|  |
| --- |
| **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 PathakamuriMingxuan 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*

**Abstract**

This study works on gender & race classification, and age prediction by using UTKFace dataset. The model is used the pre-processed dataset, which is already aligned and cropped. The model is worked with 2D CNN layers in the main architecture and the hyperparameters are tuned manually. The possible problems are observed on the dataset, which are the brightness, race similarities, and the old photo selection. On the other hand, the model is performed very well and reached 81.99% accuracy on gender classification, 62.19% accuracy on race classification, and 0.0675 MAE loss on age prediction on the test dataset. The possible future improvements are going to adapt our system to OpenCV environment for the real time detection, use more extensive dataset, and use more computing resources such as GPU.

# **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].

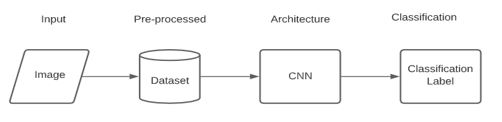


Figure 1: Flow process for the proposed 2D-CNN model

Along with the color images as an input for the final age and gender prediction, our model is also going to predict other target features. That is why, we are going to add a race detection system along with the age and gender detection for the commercial usage of this algorithm such as customized advertising options, airport security systems, embassy security protocols etc. Our approach is based on Keras and TensorFlow to create 2D convolutional layers. We used an already classified dataset for our project which is labeled with age, gender and race. We used Conv2D layers, batch normalization, MaxPooling2D, L2 Regularization and ReLU layers to detect age, gender, and race. The output layer is going to use SoftMax for gender and race detection, and use Sigmoid activation function for the age prediction model. The result will be an image with added gender, age, and other information which is race on top of the face.

# **2 Related work**

There are many research efforts and papers focused on age, gender and/or other characteristic classification in facial recognition. In the early stages, 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 near-frontal and aligned well [4]. These methods are ill-suited for unconstrained images. Other than using facial features, the Gaussian Mixture Models (GMM) use the distribution of facial patches to distinguish different ages between people. Besides the GMM, the Hidden-Markov model can be used for age classification. These models are effective for small or constrained datasets. Another solution is to replace local image intensity patches with robust descriptors, such as the Gabor image descriptors which combine with a hierarchical age classifier and support vector regression to estimate a precise age [5].

For the gender classification portion, there are numerous methods which we can compare and contrast [6] As an early example, training the neural network over a small set of front-back facial pictures is a typical approach [7]. The intensities of images can also be used for doing gender classification, especially combining SVM or AdaBoost classifiers together [8] [9]. The above-mentioned gender-separated models were using FERET benchmark dataset to develop and test the performance [10] The pictures in the FERET dataset are highly controlled because they are taken in specific planned environments, which removes the challenges of processing naturally taken image in any backgrounds. So, for the real-world pictures, the performances of these models may be severely degraded. On the other hand, there is a popular Labeled Faces in the Wild benchmark (LFW) which is used for facial gender classification in true real-world settings [11]. For our project, we considered the Adience dataset, which had unfiltered facial photos from Flickr albums and was used in the study by Levi and Hassner [2]. In the end, however, we chose the UTKFace dataset, which has an extensive dataset of over 20,000 images, wherein every file is classified for the purpose of the detection of age, gender, and race at the same time [12].

In our project, we will be using the 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. Thanks to 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)



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

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.

In Fig 2, there are some 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.

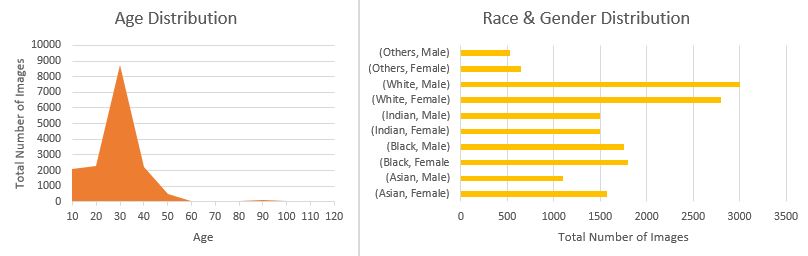


Figure 3: Dataset Distribution in terms of Age, Gender, and Race

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 64 units and 5 different 2D-CNN layers with the different filter sizes (32, 48, 64, 80, and 96). 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). Also, the L2 regularization method with the selection 0.01 is applied to the model, and prevent the overfitting on the dataset (4) [13].

= (2)

= (3)

= (4)

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 (5). The sigmoid function is used to predict the continuous target value which is the age prediction for each image which is shown in formula (6). This is a different method compared to Levi’s SoftMax classification method for age prediction.

S ( = (5)

σ(x) = (6)

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 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.

**3.2.3 Hyperparameter tuning**

The model is run with 3 different batch size selections, 3 different filter size selections, 3 different dense layer size selections, 3 different L2 regularization selections, 1 L1\_L2 regularization selection, and the main model is ran with the early stopping function to prevent overfitting problem over the training dataset split. All the hyperparameter optimization is calculated by trial & error approach.

Table 1: Hyperparameter selection for baseline, experiment #3, experiment #4, and experiment #15

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Hyperparameters** | **Baseline** | **Experiment #3** | **Experiment #4** | **Experiment #15** |
| 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 over layers | 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 | 64 | 32 | 32 |
| L2 Regularization | No | No | No | Yes (L2 = 0.01) |
| Early Stopping Function | No | No | No | Yes |
| # of epochs | 2 | 2 | 2 | 6 |
| Total Training Time | 114 min 38 sec | 175 min 47 sec | 120 min 45 sec | 198 min 36 sec |

**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 (7).

MAE = (7)

# **4 Results and discussion**

Table 2 is representing the results of gender, race classification, and age prediction. The first row of the table represents the baseline which consists of 3 different Conv2D layers modeling, and the rest of the models used 5 different Conv2D layers modeling. Batch size optimization is performed with 3 different configurations (32, 64, and 128), and 32 gave the best results. Filter size optimization is applied with also 3 different configurations. Dense layer size optimization is also ran with the 3 different configuration to understand which size is giving the highest optimal result. L2 regularization is applied with 2 different configuration, and the results show that 0.01 selection is the optimal one to maximize the accuracy over 3 different models. On the other hand, after trying flexible L1 & L2 regularization with the same starting value, which we tried on only L2 regularization selection as well, it didn’t give good results. That is why it decided to not run any more experiments for L1 & L2 regularization.

After manually tuning the hyperparameters, 32 is chosen as a batch size, (16, 32, 48, 64, 80, 96) is chosen the filter size for each Conv2D layer, respectively, 64 is chosen as an output dense layer size, 0.01 is chosen as L2 regularization parameter. By using these parameters and early stopping function over 8 epochs, the model #15 gave the best results in race prediction. Gender classification model also gave over 80% accuracy on training and test datasets. Test results from the model #3 is just a bit better than our optimized model, and probably the random selection of the test can affect this result as well. That is why, we should use 5-fold or 10-fold CV to improve the model objectivity, next time.

So, when the maximum results are observed from each of these models, our training prediction model achieved 84.03% gender prediction accuracy in experiment #3, and also reached 82.41% test classification accuracy as well. Gender accuracy level is really close to Levi’s prediction results which are around 86-87% [2]. But the results are not exactly comparable because of the dataset differences between our study and their published study, and the race classification is also not comparable with their study because their dataset is not labeled with the race feature.

On the other hand, our new proposed model has a race prediction feature which has achieved 65.51% accuracy on training set, and 62.19% accuracy on the test set over 5 different labels with our selected optimal model. These results are so promising for this step. These race classification results could be improved if the model divided by two different steps at the beginning as black and white, and then those divided models can be used to predict the final race prediction, separately. Also, this new model predictions could be ensemble with our main model predictions to increase the prediction accuracy. For the comparison, there is not much literatures to compare our results. *Greco et al.* reached 94.30% accuracy level by using 3 million images with different various of VGG architectures [14]. On the other hand, our model is only trained over 16,000 images with our Keras library used architecture. So, the race classification results are still definitely open for any improvements in the literature. By using the kinetic and kinematic information from motion capture systems, we can even predict the races and it could be a great study as a next step.

On the other hand, 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 MAE in experiment #3 over the normalized age image test dataset, and reached 0.0675 MAE in experiment #15 which is also so close to results of experiment #3.



Figure 4: Sample pictures for the test results

During checking the results from the output images, the brightness problem is arisen. This problem affected the prediction results on the ground truth black race, and it turns out the results as the white race labeled. However, the biggest problem is observed on Indian race. Indian race is mostly turned out as white race, and possible problem could be the skin color, which is so close to the white race’s skin color. The age prediction on the old pictures also turned out negative results. For example, from the last two pictures on Fig 4, our model is predicted ages too old comparing to the ground truth labels.

The results on gender and race classification models are open for the improvement by using the transfer learning, and ensemble learning methods. However, our limited time and the limited computing resources (3rd generation 3 GHz Intel i7 dual-core, and 8GB 1600 MHz DDR3 memory on macOS Catalina) affected our classification and prediction results. Also, the results clearly showed that more trainable parameters selection give better accuracy on classification for age and gender. On the other hand, more trainable parameters mean that the computing resources should be superior to run these models in a short period of time.

Table 2: Results for gender, race, and age prediction models

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Hyperparameter Tuning Optimization | | Gender Accuracy (%) | | Race Accuracy  (%) | | Normalized Age  MAE (0-1) | |
| Exp Num. | Training | Test | Training | Test | Training | Test |
| #1 | Baseline | | 0.6352 | 0.4854 | 0.4164 | 0.3662 | 0.0926 | 0.0698 |
| #2 | Batch Size Optimization | 128 | 0.7324 | 0.4854 | 0.542 | 0.3511 | 0.1077 | 0.1176 |
| #3 | 64 | NA | 0.4866 | NA | 0.3950 | NA | **0.0667** |
| #4 | 32 | **0.8403** | **0.8241** | 0.6468 | 0.6151 | 0.0879 | 0.0960 |
| #5 | Filter Size Optimization | (16,32…,96) | 0.8155 | 0.8407 | 0.6392 | 0.5702 | 0.0869 | 0.0682 |
| #6 | (8,16…,48) | 0.8029 | 0.7829 | 0.6236 | 0.5956 | 0.0749 | 0.0772 |
| #7 | (4,8…,24) | 0.7524 | 0.7506 | 0.5651 | 0.4506 | 0.0724 | 0.1207 |
| #8 | Dense Layer Size Optimization | 64 | 0.7686 | 0.7210 | 0.5591 | 0.5463 | 0.0748 | 0.1112 |
| #9 | 32 | 0.7439 | 0.7588 | 0.5562 | 0.5595 | 0.0729 | 0.0741 |
| #10 | 16 | 0.7216 | 0.5565 | 0.5284 | 0.3693 | 0.0772 | 0.0739 |
| #11 | L2 Regularization Optimization | 0.01 | 0.7575 | 0.6772 | 0.5444 | 0.4912 | 0.0715 | 0.0751 |
| #12 | 0.1 | 0.7881 | 0.7175 | 0.5204 | 0.5188 | 0.0779 | 0.1092 |
| #13 | 0.5 | 0.7705 | 0.5561 | 0.5608 | 0.3491 | 0.0770 | 0.2470 |
| #14 | L1\_L2 Model | (0.01), (0.01) | 0.7663 | 0.5288 | 0.5666 | 0.4142 | 0.0785 | 0.1144 |
| #15 | Epoch Size Optimization | Early Stopping Function (Patience=1) | 0.8284 | 0.8199 | **0.6551** | **0.6219** | **0.0695** | 0.0675 |

# **5 Conclusion and future works**

**5.1 Conclusion**

In this study, the image classification and the prediction methods are applied to detect race, gender, and age of the given input data which is 200 X 200 pixels. For the optimal neural network modeling’s parameters, manual hyperparameter tuning is applied, and the final parameters are found as 32 for Batch Size, (16, 32, 48, 64, 80, 96) for filter sizing, 64 for the dense layer output, “0.01” for the L2 regularization, and the early stopping function with the patience = 1 epoch. This chosen model, which is model #15 in Table 2, is achieved 81.99% accuracy on gender classification, 62.19% accuracy on race classification, and 0.0675 MAE loss on age prediction. The best Gender prediction is reached on model #4 with the accuracy of 82.41%. The results showed that normalized age prediction could be performed better with the more trainable parameters with the more computing resources. For the gender and race prediction, we need to implement more training, and use extensive dataset comparing to our chose dataset which is UTKFace dataset. On the other hand, the brightness, and the old photos observed as possible source of problems and could be the possible improvement can perform with more pre-processing in this side of the study.

**5.2 Future Works**

Given our results, the next steps for our group will be to fully transition to video inputs and to complete even further training and testing. We were not able to successfully implement the OpenCV classifier, an objective we could work towards. Moving forward, our project will try to further minimize current sources of error such as lighting, facial emotions, pose, etc. We are not fully satisfied with the results on race accuracy, which represents a widespread problem within facial recognition and leads to biases. As such, we will try to root out any sources of racial biases within the algorithm to solve this problem.

**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] C. Liu and H. Wechsler, “Gabor feature based classification using the enhanced Fisher linear discriminant model for face recognition,” *IEEE Trans. Image Process.*, vol. 11, no. 4, pp. 467–476, 2002, doi: 10.1109/TIP.2002.999679.

[6] E. Makinen and R. Raisamo, “Evaluation of gender classification methods with automatically detected and aligned faces,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 3, pp. 541–547, 2008, doi: 10.1109/TPAMI.2007.70800.

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

[8] 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.

[9] 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.

[10] P. J. Phillips, H. Wechsler, J. Huang, and P. J. Rauss, “The FERET database and evaluation procedure for face-recognition algorithms,” *Image Vis. Comput.*, vol. 16, no. 5, pp. 295–306, 1998, doi: 10.1016/s0262-8856(97)00070-x.

[11] N. Zhang and W. Deng, “Fine-grained LFW database,” *2016 Int. Conf. Biometrics, ICB 2016*, pp. 1–11, 2016, doi: 10.1109/ICB.2016.7550057.

[12] 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-Janua, pp. 4352–4360, 2017, doi: 10.1109/CVPR.2017.463.

[13] R. Grosse, “Lecture 9 : Generalization,” *Lect. Note*, vol. 1, no. 1, pp. 1–11, 2018.

[14] A. Greco, G. Percannella, M. Vento, and V. Vigilante, “Benchmarking deep network architectures for ethnicity recognition using a new large face dataset,” *Mach. Vis. Appl.*, vol. 31, no. 7–8, pp. 1–13, 2020, doi: 10.1007/s00138-020-01123-z.

**Contributions**

**Goksu Avdan:** Meeting organization and communication over Zoom, and all code writings. Methodology and Experiments writing in the proposal. Materials & Methods, Preliminary Results writing in the midpoint report. Dataset & Features, Models & Methods, Results, and Conclusion & Discussion writing in the poster presentation. Abstract, Materials & Methods, Results & Discussion, Conclusion writing in the final report. All formatting and submission processes for the proposal, midpoint report and final report.

**Dipak Subramaniam:** Team scribe, initial research, midpoint report and power-point coordination, poster introduction and future works, final report related works section and future works section.

**Hari Krishna Pathakamuri:** Dataset gathering, research on gathering reference code for CNN, introduction writing on midpoint report, and final report, and helping models & methods section on poster.

**Mingxuan Bai:** Helped writing to related works section in midpoint report.