Ethnicity, Age and Gender Recognition using Multi-Task Learning

Soumyo Dey
Northeastern University
Boston, MA
dey.soum@northeastern.edu

Kartik Aggarwal
Northeastern University
Boston, MA
aggarwal.kart@northeastern.edu

Abstract

This project focuses on predicting age, race, and gender using pre-trained models using multitasking and transfer learning techniques. The proposed model employs a multitask learning approach that allows the model to learn to recognize multiple attributes simultaneously, while also leveraging transfer learning to improve performance by utilizing pre-trained models on large-scale datasets. The project utilizes publicly available UTKFace dataset and evaluates the proposed model's performance using various metrics such as accuracy, precision, recall, and F1-score. The developed model has several potential applications in various domains such as security, marketing, and healthcare, where recognizing the age, race, and gender of individuals can be crucial.

1 Introduction

Recognition of age, ethnicity, and gender from facial pictures is a crucial computer vision and deep learning task with several applications in security, entertainment, and healthcare. However, the performance of face recognition algorithms is affected due to some controlled and uncontrolled covariates. Controlled variates such as lighting, position, occlusion, facial expressions and image resolution, and uncontrolled covariates such as aging, gender and ethnicity [1]. It can be difficult to precisely identify these characteristics from photos.

Previously a lot of work has been done in gender recognition but age and ethnicity recognition still remain a challenging and not so worked upon task. Multitasking and transfer learning strategies can be used to get around these obstacles. Transfer learning involves applying what is learnt from one task to perform better on another task that is related, whereas multitask learning involves training a neural network to execute multiple tasks at once. These methods can be combined to produce a more reliable model that can correctly identify age, race, and gender from facial photos.

In this paper, we will look at several architectures, such convolutional neural networks (CNNs) and pre trained models such as ResNet, VGG19, Xception, MobileNet and also transformers, and assess the efficacy of various transfer learning strategies, like feature extraction and fine-tuning. We will also assess the model's performance on the UTK dataset and contrast it with baseline model’s performance. Our ultimate objective is to create a model that is both very accurate and effective and can be used in situations that occur in the actual world.

2 Literature Review

Convolutional networks have been studied by several for age estimation like [2] and [3]. A multi-scale convolutional neural network with 23 sub-networks is used by Yi et al. [4]; each sub-network receives an input of an image patch. Each of the sub-networks independently learns features, which are then combined and supplied to a fully connected layer with a square difference cost function that resembles a linear regressor. A convolutional layer, a max-pooling layer, and a locally connected layer make up the sub-networks. They also use a unique loss function for predicting age, gender, and ethnicity to train the network to do multitask learning.

For the purpose of determining age and gender, Levi et al. [5] employ shallow CNNs. Three convolutional layers and two fully connected layers make up their network. When leveraging deep feature representations learned by the CNN, they exhibit a noticeably better level of performance on the Adience benchmark for age and gender.

Ranjan et al. [6] propose a multi-task learning scheme using CNN's for concurrently performing face detection, face alignment, pose estimation, gender recognition, smile detection, age estimation, and face recognition. Parameters are shared between the five sub-networks of the CNN for the purpose of concurrent learning. They evaluate their approach on multiple unconstrained datasets and show significant improvements in performance.

Using a deep learning-based classifier, Narang et al. [7] accomplish ethnicity classification on near-infrared images (NIR). A deep convolution neural network that conducts gender and ethnicity classification is part of the suggested system. It has been demonstrated that the system uses soft biometrics to enhance face recognition performance.

Alyaa et al. [8] proposed a method for gender classification using convolutional neural networks (CNNs) on passive infrared (PIR) images. The accuracy of gender classification was tested in studies to see how different CNN architectures affected it. They demonstrate that ResNet-50 outperforms other CNN designs on their dataset. In addition, they investigate how the quantity of training samples affects the gender classification accuracy and demonstrate that adding more training samples raises the precision of their CNN-based approach.

3 Proposed Method

3.1 Dataset

A large-scale face dataset with a wide age range (from 0 to 116 years old) is the UTKFace dataset. Over 20,000 face photos with annotations for age, gender, and ethnicity make up the dataset. There is a wide range of poses, facial expressions, lighting, occlusion, resolution, etc. in the photographs. The dataset consists of 20k+ face images in the wild (only single face in one image). The images are labelled by age, gender, and ethnicity. The labels of each face image are embedded in the file name where 'age' is an integer from 0 to 116, indicating the age, 'gender' is either 0 (male) or 1 (female) and finally 'race' is an integer from 0 to 4, denoting White, Black, Asian, Indian, and Others (like Hispanic, Latino, Middle Eastern).

3.2 Data Preparation

We used a generator function to prepare our data that takes in a data frame, indices, and other parameters, that is used for training, validation and testing of the model. The generator will produce batches of images along with their corresponding age, gender, and race labels.

The filename of the image is used to extract age, gender, and race. The images were resized to (224x224) to have a consistent image size since the size of the images in data were inconsistent. The images are then normalized by dividing each pixel value by 255. This step ensures that the pixel values are in the range of 0 to 1, making it easier for the model to learn. The image, gender, and race information are stored in separate lists. The age is normalized by dividing it by the maximum age value in the dataset. The gender and race labels are one-hot encoded.

3.3 Exploratory Data Analysis

We performed various exploratory data analysis (EDA) to understand our dataset better. In total we have 23725 images and each image has three labels – 'age', 'race' and 'gender'. We plotted a boxplot for understanding 'gender' distribution and we found that number of photos of 'male' is higher than that of 'female'. Similarly, we plotted a boxplot for 'race' distribution and found out that number of photos of ‘white’ people were most followed by ‘Indians’ and the least were that of ‘others. We also plotted a boxplot for finding out race distribution for each gender separately and found out that the distribution was similar where ‘white’ was the most common label and ‘others’ was the least. To understand how ‘age’ is distributed we used histogram instead of boxplot as we could see that age was a continuous data. We can see that most of the ‘age’ lies between age ‘10’ and ‘65’ with mean ‘age’ coming out to be ‘33’ and thus rest can be considered as outliers.

3.4 Base Models

In transfer learning, a base model is a pre-trained model that has been trained on a large dataset for a particular task such as image classification. These base models provide a provide a starting point for the task in-hand which can be fine-tuned on a smaller dataset specific to the task in-hand. This approach can help to improve the accuracy of the model on the task in-hand, even with limited data.

3.4.1 VGG19

VGG19 is a convolutional neural network architecture introduced by Simonyan et. al. [9]. VGG19 comprises of sixteen convolutional layers, five max pooling layers, and three dense layers in total, for a total of 24 layers, but only nineteen weight layers.

In this project VGG19 model architecture was used as the base model to extract features. The top layers of the model were removed and it was trainable, which means that its weights can be updated during training and fine-tuned according to the data. The model was initialized with pretrained weights of ‘ImageNet’. A global max pooling layer is added after the base model to reduce the spatial dimensions of the output.

3.4.2 ResNet50

ResNet50 is a convolutional neural network architecture introduced in 2015 by Kaiming He et al. [10]. "ResNet" stands for "Residual Network," and it refers to the use of residual connections to train far deeper networks than was previously achievable. ResNet50 is made up of 50 layers, which include convolutional layers, batch normalization layers, activation layers, and fully linked layers. It employs a revolutionary architecture with residual connections that allow data to bypass some layers and travel directly to deeper layers. This helps to avoid the "vanishing gradient" problem, which can arise in very deep networks when gradients become too small to propagate back through the network during training.

In this project ResNet50 model architecture was used as the base model to extract features. The top layers of the model were removed and it was made trainable, which means that its weights can be updated during training and fine-tuned according to the data. The model was initialized with pretrained weights of ‘ImageNet’. A global max pooling layer is added after the base model to reduce the spatial dimensions of the output.

3.4.3 Inception v3

Inception v3 is a convolutional neural network architecture introduced in 2015 by Szegedy, Christian, et al. [11] as an enhancement to the original Inception model. It was meant to be more computationally efficient while still performing well on image identification tests. The Inception v3 model is made up of a sequence of convolutional layers, followed by Inception modules, which are blocks of convolutional layers having numerous routes for feature extraction.

Inception v3 additionally incorporates batch normalization layers and global average pooling, which serve to increase the model's accuracy and efficiency. One of the primary characteristics of Inception v3 is the use of factorization, which decreases the number of parameters in the model while retaining accuracy. Inception v3 employs factorized 7x7 convolutions, which split the convolution operation into separate 1x7 and 7x1 convolutions, and factorized 3x3 convolutions, which break the convolution action into separate 1x3 and 3x1 convolutions. This helps to lower the model's computing cost while keeping its capacity to extract relevant results.

In this project Inception V3 model architecture was used as the base model to extract features. The top layers of the model were removed and it was made trainable, which means that its weights can be updated during training and fine-tuned according to the data. The model was initialized with pretrained weights of ‘ImageNet’. A global max pooling layer is added after the base model to reduce the spatial dimensions of the output.

3.4.4 Xception

Xception is a convolutional neural network architecture that was introduced by Chollet, François et. al. [12]. The name "Xception" stands for "Extreme Inception," which refers to the use of an "extreme" version of the Inception module, a popular convolutional neural network module used for feature extraction. Although there are some significant differences, the Xception model is comparable to the Inception architecture. Xception uses depthwise separable convolutions, which divide the convolution operation into distinct depthwise and pointwise convolutions, in place of normal convolutional layers. This enhances the network's effectiveness by lowering the number of parameters in the system.

In this project Xception model architecture was used as the base model to extract features. The top layers of the model were removed and it was made trainable, which means that its weights can be updated during training and fine-tuned according to the data. The model was initialized with pretrained weights of ‘ImageNet’. A global max pooling layer is added after the base model to reduce the spatial dimensions of the output.

3.4.5 MobileNet

MobileNet is a convolutional neural network architecture that was introduced by Howard, Andrew G., et al. [13]. The MobileNet architecture was created to be quick and light, making it appropriate for use in contexts with limited resources like mobile devices.

Similar to the Xception model, the MobileNet model makes use of depthwise separable convolutions. By dividing the usual convolutional process into a depthwise convolution and a pointwise convolution, MobileNet, however, expands on this idea. This enhances the network's effectiveness and further reduces the number of parameters in use. A set of convolutional layers, depthwise separable convolutions, batch normalization layers, and activation layers make up the MobileNet model. Additionally, skip connections are included to aid in gradient propagation and information flow during training.

In this project MobileNet model architecture was used as the base model to extract features. The top layers of the model were removed and it was made trainable, which means that its weights can be updated during training and fine-tuned according to the data. The model was initialized with pretrained weights of ‘ImageNet’. A global max pooling layer is added after the base model to reduce the spatial dimensions of the output.

3.4.6 Transformer

The concept of transformers was introduced in a research paper titled "Attention Is All You Need," which was published by Vaswani et al. in 2017. The transformer model relies on a mechanism called self-attention. Self-attention enables the model to process each element while concentrating on different portions of the input sequence, effectively capturing both local and global interdependence.

The encoder and the decoder are the two primary parts of the transformer design. The encoder takes an input sequence and processes it step-by-step. Each input element is embedded into a high-dimensional vector representation. These embeddings are then processed through a stack of identical layers, typically consisting of two sub-layers Self-Attention Layer and Feed-Forward Layer. The decoder also consists of a stack of identical layers, but it has an additional sub-layer compared to the encoder. The layers are Self-Attention Layer, Encoder-Decoder Attention Layer and Feed-Forward Layer. Similar to the encoder, the decoder employs self-attention to capture dependencies within the target sequence. Encoder-Decoder Attention Layer allows the decoder to attend to the encoder's output. Similar to the encoder, a feed-forward neural network is applied to the decoder's representations.

In this project, we defined two classes, first class had the self-attention and feed forward layers and in the second class we had the encoder. We used MobileNet as the base model before using transformer. So, the input was passed to MobileNet and the output of the MobileNet was passed to transformer. Finally, a global max pooling layer is added after the base model to reduce the spatial dimensions of the output.

3.5 Classification Heads

3.5.1 Age Classifier

In the age classifier block the output from the global max pooling layer is passed through two dense layers with 128 hidden units each, using ReLU activation, and a single output neuron with sigmoid activation, producing a continuous value for age.

3.5.2 Race Classifier

In the race classifier block the output from the global max pooling is passed through two dense layers with 128 hidden units each and ReLU activation are used to generate the race prediction. The output layer has a SoftMax activation function and produces a probability distribution over five different races.

3.5.3 Gender Classifier

In the gender classifier block the output from the global max pooling is passed through two dense layers with 128 hidden units each and ReLU activation are used to generate the race prediction. The output layer containing a probability distribution over two different genders.

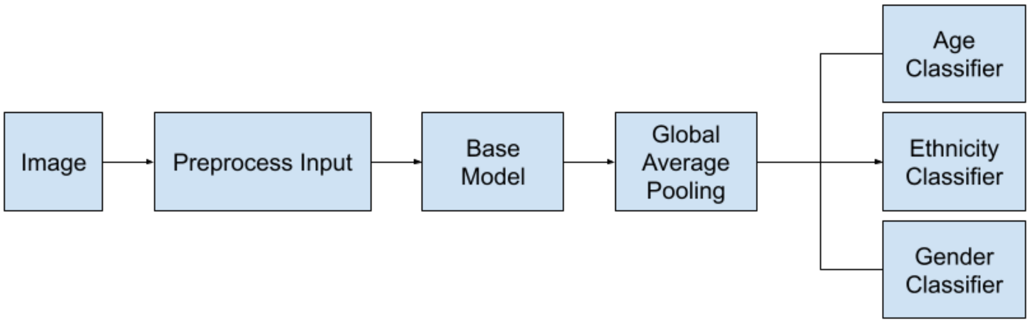

Figure 1: Proposed Model

4 Experiments and Results

4.1 Training, Validation and Testing Dataset

During EDA, we found that the data was unbalanced and most of the ‘age’ labels were in range 10 to 65. Thus, the rest of the labels were considered outliers and were removed in order to get a balanced dataset. Once the dataset was reduced, 9115 images were used as training data, 3907 images as validation data and finally 5581 images as test data.

4.2 Experimental Setup

RMSprop is used as the optimizer in all the models. MSE loss is used to calculate the loss of age prediction because we are treating age as a continuous value. For gender and ethnicity, we used categorical cross entropy loss for loss calculation. The evaluations metrics for age is MAE (Mean Absolute Error) and for gender and ethnicity is accuracy.

For all the models, batch size was set to be 16 and model checkpoint was used where we monitored the validation loss. In total all the model was trained for 10 epochs.

4.3 Results

We evaluated the performance of all the proposed models on UTKFace dataset. Table 1 represents the total training loss, training loss of each task and the evaluation metrics of each task for all the models. Table 2 represents the total validation loss, validation loss of each task and the evaluation metrics of each task for all the models. Table 3 represents the total test loss, test loss of each task and the evaluation metrics of each task for all the models.

The multi-task classifier with Xception as the fine-tuned pre-trained model provided the best evaluation metrics in all the three datasets. The MobileNet model's total validation and test loss was less than that of Xception model but still Xception model provide better results.

Table 1: Training loss and evaluation metrics

Models	Loss				Evaluation Metric		
	Total	Age	Ethnicity	Gender	Age MAE	Ethnicity Accuracy	Gender Accuracy
VGG19	2.94	0.03	1.45	0.69	0.15	0.40	0.52
ResNet50	2.94	0.03	1.45	0.69	0.15	0.40	0.52
Inception v3	2.03	0.03	1.07	0.35	0.14	0.57	0.84
Xception	**0.36**	**0.02**	**0.17**	**0.06**	**0.11**	**0.94**	**0.98**
MobileNet	0.66	0.02	0.33	0.11	**0.11**	0.89	0.96
Transformer	3.39	0.25	1.45	0.69	0.47	0.40	0.52

Table 2: Validation loss and evaluation metrics

Models	Loss				Evaluation Metric		
	Total	Age	Ethnicity	Gender	Age MAE	Ethnicity Accuracy	Gender Accuracy
VGG19	2.93	0.03	1.44	0.69	0.15	0.41	0.52
ResNet50	2.94	0.03	1.45	0.69	0.15	0.40	0.53
Inception v3	1.97	0.03	1.05	0.32	0.14	0.59	0.86
Xception	1.93	**0.02**	**0.08**	0.27	**0.11**	**0.79**	**0.93**
MobileNet	**1.58**	0.02	0.87	**0.22**	**0.11**	**0.79**	**0.93**
Transformer	3.38	0.25	1.45	0.69	0.46	0.40	0.52

Table 3: Testing loss and evaluation metric

Models	Loss				Evaluation Metric		
	Total	Age	Ethnicity	Gender	Age MAE	Ethnicity Accuracy	Gender Accuracy
VGG19	2.94	0.03	1.45	0.69	0.15	0.40	0.52
ResNet50	2.93	0.03	1.44	0.69	0.15	0.41	0.52
Inception v3	2.22	0.03	1.09	0.52	0.14	0.59	0.76
Xception	1.78	**0.02**	1.02	0.21	**0.11**	**0.81**	**0.94**
MobileNet	**1.55**	**0.02**	**0.87**	**0.19**	**0.11**	0.79	**0.94**
Transformer	3.39	0.25	1.45	0.69	0.46	0.41	0.53

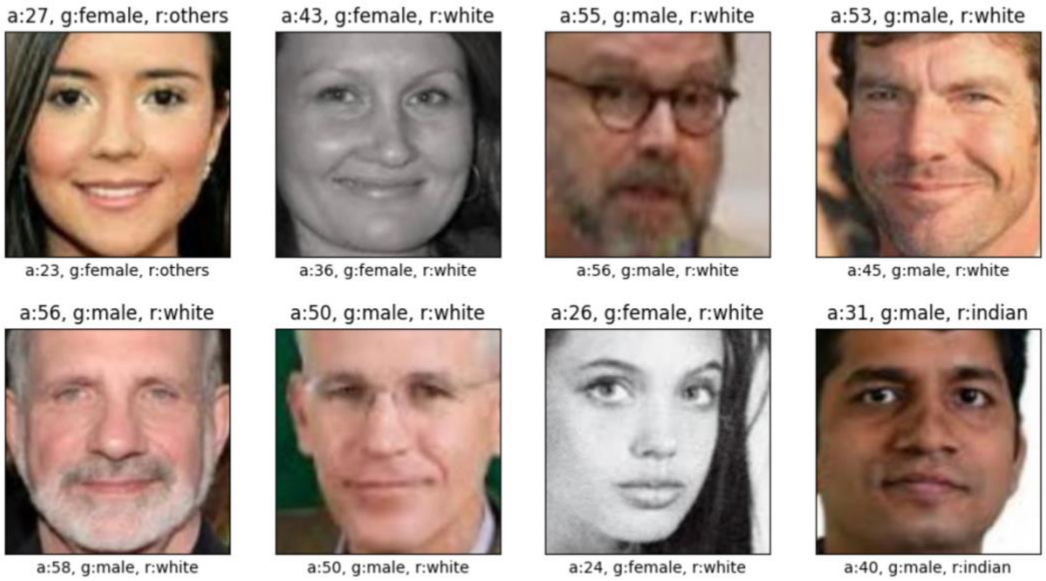

Figure 1: Results when Xception as base model

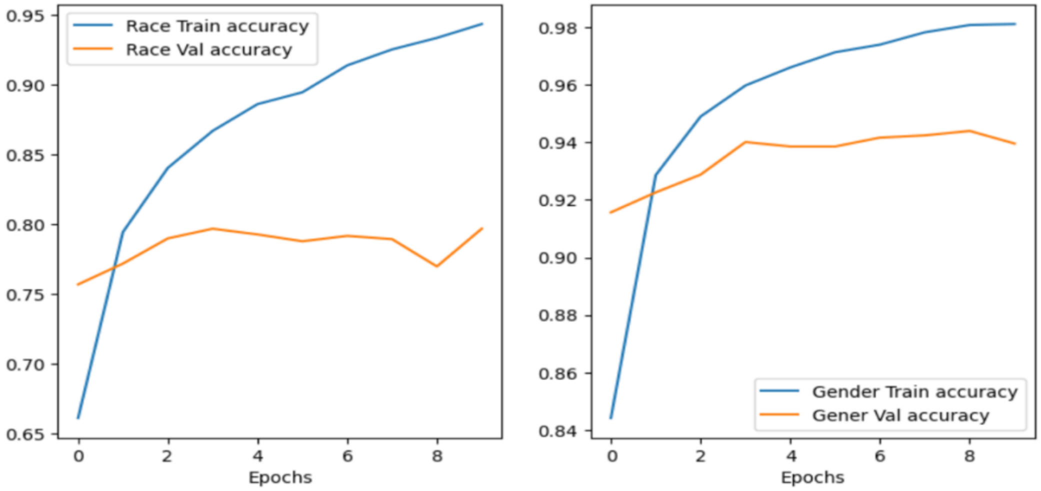

Figure 2: Comparison between train and validation accuracy for race and gender for Xception model

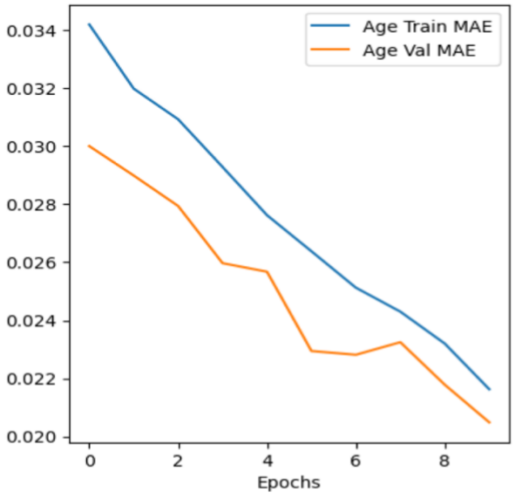

Figure 3: Comparison between train and validation MAE for age for Xception model

The predictions of the best model are depicted in Figure 1. The above labels are the predicted labels and the bottom ones are the original labels.

5 Conclusion

Our project demonstrates the effectiveness of Multi-Task Learning in Ethnicity, Age, and Gender Recognition. The Xception model emerged as the top-performing architecture, achieving impressive results with an age MAE (Mean Absolute Error) of 0.11, gender classification accuracy of 0.94, and ethnicity classification accuracy of 0.81. These findings pave the way for future advancements in computer vision and offer valuable insights for applications such as facial recognition systems, demographic analysis, and personalized user experiences.

References

[1] Abdurrahim, Salem Hamed, Salina Abdul Samad, and Aqilah Baseri Huddin. "Review on the effects of age, gender, and race demographics on automatic face recognition." *The Visual Computer* 34 (2018): 1617-1630.

[2] X. Wang, R. Guo and C. Kambhamettu, "Deeply-learned feature for age estimation", 2015 IEEE Winter Conference on Applications of Computer Vision, pp. 534-541, Jan 2015.

[3] X. Yang, B.-B. Gao, C. Xing, Z.-W. Huo, X.-S. Wei, Y. Zhou, et al., "Deep label distribution learning for apparent age estimation", The IEEE International Conference on Computer Vision (ICCV) Workshops, December 2015.

[4] D. Yi, Z. Lei and S. Z. Li, "Age estimation by multi-scale convolutional network", Asian Conference on Computer Vision, pp. 144-158, 2014.

[5] R. E. Eran Eidinger and T. Hassner, "Age and gender estimation of unfiltered faces", Transactions on Information Forensics and Security (IEEE-TIFS) special issue on Facial Biometrics in the Wild, vol. 9, no. 12, pp. 2170-2179, Dec. 2014.

[6] R. Ranjan, S. Sankaranarayanan, C. D. Castillo and R. Chellappa, "An all-in-one convolutional neural network for face analysis", CoRR abs/11611.00851, 2016.

[7] N. Narang and T. Bourlai, "Gender and ethnicity classification using deep learning in heterogeneous face recognition", 2016 International Conference on Biometrics (ICB), pp. 1-8, June 2016.

[8] Alyaa J. Jalil 1and Naglaa M. Reda2 “Infrared Thermal Image Gender Classifier Based on the Deep ResNet Model” 8 July, 2022.

[9] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." *arXiv preprint arXiv:1409.1556* (2014).

[10] He, Kaiming, et al. "Deep residual learning for image recognition." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.

[11] Szegedy, Christian, et al. "Rethinking the inception architecture for computer vision." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.

[12] Chollet, François. "Xception: Deep learning with depthwise separable convolutions." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017.

[13] Howard, Andrew G., et al. "Mobilenets: Efficient convolutional neural networks for mobile vision applications." *arXiv preprint arXiv:1704.04861* (2017).