|  |
| --- |
| **Visual Age and Gender Classification by Using Convolutional Neural Network in Real Time Applications** |

**Dipak SubramaniamGoksu Avdan**

Department of Computer Science Department of Industrial Engineering

Southern Illinois University Edwardsville Southern Illinois University Edwardsville

Edwardsville, IL 62026 Edwardsville, IL 62026

[*dsubram@siue.edu*](mailto:dsubram@siue.edu)[*agoksu@siue.edu*](mailto:agoksu@siue.edu)

**Hari Krishna PathakamuriMinxuan Bai**

Department of Computer Science Department of Computer Science

Southern Illinois University Edwardsville Southern Illinois University Edwardsville

Edwardsville, IL 62026 Edwardsville, IL 62026

[*hpathak@siue.edu*](mailto:hpathak@siue.edu) *mbai@siue.edu*

# **1 Introduction**

The number of images uploaded to the internet has increased exponentially over the decade. Due to unconstrained imaging conditions, accuracy for age and gender detection were dropped. Age and gender detection is useful for identity recognition, security, and verification purposes in public spaces such as airports, casinos, etc. It is also useful for social platforms to prevent ID verification [1]. This project was inspired by the research of Levi & Hassner on age and gender classification by using Convolutional Neural Networks (CNNs) [2].

Instead of using images as an input for the final model, our model is going to use the video-based image for the real time detection. We are going to add race detection, and also in the further process we are going to add the smile detection systems along with the age and gender detection for the commercial usage of this algorithm such as customized advertising options etc. Our approach is based on Keras and TensorFlow to create convolutional layers. We use the OpenCV classifier to detect the face. We then use Conv2D layers, batch normalization, MaxPooling2D and ReLU layers to detect age, gender, and race. The result will be an image with added gender, age and other information on top of the face such as race.

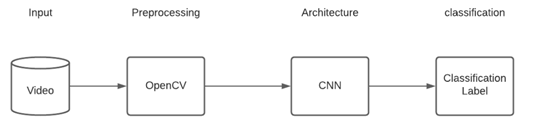


Figure 1: Flow process for the proposed 2D-CNN model

# **2 Related work**

There are many researches focused on age, gender and/or other characteristic classification in face recognition. At early days the age classification was calculated by fitting between man-made facial feature datasets which derived from their geometry parameters and required accurate measurements[3]. Treating the aging process as a subspace structure also works but it needs the images to be aligned [4]. Other than using facial features, the Gaussian Mixture Models (GMM) use the distribution of facial patches to distinguish different ages between people. Not only GMM, but also Hidden-Markov model can be used for age classification. These models are effective for small or constrained datasets. For the gender classification part, as an early example, training the neural network over a small set of front-back facial pictures is a typical approach [5]. The intensities of images can also be used for doing gender classification, especially combining SVM or AdaBoost classifiers together [6] [7]. The above mentioned gender separated models were using FERET benchmark dataset to develop and test the performance. The pictures in the FERET dataset are highly controlled. So, for the real world pictures the performances of these models may be severely degraded. On the other hand, UTKFace dataset has an extensive dataset which is over 20,000 images, and every file is classified for the purpose of the detection of age, gender, and race at the same time [8].

In our project, we will be using Keras library to create a series of layers such as Conv2D, batch normalization, Maxpooling 2D, ReLU, softmax function, etc to construct and train our neural network, and thanks to the growing computational power our network may achieve better accuracy and a better performance on real-world or in-the-wild pictures.

# **3 Materials and methods**

**3.1 Dataset and features**

Our project utilizes UTKFace, which is a publicly available dataset containing over 20,000 total face images; however, some loading problems forced us to use 16,832 images in our experiment. The images split into training, validation and test sets. The images are split based on the 70/30 rule first as training/test, and then the training set is further split into 70/30 rule for training/validation dataset creation. The labels of each face image are embedded in the file name, formatted like “[age]\_[gender]\_[race]\_[date&time].jpg”.

The age factor jumps between 0 to 116, gender is labeled as 0 (male) and 1 (female), and the race factor is denoted in integer from 0 to 4 (0 = White, 1 = Black, 2 = Asian, 3 = Indian, 4 = Others). The publicly available dataset is already cropped and aligned for the final usage. However, the normalization is done by scaling the pixel value between 0 and 1. The normalization is calculated as,

Norm(ei) = where Emin and Emax are minimum value and maximum value of E, respectively. (1)

On the resolution side of the dataset, the dataset covers the image in 200 x 200 pixels. The resolution on vertical and horizontal directions is 96 dpi. Each file is around 5-7 kb per image.

Here are the examples from the non-aligned and cropped dataset on the left, and the example from the aligned and cropped dataset on the right which is used in 2D CNN architecture.



Figure 2: Sample images after aligned and cropped pre-processing

Our dataset is going to use the aligned and cropped dataset as input features in which their pixel values are normalized between 0 and 1 by the normalization function. The target features are going to be age, gender, and race. Age and gender output features are defined as discrete values, and the age is defined as continuous in the final dataset revision.

**3.2 Methods**

Anaconda navigator is used as a desktop graphical user interface (GUI). The scientific python development environment (Spyder) is also used as a free integrated development environment (IDE). Keras library is selected to help for the creation of the network.

**3.2.1 Model description**

We can summarize the algorithm working principles like the following: The input data is aligned and cropped image dataset. The training set is trained over 1 input layer size with 32 units and 5 different 2D-CNN layers with the different filter sizes (64, 96, 128, 160, and 192). Each 2D-CNN layer architecture has 2D MaxPooling which is for the down-sampling and shown in formula (2), batch normalization which accelerates the performance, and ReLU which is the most common selection as a nonlinear activation function because the inputs of each layer are non-negative numbers. ReLUs working principle is shown in formula as well (3).

= (2)

= (3)

After 2D-CNN layers, the model is reduced the dimensionality by using the 2D global max pooling in the bottleneck layer. After the dimensionality reduction, the model is split into the three different sections to predict the age, gender, and race target values separately. The SoftMax activation functions are used to predict the gender (gender = 0, 1) and race (race = 0, 1, 2, 3, 4) discrete target values based on the probability estimation for each category (4). The sigmoid function is used to predict the continuous target value which is the age prediction for each image which is shown in formula (5). This is a different method compared to Levi’s SoftMax classification method for age prediction.

S ( = (4)

σ(x) = (5)

Baseline model is also created by using 3 different 2D-CNN layers instead of using 5 different 2D-CNN layers after the input layer in our main architecture.

**3.2.2 Hyperparameter selection and varying batch sizes**

The model is run with 3 different batch size selections, and the hyperparameter selection and other experiment details for each model are shown in table 1. The hyperparameter optimization is just done on the batch sizes by using trial & error method. The other hyperparameter optimization methods are going to run in the final step of the project to find an optimal parameter selection. Early stopping function is not added to our models because of the selection of the total of 2 epochs for each model.

Table 1: Hyperparameter selection for each model

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Hyperparameters** | **Baseline** | **Experiment 1** | **Experiment 2** | **Experiment 3** |
| 2D CNN Activation Func. | ReLU | ReLU | ReLU | ReLU |
| Pooling Method | 2D MaxPooling | 2D MaxPooling | 2D MaxPooling | 2D MaxPooling |
| Kernel Size | (3, 3) | (3, 3) | (3, 3) | (3, 3) |
| Filter Size | 32 | 32 | 32 | 32 |
| Train/Validation/Test | 49/21/30 | 49/21/30 | 49/21/30 | 49/21/30 |
| Target Activation Func. | SoftMax / Sigmoid | SoftMax / Sigmoid | SoftMax /Sigmoid | SoftMax /Sigmoid |
| Batch Size | 64 | 128 | 64 | 32 |
| # of epochs | 2 | 2 | 2 | 2 |
| Total Training Time | 114 min 38 sec | 117 min 57 sec | 175 min 47 sec | 120 min 45 sec |

**3.2.3 Dataset splitting**

First, the dataset is split into 2 different parts based on 70/30 rule which is the most common usage for training/test dataset selection then 70/30 rule is applied onto the training dataset to create the training/validation dataset for the experiments, the exact split for each training, validation, and test sets are shown in table 1.

**3.2.4 Result metrics**

For the race and the gender classification methods, the loss function is selected as categorical-cross entropy. For the age prediction method, the loss function is calculated by mean squared error (MSE).

The final accuracy is calculated with the percentage based for race and gender which are shown under the preliminary results section. The final accuracy for the age target values, we use the mean absolute error (MAE) method (6).

MAE = (6)

# **4 Preliminary results and future steps**

Our prediction model achieved 82% gender prediction accuracy in experiment 3. Gender accuracy level is really close to Levi’s prediction results which are around 85-86%. But the results are not exactly comparable because of the dataset differences between our study and their published study.

Table 2: Gender classification results

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Gender** | **Baseline** | **Experiment 1** | **Experiment 2** | **Experiment 3** |
| 0 (Male) | 0.48 | 0.42 | NA | 0.71 |
| 1 (Female) | 0 | 0.20 | NA | 0.86 |
| Average | 0.23 | 0.18 | 0.48 | 0.82 |

On the other hand, our new proposed model has a race prediction feature which has achieved 62% accuracy over 5 different categories in the selection of 32 batch sizes. These results are promising for the preliminary results step and will be definitely improved after extensive hyperparameter optimization for each feature.

Table 3: Race classification results

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Race** | **Baseline** | **Experiment 1** | **Experiment 2** | **Experiment 3** |
| 0 (White) | 0.36 | 0.32 | NA | 0.63 |
| 1 (Black) | 0 | 0 | NA | 0.88 |
| 2 (Asian) | 0 | 0 | NA | 0.61 |
| 3 (Indian) | 0 | 0 | NA | 0.84 |
| 4 (Others) | 0 | 0 | NA | 0 |
| Average | 0.13 | 0.10 | 0.39 | 0.62 |

Age prediction method has a different activation function, sigmoid activation function, compared to Levi’s model which uses the SoftMax function. So, the results here are not comparable with his results; however, our method has reached the minimum 0.066 loss in experiment 2 over the normalized age image dataset.

Table 4: Age prediction results

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Age** | **Baseline** | **Experiment 1** | **Experiment 2** | **Experiment 3** |
| MAE Loss | 0.097 | 0.117 | 0.066 | 0.096 |

Baseline and experiment 1 models are not working very well as we can see in the gender and race prediction classification results. For experiment 2, the data is missing in the race and gender classification results because of the technical difficulty on Spyder IDE but this will be fixed later in the second report. On the other hand, experiment 3 was given great preliminary results for the next step of this project.

**4.1 Future steps**

Given our results, the next steps for our group will be to fully transition to video inputs and to complete further training and testing. Moving forward, our project will try to understand current sources of error (i.e. lighting, facial emotions, pose, etc.) and also will try to tackle racial biases within facial recognition. Also, we will focus on the hyperparameter optimization to obtain the optimal features based on our UTKFace dataset. This will result in a few more new models which we will detail in our final report.

**References**

[1] I. Amerini, C. T. Li, and R. Caldelli, “Social Network Identification Through Image Classification with CNN,” *IEEE Access*, vol. 7, pp. 35264–35273, 2019, doi: 10.1109/ACCESS.2019.2903876.

[2] G. Levi and T. Hassncer, “Age and gender classification using convolutional neural networks,” *IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work.*, vol. 2015-Octob, pp. 34–42, 2015, doi: 10.1109/CVPRW.2015.7301352.

[3] Y. H. Kwon and N. Da Vitoria Lobo, “Age classification from facial images,” *Comput. Vis. Image Underst.*, vol. 74, no. 1, pp. 1–21, 1999, doi: 10.1006/cviu.1997.0549.

[4] X. Geng, Z. H. Zhou, and K. Smith-Miles, “Automatic age estimation based on facial aging patterns,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 12, pp. 2234–2240, 2007, doi: 10.1109/TPAMI.2007.70733.

[5] T. J. S. B.A. Golomb, D.T. Lawrence, “Sexnet: A neural network identifies sex from human faces.” .

[6] B. Moghaddam and M. H. Yang, “Learning gender with support faces,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 5, pp. 707–711, 2002, doi: 10.1109/34.1000244.

[7] S. Baluja and H. A. Rowley, “Boosting sex identification performance,” *Int. J. Comput. Vis.*, vol. 71, no. 1, pp. 111–119, 2007, doi: 10.1007/s11263-006-8910-9.

[8] Z. Zhang, Y. Song, and H. Qi, “Age progression/regression by conditional adversarial autoencoder,” *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, vol. 2017-January, pp. 4352–4360, 2017, doi: 10.1109/CVPR.2017.463.