Age Classification Using an Optimized CNN Architecture

M. Fatih Aydogdu
Department of Computer Engineering
TOBB University of Economics and Technology
Ankara, Turkey
E-mail: mfaydogdu@etu.edu.tr

M. Fatih Demirci
Department of Computer Engineering
TOBB University of Economics and Technology
Ankara, Turkey
E-mail: mfdemirci@etu.edu.tr

ABSTRACT

With growing data size in multimedia systems, the need for successful image classification and retrieval systems becomes vital. Nevertheless, the performance of such systems is still limited for real-world applications. In this paper, we propose an optimized Convolutional Neural Network (CNN) architecture for the age classification problem. In order to justify the structure and depth of the proposed CNN-based framework, comprehensive experiments on a number of different CNN architectures are conducted. Based on the fitness of the age classification results with respect to success-error ratios, training times, and standard deviations of success rates; using exact, top-3 and 1-off criterion, the CNN architecture involving 4 convolutional layers and 2 fully connected layers is found to be superior to the other CNN-based architectures with different number of layers. We evaluate our method on a face database consisting of more than 55,000 images.

CCS Concepts

• **Computing methodologies Neural networks; Object identification; Visual inspection.**

Keywords

Age estimation; deep learning; convolutional neural networks

1. INTRODUCTION

A face image carries a number of human characteristics useful for several purposes including face recognition, synthesis, and verification. Gender, age, expression, and ethnicity are a few examples for such characteristics. Although age is less studied than the others, automatic age classification using facial images has become increasingly important to a large amount of applications in both industry and academia [3]. Given a face image, the objective of this problem is to estimate or classify the age rather than identify the individual. As an example, application, age classification may allow vendine machines to decide whether tobacco products should be sold to the current customer. Moreover, which age groups are most interested in a specific product may also be reported by applications using automatic age classification. Despite the fact that companies know exactly their sales figures, identifying their customer ages cannot be done without such an application.

Although people have the ability to perform the accurate age classification with high accuracy [6], building an automatized classification system using only facial images remains difficult. This is mainly due to the fact that face aging is personalized and people have different aging patterns, which depend on several internal and external factors, e.g., gender, race, health condition, makeup, and ethnicity [5]. A number of frameworks in the literature performs age classification/estimation by modeling facial appearance and aging pattern. One such approach is presented by Lanitis et al. [10], who generate a statistical model of facial appearance used as the basis for obtaining a compact parametric description of face images. Computing the aging pattern based on face images without taking into consideration the other internal and external factors is the main limitation of this technique. Given a set of face images that belong to the same person, the AGES framework [5] models the aging pattern by computing a representative subspace. Although AGES does not use the factors affecting the aging patterns either, the sequence of a person's face images taken in different times is a lot more helpful in modeling the aging pattern. The aging pattern of an unseen face image is computed by the projection in the subspace that reconstructs the face image with minimum reconstruction error. The lack of complete aging patterns yields incomplete training data. To deal with this issue, an iterative learning algorithm that estimates a part of the missing personal aging pattern using the global aging pattern model is adopted.

The existing techniques on face classification in the literature usually consists of two steps: image representation and age classification. Besides the ones discussed above, the active appearance model (AAM) [17, 13, 20], patch-based model [24], anthropometric model [9], and age manifold [4] are the most popular. Several approaches formulate the age classification as that of multi-class classification [21] or regression [4, 23].

Instead of performing age classification using a few selected features extracted from the image representation step as done by many of the previous approaches, we use convolutional neural networks to learn deep features for the age classification task, and establish a state-of-the-art result on a face database consisting of more than 55k near-frontal face images. In recent years, CNN have attracted substantial research attention due to their superior performance on several computer vision tasks such as image classification [26, 19] and object detection [7, 15]. We demonstrate a similar performance gain with a simple network architecture.

In our previous work [1], we compare the performance of three existing CNN architectures consisting of 6 layers [11], 18 layers

ICIET '17, January 10-12, 2017, Tokyo, Japan

DOI: http://dx.doi.org/10.1145/3029387.3029392

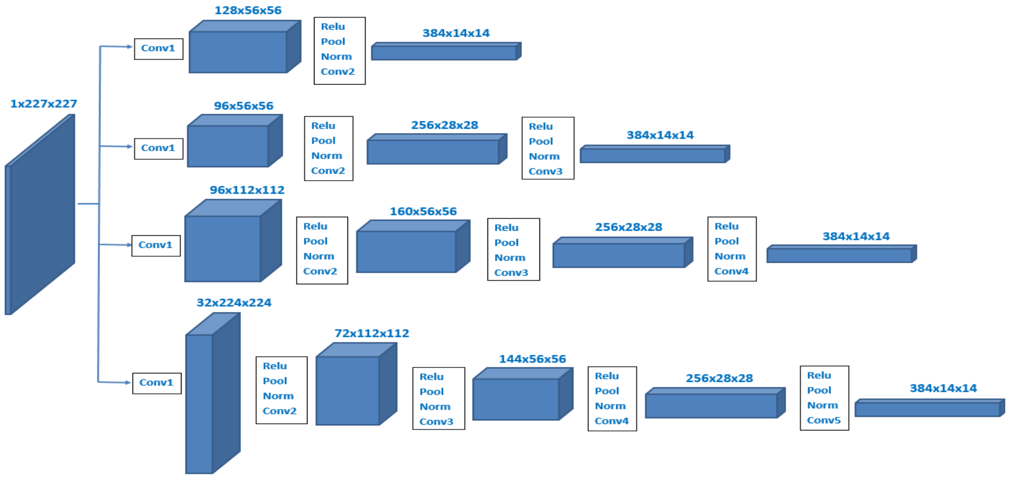

Figure 1. The structure of the convolutional stages involving convolutional, rectified linear unit, normalization and pooling layers. The data volumes are shown after each convolutional layer.

(ResNet-18), and 34 layers (ResNet-34) [8] for age classification. The results show that the CNN with 6 layers [11] outperforms the ResNet architectures although ResNet18 and ResNet-34 are successfully used on the ImageNet classification dataset [18] with 1000 classes. Our proposed framework obtains even better classification scores mainly due its optimized architecture for the age classification problem with 6 different classes.

The rest of the paper is organized as follows. After introducing the networks in the next section, we present the dataset and preprocessing steps applied to each face image in Section 3. We describe the experiments and present the performance results along with our observations in Section 4. We conclude the paper in Section 5.

2. NETWORKS

In our previous work [1], we show that for an age classification problem dealing with 6 classes, 6-layer plain network [11] composing of 3 convolutional layers and 3 fully connected layers outperforms the residual ResNet-18 and ResNet-34 architectures [8]. Following our observations, we concentrate on the plain networks around 6 layers and carry out different experiments to find the most suitable CNN architecture for age classification in terms of both performance and training time in this work. Instead of dealing with the effects of different types of CNN layers on the classification performance, we examine the effect of the number convolutional layers and fully connected layers on the classification performance. To do this, different architectures are trained from scratch and their performance scores are computed. The CNN architectures studied are composed of two different network stages, namely the convolutional stage and fully connected stage. Since convolutional layers and fully connected layers are the key elements of the defined stages, the stages are named by these layers.

2.1 Convolutional Stages

In the proposed work, a number of convolutional stages with varying depths are generated. The depth of a convolutional stage is determined based on the number of convolutional layers in the stage. The number of the convolutional layers of the convolutional stages used in this paper starts from 2 and extends up to 5. The structure of the generated convolutional stages are shown in Figure 1 where the number of convolutional layers included in convolutional stages increases from top to bottom. Except the first convolutional layer, all such layers are followed with Rectified Linear Unit (ReLu) layers, Maximum Pooling layers, and Normalization (Norm) layers. All of the convolutional stages are structured so as to process gray scale images in 227x227 resolution. The dimensions of the output volumes for all the convolutional stages are 384x14x14. Since this volume is equal for each convolutional stage, it is possible to connect the convolutional stages to the same fully connected stages. This allows us to make fair comparisons among the performances for all generated CNNs.

2.2 Fully Connected Stages

The convolutional stages are followed by 4 different fully connected stages. Figure 2 shows this structure. As shown at the bottom of this figure, the maximum depth of the fully connected stage is 4 and the depth decreases to 1 as moving from bottom to top. In the generated fully connected stages, while connecting a fully connected layer to the proceeding one, ReLu and Dropout layers with a drop out ratio of 0.5 are used. All of the fully connected stages are capable of processing 384x14x14 input data volumes. The number of the output nodes of the fully connected stages is equal to 6, which is the number of the age classes used in the proposed work.

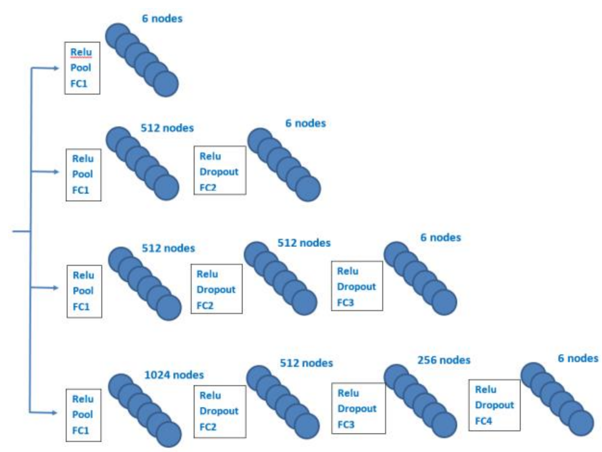

Figure 2. The structure of the fully connected stages involving rectified linear unit, dropout and fully connected layers and the nodes after each fully connected layers

A total of 16 CNN architectures are generated by uniting each of the 4 convolutional stages with each 4 fully connected stage mentioned above. The training times of the CNN architectures and performance results according to various criteria along with description of the face dataset and our preprocessing steps are presented in the following sections in detail.

3. DATASET AND PREPROCESSING

Although the dataset and preprocessing steps we used in this paper are the same as our previous study [1], we briefly mention them in this part to make the paper self-contained. The proposed work is evaluated on the Album-2 of Craniofacial Longitudinal Morphological Face Database (MORPH) [16]. As shown in the upper two rows of Figure 3, the images in the dataset contain frontal color photographs of individuals. The images are recorded in varying zoom ratios; and the vertical and horizontal resolution of the images are not constant. Therefore, the number of pixels belonging to the face zones differ excessively in the recordings. Moreover, the directions of the faces with respect to image edges are different. A total of 55,134 records of 13k different individuals are present in the dataset and the ages of the individuals vary from 16 to 77. The dataset provides data diversity by containing records of individuals from different genders, moods and races.

In deep learning, it is desirable to have an uncomplicated binding function between input and output in order to simplify the training procedure. For the age classification problem in this study, this is done with simple preprocessing operations whose objective is to equalize the amount of relevant information taken from each image in the dataset. Furthermore, assimilating the structure of images to each other is another purpose of the preprocessing step. After the images are converted to grey scale, they are rotated in such a way that the vertical direction of the faces in the images become parallel to horizontal edge of the images. While determining the rotation angles needed for each image, four variants of Viola Jones algorithm [22] are used. These are classification and regression tree (CART) based [25] and Decision stump based [2] eye detections algorithms and Local binary patterns (LBP) based [14] and CART based face detection [12] algorithms. By varying the eye detection and face detection thresholds of the algorithms; and using them in a nested fashion, it is possible to find the rotation angles of each image.

Once the image rotation is done, the visages are cropped from the images to equalize the amount of relevant information taken from each image. To detect the visages, we use LBP based [14] and CART based [12] face detection algorithms. The crops taken from each image are rescaled to have a 227x227 resolution. The third and fourth rows of Figure 3 presents the preprocessed versions for the corresponding images shown in the first and second rows, respectively. One may observe that the ratio of facial pixels to non-facial pixels in each preprocessed image is closer to each other than the one obtained in the original images.

After the preprocessing step, the horizontal flips of the preprocessed images are added to the dataset to double its size. 6 different age classes are then generated from the database. Table 1 presents the age borders of the classes and the number of images in each class. While 80% of the images are used in training, 20% is used in tests and half of the test images are used in validation. If an image belonging to an individual is selected in the training or test set, its horizontal flip and all other images belonging to the same individual are placed to the same set. As detailly described in the previous study [1], some of the recordings belonging to

Figure 3. The first two rows present the original images, while the last two rows show their preprocessed versions.

Table 1. The Number of Images in Classes

16-20	21-27	28-34	35-41	42-48	49-
18910	23638	18990	22784	16654	9292

the same individual have different born dates. Thus, some corrections are applied to these recording to have more reliable age information to be used in the training and test stages.

4. EXPERIMENTS AND PERFORMANCE EVALUATION

As mentioned before, our previous work [1] shows that the CNN architecture consisting of 3 convolutional and 3 fully connected layers obtains better age classification scores for the face database with 6 classes than the ones with a higher number of layers. Thus, we focus on CNNs with around a total of 6 layers. Specifically, we generate 16 CNNs starting from 2 convolutional and 1 fully connected layers to 5 convolutional and 4 fully connected layers. We train each CNN using the same train, validation and test sets. The learning rate is fixed to 0.001 while the momentum and weight decay values are selected as 0.9 and 0.0001 respectively. Each CNN is trained for 70 epochs since error rates for all of the CNNs trained tend to increase before reaching this point mainly due to the overfit caused by the size of the training dataset. The processes run on a single NVIDIA Quadro K4000 192 bit GPU with 3GB memory. Figure 4 depicts the average training duration for each network. According to the training times, the duration of the training increases as the number of the convolutional layers grows. This is consistent with our expectations since adding convolutional layers increases the data volume to be processed.

Increasing the number of fully connected layers does not always lengthen the training time. As opposed to the data volumes, fully connected layers insert nodes to the architectures, which can diversify memory access times. Therefore, it is possible to have longer training times for CNNs with less number of fully connected layers.

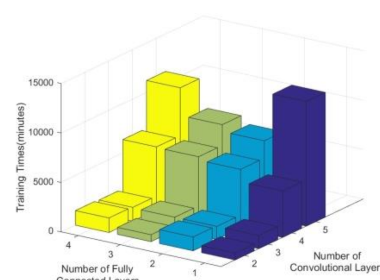

Figure 4. Training times for 16 CNNs using 70 epochs

To measure the performance for each CNN architecture, we use 3 different criteria:

• **Exact Success:** This criterion is successful when the top scored class is the same as the exact class of the query.

• **Top-3 Success:** Determines whether the one of the top 3 classes consists of the query class.

• **1-Off Success:** When the top scored class is the same as the exact class of the query or it belongs to one of the neighbor classes of the query, this criterion is successful.

The validation errors for each CNN with respect to exact success and top-3 success are shown in Figure 5 and Figure 6, respectively. One may observe that CNNs with 1 fully connected layer and 4 fully connected layers have less exact success and top-3 success rates than CNNs with 2 fully connected layers and 3 fully connected layers. Among the CNNs with 2 fully connected and 3 fully connected layers, the former ones have smaller error score in both exact success and top-3 success criteria. Thus, the CNN with 2 fully connected layers tend to perform better while the CNN with 1 or 4 fully connected layers tend to perform worse.

Validation curves shown in Figures 5 and 6 are also used to compare the performance of CNNs having different number of convolutional layers. Regarding to the exact success criterion,

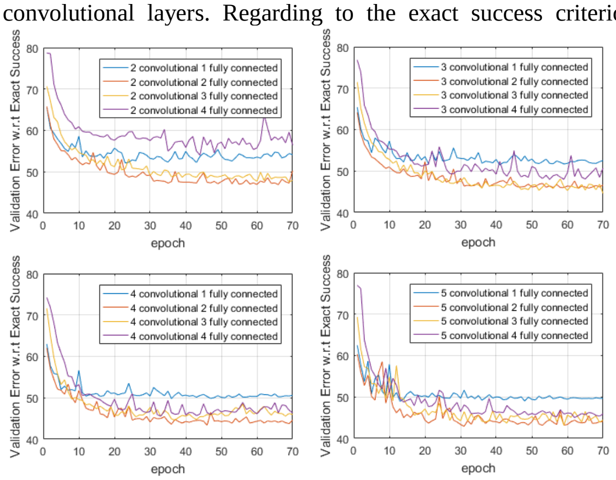

Figure 5. Validation errors of the generated 16 CNNs during 70 epoch training with respect to exact success criterion

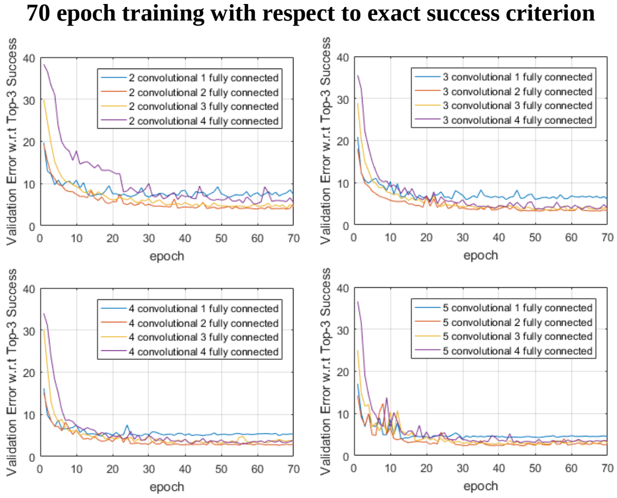

Figure 6. Validation errors of the generated 16 CNNs during 70 epoch training with respect to top-3 success criterion

CNNs with 2 convolutional layers and 3 convolutional layers do not have error rates less than 45%. On the other hand, CNNs with 4 and 5 convolutional layers achieve error rates less than 45%. Moreover, CNNs with 4 and 5 convolutional layers obtain almost the same error rate for both exact success and top-3 success criteria. Since CNNs with 4 and 5 convolutional layers have similar error rates, we do not prefer to use deeper ones requiring longer training times as observed from Figure 4. Therefore, by examining the validation curves, the optimum CNN we infer for our problem is the CNN with 4 convolutional and 2 fully connected layers.

A good classifier, in general, must have a similar level of classification capability for each different class. As a result, among CNNs with similar age classification success, the one with the lower standard deviation value with respect to age classes should be considered as a better choice. Figures 7 and 8 present the standard deviations computed using the exact and 1-off success scores for each CNN. While only validation images are used to generate the previously discussed learning curves, both validation and test images are used to generate the success and standard deviation results shown in these figures. When choosing the trained CNN versions, the epochs having minimum validation errors for the exact and top-3 success criteria are used. After validation and test images are tested with the selected CNN versions, the most successful test results using the exact and 1-off success criteria are shown in the corresponding figures.

Having a close look at Figure 7, we observe a similar behavior to the previously discussed learning curves for CNNs with 1 and 4 fully connected layers, i.e., CNNs with 2 and 3 fully connected layers are more successful than CNNs with 1 and 4 fully connected layers. It is also possible to infer that although CNNs with 2 fully connected and 3 fully connected layers have similar exact success rates, the former ones have less standard deviation values. Therefore, regardless the number of convolutional layers, CNNs with 2 fully connected layers are preferable to CNNs with 3 fully connected layers in terms of exact success. In addition, CNNs with 4 and 5 convolutional layers are more successful than their counterparts having a less number of convolutional layers. We observe that adding one more convolutional layer to the CNN with 4 convolutional layers does not increase the exact success significantly. Therefore, taking into consideration the exact success and standard deviation scores and the network training times, the CNN having 4 convolutional layers and 2 fully connected layers is preferable.

According to 1-off success results shown in Figure 8, the performance of the CNN with 1 fully connected layer is less than their variants with the same number of fully connected layers but different number of convolutional layers. When only CNNs having equal number of convolutional layers and more than 1 fully connected layer are examined in groups, we observe that the number of fully connected layers do not affect the 1-off success dramatically. Instead, the standard deviation varies with the number of fully connected layers. For all 4 cases where the number of convolutional layers varies from 2-5, CNN with 2 fully connected layers have the least standard deviations. Hence, in terms of 1-off success and the standard deviation, the CNN with 2 fully connected layers are preferable. Moreover, we also observe from this figure that CNNs with 4 and 5 convolutional layers are almost equally successful and perform better than CNNs with 2 and 3 convolutional layers. Although in terms of 1-off success and standard deviation, the CNN with 5 convolutional and 2 fully connected layers slightly outperform that with 4 convolutional and

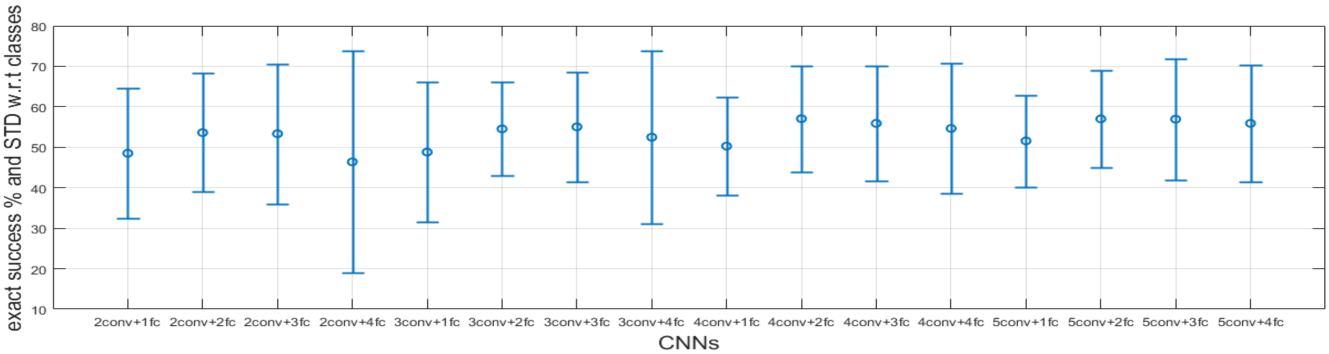

Figure 7. Exact success rate of each CNN with standard deviation while processing test and validation images

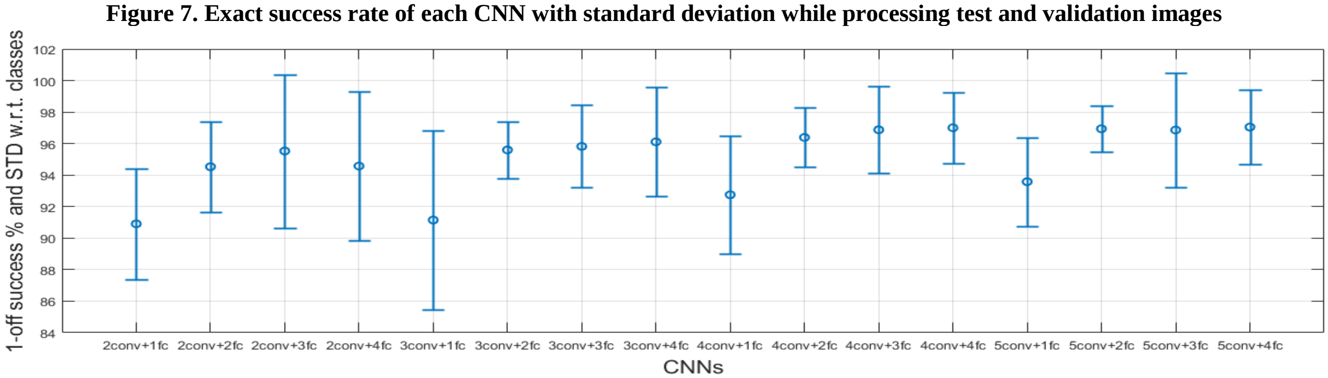

Figure 8. 1-off success rate of each CNN with standard deviation while processing test and validation images

2 fully connected layers, we still consider the CNN with 4 convolutional and 2 fully connected layers as the optimal architecture since it achieves the same performance based on the exact success and has the less training time.

Examining the result of the CNN with 2 convolutional stages and 4 fully connected stages in Figure 7, it is observed that the network has a great standard deviation in terms of exact success. In order to discuss this situation, the detailed test results of the network are shown in Tables 2 and 3. In the Table 2, while the columns contributing to both exact success and 1-off success are shown in dark green, the columns contributing to only 1-off success are shown in light green. According to the results, with the network, none of the images of which exact class is [49-] is classified correctly and only 22.56% of the images of the class [28-34] are classified correctly. Moreover, most of the images in these classes are classified as one of their neighboring classes. As shown in Figure 7, it is clear that for the CNNS with 4 fully connected layers as the number of convolutional layers increases the standard deviations decreases. Therefore, we conclude that adding fully connected layers to the CNNs which are shallower in terms of convolutional layers increases the standard deviation with respect to exact success by increases the tendency of the CNNs to classify images in some classes as one of their neighboring classes.

5. CONCLUSION

In this paper, an optimized CNN architecture for the problem of age classification is proposed. Rather than focusing on the performance effects of the network layers, the effect of having a different number of convolutional layers and fully connected layers is investigated. In order to find the optimal depth, 16 different CNN architectures are generated. Each architecture contains one of the 4 proposed convolutional stages and one of the 4 proposed fully connected stages. According to the test and validation results, the CNN containing 2 fully connected layers is the optimum considering exact success, top-3 success, 1-off success criteria and standard deviation values. The CNN containing 4 convolutional layers achieves similar performance results with the one containing 5 convolutional layers. Considering the longer training time for the CNNs with 5 convolutional layers, we conclude that the convolutional stage with 4 convolutional layers is the optimum one. Therefore, the proposed CNN architecture consists of 4 convolutional layers with 2 fully connected layers. We should note that the CNN architecture with 3 convolutional and 3 fully connected layers has

Table 2. Test results of CNN with 2 convolutional layers and 4 fully connected layers with respect to age classes

Image Classes	Number of Estimations w.r.t. Classes					
	16-20	21-27	28-34	35-41	42-48	49-
16-20	1547	2033	18	2	0	0
21-27	481	3826	310	115	10	0
28-34	37	1725	804	861	137	0
35-41	19	756	907	2111	973	0
42-48	8	154	278	1128	1734	0
49-	1	24	31	209	1365	0

Table 3. Exact Success % of CNN with 2 convolutional layers and 4 fully connected layers with respect to age classes

Image Classes	Exact Success %
16-20	42.97
21-27	80.68
28-34	22.56
35-41	44.29
42-48	52.51
49-	00.00
Mean	46.39
STD	27.35

been presented for age estimation before [11] and it is one of the architectures experimented in this study. As shown by the experiments, the proposed architecture outperforms this work according to the exact success and 1-off success criterion. In addition, our architecture results in smaller standard deviation with respect to exact success and 1-off success criterion, which is also one of the key features of a good image classifier.

For the future work of this study, we plan to use larger datasets to perform more comprehensive tests. Although, by using horizontal flips of 55k images in the dataset, the experiments are carried on more than 110k samples, the number of samples can still be increased to prevent the risk of overfitting while measuring the performance scores of the architectures.

6. REFERENCES

[1] M. F. Aydogdu, V. Celik, and M. F. Demirci. Comparison of three different cnn architectures for age classification. In *Semantic Computing (ICSC), 2017 IEEE 11th International Conference on*, pages 372–377. IEEE, 2017.

[2] M. Castrillo´n, O. D´eniz, C. Guerra, and M. Herna´ndez. Encara2: Real-time detection of multiple faces at different resolutions in video streams. *Journal of Visual Communication and Image Representation*, 18(2):130–140, 2007.

[3] Y. Fu, G. Guo, and T. S. Huang. Age synthesis and estimation via faces: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 32(11):1955–1976, 2010.

[4] Y. Fu and T. S. Huang. Human age estimation with regression on discriminative aging manifold. *IEEE Transactions on Multimedia*, 10(4):578–584, 2008.

[5] X. Geng, Z.-H. Zhou, and K. Smith-Miles. Automatic age estimation based on facial aging patterns. *IEEE Transactions on pattern analysis and machine intelligence*, 29(12):2234–2240, 2007.

[6] P. A. George and G. J. Hole. Factors influencing the accuracy of age estimates of unfamiliar faces. *Perception*, 24(9):1059–1073, 1995.

[7] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[8] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.

[9] Y. H. Kwon and N. da Vitoria Lobo. Age classification from facial images. *Computer Vision and Image Understanding*, 74(1):1–21, 1999.

[10] A. Lanitis, C. Draganova, and C. Christodoulou. Comparing different classifiers for automatic age estimation. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 34(1):621–628, 2004.

[11] G. Levi and T. Hassner. Age and gender classification using convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 34–42, 2015.

[12] R. Lienhart, A. Kuranov, and V. Pisarevsky. Empirical analysis of detection cascades of boosted classifiers for rapid object detection. In *Joint Pattern Recognition Symposium*, pages 297–304. Springer, 2003.

[13] K. Luu, K. Ricanek, T. D. Bui, and C. Y. Suen. Age estimation using active appearance models and support vector machine regression. In *Biometrics: Theory, Applications, and Systems, 2009. BTAS'09. IEEE 3rd International Conference on*, pages 1–5. IEEE, 2009.

[14] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on pattern analysis and machine intelligence*, 24(7):971–987, 2002.

[15] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[16] K. Ricanek and T. Tesafaye. Morph: A longitudinal image database of normal adult age-progression. In *7th International Conference on Automatic Face and Gesture Recognition (FGR06)*, pages 341–345. IEEE, 2006.

[17] *K. Ricanek, Y. Wang, C. Chen, and S. J. Simmons. Generalized multi-ethnic face age-estimation. In Biometrics: Theory, Applications, and Systems, 2009. BTAS'09. IEEE 3rd International Conference on, pages 1–6. IEEE, 2009.*

[18] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[19] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv:1312.6229*, 2013.

[20] A. Sethuram, K. Ricanek, and E. Patterson. A hierarchical approach to facial aging. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*, pages 100–107. IEEE, 2010.

[21] K. Ueki, T. Hayashida, and T. Kobayashi. Subspace-based age-group classification using facial images under various lighting conditions. In *7th International Conference on Automatic Face and Gesture Recognition (FGR06)*, pages 6–pp. IEEE, 2006.

[22] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–511. IEEE, 2001.

[23] S. Yan, H. Wang, X. Tang, and T. S. Huang. Learning auto-structured regressor from uncertain nonnegative labels. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8. IEEE, 2007.

[24] S. Yan, X. Zhou, M. Liu, M. Hasegawa-Johnson, and T. S. Huang. Regression from patch-kernel. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[25] S. Yu. Shiqi Yu's Homepage, 2016.

[26] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. *In European Conference on Computer Vision*, pages 818–833. Springer, 2014.