Age estimation from face images with deep leaning

Dario d'Andrea
University of Twente
The Netherlands
d.dandrea@student.utwente.nl

Diego Antonelli
University of Twente
The Netherlands
d.antonelli@student.utwente.nl

Abstract

The aim of this paper is to describe how we faced the problem of age estimation from facial images with deep learning. Our work relies on DEX, an already existing paper in which is shown an analogous study [1]. We rebuilt a similar model with a convolutional neural network (CNN) using the VGG-16 architecture pretrained on ImageNet for image classification. Experiments and results are obtained using IMDB-WIKI, the largest public dataset of face images with age labels that the owners released for academic research purpose. We considered age prediction as a deep classification problem with the softmax function in the final layer of the neural network. Moreover, we investigated which part of the face has more influence in age estimation. In particular, we examined forehead, eyes, nose, mouth and chin.

Keywords - age estimation, deep learning, CNN

1. Introduction

Age estimation from facial images is a challenging task for research in the field of computer vision and pattern recognition that has been tackled in the recent years. Many studies have been proposed mostly on real-biological age, but also on apparent age that is the age as perceived by other humans. In addition, there has been considerable research work on the prediction of other facial features such as gender, ethnicity, hair color and expressions.

The motivations associated with these studies are related to a growing demand of extracting biometric information from face images for many of the new intelligence applications [2]. Some examples to understand in which contexts age estimation has a key role are as follows: (i) access control, i.e. restricting the access of minors in places where only adults are allowed or to avoid selling them alcohol and tobacco; (ii) human-computer interaction, i.e. adapting information of a board, such as an advertising panel, according to the age of the person; (iii) law enforcement, i.e. automatic identification of the age of a suspect; (iv) surveillance, i.e. automatic detection of minors in places where they are not allowed to be [2]. There are many aspects that make age prediction a difficult task: (i) aging processes are affected by external factors and human genetics; (ii) males and females can have different aging characteristics; (iii) people of different races may have different aging cues [3]. In any case, as it is reasonable to imagine, the most informative features are typically located where wrinkles appear, such as the eye and mouth corners, nasolabial folds, and cheeks [4].

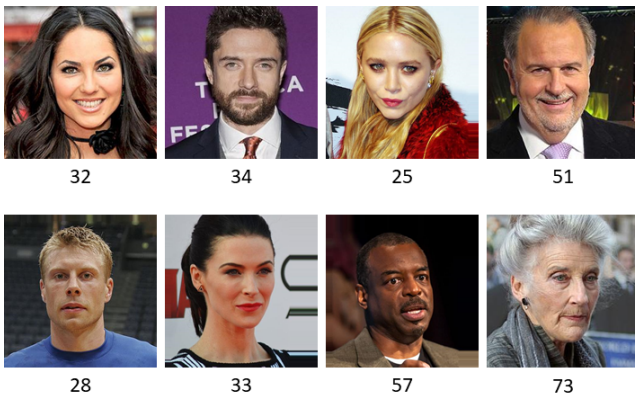

Figure 1. Facial images with ages from IMDB-WIKI dataset

Another factor to consider is related to the quality of the pictures used for the research study and how well they show the face of the person. Ideally it is better to deal with pictures containing a frontal image and an appropriate margin. Considering that, it is necessary to carefully preprocess the pictures in order to detect and align the face correctly before to give them as input to the age estimator. In literature there are many techniques used for face detection as the off-the-shelf

Mathias *et al.* [5] detector used in DEX system or other approaches based on machine learning [6, 7].

In this paper we used IMDB-WIKI [8], the largest public dataset of face images with age labels available that contains more than 0.5 million images of celebrities crawled from IMDb and Wikipedia. Some of the images are shown in Fig. 1.

The aim of our work is to tackle the estimation of real-biological age with deep learning. We are motivated by recent related studies, such as DEX [1], thus we rebuilt a similar model with a convolutional neural network (CNN) using the VGG-16 architecture [9] pretrained on ImageNet [10] for image classification. We started from pretrained CNN on the large ImageNet dataset to benefit from the representation learned to discriminate 1000 object categories in images, and to have a meaningful representation and a warm start [2].

In the common assumption age estimation is a regression problem as the age is a continuous variable. However we considered it as a deep classification problem where the age values are associated to n classes and the softmax function is used in the final layer of the neural network. Training our CNN for classification improved the results over standard regression.

Moreover, extracting region-specific features of local aging cues we investigated which part of the face has more influence in age estimation. We split the images in 4 parts and we evaluate which of them (forehead, eyes, nose, mouth and chin) are more relevant to predict the age.

In the next section we describe the details of our method including face detection, dataset split, the architecture of the convolutional neural network (CNN) and the evaluation protocol. At the end of Section 2 we describe our technique to investigate which part of the face has more influence in age prediction. Section 3 contains a brief description of the dataset and the experiments with the results obtained. Section 4 concludes the paper.

2. Method

Our solution to predict the age from facial images consists in a convolutional neural network (CNN) using the VGG-16 [9] architecture as it is shown in Fig. 2. In this section it is explained in details the structure of the network. Moreover, we describe our approach to understand which part of the face has more influence in age estimation.

2.1. Face detection and alignment

The IMDB-WIKI [8] dataset is released in two versions: (i) wide images containing the whole body of the subject and other objects; (ii) cropped images containing central frontal faces with 40% of external margin. In our work we used the version with already cropped images showing central frontal faces. In this paragraph we want to show how the authors of the dataset extract these faces from wider images containing other parts of the body and objects.

The detector used to obtain the location of the face is the off-the-shelf Mathias *et al.* [5]. Cropping the detected face for the following age estimation process instead of using the entire image leads to a massive increase in performance [2]. This is reasonable because in this way unuseful information is discarded and only a clear image of the face is kept.

The problem of alignment has been solved without facial landmark. The procedure used consisted in running the face detector not only on the original image but also on all rotated versions between $-60°$ and $60°$ with steps of $5°$. To deal with flipped and rotated images the detector was also run at $-90°$, $90°$, and $180°$. Finally the face with the highest detection score across all rotation is picked and then it is rotated accordingly to the up-frontal position [2].

Furthermore, the detected face was extended by taking an additional 40% of its width and height on all sides. This margin ensures that the face is always placed in the same location of the image. The resulting image is then squeezed to 224x224 pixels and used as input of our convolutional neural network (CNN).

2.2. Dataset split

To train our model and to evaluate its performances correctly we split the dataset in 3 parts: training set, validation set and test set respectively with a percentage of 80%-10%-10%. The training set is used to train the model, the validation set to tune the model's hyperparameters (i.e. learning rate, dropout, etc.) and eventually the test set to evaluate the results in terms of mean absolute error (MAE).

This technique is known in literature as *hold out method* [11]. We used it in an effort to limit overfitting and to improve the model's ability to generalize.

The split is realized with stratification. It means that the random split is done in a way that guarantees that each age values is properly represented in training, validation and test sets. Thus, the age distribution is similar in each of the 3 sets.

2.3. CNN architecture

Our method uses a CNN with the VGG-16 [9] architecture shown in Fig. 2. One of reason why we chose this particular architecture is due to its remarkable results on the ImageNet challenge [10]. Moreover, it also

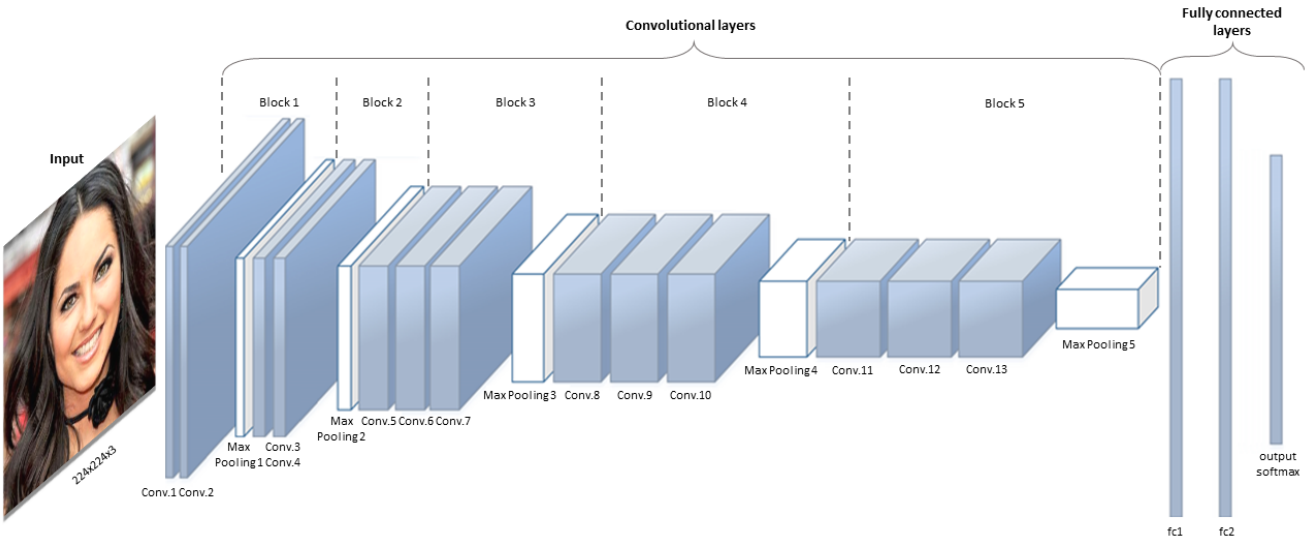

Figure 2. Convolutional neural network (CNN) with VGG-16 architecture

allowed us to have a warm start for the training process using pretrained models for classification that are publicly available.

Our CNN starts from an input image of medium resolution 224x224. The VGG-16 architecture is composed by 13 convolutional and 3 fully connected layers organized in 5 blocks. The convolutional layers in each of the 5 block have respectively 64, 128, 256, 512, 512 filters (the number of filters used for the operation of convolution is usually referred as *depth* of the convolutional layer). The filters have a shape of 3x3 pixels with a stride of 1. The operation of reducing the volume size is handled by max pooling at the end of each block with 2x2 filters and a stride of 2. Eventually there are two fully-connected layers with 4096 neurons each, then the output layer is characterized by the softmax function. We changed the number of neurons in the output layer according to our needs as explained in paragraph 2.4.

For all experiments the CNN is initialized with the weights from training on ImageNet [10]. Pretraining on ImageNet for classification gives a warm start for the training process.

We set the model's hyperparameters according to the best performances obtained on the validation set. The learning rate is set to 0.01. We used ReLU as activation function in both convolutional and fully connected layer, except the output layer which is provided with the softmax function. We trained with a momentum of 0.9 and a weight decay of 0.0005 using L_2 regularization [12]. We also added dropout [13] for regularization with a rate of 0.5 in an effort to limit overfitting and improve the model's ability to generalize. The gradient descend algorithm is computed on using batches whose size is 32.

In Table 1 we summarized the VGG-16 model architecture.

2.4. Output layer

The output layer of the convolutional neural network with VGG-16 [9] architecture used for ImageNet classification [10] has 1000 softmax-normalized neurons, because the goal is to discriminate 1000 object categories in images.

In the case of considering age estimation as a regression problem the last layer need to be replaced with only 1 output neuron. However the approach to pose age estimation as a regression problem has shown low results because training the CNN directly for regression is relatively unstable as outliers cause a large error term [2].

Thus, we preferred to consider it as a classification problem. This means that the continuous age values need to be discretized. When training for classification, the output layer is adapted to have n output neurons, where we set n according to our experiments.

One of our approaches is to associate each age value to a separate output neuron. Thus, if the dataset contains age values from y_{min} to y_{max}, we set a number of neurons in the output layer equals to:

$$n = y_{max} - y_{min} + 1$$

In this case we consider n classes, each of them corresponding to an age value.

Block	Layer	Filter shape	Output shape
Block 1	Conv.1	64@3x3	224x224x64
	Conv.2	64@3x3	224x224x64
	Max Pooling 1	2x2	112x112x64
Block 2	Conv.3	128@3x3	112x112x128
	Conv.4	128@3x3	112x112x128
	Max Pooling 2	2x2	56x56x128
Block 3	Conv.5	256@3x3	56x56x256
	Conv.6	256@3x3	56x56x256
	Conv.7	256@3x3	56x56x256
	Max Pooling 3	2x2	28x28x256
Block 4	Conv.8	512@3x3	28x28x512
	Conv.9	512@3x3	28x28x512
	Conv.10	512@3x3	28x28x512
	Max Pooling 4	2x2	14x14x512
Block 5	Conv.11	512@3x3	14x14x512
	Conv.12	512@3x3	14x14x512
	Conv.13	512@3x3	14x14x512
	Max Pooling 5	2x2	7x7x512
Fully connected	fc1	-	4096
	fc2		4096
	output softmax		*num_classes*

Table 1. CNN model summary

Another method we explored consists in partition the age values in m ranges of ages. Each of them covers a range of age from y_i^{min} to y_i^{max} and votes the mean of all training example in the *i-th* age range . In this case the number of neurons in the output layer is set to m. We set boundaries between ranges to create both uniform and balanced ranges: (i) uniform means that age range covers the same number of years; (ii) balanced indicate that each age range covers approximately the same number of training samples and, thus, fit the data distribution.

Since we set the problem as multiclass classification, the softmax function is used in the final layer of the neural network. The softmax function squashes the outputs of each neuron to be between 0 and 1 such that the total sum of the outputs is equal to 1. The output of the softmax function is equivalent to a categorical probability distribution, it gives the probability that any of the classes are true. Mathematically the softmax function is shown below:

$$\sigma(\mathbf{z})_j = \frac{e_j^z}{\sum_{k=1}^{K} e_k^z}$$

for $j = 1, 2, ..., K$, where z is the vector of the inputs to the output layer and j indexes the output units.

2.5. Final predictions

We trained our CNN for classification and at test time we computed the age prediction with 3 different methodologies: (i) we consider as predicted age value the age associated with the output neuron having the highest probability after applying the softmax function; (ii) we compute the expected value over the softmax-normalized output probabilities of the neurons [2]. It is the sum of the age values associated to the n neurons weighted with their probabilities:

$$E(O) = \sum_{i=1}^{n} y_i \cdot o_i$$

where $O = 1, 2, ..., n$ in the n-dimensional output layer, $o_i \in O$ is the softmaxed-normalized output probability of the neuron i, y_i is the age value associate to the neuron i. Thus, we consider the expected value as predicted age; (iii) we take as predicted age value the mean (or median) between the ages associated with the neurons that have the n highest probabilities.

2.6. Evaluation protocol

The metric we used for quantitative evaluation in our experiments is the mean absolute error (MAE). It measures the average magnitude of the errors in a set of predictions, without considering their direction. It is

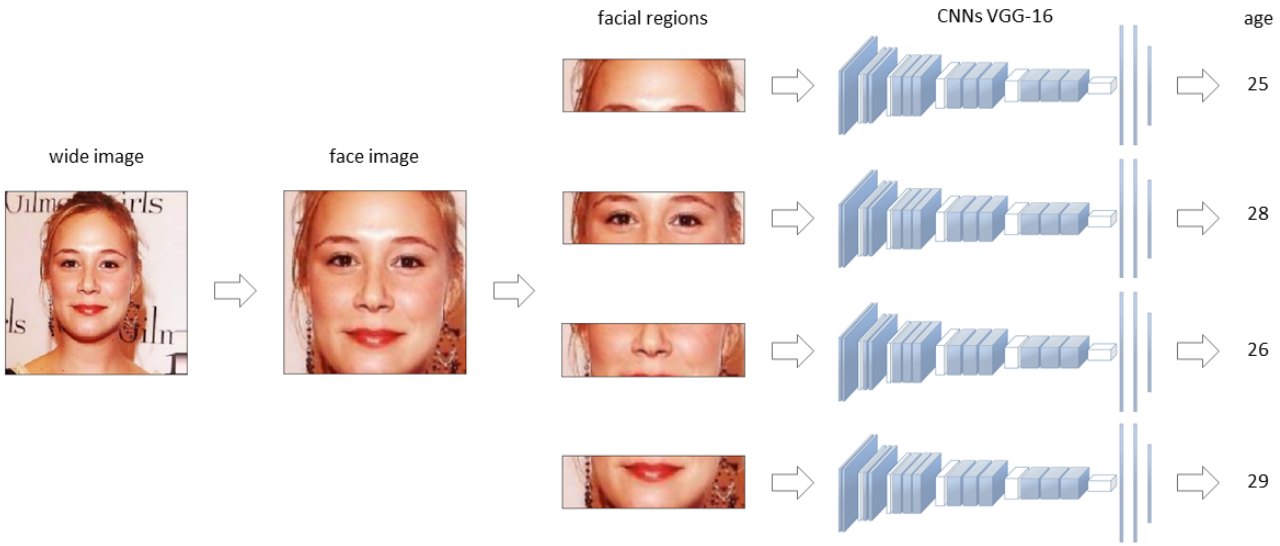

Figure 3. Pipeline region-specific local features

the average over the test sample of the absolute differences between predictions $\hat{y}_i$ and actual observations y_i where all individual differences have equal weight [14].

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

where n is the number of test samples.

In our case the MAE is the average of the absolute error between the predicted age and the ground truth age. We chose to use this metric for our evaluation because it represents a *standard de facto* for age estimation and in this field it is the most used metric in literature.

2.7. Region-specific analysis

Our method to understand which part of the face gives more contribution in age prediction consists in extracting local-level features from specific regions of the face. In particular we focused our attention on forehead, eyes, nose, mouth and chin.

The technique we used is shown in the pipeline at Fig. 3 which consists of the following steps: (i) face detection from the wide input image; (ii) image split into regions containing forehead, eyes, nose, mouth and chin; (iii) each region is given as input in 4 different CNNs; (iv) evaluation of which region achieves better results in terms of MAE. In the following we will analyze these phases in more details.

As first step, since the dataset we used in our experiments is provided with wide images containing central frontal faces with 40% of margin, we needed to discard this external margin because it has no meaningful information. The idea is essentially to detect only the face of the person because it contains the specific regions we want to select. In particular, we used the face detector of the *OpenCV* library [15] to obtain clear images containing only frontal faces. We only took into account images where the detector localized one face and we discarded the others.

After this preliminary step, we split the facial image horizontally into 4 parts with five-pixels of overlap between adjacent regions. The idea to use overlap between adjacent regions comes from an existing study [3] where they apply this technique in a similar context for another purpose. Each region contains respectively: (i) forehead; (ii) eyes; (ii) nose; (iv) mouth and chin. Since splitting and displaying images to evaluate if the split truly gives regions containing these facial part is computationally costly, we checked it only on a subset of 100 images. It performed well, thus we can reasonably suppose that it also works well on the rest of the images.

Therefore, we trained 4 separate convolutional neural networks with the VGG-16 [9] architecture (described in paragraph 2.3) in which each of them has as input one of the 4 regions cropped.

Finally evaluating the results in terms of MAE we can see which has better performances, thus we can claim which part is more essential for age prediction.

2.8. Discussion

Once described our method we can clarify more in details the purposes of our study. The high-level goal is to rebuild a system similar to DEX [1] with a CNN able to address the problem of age estimation from facial images. We approached this problem as classification. To be more specific we want to evaluate the results both (i) considering each age value as a separate output class; (ii) discretizing age values into n ranges of age and voting the mean of all training example in the *i-th* rage (with uniform and balanced ranges).

Moreover at test time we computed the age prediction with 3 different methodologies to check which one is more beneficial: (i) highest probability, (ii) expected value, (iii) top n highest probabilities.

Eventually, selecting different parts of the face from the images we investigate which of them has more influence in age estimation.

Face score	IMDb	Wikipedia	IMDB-WIKI
> 0	460,723	62,328	523,051
> 5	34,695	3,549	38,244

Table 2. Number of images in the dataset

3. Experiments and Results

In this section, we report experimental results to evaluate the performance of our method for age prediction. We first provide information about the dataset and implementation details. Then we present the experiments with both quantitative and qualitative results. We conclude the description of each experiment with a discussion about the results measured in terms of mean absolute error (MAE).

3.1. Dataset

In our experiments we use the IMDB-WIKI [8] dataset. It is the largest dataset of face images with age labels publically available that contains more than half a million labelled images of celebrities crawled from IMDb[1] and Wikipedia[2].

To realize this dataset the authors used the list of the 100,000 most popular actors as listed on the IMDb website and automatically crawled from their profiles date of birth, name, gender and all the images related to that person. Additionally, they crawled all profile images from pages of people from Wikipedia with the same meta information. In total they obtained 523,051 face images: respectively 460,723 from IMDb and 62,328 from Wikipedia.

[1]www.imdb.com
[2]en.wikipedia.org

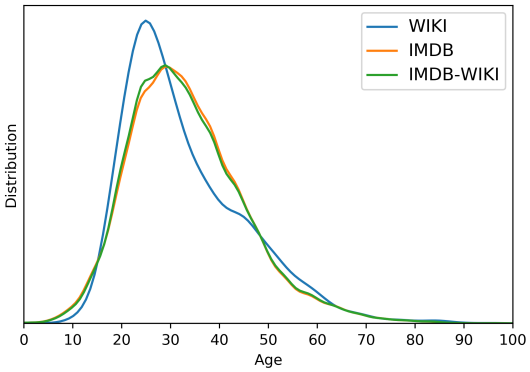

Figure 4. Age distribution of images with face score > 5

The authors also provide a version of the dataset containing cropped faces with 40% of external margin [8]. This is the dataset they obtain after preprocessing the images as described in section 2.1 and it is the dataset we used for our experiments.

Each image is provided with metadata containing information related to it. One of them, called face score, measures the accuracy with witch the detector finds a face in the picture. Practically, the face score quantifies how clear the face is in the picture (the higher the better).

For our experiments we only used images with face score > 5 and we discarded the rest. This allow us to deal with images where the face is shown clearly and it leaves us with a reasonable number of images for the training process.

As it is shown in the distribution Fig. 4, the dataset covers the 9-89 years interval best, while for the 0-9 and 89-100 intervals it suffers from small number of samples per year. Moreover, the few samples contained in those external intervals are mostly incorrectly labeled, thus we decided to discard the images whose age is lower then 9 and higher then 89. Hence, we only considered the age range from 9 to 89 in our experiments.

In Table 2 we summarize the number of images contained in the IMDB-WIKI dataset and the amount of images we used for our experiments.

3.2. Implementation details

We implemented our CNN with the Python neural network library *Keras* [16] using *Tensorflow* [17] backend. We modified the preexisting Keras implementation of the CNN with VGG-16 [9] architecture according to our needs. Therefore, we chose a different framework than the authors of DEX [1] who used *Caffe* [18].

Method	MAE	MAE w/o 2nd face detected
Highest probability	9.67	8.65
Expected value	9.49	8.62
Top n highest probabilities (mean)	9.22 (n=3)	8.61 (n=20)
Top n highest probabilities (median)	9.16 (n=3)	8.60 (n=20)

Table 3. Results table considering each age value has a separate output class (81 classes represent the interval 9-89)

The CNN is trained on Tesla P100/16GB GPUs, but we also used Titan-X/12GB GPUs for smaller tests. Since training on the whole dataset takes around 10 hours, we made our experiments first on a smaller subset to verify their effectiveness.

3.3. Insight experiments

As discussed before we considered age estimation as a classification problem rather then regression. We performed several experiments with different number of neurons in the output layer to inspect how this effects the results.

The outcomes of our experiments cannot be directly compared to DEX [2] (the study we rely on) because we used different datasets to evaluate the results. In particular, we just used IMDB-WIKI [8] as dataset, while in DEX study IMDB-WIKI is used to pretrain the model and then the CNN is finetuned on other datasets. Thus, compared to DEX we had less images available to work with and the results in terms of MAE are computed on completely different images. Moreover, we want to mention that the pictures of IMDB-WIKI contain facial images of celebrities. Mostly, they look younger than their real-biological age, if compared to common people.

3.3.1 Classifying each age value

The most intuitive way to evaluate our model is to verify how good are its performances considering each age value as a separate output class. In this case the interval 9-89 is converted into 81 classes, each of them corresponding to an age value.

Examining the result obtained by DEX we hypothesized the MAE should not have been higher than 10. In fact, using the dataset preprocessed as explained in paragraph 3.1 with images that have a face score > 5, we computed the MAE looking at the output probability of the neurons in 3 different ways: (i) We considered as predicted age value the age associated with the output neuron having the highest probability after the softmax function. Thus, we achieved a MAE of 9.67. (ii) We ran the same experiment considering as predicted age value the expected value over the softmax-normalized output probabilities of the neurons (as described in paragraph 2.5). In this case the MAE achieved is 9.49. (iii) On the other hand, if we take as predicted age value the mean between the ages associated with the neurons that have the 3 highest probabilities, the MAE obtained is 9.22 (9,16 with the median).

Moreover, since we believe a clear image with frontal face is essential to achieve good results, we accomplish experiments to verify our assumption. Once again we considered 81 neurons in the output layer, but we just considered images with face score > 5 in which the detector did not localize a second face in the picture. Computing the MAE in the same ways described above we achieved improvements: (i) 8.65 taking the age associated with neuron with the highest probability; (ii) 8.62 considering the expected values over the output probabilities of the neurons; (iii) 8.61 taking the ages mean of the neurons associated with the 20 highest probabilities (8.60 with the median). As expected, using only images showing a clear frontal face we obtained better results improving the MAE of about 1 year.

The results are summarize in Table 3. With the three different methodologies used the results are similar between each other.

3.3.2 Discretizing age values in ranges

One of our goals was also to analyze how the discretization of age values into n ranges of age (where each range covers an interval from from y_i^{min} to y_i^{max}) could effect the final results. In particular we set boundaries between ranges to create both uniform and balanced ranges: (i) uniform means that age range covers the same number of years; (ii) balanced indicate that each age range covers approximately the same number of training samples. Recalling that the distribution of ages is not uniform as it is shown in Fig. 4, it is interesting to see how discretization can impact on the results.

We evaluated both: (i) the MAE when an instance is classified in the wrong bin, thus the predicted value is the number of the predicted bin and the real value is the actual number of the bin the sample belongs; (ii) the MAE when each bin is associated to the mean of all

Discretization	Bin 1	Bin 2	Bin 3	Bin 4	Bin 5	Bin 6	Bin 7	Bin 8	Bin 9	Bin 10
Uniform	9-16	17-24	25-32	33-40	41-48	49-56	57-64	65-72	73-80	81-89
Balanced	9-20	21-24	25-26	27-29	30-32	33-35	36-38	39-43	44-49	50-89

Table 4. Age range discretization

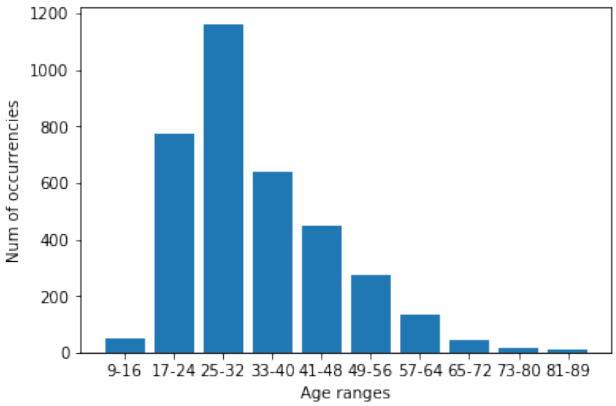

Figure 5. Uniform ranges of images with face score > 5

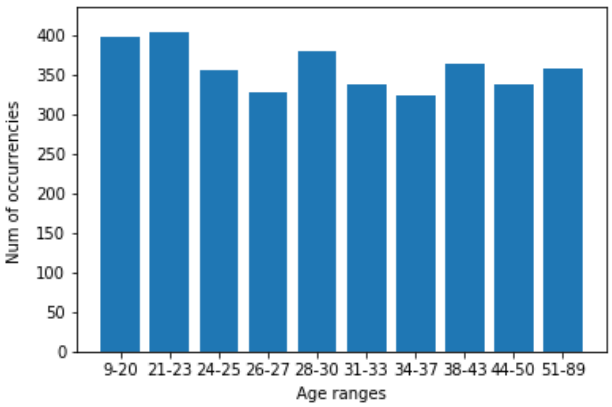

Figure 6. Balanced ranges of images with face score > 5

training samples in the bin, in this case the predicted value is the mean associate to the bin predicted and the real value is the age associated to the sample.

We hypothesized that with discretization, despite the large distance in age between the ranges, we could achieve better results than the approach described in the previous paragraph, as it happens in DEX [2].

The dataset used for these experiments is again the one explained in paragraph 3.1 with images that have a face score > 5.

For uniform ranges we considered 10 groups of ages where each of them covers an interval of 8 years. Since the dataset contains images in the interval 9-89 the ages are discretized as follow: 9-16, 17-24, 25-32, 33-40, 41-48, 49-56, 57-64, 65-72, 73-80, 81-89. What we achieved is a MAE of 1.17 between bins and 16.19 if we consider the mean of all training samples in the bin.

For balanced ranges we considered 10 groups of ages where each of them covers a different interval of years. Since the dataset contains images in the interval 9-89, the ages are discretized as follow according to the number of training instances: 9-20, 21-24, 25-26, 27-29, 30-32, 33-35, 36-38, 39-43, 44-49, 50-89. What we achieved is a MAE of 2.01 between bins and 18.21 if we consider the mean of all training samples in the bin.

Range	MAE b/w bins	MAE b/w age values
Uniform	1.17	16.19
Balanced	2.01	18.21

Table 5. Results table discretizing age values in ranges

Analyzing the results, summarized in Table 5, uniform ranges seems to perform slightly better than balanced ranges. Eventually, our hypothesis was not proved because we achieved better results with the method described in the previous paragraph.

3.3.3 Region-specific analysis

As mentioned before, the other purpose of our study is to understand which part of face has more influence in age estimation. To achieve this goal we run experiments giving as input to our CNN different parts of the face: (i) forehead; (ii) eyes; (ii) nose; (iv) mouth and chin.

In this case the output layer of the neural network has 81 neurons corresponding to age values in the interval 9-89. As it is reasonable to imagine, we hypothesized that the most informative features are typically located where wrinkles and age cue appears, for instance near the eyes.

The dataset used for this experiments is described in paragraph 3.1, we took into account images with face score > 5 where the detector has localized only 1 face.

Considering as predicted age value the age associated with the output neuron having the highest probability after the softmax function, the results observed have a MAE of 9.16 (eyes), 11.40 (forehead), 10.01 (nose), 9.44 (mouth and chin). On the other hand, there is a improvement of the results considering as pre-

MAE	Eyes	Forehead	Nose	Mouth and chin
Highest probability	9.16	11.40	10.01	9.44
Expected value	8.93	10.93	9.76	9.31

Table 6. Results table with different parts of the face

dicted age value the expected value over the softmax-normalized output probabilities of the neurons (as described in paragraph 2.5). In this case the results obtained have a MAE of 8.93 (eyes), 10.93 (forehead), 9.76 (nose), 9.31 (mouth and chin).

In any case with both techniques we achieved the results we were expecting: the part of the face that has more influence in age estimation is the region near the eyes where age cues are more visible. Moreover, the value of the MAE obtained using only the eyes is almost similar to the one obtained using the whole face. This strengthens the fact that the region near the eyes has a strong influence in age estimation.

4. Conclusions

In this paper we presented a solution to address the problem of age estimation from facial images with deep learning. We succeeded in rebuilding a system similar to DEX [2] using a convolutional neural network (CNN) with the VGG-16 architecture pretrained on ImageNet. We posed age estimation as a deep classification problem and we explored different solutions changing the number of neurons in the output layer of the neural network. At test time we investigated how different methodologies to make the final predictions can effect the results.

Moreover, we analyzed which part of the face has more influence in age estimation. Experimental results on the IMDB-WIKI dataset have shown that the region near the eyes, where age cues are more visible, plays a key role to predict the age.

To conclude we want to mention that our model could be used for other prediction tasks of facial features including gender, ethnicity, hair color, presence or absence of beard/glasses and facial expressions

References

[1] R. Rothe, R. Timofte, and L. V. Gool, "Dex: Deep expectation of apparent age from a single image," in *IEEE International Conference on Computer Vision Workshops (ICCVW)*, December 2015.

[2] ——, "Deep expectation of real and apparent age from a single image without facial landmarks," *International Journal of Computer Vision (IJCV)*, July 2016.

[3] J. Qiu, Y. Dai, Y. Zhang, and J. M. Alvarez, "Hierarchical aggregation based deep aging feature for age prediction," in *2015 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, Nov 2015, pp. 1–5.

[4] H. Han, C. Otto, X. Liu, and A. K. Jain, "Demographic estimation from face images: Human vs. machine performance," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 6, pp. 1148–1161, June 2015.

[5] M. Mathias, R. Benenson, M. Pedersoli, and L. Van Gool, "Face detection without bells and whistles," in *Computer Vision – ECCV 2014*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham: Springer International Publishing, 2014, pp. 720–735.

[6] P. Viola and M. J. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, May 2004. [Online]. Available: https://doi.org/10.1023/B:VISI.0000013087.49260.fb

[7] H. A. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 1, pp. 23–38, Jan 1998.

[8] L. V. G. Rasmus Rothe, Radu Timofte, "Imdb-wiki - 500k+ face images with age and gender labels." [Online]. Available: https://data.vision.ee.ethz.ch/cvl/rrothe/imdb-wiki/

[9] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *ArXiv e-prints*, Sep. 2014.

[10] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, Dec 2015. [Online]. Available: https://doi.org/10.1007/s11263-015-0816-y

[11] R. Kohavi *et al.*, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *Ijcai*, vol. 14, no. 2. Montreal, Canada, 1995, pp. 1137–1145.

[12] A. Y. Ng, "Feature selection, l 1 vs. l 2 regularization, and rotational invariance," in *Proceedings of the twenty-first international conference on Machine learning*. ACM, 2004, p. 78.

[13] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[14] M. R. Karim, "Scala machine learning projects: Build real-world machine learning and deep learning project with scala." Packt, Jan. 2018, pp. 27–28.

[15] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.

[16] F. Chollet *et al.*, "Keras," https://keras.io, 2015.

[17] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: https://www.tensorflow.org/

[18] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," *arXiv preprint arXiv:1408.5093*, 2014.