balance class distribution: Handle unbalanced data sets by applying class weights to ensure that all classes are treated equally during training.

4. Evaluate different models: Train multiple models for each task and compare their performance to select the most suitable model.

Results Presentation

Figure 3.1.1

Figure 3.1.2

Figure 3.1.3

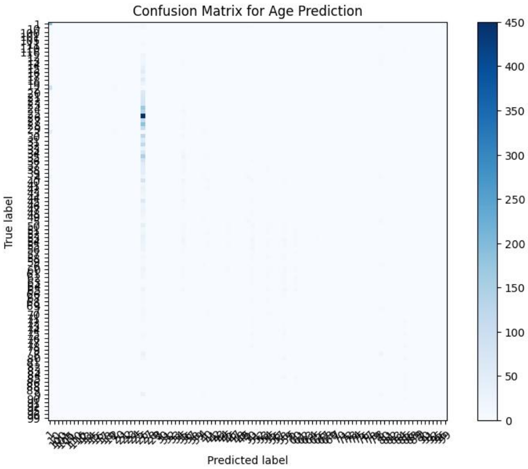

Figure 3.1.1

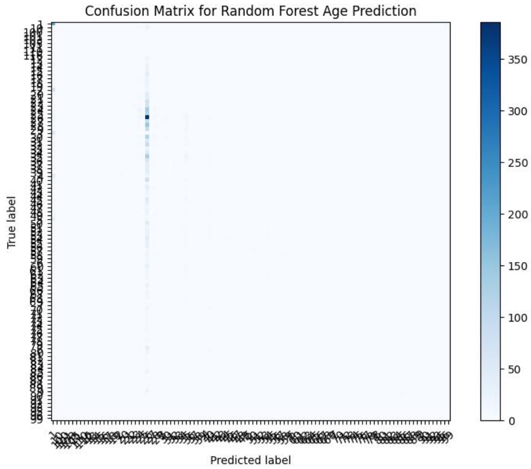

Figure 3.1.2

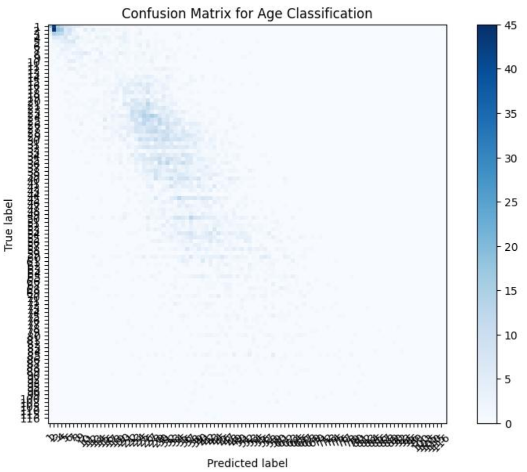

Figure 3.1.3

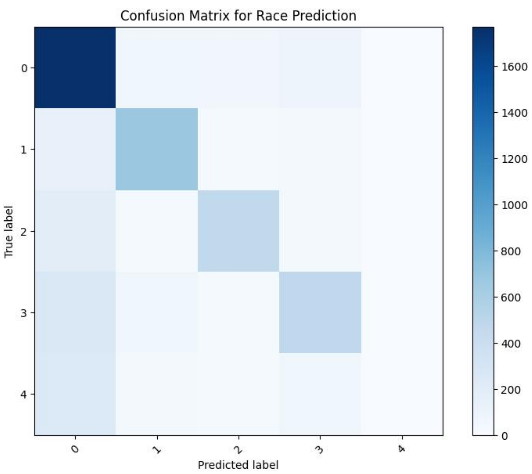

Figure 3.2.1

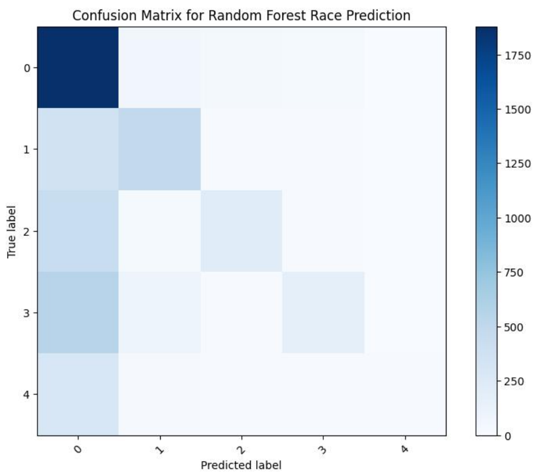

Figure3.2.2

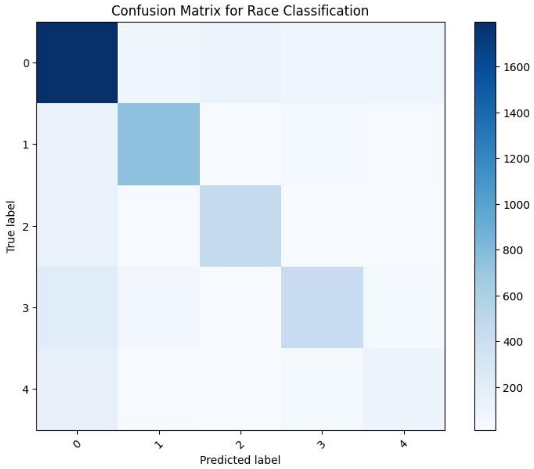

Figure 3.2.3

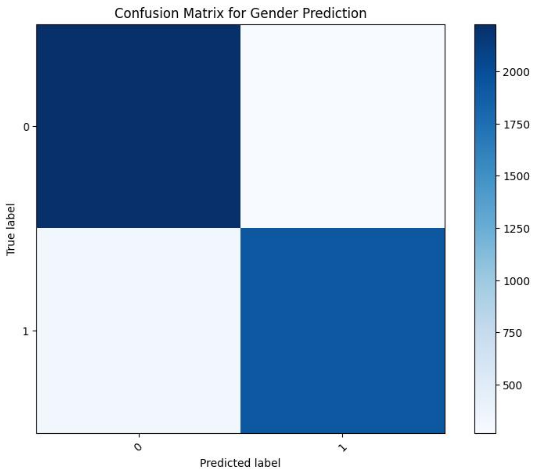

Figure 3.3.1

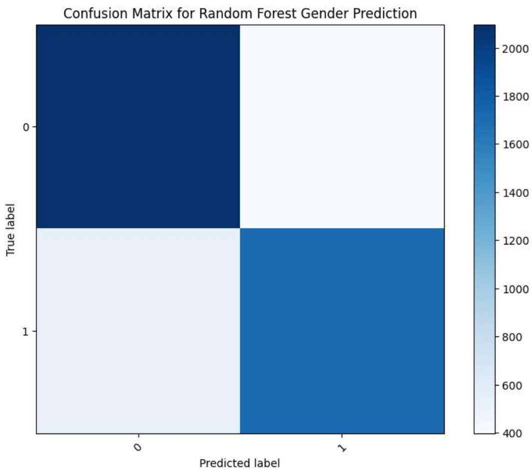

Figure 3.3.2

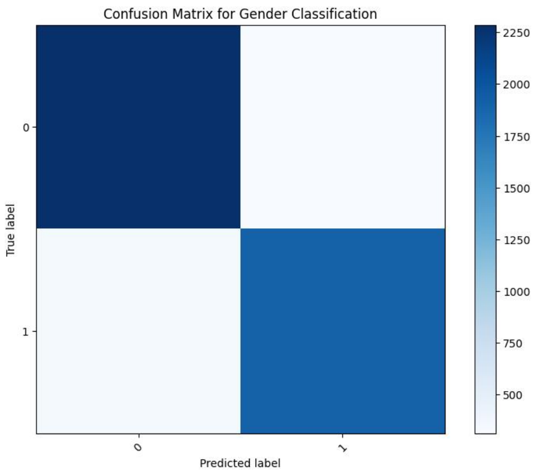

Figure 3.3.3

The performance of the age and race prediction models appears to be quite consistent across a certain range, which may suggest that they are all achieving similar results. This could be due to the characteristics of the dataset itself, which may not be diverse or high-quality enough to allow for clear differentiation between the models' capabilities. As a result, the models tend to perform similarly well in a limited age range, but their overall effectiveness could be constrained by the limitations of the dataset.

Conclusion

In summary, our project builds a CNN model to do facial recognition to predict age, gender, and race based on give facial picture. This project can be used in many areas like the carme in store can have one to do better recognition. Or it can be a function in some social media platform that people can take a photo of each other can guess their age or race which is good way to short the distance in social communication.

In our project, we improve our model's performance step by step by doing, traditional model, un–Optimized CNN model and Optimized CNN model. However, there are more things that can be done like we can even cut some pictures of white because it is too large compared with others. Mybe we can do better data cleaning as the cleaning now does improve performance a lot, or we can do more work in CNN model to improve the performance like we can do classification for group that has lower accuracy.

References

Bhattacharyya, S. (2011). A brief survey of color image preprocessing and segmentation techniques. Journal of Pattern Recognition Research, 1(1), 120-129.

Boughorbel, S., Tarel, J. P., & Boujemaa, N. (2005, July). Conditionally positive definite kernels for svm based image recognition. In 2005 IEEE International Conference on Multimedia and Expo (pp. 113-116). IEEE. https://doi.org/10.1109/ICME.2005.1521373

Chauhan, R., Ghanshala, K. K., & Joshi, R. C. (2018, December). Convolutional neural network (CNN) for image detection and recognition. In 2018 first international conference on secure cyber computing and communication (ICSCCC) (pp. 278-282). IEEE. https://doi.org/10.1142/S0218127497001618

Kaur, P., Krishan, K., Sharma, S. K., & Kanchan, T. (2020). Facial-recognition algorithms: A literature review. Medicine, Science and the Law, 60(2), 131-139. https://doi.org/10.1177/0025802419893168

Kremic, E., & Subasi, A. (2016). Performance of random forest and SVM in face recognition. Int. Arab J. Inf. Technol., 13(2), 287-293.

Liu, Y., Wang, Y., Yu, R., Li, M., Sharma, V., & Wang, Y. (2019). Optimizing {CNN} model inference on {CPUs}. In 2019 USENIX Annual Technical Conference (USENIX ATC 19) (pp. 1025-1040).

Poojary, R., & Pai, A. (2019, November). Comparative study of model optimization techniques in fine-tuned CNN models. In 2019 International Conference on Electrical and Computing Technologies and Applications (ICECTA) (pp. 1-4). IEEE.

http://doi.org/10.1109/ICECTA48151.2019.8959681.