ACCURACY OF A FACE ANALYSIS SYSTEM IMPLEMENTED IN PROCESSING

Grigore-Adrian Iordachescu
University of Pitești, Romania
adi_iord@yahoo.com

Keywords: face analysis, age detection, gender detection, Processing, neural networks

Abstract: *This paper aims to measure the accuracy of a face analysis system, implemented in Processing using deep neural networks. The face analysis targets age and gender identification. If proven efficient, such a system can be used in on-line security applications. The tests are implemented using an on-line image dataset, containing hundreds of different faces in over 4000 different poses. The resulting accuracy of the system is expounded as a function of age. For age identification, the accuracy of the system was 48% across all age groups, while for gender identification the measured accuracy was 80%.*

1. INTRODUCTION

Computer vision is an umbrella term which spans a great array of techniques and technologies which aim to bring the capabilities of human vision and human visual processing to the world of computers. It can refer to simple applications such as QR code reading [1], fingerprint recognition [2], SnapChat filters [3], Magic Wand tool in Photoshop [4] or it can refer to complex applications like face recognition [5], assistive medical diagnosis [6], egomotion [7], driver-less cars [8], soccer match statistics [9] and so on. The huge array of applications makes this field a hot spot for research. One of its most important areas of research is the field of face recognition, having found uses in forensic science, state control, home security or on-line security with its most important challenge, prevention of child pornography [10]. In this context, this paper aims to test a system for age and gender identification on the basis of face analysis. This type of systems can be used for on-line security to protect the access of children to certain sites or for any other form of age or gender-based verification system, like for example on dating sites. This paper does not intend to create a novel classification algorithm, but only to test an already optimized classifier. The tests will be using an on-line database of over 4000 images.

The biggest challenge in testing a face analysis system is finding the right dataset of images. Datasets of faces in real world scenarios are difficult to come by because of copyright laws. Assembling a collection of such images under varying levels of noise, brightness and tilting angle is thus a painstaking task. One such image collection is the Adience database, found at the Open University of Israel Face Image Project [11-14]. The photos assembled in this collection are taken using smart-phones in real-world scenarios and uploaded by their respective authors on Flickr under the Creative Commons license. The faces presented in this collection have been cropped and tilted at varying angles relative to the raw images. In total, the subset chosen for the tests described later in the present paper contains 4484 images obtained from 481 unique faces. From this subset a further screening permitted the removal of all the photos for which the age of the subject was unknown or was outside of the targeted age intervals, arriving at a total of 3888 images containing 460 unique faces.

Before proceeding to the next sections of this paper, in which the face analysis system is presented and tested, a few introductory words are warranted about the environment on which the system is implemented. Processing is an integrated development environment (IDE) running on Java and containing many libraries with uses in computer vision [15]. For example, the Video library allows processing of video streams from computer or webcam. This can be used to access the local webcam in on-line security applications, before establishing the identity or age of the person looking into the webcam. Another useful library found in Processing IDE is the OpenCV library, one of the most comprehensive libraries for real-time computer vision applications [16]. But the most useful library for the field of face detection, used to implement the face analysis system described in the next section, is the DeepVision library [17], which uses the power of deep neural networks in many vision related tasks, like object detection and classification, gesture recognition, emotion detection, etc.

Similar efforts have been made to measure the neural networks accuracies in detecting ages [18] or gender [19]. In the first of these papers [18], a state of the art pre-trained deep Convolutional Neural Network (VGG-Face CNN) was used on a photo database different from the one on which the training was performed. The VGG-Face CNN was initially trained for face recognition but was modified by the authors for age classification. Another pre-trained convolutional network (GoogLeNet CNN) initially trained for image classification was also modified by the authors for age classification. The database on which the tests were performed is a larger subset of the Adience database also used in the present paper. The resulting accuracy for age classification varies wildly across age groups, with an obtained average accuracy of 45% for GoogLeNet CNN and of 60% for VGG-Face CNN [18]. In the second paper [19] the authors devised a method based on a multi-purpose neural network architecture named MobileNets and obtained a very accurate gender recognition system with an accuracy of over 98% when faced with a simple photo dataset, and an accuracy of 84.49% when faced with the very challenging Adience photo database also used in the present paper. An exhaustive review of gender classification using face images [20] shows that gender classifiers can arrive at accuracies varying from 88% to 99.3%, but the photos datasets used in the reviewed papers do not use real-life challenging photos as the one used in the present paper. For example, the 99.3% gender accuracy was obtained when using a KNN classifier on the Stanford University Medical Student (SUMS) frontal facial images database. The way a photo database is chosen has a large influence on the measured accuracy of an age or gender classification system.

2. FACE ANALYSIS SYSTEM

The application whose efficiency is aimed to be measured in the current paper is a benchmark application in Processing. Similar age and gender detection systems are to be found in the examples included with the DeepVision library of Processing. The standard application was slightly modified as in the logical flow diagram of Fig. 1, for to include automatic loading of new images and a comparator between the answer given by the neural network used to detect age (or gender) and the real answer. The real age and gender for each photo were found in a text file included with the original images dataset downloaded from the Open University of Israel Face Image Project [11].

First stage of the program (Fig. 1) is the importing of necessary libraries. The only library needed is DeepVision. Because the program uses images which are on the local drive, there is no need for the Video library. Next step is the declarations and initializations of variables. This step includes the instantiation of the DeepVision class. This class has many member functions which can automatically create classifiers using already trained deep neural networks. For example, if one needs a general object detection and classification, one can use the member functions *.createYOLOv3()* or *.createYOLOv4()* to generate an already trained classification neural network of the YOLO ("You Only Look Once") type. For the present application, there is no need for such a classifier. The DeepVision class member functions which are needed are instead *.createCascadeFrontalFace()* for a face detection algorithm based on a neural network, *.createGenderClassifier()* for a gender classifier based on a neural network and *.createAgeClassifier()* for an age classifier. The classifiers defined previously will be setup at the next stage of the flow diagram. Then each image in the dataset folder is loaded and analyzed.

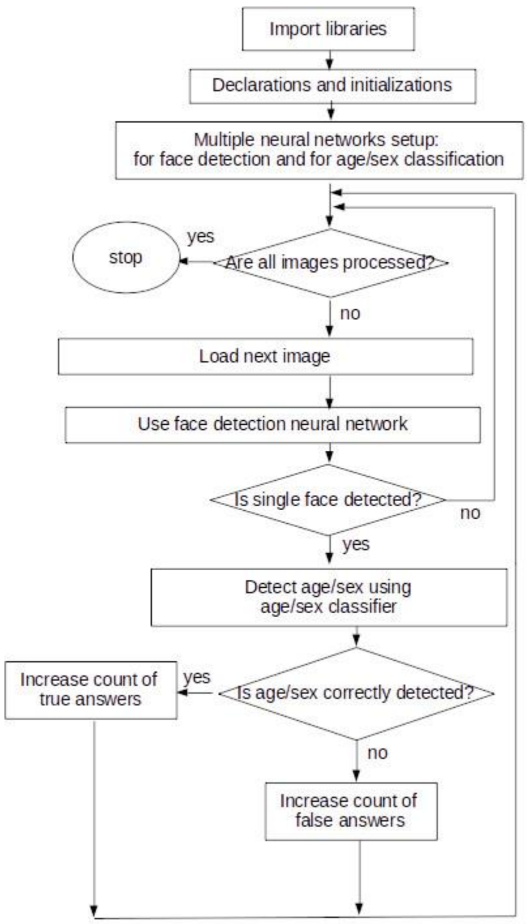

Fig. 1 Logical flow diagram for the face analysis testing system

For each image the first DeepVision algorithm used is for face detection. If no face is detected in the image or if more than one face is detected, then the algorithm goes to the next image in the dataset. If only one face is detected, then the algorithm can use the age classifier (or the gender classifier respectively). If the age is correctly detected by comparison with the information found in the text file of the dataset, then the counter memorizing the number of correct answers is increased. If the age is not correctly detected, the counter memorizing the number of false answers is increased instead.

3. RESULTS

The initial image dataset contained 4484 different images. From this dataset all photos of faces for which the age was unknown were removed. The rest of the photos were divided in 8 different groups according to age, expressed in years: (0-2), (4-6), (8-12), (15-20), (25-32), (38-43), (48-53), (60-100). As it can be noticed, the span between two consecutive age groups must be large enough to clearly differentiate the two groups. All images with faces whose ages were in the borders between two consecutive groups were also removed from the dataset. The final result is a set of 3888 photos of 460 different faces whose age distribution is that of Fig. 2. Because some of the faces appeared in more photos than others, the distribution of photos by age group (Fig. 3) is different than the distribution of faces by age group (Fig. 2).

In Fig. 4 one can notice the measured accuracy of the answers regarding the age of faces in the photos for two cases: a. for the age classifier implemented with neural networks in Processing (solid blue) and b. for a random-answer algorithm which chooses one of the age intervals with equal probability (hatched yellow). The problem with this specific analysis is that the number of photos in each age group (Fig. 3) has a direct influence on the accuracy of the answers in both cases. It is obvious that when choosing an age group having more photos in the dataset the answers will be true more times than when choosing an age group having fewer photos in the dataset. But what can be noticed is that accuracy is always higher for answers given by age classifier than for answers given randomly. And this statement is valid across all age groups.

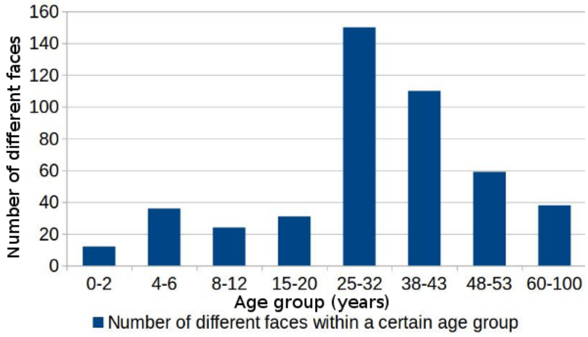

Fig. 2 Distribution of available faces by age groups

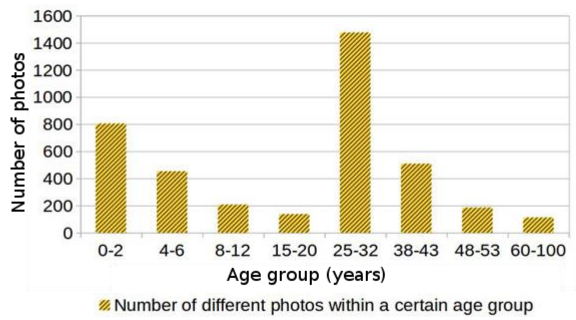

Fig. 3 Distribution of available photos by age group

Another thing worth noting is that the increase in efficiency between the random algorithm and the age classifier is highest for [0-2] and [60-100] age groups. On the horizontal axis of Fig. 4 the age groups correspond to the age groups returned as answers by the algorithms, and not the real age groups of the analyzed faces. If one makes the weighted average of the accuracies measured for the age classifier in Fig. 4, the result obtained is 48%. If one makes the weighted average of the accuracies measured for the random classifier in Fig. 4, the result obtained is 12.5%. The weights in these calculations are the number of times an age group is chosen by each algorithm.

If on the contrary, an analysis is done from the point of view of the real age groups of the analyzed faces, then the measured accuracy is represented in Fig. 5. In this figure, the age groups on the horizontal axis represent the real age groups of the faces and thus this analysis answers the question “What is the probability that the algorithm will give us the right age for a person appearing in the photo?” The advantage of this type of analysis is that the measured efficiency is not directly influenced by the number of photos in each age group. As in the previous example, the measurements were done in two scenarios: for the age classifier implemented with neural networks (solid blue) and for a random type answering algorithm which chooses one of the age intervals with equal probability (hatched yellow). An interesting fact is that if one makes the weighted average of the accuracies measured for the age classifier in Fig. 5, the result obtained is still 48%, the same accuracy calculated earlier when discussing the confidence in the answers returned by this classifier (Fig. 4). If one makes the weighted average of the accuracies measured for the random classifier in Fig. 5, the result obtained is still 12.5%, the same value as for the average calculated in Fig. 4. For the case of Fig. 5 the weights used for averaging are the number of faces in each age group (Fig. 3). It can be said that on average, the accuracy in finding the right age is increased by using the age classifier, from 12.5% by randomly choosing one of the 8 groups, to 48% for the case of the age classifier.

Another interesting fact of Fig. 5 is that the probability for a right answer is not uniformly increased for all age groups relative to the random algorithm.

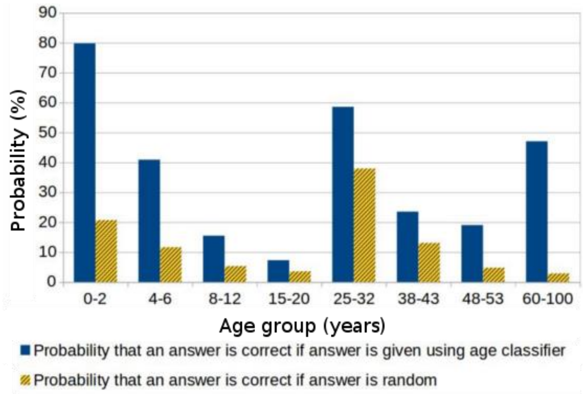

Fig. 4 Probability an answer is correct when testing the Adience photo dataset: solid blue - answer is given by age classifier; hatched yellow - answer is given randomly

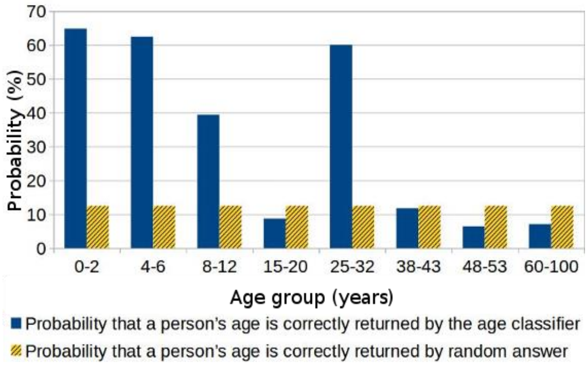

Fig. 5 Probability the real age is determined when: answer is given by age classifier (solid blue); answer is given randomly (hatched yellow)

It can be noticed that the age classifier is most accurate for recognizing certain age groups: (0-2), (4-6) and (25-32). For example, for the age group (0-2) it can be noticed that an answer for this age group is true 80% of the time in the given dataset (Fig. 4) and a person having this age is recognized by the classifier 65% of the time (Fig. 5). For the next age group (0-4) an answer indicating this age group is true 40% of the time (Fig. 4) while a person of this age group is correctly recognized 65% of the time (Fig. 5). The results after measuring the efficiency of the gender classifier are presented in Fig. 6. It can be seen that across all age groups the gender classifier (solid blue) is more efficient than a random answer (hatched yellow) in recognizing the gender of the person in a photo. The accuracy of the classifier is almost constant on all age groups and restricted to the 70%-90% interval, which is not similar to the human experience of evaluating the gender, whose efficiency has a sharp decrease when evaluating babies (0-2 age group).

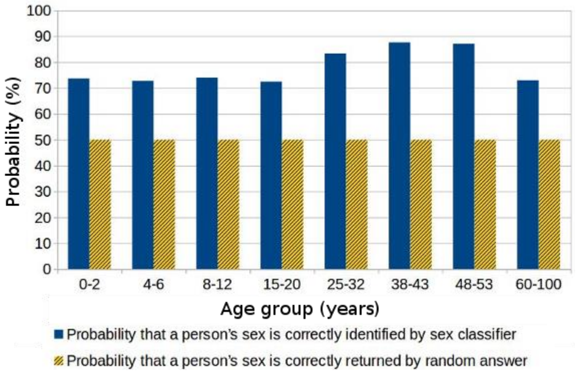

Fig. 6 Probability that a person's gender is correctly identified as a function of the age of that person, for two cases: solid blue - answer is given by gender classifier; hatched yellow - answer is given randomly

4. CONCLUSIONS

This paper presents the process and the results for measuring the accuracy for two benchmark classifiers based on neural networks. The classifiers for age and gender are included in one of the libraries of Processing IDE (the DeepVision library). The classifiers themselves are already optimized and there is no way for the user to modify the number of layers or the number of nodes of the neural networks. Besides, the networks are already trained, so the weights of the networks are also fixed. Before using these optimized classifiers in on-line security applications, one must first know their accuracy. As there is no information regarding these accuracies in the literature, this paper fills this deficiency by a series of measurements applied on a real-life and thus challenging faces dataset. After measuring the accuracy of the age classifier, it was inferred that it is best used for targeting persons of certain age groups: (0-2), (4-6) and (25-32) for which the algorithm identifies the correct age in more than 60% of cases (Fig. 5). Across all age groups, the algorithm has an average accuracy of 48%, similar to the accuracies (45% and 60%) of other neural networks which were tested on similarly difficult real-life photo databases and which were cited in the introduction [18]. The accuracy of the gender classifier is relatively uniform and has an average of 80.3% across all age groups, always higher than the random-guess approach which has 50% accuracy (Fig. 6). This result is also in tune with the 84.49% accuracy obtained using a state-of-the-art gender classifier tested on the same type of real-life photos as those of the present paper [19].

5. REFERENCES

[1] Alex Karpus, *"An OpenCV implementation of a QR barcode reader for the iPhone and Windows mobile platforms"*, Honours Project at Carleton University, December 2009

[2] Le Hoang Thai and Ha Nhat Tam, *"Fingerprints Classification through Image Analysis and Machine Learning Method"*, IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 3, No 7, May 2010

[3] Swapnil Bhardwaj, MadhurimaHooda and Saru Dhir, *"An Era of Face Filters"*, International Journal of Innovative Technology and Exploring Engineering (IJITEE), Vol. 9 Issue 1S, November 2019

[4] Eric N. Mortensen, *"Vision-assisted image editing"*, ACM SIGGRAPH Computer Graphics, Volume 33, Issue 4, November 2000

[5] Shervin Emami and Valentin Petrut Suciu, *"Facial Recognition using OpenCV"*, Journal of Mobile, Embedded and Distributed Systems, vol. IV, 2012

[6] Jerome Thevenot, Miguel Bordallo Lopez and Abdenour Hadid, *"A Survey on Computer Vision for Assistive Medical Diagnosis From Faces"*, IEEE Journal of Biomedical and Health Informatics, Vol. 22, No. 5, September 2018

[7] Yusuf Sait Erdem, Feyza Galip, Ibrahim Furkan Ince and Md. Haidar Sharif, *"Estimation of Camera Ego-Motion for Real-Time Computer Vision Applications"*, International Journal of Scientific Research in Information Systems and Engineering (IJSRISE), Vol. 1, Issue 2, December 2015

[8] Balika J. Chelliah,Vishal Chauhan, Shivendra Mishra and Vivek Sharma, *"Advancement of Driverless Cars and Heavy Vehicles using Artificial Intelligence (Object Detection) "*, International Journal of Engineering and Advanced Technology (IJEAT),Vol. 9 Issue 1, October 2019

[9] Hayk H. Tepanyan, *"Soccer Stats with Computer Vision"*, Stanford University Publishing House, 2017

[10] N. Sae-Bae, X. Sun, H. T. Sencar and N. D. Memon, *"Towards automatic detection of child pornography"*, 2014 IEEE International Conference on Image Processing (ICIP), 2014

[11] Face Image Project Data, The Open University of Israel - Adience Face Image Project, https://talhassner.github.io/home/projects/Adience/Adience-data.html , last accessed 01.09.2021

[12] Eran Eidinger, Roee Enbar, and Tal Hassner, *"Age and Gender Estimation of Unfiltered Faces"*, Transactions on Information Forensics and Security (IEEE-TIFS), special issue on Facial Biometrics in the Wild, Volume 9, Issue 12, pages 2170 - 2179, Dec. 2014

[13] Gil Levi and Tal Hassner, *"Age and Gender Classification Using Convolutional Neural Networks"*, IEEE Workshop on Analysis and Modeling of Faces and Gestures (AMFG), at the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Boston, June 2015

[14] Tal Hassner, Shai Harel, Eran Paz and Roee Enbar, *"Effective Face Frontalization in Unconstrained Images"*, IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Boston, June 2015

[15] S.V. Viraktamath, Mukund Katti, Aditya Khatawkar and Pavan Kulkarni, *"Face Detection and Tracking using OpenCV"*, The SIJ Transactions on Computer Networks & Communication Engineering (CNCE), Vol. 1, No. 3, July-August 2013

[16] Daniel Lelis Baggio, *"Mastering OpenCV with Practical Computer Vision Projects"*, Packt Publishing, 2012

[17] GithHub – cansik/deep-vision-processing: Deep computer-vision algorithms for the Processing framework, https://github.com/cansik/deep-vision-processing , last accessed 01.09.2021

[18] Zakariya Qawaqneh, Arafat Abu Mallouh and Buket D. Barkana, *"Deep Convolutional Neural Network for Age Estimation based on VGG-Face Model"*, 2017, https://arxiv.org/abs/1709.01664 , last accessed 01.09.2021

[19] A. Greco, A. Saggese, M. Vento and V. Vigilante, *"A Convolutional Neural Network for Gender Recognition Optimizing the Accuracy/Speed Tradeoff"*, in IEEE Access, vol. 8, pp. 130771-130781, 2020

[20] Dhanashri P. Lale, Kailash J. Karande, “Gender Classification using Face Images: A Review”, International Journal of Latest Trends in Engineering and Technology (IJLTET), Vol 7, issue 2, July 2016