Predict Age, Gender and Ethnicity (December 2020)

Debsankar Mukhopadhyay, Vasim Saikh, Pravanjan Trivedi, William Yerkes

***Abstract*—A simplified version of the UTKFace dataset have been used to determine gender, ethnicity, and age of the images based on a multi-output Convolutional Neural Network Model. An attempt has also been made to identify the facial expressions using OpenCV Haar Cascade models. Our CNN model runs successfully and exhibits an ability to identify the features from the images within reasonable accuracy and error metrics.**

***Index Terms*— CNN, Image Analysis, Gender recognition, Age estimation, Ethnicity determination, Expression determination.**

I. Introduction

THE human face contains a variety and veracity of semantic information about a person. The prediction of identity, age, gender, ethnicity, and expression constitutes the discipline of facial biometrics that is crucial for business campaigns, security analysis, and academic interests.

Artificial neural networks offer an efficient solution to detect and classify images of different types. Convolutional Neural Network (CNN) is a special class of artificial neural network that dominates analyzing and resolving computer vision tasks. CNN provides a framework to automatically and adaptively learn spatial features in multiple layers. This network provides us with a deep learning model to process grided data, such as, the images. CNN is based on a mathematical structure describing three types of layers or building blocks: convolution, pooling, and fully connected layers. The features inherent in the images are extracted by convolutional and pooling layers. The final output is reconstructed from the extracted features in the pooling layers [1]. In a digital image, the pixel values are stored in a two-dimensional grid, and a small grid of parameter, called kernel, is applied at each image position to extract the features from them. The extracted features are passed from one layer to another to accomplish a progressively optimized pattern recognition. This process is called training. The training is performed to nullify the difference between outputs and ground truth labels using an optimization algorithm, like, backpropagation and gradient descent.

In this project, we shall implement a simplified CNN architecture to extract age, gender, ethnicity, and expression features from a set of the images of human faces. The business domain of our application is a retail clothing store where the images of the customers are extracted using the cameras at the entrance points. Those images can be analyzed with our proposed model to extract the facial biometric data of the clients. That data can, intern, be utilized by the business teams for optimizing the products or streamlining the advertisement campaigns for the business.

The reminder of this report is organized as follows: the background and related works are reviewed in Section II. The architecture and methodology are described in Section III. After that, the Section IV will concentrate on the data, results, and discussions. Conclusion and future outlook will be presented in Section V.

II. Background and Related Works

The structure of a CNN architecture is divided into multiple learning stages composed of the convolutional layers, non-linear processing units, and subsampling layers A typical block diagram of a Machine Learning (ML) system is shown in Fig 1. A comprehensive survey of the recent CNN architecture has been described by Khan et al [2] that includes a concise discussion on the basic CNN components. A typical CNN architecture is made up of alternate layers of convolution and pooling followed by one or more fully connected layers at the end. The CNN performance can be optimized by introducing mapping functions, batch normalization and dropout components. The pattern learning activities are aided by activation functions.

A comparative analysis of the age estimation techniques using deep learning was done by Othmani et al [3]. Their results demonstrated the high performance of the popular CNNs frameworks against the state-of-the-art methods of automated age estimation. A joint gender, ethnicity and age estimation from 3D dataset was performed by Xia et al [4] where the authors performed a morphology-driven analysis and emphasized on the correlation between these three demographic features. A hybrid neural network model obtained by mixing CNN and Extreme Machine Learning (ELM) architectures, implemented by Duan et al [5], exhibited about 90% accuracy in predicting age and gender from MORPH-II and Adience Benchmark datasets. The solutions of age, gender and/or race determination vary in different aspects on choosing an optimal CNN architecture and training strategy. CNN depth, pretraining, mono or multi-task training strategies, and the target age encoding and loss functions were carefully tuned by Antipov et al to accomplish as good as 100% gender and age prediction accuracies [6]. The facial expressions can also be

extracted from the images using the existing frameworks, such as, OpenCV, designed for computational efficiency on real-time applications [7]. In the realm of OpenCV framework, we can extract facial expressions using the Haar Cascades method. Haar Cascades are classifiers that are used to detect features (of face in this case) by superimposing predefined patterns over face segments and are used as XML files. Different pre-defined classifiers can be generated using training dataset that can be applied to the images under experiment for a facial mood or expression determination.

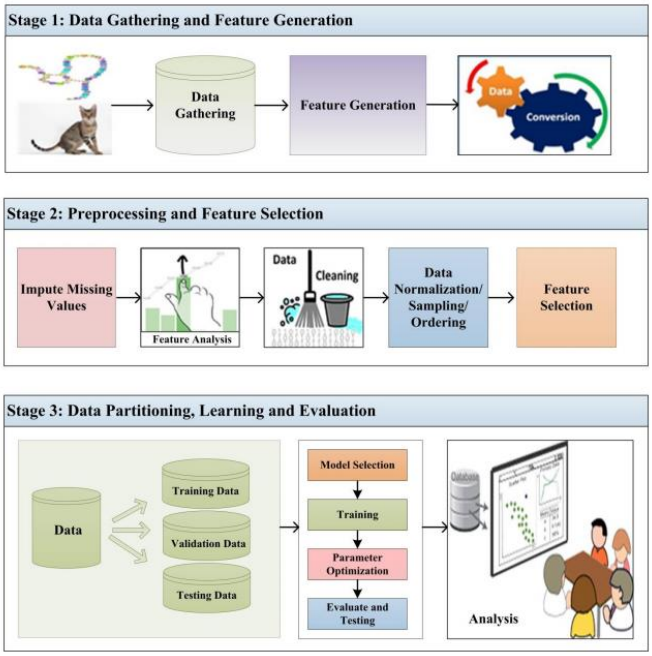

Figure 1: Layout of a multi-stage ML system.

III. ARCHITECTURE AND METHODOLOGY

Our task involves a simultaneous prediction of age, gender and ethnicity prediction from a labeled image dataset provided by a Kaggle Challenge [8] to predict age, gender and ethnicity from images. Besides, we also study a method of the expression estimation from the face images.

In the first study, we have involved our group members into two main directions. Those are (a) a CNN multi-output model to estimate the data features, i.e, the age, ethnicity and gender, and (b) define a mixed variable combining a scaled distribution of the age, gender, and ethnicity and then build the CNN architecture on that derived feature.

The mixed variable was constructed by scaling the gender and ethnicity variables by multiplying them by 1000 and 10,000, such that gender of 1 and ethnicity of 2 with age of 50 would be converted to 12050. This was attempted with various permutations, (gender, ethnicity, age,) (ethnicity, gender, age), (age, gender, ethnicity), (age, gender), (gender, ethnicity) etc. Each permutation which included age failed to produce accurate results, the gender and ethnicity permutation produced results which were not as accurate as the individual models for age and ethnicity on their own. Based upon these results we abandoned this model.

A comprehensive survey of multiple output learning was presented by Xu et al [9] where the multi-output architectures were reviewed from four fundamental point of views, namely, volume, velocity, variety, and veracity. It was noted that such a CNN model is often required to be tuned with a multi-variate loss function to address the variety issue of the data. A difference in the quality of the output is often found to be an issue with this approach associated with the data veracity. In our case, in particular, the inherent noises and biases in the dataset can affect the quality of our final predictions. However, within a realm of Keras architecture, it is straight forward to define separate loss functions with different weights for different outputs.

We defined three branches of our CNN model – age, gender, and ethnicity. In one of our studies, we have defined a defauklt structure of our convolutional layers based on a Conv2D layer with a RELU activation function, a BatchNormalization layer,a MaxPooling and a Dropout layer. For each branch, these default layers are then followed by a Dense layer. The architecture of the model is schematically shown in Fig 2.

Figure 2: Multiple-Output CNN architecture for age, ethnicity and gender prediction.

We have used a "Softmax" activation function for ethnicity output as it is expected to yield different classes in the data. The gender branch has a sigmoid activation function to emphasize the binary nature of the gender. The age branch is associated with a linear activation function.

The model, created in this architecture, is compiled with multi-variate loss functions: Mean Squared Error (MSE) for age, categorical_crossentropy for ethnicity, and binary_crossentropy for gender features. The choice of loss functions, again, emphasizes the respective natures of the features. We have assigned a set of loss-weights to each output branch. As a future tune up, we expect to study a variance of these loss functions and their weights as shown in the code snippet in Fig 3.

```
model.compile(optimizer=opt,
              loss={
                  'age_output': 'mse',
                  'ethnicity_output': 'categorical_crossentropy',
                  'gender_output': 'binary_crossentropy'},
              loss_weights={
                  'age_output': 4.,
                  'ethnicity_output': 1.5,
                  'gender_output': 0.1},
              metrics={
                  'age_output': 'mae',
                  'ethnicity_output': 'accuracy',
                  'gender_output': 'accuracy'})
```

Figure 3: Configuration of multi-variate loss functions.

The goal to select different loss weights is to account for different scales of values for different features in the data (variety) [10].

IV. Data, Results, and Discussions

Dataset: We have used a simplified version of the UTKFace Dataset which is a "large-scale face dataset with long age span" [11]. There are over 20000 images of the faces in the dataset with information on age, gender and ethnicity for each of the images in it. The Kaggle Challenge [8] that we used for our data mining, has simplified the data to include the age, gender, and ethnicity labels for each of the face images along with the image name and the pixel for each images stored in a column as a string. A brief snapshot of the data is shown in Fig 4.

	age	ethnicity	gender	img_name	pixels
0	1	2	0	20161219203650636.jpg.chip.jpg	129 128 128 126 127 130 133 135 139 142 145 14...
1	1	2	0	20161219222752047.jpg.chip.jpg	164 74 111 168 169 171 175 182 184 188 193 199...
2	1	2	0	20161219222832191.jpg.chip.jpg	67 70 71 70 69 67 70 79 90 103 116 132 145 155...
3	1	2	0	20161220144911423.jpg.chip.jpg	193 197 198 200 199 200 202 203 204 205 208 21...
4	1	2	0	20161220144914327.jpg.chip.jpg	202 205 209 210 209 209 210 211 212 214 218 21...

Figure 4: A snapshot of the dataset.

While the simplified data uses numeric labels for ethnicity and gender, we have retrieved the actual ethnicity and gender specification from the UTKFace descriptions as shown in Table 1-2.

Table 1: Gender descriptions

Gender Numeric ID	Gender Description
0	Male
1	Female

Table 2: Ethnicity Descriptions

Ethnicity Numeric ID	Description
0	White
1	Black
2	Asian
3	Indian
4	Others

We have explored the data by plotting the ethnicity, age, and gender distributions as Pie-Charts. The distributions are shown in Fig 5-7.

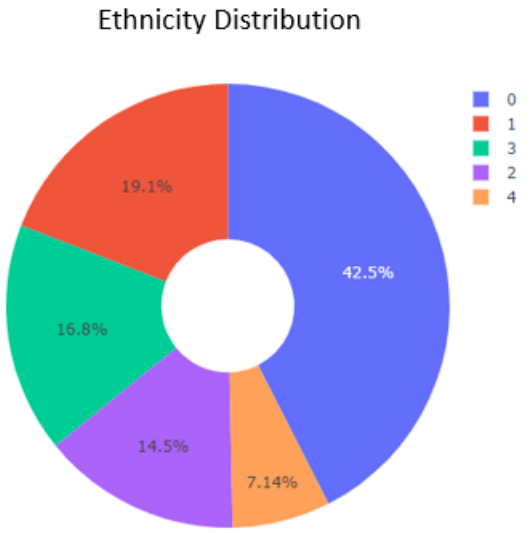

Figure 5: Ethnicity Distribution

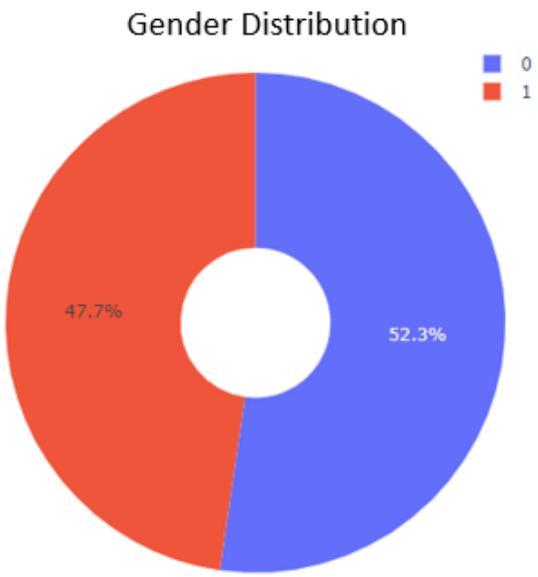

Figure 6: Gender distribution.

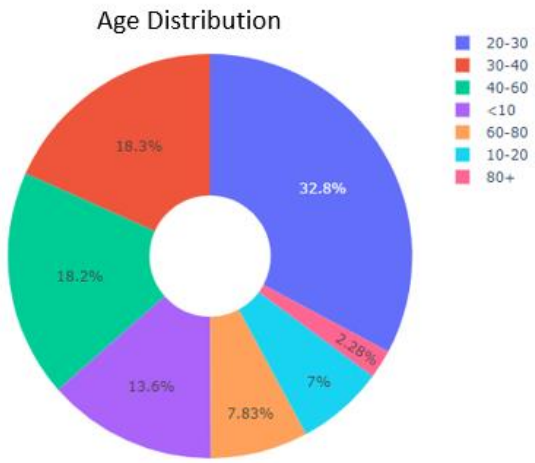

Figure 7: Age distribution.

The data suggests some inherent biases of disproportional ethnicity, age and genders as shown in the Figs. The disproportionality may affect with our statical efficiencies in training and/or testing the models. These biases should be factored in when we benchmark the prediction accuracy metrics.

Results and Discussions: In this project, we split our data into three ensembles, namely, train, validation, and test. The dataset used for training contains 70% of the data making our train_test_split to be 0.7. The test samples contain the remaining 30% of the data. The training dataset is further split in 70:30 proportions for the actual training data and validation datasets. As a next step, we have used our model described in the last section.

Following metrics are defined on our model for evaluation.

```
'loss',
'age_output_loss',
'ethnicity_output_loss',
'gender_output_loss',  'age_output_mae',
'ethnicity_output_accuracy',
'gender_output_accuracy',
'val_loss',
'val_age_output_loss',
'val_ethnicity_output_loss',
'val_gender_output_loss',
'val_age_output_mae',
'val_ethnicity_output_accuracy',
'val_gender_output_accuracy'
```

The keys prefixed with “val” denote the validation metrics keys while the ones without any “val” prefix are for training process.

The accuracy for ethnicity estimation is shown in Fig 8. The plot shows an improvement of accuracy to measure the ethnicity with increasing epochs for both training and validation model fits.

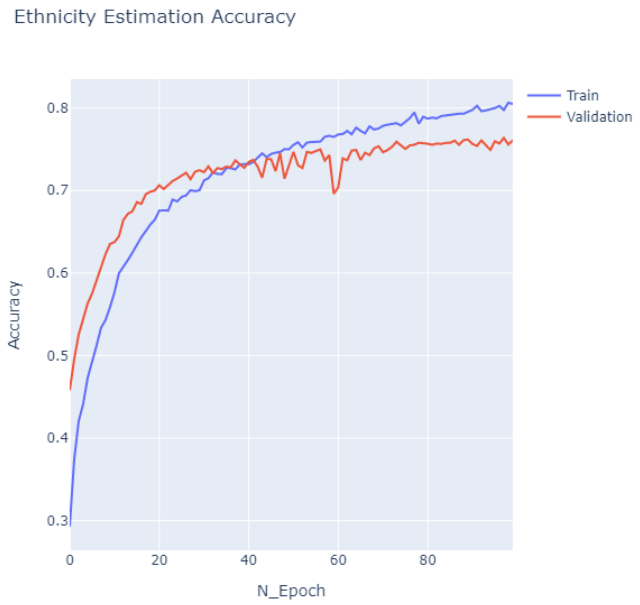

Figure 8: Ethnicity estimation accuracy.

The gender estimation accuracy is shown in Fig 9.

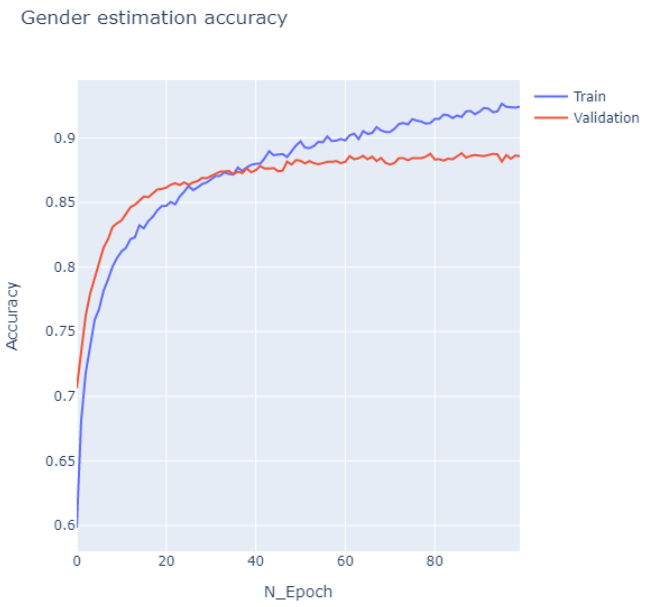

Figure 9: Gender estimation accuracy.

The gender estimation shows a similar trend as ethnicity as the accuracy increases with increasing epochs. Notably, however, the gender estimation accuracy is above or close to 90% in contrast with the same for the ethnicity, which was less than or equal to 80% for both training and validation sets.

The prediction of age is done by a regression method involving continuous variables. A Mean Absolute Error (MAE) should be a suitable metrics to exhibit the accuracy of the age measurement with a less MAE to reveal a better accuracy. This is shown in Fig 10 which clearly describes that our MAE for both training and validation datasets reaches to 0.1 for number of epochs more than 30.

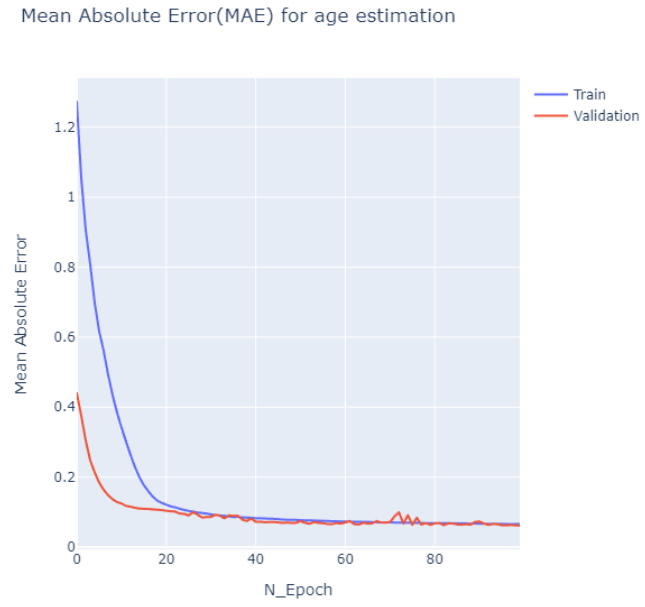

Figure 10: MAE as a function of number of epochs for measurement of the Age.

The data analysis also reveals an overall loss of about 10% or 1.0 at higher epochs number (Fig 11).

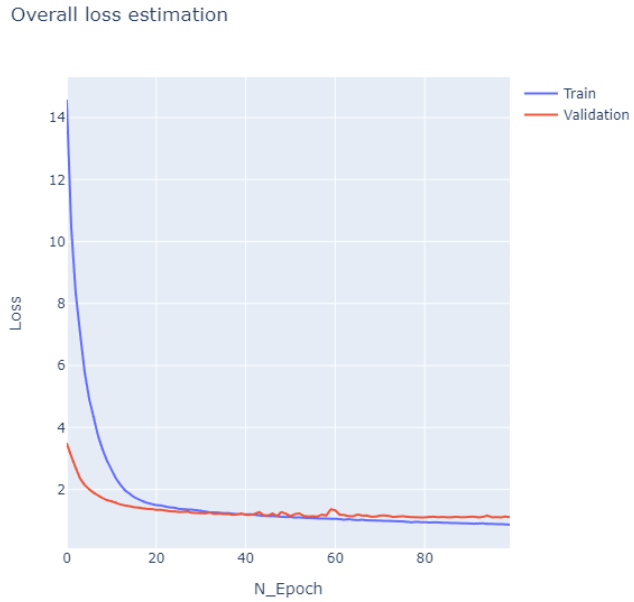

Figure 11: Overall losses.

We have evaluated our model based on the test dataset. Below is the metrics for our ethnicity prediction.
Figure 12

	precision	recall	f1-score	support
white	0.80	0.83	0.81	2996
black	0.79	0.84	0.82	1322
asian	0.73	0.80	0.76	981
indian	0.69	0.72	0.70	1206
others	0.37	0.12	0.18	535
accuracy			0.76	7040
macro avg	0.68	0.66	0.66	7040
weighted avg	0.74	0.76	0.74	7040

Figure 13: Metrics for the model evaluation for ethnicity branch.

Note that our model is not performing so well for the “others” ethnicity. That may be due to several factors, namely, the noises in the image and limitation of the statistics as shown in the Fig 5. The gender classification is, however, found to be very accurate as shown in the Fig 13.

	precision	recall	f1-score	support
0	0.90	0.88	0.89	3742
1	0.86	0.89	0.88	3298
accuracy			0.88	7040
macro avg	0.88	0.88	0.88	7040
weighted avg	0.88	0.88	0.88	7040

Figure 14: Gender estimation metrics.

Age, being a continuous variable, can not be described within the “classification_report”. We opted to measure mean squared

error (MSE), R2 score, explained variance score, max_error, and mean absolute error (MAE) to measure the accuracy for Age prediction as shown in Fig 14.

```
Age prediction r2 score:  0.7511268542443637
Age prediction Mean Squared Error  97.90986586385347
Age prediction Explained Variance Score  0.7523926872161244
Age prediction Max Error  68.80561065673828
Age prediction Mean Absolute Error  7.2897535849531945
```

Figure 15: Age Prediction metrics.

Finally, we ran our model to some random image samples within our dataset to test if our model can predict the age, gender, and ethnicity accurately. The images where we can predict the ethnicity and gender accurately and the age within a ±10 years of accuracy are marked by a green label in the bottom.

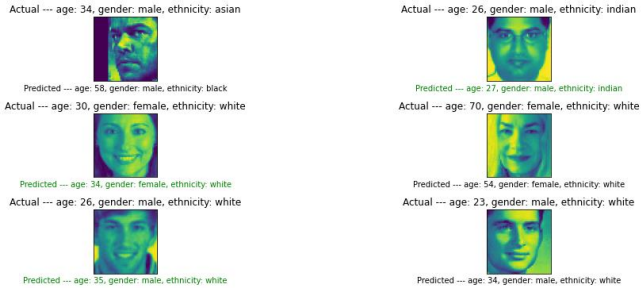

Figure 16: Application of the model on random image samples.

Fig 15 shows predicted and actual values of the features for the images.

The age is determined to be the predicted feature with a relatively worse certainty compared to gender and ethnicity. We have further performed our error analysis on age predictions by studying the difference

$$\Delta age \ = Actual - Predicted$$

Figure 16: Distribution of Δage = Actual - Predicted for all images

Fig 16 shows the distribution of Δage for all images. The distribution has been fitted with a Gaussian Function that gives us a mean of 3.11 years and a σ = 10.30 years. The value of the fitting parameter σ justifies the precision that we set to determine the accuracy of our age prediction.

To explore the age prediction uncertainties in different age groups, we have extended our analysis to the following age ranges:
0 – 10 years, 10 – 20 years, 20-30 years, 30 – 40 years, 40 – 60 years, 60 – 80 years and more than 0 years. The respective fitted distributions are shown in Fig 17.

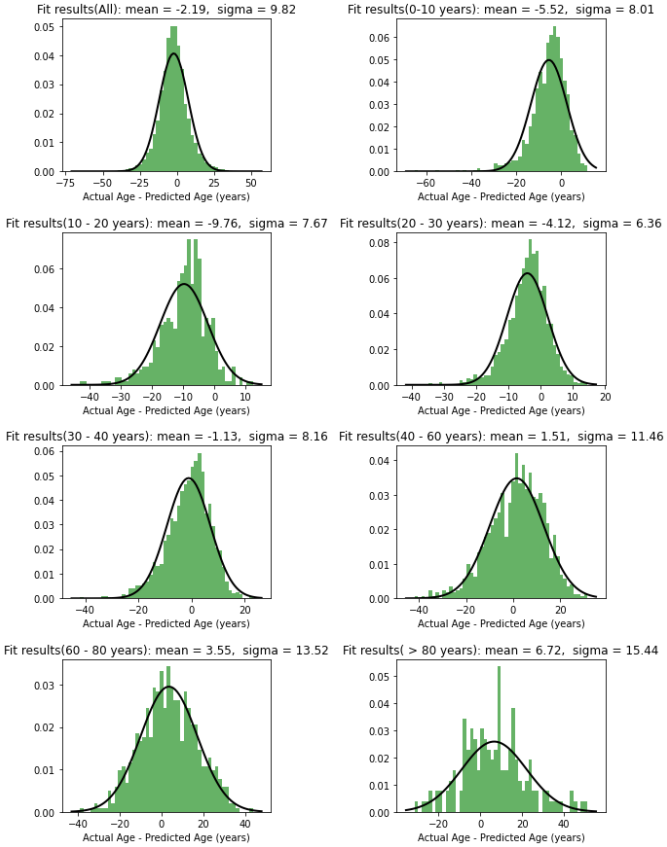

Figure 17: Δage distribution for different actual age ranges.

The mean and σ of the Gaussian Fitting parameters of the above distributions are exhibited in Fig 24 – 25.

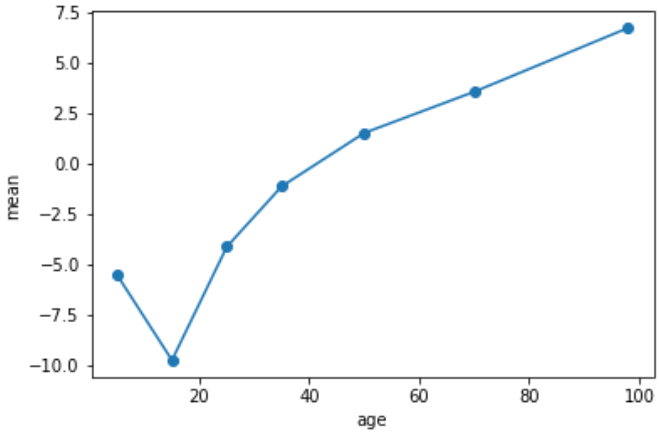

Figure 18: Mean vs age range.

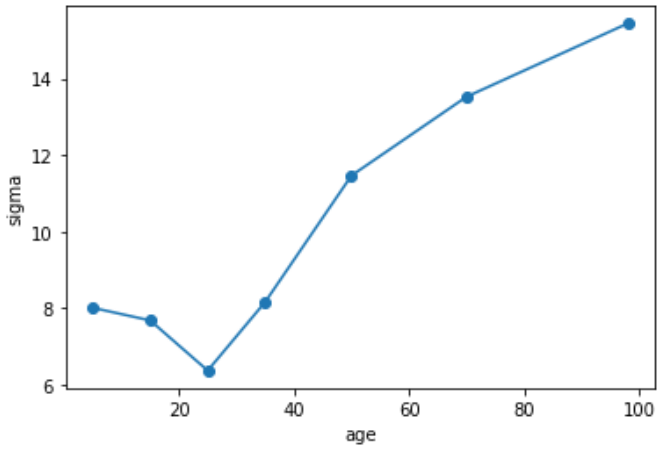

Figure 19: σ vs age range.

Fig 18 indicates that the age tends to be underestimated from 0 – 30 years and then it is gradually overestimated. The value of the standard deviation of σ tends to decrease with increasing age up to 20 – 30 years age range, then it increases gradually (Fig 19). We can correlate this variation of σ with the statistics for different age ranges as shown in Fig 7. The change in σ with age range is apparently a statistical artifact of the data. The data has the best statistics in an age range of 20 – 30 years, which shows the lowest value of σ.

In the second part of our data analysis, we have applied OpenCV architecture to our images with a "Smiley" Haar Cascade configuration. That enables us to predict only the happy moods identified by the smiley faces in our dataset. Haar Cascade is a pattern recognition algorithm used in Machine Learning that can be applied to recognize artifacts in the images. The OpenCV is an open source computer vision library that provides a set of predefined Haar Cascades with specific implementation APIs [12]. We utilized OpenCV2 as shown in Fig 16.

```
image = cv2.imread(fullpath)
smile_cascade = cv2.CascadeClassifier(path2)
smiles  = smile_cascade.detectMultiScale(image, scaleFactor = 1.2, minNeighbors =2)
for (sx, sy, sw, sh) in smiles:
  cv2.rectangle(image, (sx, sy), ((sx + sw), (sy + sh)), (0, 255,0), 5)
  plt.imshow(image)

cv2_imshow( image)
# plt.imshow(im.resize((198,198)))
cv2.waitKey(0)
cv2.destroyAllWindows()
```

Figure 17: OpenCV2 API implementation to detect a Smiley face.

A set of smiley expressions using the above API are shown in Fig 17.

Figure 18: Mood recognition using OpenCV.

V. Conclusions and Future Outlook

In this project, we successfully implemented a multi-output multi-layered CNN to identify age, gender, and ethnicity of facial images using a simplified dataset originally retrieved from UTKFaces. Our multi-output model shows maximum accuracy of about 90% in determination of the gender associated with the image. The ethnicity and age are also determined using our algorithm with an accuracy of 76% for ethnicity and R_2 score of 0.75 for age. A recent publication that optimized the CNN models to mitigate biases and loss weights showed an overall race and gender classification accuracy of 84% and 93% respectively [13]. There seems to be a statistical bias in the age determination which was determined by studying the difference between the actual and predicted ages at different age ranges. The resulting distributions show that the age range between 20 – 30 years has a minimum standard deviation as this age range has the best statistics of the data.

A better accuracy in age determination can possibly be materialized if we slice the age data into several broad bins and convert that to a classification problem rather than a regression problem.

The mood prediction scheme in our work is based on the OpenCV architecture that, at this time, is predicting only the happy or smiley faces. The mood recognition challenge can be extended to a process where we will train a model with mood-annotated dataset (not the UTKFace dataset which does not include any mood annotation) and apply that model to the UTKFace dataset.

VI. References

[1] R. Yamashita, M. Nishio, R. K. G. Do and K. Togoshi, "Convolutional neural networks: an overview and application in radiology," *Insights into Imaging*, vol. 9, pp. 611-629, 2018.

[2] A. Khan, A. Sohail, U. Zahoora and A. S. Qureshi, "A survey of the recent architectures of deep convolutional neural networks," *Artificial Intelligence Review*, vol. 53, pp. 5455-5516, 2020.

[3] AliceOthmani, A. R. Talebb, H. Abdelkawya and A. Hadidc, "Age estimation from faces using deep learning: A comparative analysis," *Computer Vision and Image Understanding*, vol. 196, no. 1077-3142, p. 102961, 2020.

[4] B. Xia, B. B. Amor and M. Daoudi, "Joint gender, ethnicity and age estimation from 3D faces: An experimental illustration of their correlations," *Image and Vision Computing*, vol. 64, pp. 90-102, 2017.

[5] M. Duan, K. Li, C. Yang and K. Li, "A hybrid deep learning CNN–ELM for age and gender classification," *Neurocomputing*, vol. 275, no. 0925-2312, pp. 448-461, 2018.

[6] G. Antipov, M. Baccouche, S.-A. Berrani and J.-L. Dugelay, "Effective training of convolutional neural networks for face-based gender and age predictio," *Pattern Recognition*, vol. 72, pp. 15-26, 2017.

[7] A. Sun, Y. Li, Y.-M. Huang, Q. Li and G. Lu, "Facial expression recognition using optimized active regions," *Human-centric Computing and Information Sciences,* vol. 8, no. 33, 2018.

[8] V. Birla, "AGE, GENDER AND ETHNICITY (FACE DATA) CSV," September 2020. [Online]. Available: https://www.kaggle.com/nipunarora8/age-gender-and-ethnicity-face-data-csv/tasks?taskId=2154.

[9] D. Xu, Y. Shi, I. W. Tsang, Y. -S. Ong, C. Gong and X. She, "Survey on Multi-Output Learning," *IEEE Transactions on Neural Networks and Learning Systems,* vol. 31, no. 7, pp. 2409-2429, 2020.

[10] F. Chollet, Deep Learning with Python, Second ed., Manning Publications, 2017.

[11] Z. Zhang, Y. Song and H. Qi, "Age Progression/Regression by Conditional Adversarial Autoencoder," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* pp. 5810-5818, 2017.

[12] "OpenVC," [Online]. Available: https://github.com/opencv/opencv.

[13] A. Das, A. Dantcheva and F. Bremond, "Mitigating Bias in Gender, Age and Ethnicity Classification: a Multi-Task Convolution Neural Network Approach," in *ECCVW 2018 - European Conference of Computer Vision Workshops*, Munich, Germany., 2018.