Vec2UAge: Enhancing Underage Age Estimation Performance through Facial Embeddings

Felix Anda[a], Edward Dixon[b], Elias Bou-Harb[c], Nhien-An Le-Khac[a], Mark Scanlon[a]

[a] *Forensics and Security Research Group, University College Dublin, Ireland*
[b] *Intel Ireland*
[c] *The University of Texas at San Antonio, Texas, United States*

Abstract

Automated facial age estimation has drawn increasing attention in recent years. Several applications relevant to digital forensic investigations include the identification of victims, suspects and missing children, and the decrease of investigators' exposure to psychologically impacting material. Nevertheless, due to the lack of accurately labelled age datasets, particularly for the underage age range, sufficient performance accuracy remains a major challenge in the field of age estimation. To address the problem, a novel regression-based model was created, Vec2UAge. FaceNet embeddings were extracted and used as feature vectors to train the model from the VisAGe and Selfie-FV datasets. A balanced, unbiased dataset was created for testing and validation. Data augmentation techniques were evaluated to further be used to expand the training dataset. The learning rate (*lr*) is one of the most important hyper-parameters for deep neural networks; a cyclic learning rate approach was used to find the optimal initial value for *lr* and the performance was evaluated. The distribution of model performance was presented per optimiser and one of the winning models with a Stochastic Weight Averaging (SWA) optimised training run reached a mean absolute error rate as low as 2.36 years. Additionally, the time of convergence using SWA was significantly faster than other optimisers evaluated, i.e., ADAGRAD, ADAM and Stochastic Gradient Descent. The evaluation model metric is presented in a form of a distribution rather than a single value, giving more insights into the effects of the random initialisations, optimisers and the learning rate on the outcome.

Keywords:
Child Sexual Exploitation/Abuse Material (CSEM/CSAM), Age Estimation, Underage Facial Age, Deep Learning

1. Introduction

Facial recognition is a well-known topic studied across several fields. The race to obtain a highly accurate tool to recognise, verify and cluster similar faces is a common task within the topic of computer vision. The deluge of facial photographs on the cloud has allowed the creation of robust facial recognition systems. Nevertheless, labels for soft biometric traits such as age, gender, ethnicity, height, weight, eye colour, marks, etc. are scarce. While the consideration of these traits are able to improve the accuracy of a biometric system [1], there are few datasets that contain such information accurately annotated. Age is a cue for face verification and facial recognition widely used in forensics. Automated age prediction could be valuable as an aid to live and post-mortem triage of collected evidence, while assisting alleviate digital forensics backlogs that have become commonplace throughout the world [2]. Age prediction can also assist in the identification of victims or suspects in CCTV footage, photographs, or child sexual exploitation material (CSEM). Moreover, Generative Adversarial Networks (GANs) are able to estimate images of victims by creating aged versions from an input image [3].

Age estimation models rely on good quality images with the relevant age labels. Nonetheless, accurate age annotations in facial datasets are also inadequate; certain age groups have few samples – particularly the underage age range. Datasets for this age range are difficult to find due to legal restrictions and ethical implications.

The IMDB-WIKI dataset [4] and Adience dataset [5] are amongst the most popular datasets for facial age estimation. The former is a large dataset of over 500k images that have automatic labels based on crawled age information. The latter is a Flickr dataset of over 26k images that have been labelled by humans, categorised in several age groups. Both labelling techniques have produced models with performance challenges that limit their usefulness for digital forensic investigations, as evidence based on their outputs would be unlikely to stand up in the courtroom.

For this research, two datasets were selected and FaceNet facial embeddings were calculated for each image. FaceNet learns mappings directly from facial images to a compact Euclidean space [6]. These facial embeddings were used in this work as feature vectors in the input of a four layered neural network. The first dataset is a recently-released underage dataset named VisAGe [7]. This dataset focuses on underage subjects

Email addresses: felix.andabasabe@ucdconnect.ie (Felix Anda), edward.dixon@intel.com (Edward Dixon), elias.bouharb@utsa.edu (Elias Bou-Harb), an.lekhac@ucd.ie (Nhien-An Le-Khac), mark.scanlon@ucd.ie (Mark Scanlon)

and has accurate age and gender labels assigned by consensus among human annotators. The images were pre-processed and the feature vectors were computed. The second dataset is Selfie-FV [8], which contains calculated FaceNet facial vectors and accurate age for female subjects ranging from 8 to 38 years old. The facial vectors were merged excluding adult images, and later relevant data augmentation techniques were applied (only on the training dataset). The image-augmentation facial vectors were also computed and compared with the original image for their consideration. In the making of the machine learning dataset for age estimation of minors, a balanced unbiased collection of accurately labelled faces was gathered for the whole age range of underage subjects including 18 year-olds (1 to 18). This subset contains 5,000 images evenly distributed within the different age bins and is ideal for testing and validating an underage age estimation model based on facial vectors.

Several optimisers were tested such as Adaptive Moment Estimation (ADAM), Adaptive Gradient Algorithm (ADAGRAD), Stochastic Gradient Descent (SGD) and Stochastic Weight Averaging (SWA). A simple regression network was configured rather than treating the problem as a classification task, in order to fully exploit the accuracy of the age labels. Finally, 20 models were trained with each optimiser (a total of 80 models). Each model corresponds to a setting with a different optimiser and random initialisation. It was observed that random initialisation has a strong influence on the final quality. SWA's ability to converge rapidly was also replicated.

In this paper, a model for underage age estimation based on facial embeddings is presented. The paper is organised as follows. In Section 2, an overview of the related work is presented. Section 3 provides an overview of the design and methodology of the developed model and its derivation from the VisAGe and Selfie-FV datasets. Section 4 describes the performance of the several Vec2UAge models. Section 5 provides a discussion of the research. Finally, the last section 6 outlines the conclusions and discusses future work.

Contribution of this Work:

- Implementation of face embeddings as input vectors in a neural network to tackle an underage age estimation problem as regression.
- Evaluation and application of data augmentation guidelines to improve underage age estimation performance and obtain a robust model.
- State-of-the-Art models with an average performance in Mean Absolute Error (MAE) of 2.5 years in test and a winning model of 2.36 years.
- Balanced underage facial vector dataset for training and testing, and open code for the experiments available at `https://github.com/4ND4/Vector2UAge`
- Usage of Stochastic Weight Averaging (SWA) to improve generalisation and further obtain results faster for underage age estimation.
- Comprehensive evaluation of random initialisations, optimisers and initial learning rates to obtain the best performing models.

2. Related Work

2.1. Previous Work on Facial Age Estimation

Geng et al. reiterate the lack of sufficient and complete training data [9]; however, the authors exploit the fact that close ages look quite similar. Instead of labelling with a single age, a label distribution is considered. Smaller datasets than the ones mentioned in Section 1 were used: FG-NET, which is an ageing dataset of 1,002 subjects [10], and MORPH, which is a larger dataset of over 55k images [11]. The best performing results in terms of MAE oscillate between 4.76 and 8.06 years, The MAE in different age ranges was also evaluated in the FG-NET dataset; the best performance lies on the age range 0 to 9 (2.30 years) followed by the age range 10 to 19 (3.83 years).

Chao et al. [12] aimed to overcome the data imbalance problem. An imbalance treatment is introduced to the training phase and the connections between facial features and age labels by combining distance metric adjustment and dimensionality reduction, are explored. Performance evaluated on the most widely-used FG-NET ageing database produced MAEs ranging from 3.06 and 3.10 years for ages less than 30. The MAE in different age ranges were also evaluated with the aforementioned database; the best performance lies within the age range 0 to 9 (1.911 years) followed by the age range 10 to 19 (3.52 years) with the C-lsLPP algorithm approach.

In 2017, Liu et al. presented a Group-aware deep feature learning approach that consists in learning a discriminative feature descriptor per image of the raw pixels for face representation [13]. The main motivation is that age labels are chronologically correlated and face ageing datasets lack labelled data in certain groups. The datasets used were FG-NET, MORPH and the Chalearn Challenge dataset [14]. The corresponding MAEs are 3.93, 3.25 and 4.21 years respectively.

2.2. Underage Facial Age Estimation

Age estimation classification models tend to perform better when the age bins have been grouped; hence the number of classes decreased. The penalisation for wrong classifications are less severe; similar to the effect of using a regression-based model compared to a multi-class classification model. Furthermore, limiting the size of the evaluated age range to underage subjects can potentially create an easier problem for age estimation. If the complexity of the problem is gradually increased, this is known as curriculum learning where the speed of convergence of the training process is increased [15].

Research on underage age estimation has been studied in the past, but only recently has there been work accomplished on automated underage facial age estimation. In 2016, Antipov et al. documented their winning approach for the ChaLearn LAP competition on apparent age estimation [16]. Since the major challenge was the age estimation of children, the authors

created a separate VGG16 model for minors from 0 to 12 years old and integrated the model to the final solution.

In 2019, Anda et al. attempted to ameliorate the accuracy of underage facial age estimation within the adulthood borderline with ensemble learning [17]. A deep learning model, *DS13K*, was developed that was fine-tuned on DEX (a well-known age estimation model that was pre-trained on ImageNet for image classification [4]). An improvement was achieved on the age range from 17 to 18 year-old subjects being a key group in the detection of CSEM.

Later in 2020, a ResNet50-based deep learning model, DeepUAge, was developed, which addressed underage age predictions [7]. The model was trained on VisAGe[1], an underage dataset. A novel pre-processing technique based on the *Dlib* Contour Artistic approach was implemented. The approach achieved a MAE of 2.73 years and outperformed state-of-the-art cloud based services, such as Amazon Rekognition and Microsoft Azure Face API for the underage age bracket.

2.3. Facial Vectors and Soft Biometric Traits

Schroff et al. [6] proposed a system that learns mapping from facial images where the distance between the vectors produced are able to determine facial recognition, verification and clustering of similar images. The output creates embeddings of 128 dimensions per face but currently 512 dimensions are supported. Face embeddings refers to the facial features that can be extracted from a facial image. Once processed, the problem becomes a k-Nearest Neighbour (k-NN) classification problem.

Leveraging facial vector embeddings for trait related research such as age, gender, emotions and attractiveness has only been exploited in the past two years. In 2018, Jekel and Haftka used a logistic regression and a Support Vector Machine (SVM) approach to automatically review online dating profiles based on the user's historical preferences [18]. The authors discussed a possibility of the FaceNet vectors being related to attractiveness. This research was one of the first using FaceNet facial embeddings for tasks other than facial recognition.

Later in 2019, Terhörst et al. proposed a multi-algorithmic fusion for age and gender estimation based on stochastic forward passes through a dropout-reduced neural network ensemble [19]. Their approach was benchmarked on the Adience dataset [20], and achieved an age estimation accuracy of (64.6 ± 2.8)%.

Recently in 2020, Swaminathan et al. developed a method to predict gender based on several machine learning classification techniques on facial embeddings. Logistic regression, SVM, k-NN, Naive-Bayes and Decision Trees where evaluated on the UTK Face Dataset [22] and the best performer, k-NN, achieved an accuracy of 97%. In the same year, facial embeddings and facial landmark points for the detection of academic emotions such as engagement, frustration, confusion and boredom, were studied by Leong [23]. The author evaluated the use of deep learning on FaceNet embeddings and facial landmark points and hypothesised that the facial embeddings may similarly offer valuable information for the detection of emotions. A Long Short Term Memory (LSTM) network architecture was used and the accuracy to detect both boredom and frustration was 52.15 and 70.67 % respectively.

2.4. Data Augmentation for facial images

Image augmentation for facial recognition has been studied in the past and has recently become increasingly popular. Data augmentation can improve the performance of machine learning models and convert bounded datasets into exploitable big data [24]. Lv et al. proposed 5 data augmentation methods: landmark hairstlye, glasses, poses, and illumination manipulations. The approach enlarges the training dataset, which aids the impacts of misalignment, pose variance, illumination and occlusion [25]. For facial age estimation, data augmentation was proposed by Liu et al. [26]. Their augmentation approach consisted in the application of geometric and photometric transformations such as flipping, rotating, scaling, and noise addition. The method aids overfitting, enhances the robustness of the model and improves the accuracy of age estimation [26].

3. Methodology

In applying machine learning to the problem of estimating the age of a subject from an image of their face, as with many machine learning problems, the size and quality of available datasets has been a limiting factor. However, work on facial recognition has resulted in very large public datasets, featuring thousands of faces, such as those by Huang et al. [27], Rothe et al. [4] and Anda et al. [5], and large deep convolutional networks capable of producing high-quality embeddings, enabling reliable facial recognition. While an ideal facial recognition model would produce representations that are invariant with respect to age, Huang et al. [27] hypothesised that sampling biases in popular datasets, such as the *Labeled Faces in the Wild* (LFW) dataset, would lead models to use age as a recognition cue. Although datasets like LFW are built from images of celebrities that feature a wide span of ages, many subjects will be featured during a comparatively narrow span of their lives, as can be seen in Figure 1. This histogram was constructed from the IMDB-WIKI 500k dataset [4] by calculating a "fame span" for each celebrity minus the difference in years between the dates of their first and last photographs. As shown in Figure 1, "fame span" is usually 10 years or less, making age a useful clue to identify most faces.

This would make these embeddings (with far lower dimensionality than the input images) a better representation for an age estimation model, which would otherwise be attempting to learn both the structure of a human face and its ageing trajectory from a comparatively smaller dataset, such as the one proposed in Section 3.3

3.1. FaceNet

Face embeddings are high-grade features extracted usually from detected faces. They use deep convolutional neural networks (DCNN) to map a facial image to a vector. The most

[1] VisAGe: Visual Age and Gender Dataset available at `https://www.forensicsandsecurity.com/visage`

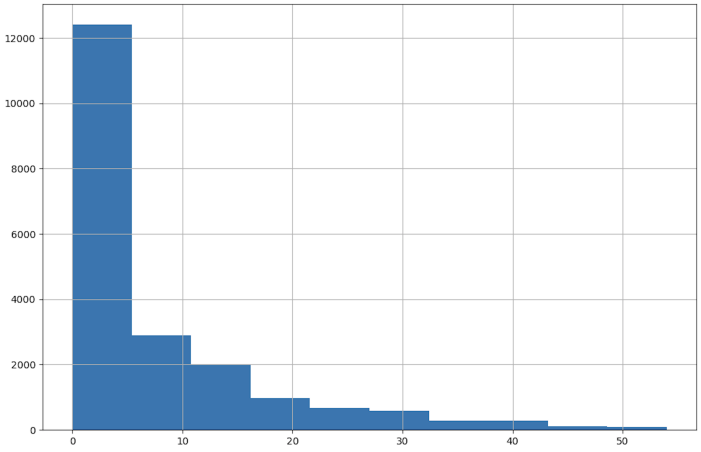


Figure 1: "Fame span": years between earliest and latest photos, per celebrity in the IMBD-WIKI 500k

used model is FaceNet, which was introduced in Section 2.3. FaceNet predicts features that are an array of 512 vector representations. The model is a Deep Neural Network (DNN) trained through a triplet loss function that influences facial embeddings for the same subject to have smaller distances and different subjects to have larger distances [6]. In this research, the cosine similarity is used to calculate with a given threshold, if the facial embedding arrays belong to the same identity in a given multidimensional space. Equation 1 is the cosine similarity between face a and face b, where the value of n is 512.

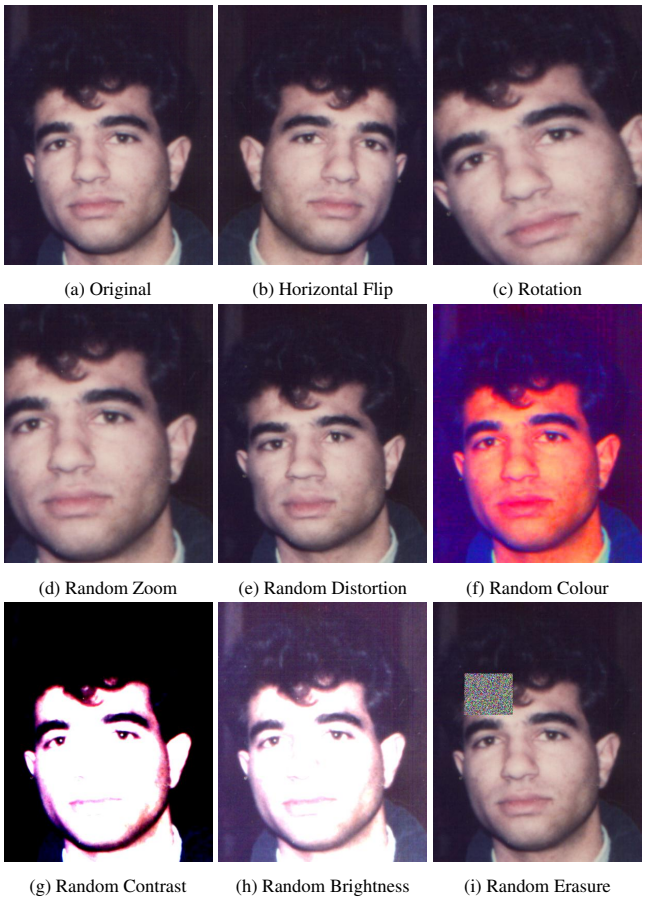
$$\cos(\mathbf{a}, \mathbf{b}) = \frac{\mathbf{ab}}{\|\mathbf{a}\|\|\mathbf{b}\|} = \frac{\sum_{i=1}^{n} \mathbf{a}_i \mathbf{b}_i}{\sqrt{\sum_{i=1}^{n} (\mathbf{a}_i)^2} \sqrt{\sum_{i=1}^{n} (\mathbf{b}_i)^2}} \quad (1)$$

Indeed, the cosine similarity formula is the standard scalar product between both 512-dimensional vectors representing respectively face a and face b. As vector components cannot be negative, the angle between those vectors is between 0 (when for each index at least one of the corresponding components in vector a or in vector b is equal to 0) and 90 degrees (when both vectors are the same or proportional). In other words, the more parallel they are, the higher the cosine similarity is.

3.2. Data Augmentation

Data augmentation is managed by the *Augmentor* machine learning python library [28]. *Augmentor* aids the image augmentation and artificial generation of data for machine learning use cases. It uses a stochastic approach using building blocks that enable operations to be pieced together in a pipeline. The data augmentation techniques used are both geometric and photometric transformations; horizontal flip, rotation, random zoom, random distortion, random colour, random contrast, random brightness and random erasing were applied. The visual effects of the several data augmentation techniques can be seen depicted in Figure 2.

The cosine similarity between the original image and the augmented image was computed with the Equation 1. As a reference, the vectors from Figure 2 were calculated and the cosine similarity was analysed. The results can be seen depicted



(a) Original (b) Horizontal Flip (c) Rotation

(d) Random Zoom (e) Random Distortion (f) Random Colour

(g) Random Contrast (h) Random Brightness (i) Random Erasure

Figure 2: Facial Image Augmentation Techniques: Original image taken from FG-NET Aging Database [29]

in Table 1. If the cosine similarity is close to 1, the augmented facial vector hasn't suffered much changes and would resemble the original image. Therefore, a more aggressive augmentation technique would be required. According to the results in the table, it can be observed that FaceNet is robust against occlusions. Therefore, the random erasure data augmentation technique should be replaced by another augmentation method.

Augmentation	**Cosine Similarity**	**Settings**
Flip	0.8599	Horizontal
Brightness	0.6845	Factor: 2
Rotation	0.6656	Angle: 25
Random Zoom	0.7856	Factor: 2
Random Distortion	0.8728	Grid width: 10 Grid height: 10 Magnitude: 8
Random Colour	0.5609	Factor: 2
Random Contrast	0.3341	Factor: 5
Random Erasure	0.9837	Rectangle : 0.2

Table 1: Cosine similarity between original image and augmented images using the *Augmentor* library

After the analysis, image augmentation was selected accordingly to a defined threshold of 0.6 – meaning that the euclidean

distance between the facial vectors predicted from the images and augmentations were slightly far from each other. However, the augmented dataset was only used for the training set as discussed in Section 3.3.3. These techniques were performed in an offline manner. Thus, creating physical images saved to the local disk.

3.3. Proposed Facial Age Dataset

It is essential to gather enough data for training and testing. Moreover, the size of the dataset required is dictated by both the complexity of the problem that is trying to be solved and the quality of the images. Both the size of the training data and its quality are influencing factors in the success of the model [30]. Furthermore, a specific rule that outputs the amount of training and testing data required has not been developed; nonetheless, the best practices are to evaluate datasets from previous research on age estimation such as the work developed by Rothe et al. [4] and Anda et al. [5, 7].

Faces accurately labelled with age and gender are significantly scarce, especially for underage subjects. Moreover, online and offline facial age estimators are challenged by images that belong to the lower age range bracket [5]. To tackle the lack of underage images, two techniques were employed. Firstly, an underage age estimation dataset was selected and an accurately labelled facial vector dataset was integrated. The proposed facial age dataset is a merge between the underage group range pertaining to VisAGe (as described in Section 3.3.1) and Selfie-FV (as described in Section 3.3.2). The counts per age distribution of VisAGe, Selfie-FV and a combination of both can be seen in Figure 3. Secondly, the relevant data augmentation techniques discussed in Section 3.2 were applied only to the training data set.

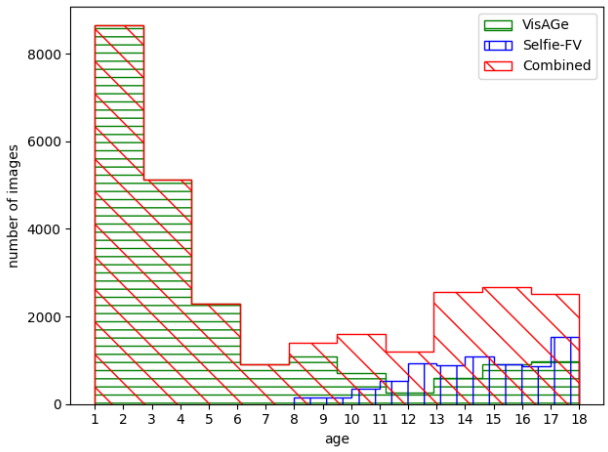


Figure 3: Histogram of Age Count Distribution: VisAGe, Selfie-FV and Combined

3.3.1. VisAGe

VisAGe is a facial single image dataset composed of mainly underage subjects. The dataset comprises of over 21k images accurately labelled by age and gender. The distribution by gender can be seen in Figure 4. In the distribution graph, it can be seen that there is an exponential decay of amount of images for both male and female subjects as the age increases. The dataset has been used in [7] and [31]. In the former study, the authors created a deep learning underage model using a contour *Dlib* artistic approach for pre-processing. This approach predicts additional landmarks pertaining to the hairlines, which enables the facial cropping technique to capture important age related data such as wrinkles on the forehead. On the latter study, VisAGe is used for the evaluation of the influence that certain human bio-metric factors, facial expressions, and image quality have on the outcome of automated age estimation.

3.3.2. Selfie-FV

Selfie-FV is a dataset of facial vectors derived from unique face images of female subjects between 8 and 38 years old. The size of the dataset also exceeds 21k subjects and the data is shared in two pickle files: one for training and the other for testing (no-one appears in both the training and test sets). The files were created with a 80/20 split and contain the pickled Pandas Dataframe objects with a unique identifier for the image, the image number, the image filename, the accurate age ground-truth and the facial embedding. The dataset is available on Github: `https://github.com/EdwardDixon/selfie-fv/`. The age distribution can be seen depicted in Figure 5. It can be observed that while the amount of images increases from age 8 onwards, a sudden peak occurs close to the 15 year-old mark. These images are in the age range of interest to develop an underage age estimation model.

The number of underage subjects used for this research is 7,419 that belong to the age group of 8 to 18 year-olds. The contribution of these images for the combined dataset can be seen in Figure 3.

3.3.3. Test/Validation and Training Dataset

A test dataset is considered optional but paramount to evaluate the final performance of the model fit on the training dataset. It is noticeable in Figure 3 that there is a decrease in the amount

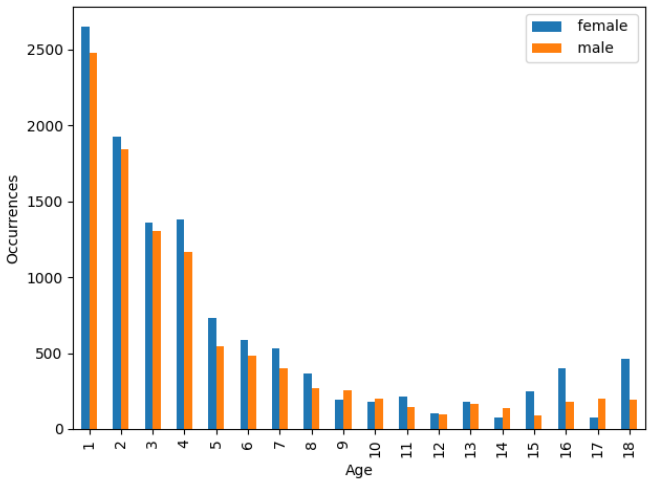


Figure 4: Distribution by Age and Gender - VisAGe

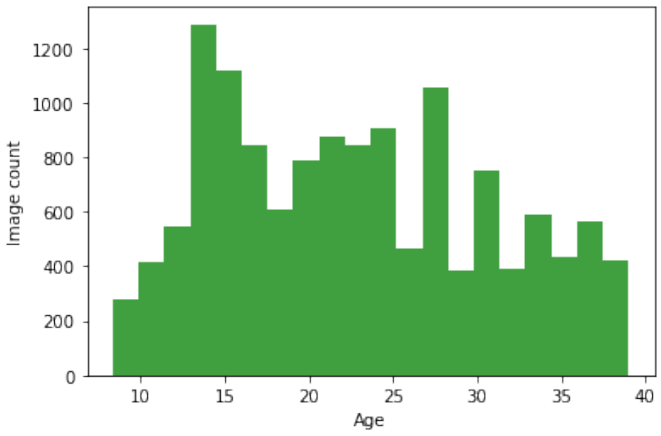


Figure 5: Distribution of Female Subjects by Age - Selfie-FV [8]

of images in the 6 to 8 and 11 to 13 bins. Nevertheless, an unbiased balanced testing/validation dataset was obtained with 500 images per class, leading to a non-augmented dataset of 9,000 images that can be used both for validation and testing or simply for validation. Test-time data augmentation has been proven to reduce appearance variations and improve face representations [32]. However, it is not considered for this research but could be a potential for future work.

A validation dataset is used to automatically select the best classifier during the training. Stratified Shuffle Split [33] was applied to divide the dataset in validation and test where the test dataset was 50% of the validation set. The shuffling technique applies stratified randomised folds – made by preserving the percentage of samples for each class.

The training dataset was formed with the remaining images of the merged dataset and the augmented images were included. An augmented dataset of 5,000 images per age was created (1 to 18 year-olds). A total of 90k images were gathered. A separate JSON file for the training dataset and the test/validation dataset was created. The file contains a unique identifier for each image as a key and its corresponding facial vector array.

3.4. Facial Image Pre-processing

The images for the selected datasets have already been curated and are predominately frontal face photographs of a single subject. Exposure, occlusion, noise and emotion are influencing factors on the accuracy of underage facial age estimation [31]. As a result, images have been discarded according to the level of these factors, thus decreasing the problem to a smaller one.

Face detection is usually needed for age estimation; while reducing the number of pixels to be evaluated, unwanted background and noise is also addressed. The *Dlib* library [34] is used to detect faces. When no face is detected, the CNN version of *Dlib* is executed. This approach lessens the processing time and increases the face recognition hits. Finally, once the face has been recognised, it is cropped to the detected face rectangle and resized to a size of 224 x 224 pixels.

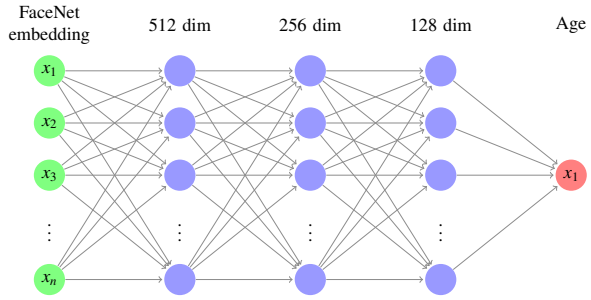


Figure 6: Proposed neural network: for every hidden layer a ReLU activation is used

3.5. Neural Network

Having the face vectors calculated previously, the inputs rather than being pixels, are float values that can be processed with a much simpler neural network. A simple 4-layer neural network with 512, 256, 128, 1 units at each layer was constructed as shown in Figure 6. For every hidden layer, a Rectified Linear Unit (ReLU) activation function was used. The input layer is not considered a layer of neurons, but rather the entry of the facial embeddings of size 512. The first hidden layer consists of 512 neurons, followed by the next layer which consists of 256 neurons and the 3rd layer is of size 128 and will generate the final results. The last layer consists of a single output neuron as an age regressor. ReLU is one of the most commonly used activation functions in neural networks. It relies on a simple calculation that returns the input if the value is greater than 0 – otherwise it returns 0. The function can be seen in Equation 2.

$$f(x) = max(0, x) \tag{2}$$

3.5.1. Regression versus Classification

In our daily life, age is treated as a discrete variable, except among the very young – a five and a half year-old will not be denied their half year. Ages are binned more coarsely as we age, e.g., young, middle-aged, old, and much previous work has treated the age estimation as a classification problem, with samples assigned to broad age buckets. In the extreme, it is binarised, i.e., minor or adult? When a more accurate age estimate is desirable, and accurate ground-truth is available, we can instead treat age as a continuous variable and make age estimation into a regression problem. Regression structures are used to estimate a value (continuous inputs) instead of a fixed class, leading to an infinite set of possible outcomes [35].

Whether modelled as regression or classification, data is the limiting factor, as with all machine learning problems. Age estimation has been addressed in the past with several machine learning regression techniques; predominantly Support Vector Regression (SVR), Multilayer Neural Networks (MNNs), Random Forests (RF) and Canonical Correlation Analysis (CCA) [36]. Conversely, commonly used classification algorithms such as k-NN, multilevel perceptron (MLP), AdaBoost and Support Vector Machine (SVM) have been studied to perform accurate age prediction and grouping [37].

The difficulty with using fine bin sizes, i.e., 1 year wide, while also taking a classification approach is that the model will score the same loss for being wrong by 1 year as it would for being wrong by 20 years. With regression, a large error, e.g., 20 years, can induce a larger weight update than a small one, e.g., 1 year. The difference would matter less if a binary classifier was trained, but for samples close to the decision boundary it would still matter. Therefore, for this research, based on data with accurate age values, a regression approach has been chosen. Hence, the last output layer of our 4-layered model contains a single neuron that is known as the age regressor.

3.5.2. Optimisation

Optimisation is one of the main components of machine learning. Gradient descent is an optimisation technique used to find the minimum of a function. It is regularly used in deep learning models to update the weights of a neural network. The following gradient-descent-based optimisation algorithms have been used in our study to lessen the error rates:

- Adaptive Moment Estimation (ADAM) [38]
- Adaptive Gradient Algorithm (ADAGRAD) [39]
- Stochastic Gradient Descent (SGD) [40]
- Stochastic Weight Averaging (SWA) [41]

ADAM and SGD are commonly used to optimise deep neural networks and are widely used in age estimation. For our research, we explore the use of several gradient-based optimisers and focus specifically on the novel SWA. SWA is a procedure that enhances generalisation in deep learning models over SGD at no additional cost. Izmailov et al. proved that the SWA procedure is able to find much flatter solutions than SGD and the solutions are wider than the optima found by SGD [41]. The authors also notice an improvement in the test accuracy versus SGD training on several state-of-the-art residual networks. It also has slightly worse train loss, but better test error.

3.6. Proposed Solution

The mixed dataset discussed in Section 3.3 was used. All faces in the images were detected and cropped, as described in Section 3.4, and the facial vectors calculated. Images for training were generated with the relevant data augmentation techniques, as explained in Section 2.4, to a total of 90k (5k per class, 18 classes in total), a stratified shuffle split was applied to the test dataset to divide it into 2 equal sub-datasets. Both the validation and testing dataset accounted to 4,500 images each.

A 4-layered neural network was selected with 512, 256, 128 and 1 units per layer respectively. Each hidden layer used a ReLU activation function. The input to the network was the array of 512 facial vectors and the output an age regressor. The optimisation algorithms chosen were ADAM, ADAGRAD, SGD and SWA. Two sets of 20 experiments each were performed with the aid of Neptune, a light-weight management tool that keeps track of machine learning experiments [42]. The first set of experiments, *D1*, correspond to a initial fixed learning rate of 1e−4 for ADAM, ADAGRAD and SGD, and 1e−5 for SWA. Conversely, the second set of 20 experiments (E1) correspond to the use of a tool as guidance for choosing an optimal initial *lr*. This tool is based on cyclic learning rates proposed by Smith [43] in 2017.

The choice of the loss function was the simplest and most common Mean Squared Error (MSE). The main metrics used to measure the loss were MSE and MAE. The MAE is the absolute mean average difference between the predicted age and the real age. Lastly, the number of epochs selected was 100, but early stopping was implemented. This is helpful to reduce the learning rate as the number of training epochs increases and therefore, a learning rate scheduler was applied.

4. Results

4.1. Evaluation of the Experiments

The set of experiments *D1* and *E1* have been logged entirely using Neptune. Up to 10 experiments can be compared simultaneously and there is an API feature that allows the integration with python through the *neptune.sessions* library. To ensure full reproducibility from run to run, *pytorch-lightning* supports deterministic experiments. Additionally, the seeds for pseudo-random generators have been logged in Neptune and the experiment results are duplicable and openly available at `https://ui.neptune.ai/4nd4/Vec2UAge/`.

4.2. D1 - Evaluation of the MAE Distribution with a Fixed Initial Learning Rate

The effects of the fixed initial values of *lr* (1e−4 for ADAGRAD, ADAM and SGD, and 1e−5 for SWA) can be seen in Figure 7. *ADAGRAD* yielded the worst performing results for training, validation and test. Nevertheless, the consistency of values of the validation and test loss is denoted by a spread of 0.05 and visible in both the figure and in Table 4. It can be seen that the *ADAM* algorithm surpassed the performance of the other optimisers for training and testing. Moreover, the data was the least sparse for all the losses. The validation and test MAE for *SGD* was consistent and the standard deviation was low. Therefore, there was not a significant spread of data. It was a stable optimiser that produced models as low as 2.51. Lastly, *SWA* although not achieving the best performing training and validation values, managed to achieve models for validation as low as 2.43, and had the best performing model and mean for testing.

Overall, the best performer for the experiment set *D1* was the *ADAM* optimiser approach with a fixed initial *lr* of 1e−4. The outcome produced models in test with a mean of 2.49 and MAEs as low as 2.46. The criteria to pick the best optimiser was to sum the count of minimum values per statistic per loss.

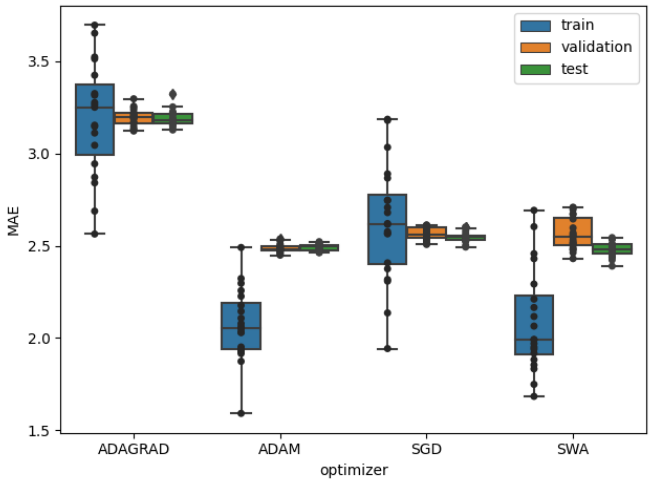


Figure 7: MAE distribution per optimiser for training/validation/test (using a fixed initial *lr*)

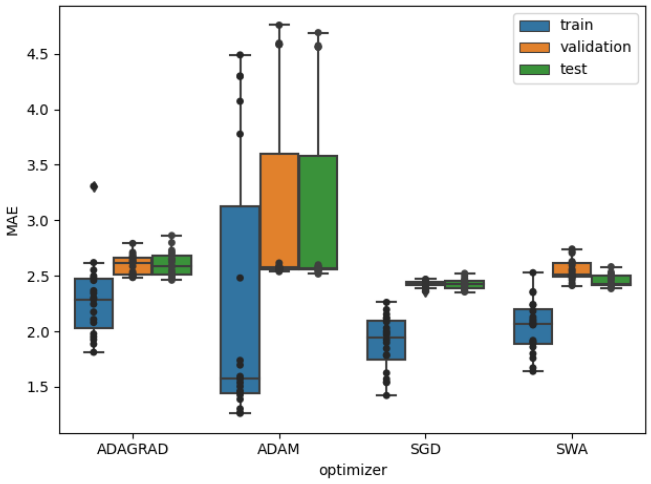


Figure 8: Mean absolute error distribution per optimiser for training/validation/test (using *lr* finder)

	mae			val_mae			test_mae		
optimiser	*min*	*mean*	*std*	*min*	*mean*	*std*	*min*	*mean*	*std*
ADAGRAD	2.56	3.19	0.31	3.12	3.20	0.05	3.13	3.19	0.05
ADAM	1.59	2.07	0.20	2.45	2.49	0.02	2.46	2.49	0.02
SGD	1.94	2.62	0.32	2.51	2.56	0.03	2.49	2.55	0.03
SWA	1.68	2.09	0.28	2.43	2.57	0.09	2.39	2.48	0.04

Table 2: Experiment D1: Statistics of MAE in training/validation/test (fixed *lr* of 1e−4 for ADAGRAD, ADAM & SGD; and 1e−5 for SWA)

4.3. *E1 - Evaluation of the MAE distribution with an initial lr finder*

The effects of the usage of the learning rate finder to obtain the initial learning rate can be seen in Figure 8. It is observed that the validation values are consistently clustered except for the ones seen in the ADAM algorithm. *ADAGRAD* had low sparse data throughout training, validation and test losses. Conversely, the performance wasn't as good as the rest of the optimisers (ADAM, SGD, SWA). It is noticeable that *ADAM* had the highest standard deviation figures with 1.23, 0.93 and 0.92 in training, validation and testing, respectively, as can be seen in Table 3. Next, *SGD* performed better than the rest of the optimisers followed by *SWA*. The winning results were consistent for all the statistical evaluations besides the standard deviation for training, which was slightly inferior than *SWA*. Experiment E1 produced models with MAEs as low as 2.36. The same criteria used in Section 4.2 to pick the best optimisers was applied.

	mae			val_mae			test_mae		
optimiser	*min*	*mean*	*std*	*min*	*mean*	*std*	*min*	*mean*	*std*
ADAGRAD	1.81	2.28	0.35	2.48	2.60	0.09	2.46	2.61	0.12
ADAM	1.26	2.24	1.23	2.54	3.11	0.93	2.52	3.09	0.92
SGD	1.42	1.89	0.25	2.36	2.43	0.03	2.36	2.43	0.05
SWA	1.64	2.04	0.24	2.41	2.55	0.09	2.38	2.46	0.05

Table 3: Experiment E1: Statistics of MAE in training/validation/test (*lr* finder executed)

4.4. *Details of Winning Optimisation approaches*

The best performer in *D1* is the experiment *VEC-403* with a validation MAE of 2.48 and a test MAE of 2.39. The next best performer is *VEC-394* with values of 2.51 and 2.42 for validation and test losses respectively. The numbers of each experiment with the corresponding seed and losses can be seen in Table 4.

	id	seed	val_mae	test_mae
1	*VEC-382*	7399	2.43	2.50
2	*VEC-383*	2125	2.71	2.48
3	*VEC-384*	7889	2.51	2.46
4	*VEC-386*	1167	2.67	2.48
5	*VEC-387*	3512	2.52	2.54
6	*VEC-388*	9970	2.65	2.47
7	*VEC-389*	7698	2.71	2.45
8	*VEC-391*	8659	2.53	2.50
9	*VEC-392*	8010	2.57	2.54
10	*VEC-393*	6904	2.49	2.47
11	*VEC-394*	4422	2.51	2.42
12	*VEC-396*	7310	2.48	2.44
13	*VEC-397*	2086	2.60	2.44
14	*VEC-398*	6547	2.56	2.51
15	*VEC-399*	3781	2.67	2.53
16	*VEC-401*	6587	2.70	2.47
17	*VEC-402*	9677	2.46	2.51
18	*VEC-403*	6569	2.48	2.39
19	*VEC-404*	3131	2.56	2.49
20	*VEC-406*	75	2.55	2.51

Table 4: Experiment D1: Validation MAE with SWA optimiser and fixed *lr* of 1e−5. The top 3 performers are highlighted.

The best performer in *E1* is the experiment *VEC-295* with a validation MAE of 2.44 and a test MAE of 2.36. The next best performer is *VEC-286* with values of 2.46 and 2.36 for validation and test respectively. Both leading experiments had

almost the same outcome; the numbers of each experiment with the corresponding seed, *lr* and losses can be seen in Table 5.

	id	seed	lr	val_mae	test_mae
1	*VEC-283*	6628	0.0251	2.36	2.52
2	*VEC-286*	7122	0.0209	2.46	2.36
3	*VEC-291*	426	0.0251	2.46	2.45
4	*VEC-295*	6532	0.0302	2.44	2.36
5	*VEC-301*	420	0.0209	2.47	2.37
6	*VEC-304*	9010	0.0251	2.44	2.42
7	*VEC-307*	958	0.0251	2.39	2.44
8	*VEC-311*	6795	0.0251	2.44	2.38
9	*VEC-314*	1298	0.0251	2.45	2.41
10	*VEC-321*	9701	0.0363	2.39	2.40
11	*VEC-325*	8329	0.0251	2.44	2.42
12	*VEC-330*	4202	0.0251	2.42	2.46
13	*VEC-334*	1532	0.0302	2.41	2.42
14	*VEC-337*	2646	0.0251	2.43	2.39
15	*VEC-340*	2952	0.0251	2.42	2.46
16	*VEC-343*	2158	0.0437	2.44	2.40
17	*VEC-346*	5482	0.0251	2.41	2.44
18	*VEC-349*	4667	0.0251	2.44	2.44
19	*VEC-352*	6527	0.0251	2.37	2.48
20	*VEC-354*	6811	0.0251	2.42	2.50

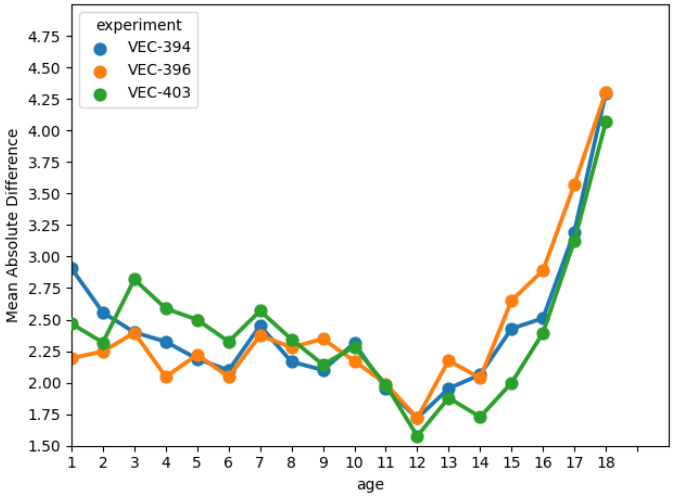
Table 5: Experiment E1: Validation MAE with SGD optimiser and *lr* finder [43]. The top 3 performers are highlighted.

The winning model (experiment *VEC-295*) was an outcome of a SGD optimisation approach with an initial optimal learning rate of 0.0302. The model produced a MAE in validation and test of 2.46 and 2.36 respectively. The performance per age for the top 3 best performing experiments *D1* and *E1* can be seen in Figure 9. The mean absolute difference (MAD) is the average absolute difference of two random variables X and Y independently and identically distributed. The formula is shown in Equation 3. This measure of statistical dispersion was used to calculate the performance per age. The winning model had a MAD performance range between 1.84 and 4.47. The performance was at its best for 2, 12 and 14 year-old subjects. The trend of the other experiments (*VEC-295, VEC-286 & VEC-311*) are similar. As can be seen in Figure 9b, they each perform well for 12-year-old subjects and the performance starts decreasing from 14 year-olds onwards in an exponential manner. In a similar way, the top 3 best performers for experiment *D1* have a behaviour inline with experiment *E1* with a good performance in 12 and 14 year-old subjects while having an exponential decrease in performance from 14 year-olds onwards, as can be seen in Figure 9a.

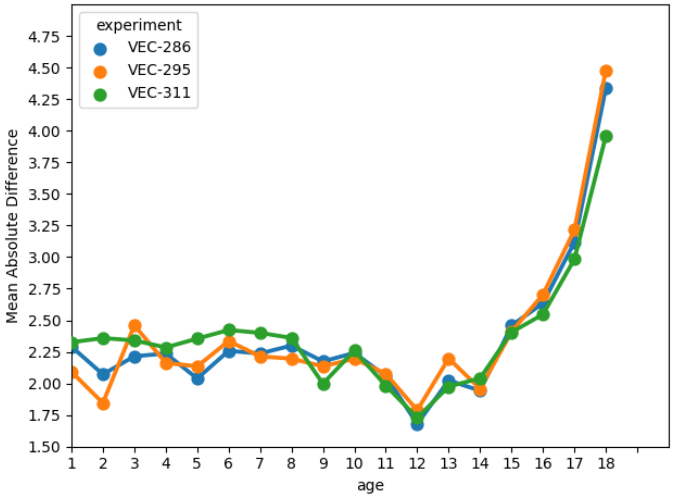
$$MAD = E[|X - Y|] \tag{3}$$

4.5. Running Times and Convergence

The runtime is associated to convergence due to the implementation of early stopping for each experiment. Once the loss function ceased to improve with a patience of 10 epochs, the



(a) Experiment *D1* - SGD fixed initial learning rate approach.



(b) Experiment *E1* - SWA *lr* finder approach

Figure 9: Performance per age for the top 3 best performers

training was stopped. Each optimiser was automatically logged to Neptune and further evaluated for both experiment *D1* and *E1*. The hardware used has a CPU processor of 2.8GHz (Quad-Core Intel Core i7), memory of 16GB 1600Mhz DDR3 and an Intel Iris Pro 1536 graphics card.

The SWA optimiser was able to converge the fastest compared to the rest of the algorithms in experiment *D1* with a mean value of approximately 7 minutes. Similar performance occurred in experiment *E1* for which its mean running time was inline with that of SGD with an approximate value of 13 minutes. Despite achieving a low runtime average in *E1*, the SGD algorithm performed the slowest of all algorithms executed for experiment *D1*. This indicates that SGD struggles with a fixed initial *lr* of 1e−4.

It is clear that in *D1* the run time average for the different algorithms varied greatly from each other particularly when compared to its *E1* counterpart, which has significantly less dispersion between the mean runtime of the algorithms as shown in

Figure 10. This suggests that *E1* has a more controlled runtime out of the two experiments. Moreover, with the exception of ADAM, the rest of the algorithms were also found to perform faster in *E1*. These outcomes were due to the automatic learning rate finder [43], which was made available in experiment *E1*.

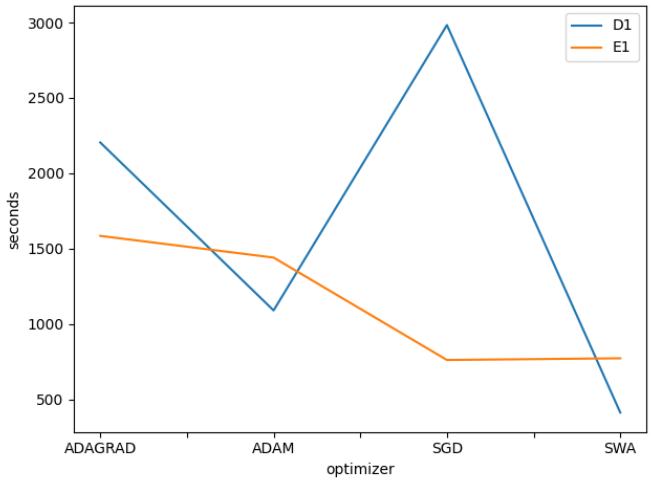


Figure 10: Average runtime per optimisation algorithms for experiments *D1* and *E1*

5. Discussion

The results presented in Section 4 describe a novel approach to obtain robust indicators of the performance of a model through distributions. The advantage of having several models that compete against each other, allows us to understand the tendency of the losses per optimisers and random initialisations. Even when all the hyper-parameters are held constant, there is a high variability in how well models generalise. By reducing the age estimation inputs to a vector of 512 (representing facial embeddings), tens to hundreds of experiments are able to be executed in a short time, enabling the evaluation of several approaches. Finding a set of experiments with means of 2.5 and below, is very encouraging. This means that the results obtained were not manipulated or obtained incidentally, but were consistent. It was observed that similar results were in line per optimiser and stable to the initial condition. Alone, the MAE of 2.5 for testing surpasses the state-of-the-art age estimation models. The best model obtained has a MAE in test of 2.36, which outperforms other age prediction models.

Prior to obtaining this performance, several data preprocessing, augmentation techniques, optimiser algorithms and learning rate initialisers were evaluated. All the correctly amalgamated methods produced promising results – particularly those engendered by the SWA optimiser with a non-fixed initial learning approach. Nevertheless, it must be considered that these models are only for underage single frontal-faced images, and will only work well for such age ranges and type of images – according to the *no-free-lunch theorem* that states that a single model cannot suit all problems. It is also observed that the models perform lower for 17 and 18 year-old subjects. This may be due to the similarity between these two ages, creating an encouraging topic to address with facial embeddings for future work. It is also worth mentioning that overall, the use of a cyclic-based learning rate finder not only improved the performance but decreased the running times of the experiments. Finally, the use of the SWA optimiser yielded optimal results where the convergence time was significantly faster than the other algorithms evaluated.

6. Conclusion and Future Work

6.1. Conclusions

Facial age estimation is still a challenging topic due to several factors including the environment, habits, ethnicity, diet, etc. Underage age estimation for digital forensics is continuously being studied and the performance has been improving, entitling digital forensic practitioners to use tools and techniques that include computational intelligence to detect and analyse evidence – particularly deep learning. Current models usually attempt to tackle several challenging factors that affect the age estimation performance such as facial occlusion, non-frontal faces, brightness, contrast, quality, etc. In our approach, a simpler challenge is addressed and a better performance is achieved. A distribution for the evaluation model metric is proposed, allowing researchers to chose a model with more confidence. The exploration of optimal learning rates was key in the influence of high performing underage age estimation models; the SWA optimiser with the cyclic-learning-rate-based approach is a promising setting that yields higher accuracy for underage age estimation than other optimisation approaches. Another factor that aided the performance was the quality of the dataset. Data augmentation techniques have been proven in the past to increase performance but the correct transformation must be chosen – specifically with facial embeddings. The calculation of the facial vectors enabled the use of simpler neural networks. And the experiments were managed swiftly without the use of a GPU. Collaborative tools were used to record and manage all the experiments. Tracking and visualising metrics such as loss and accuracy are paramount for researchers not only in digital forensics but in other areas.

6.2. Future Work

While using a fixed encoder (FaceNet) to produce facial embeddings worked well, fine-tuning the encoder by back-propagating the loss from the age estimation module with a low learning rate should produce a more optimal representation.

Trained with a regression loss, the models outlined in this work produce a point estimate of the subject's age. However, the value of such an estimate to a digital forensics end-user would be increased if the model instead produced a well-conditioned distribution (mean and variance), for example by applying the methods described in SWAG [44].

Finally, the use of a hyperparameter optimisation framework for machine learning such as Optuna [45] would aid the experiments to find improved performance for underage facial age estimation.

Acknowledgements

Our thanks to Alexei Bastidas, Eric Callanan, Hannah Jung, Rovic Perdon, Pete Pilley, David Wang and Tom Whiddett, all of whom gave generously of their time at various stages of this work.

References

1. Jain AK, Dass SC, Nandakumar K. Can soft biometric traits assist user recognition? In: *Biometric technology for human identification*; vol. 5404. International Society for Optics and Photonics; 2004:561–572.
2. Scanlon M. Battling the Digital Forensic Backlog through Data Deduplication. In: *Proceedings of the 6th IEEE International Conference on Innovative Computing Technologies (INTECH 2016)*. Dublin, Ireland: IEEE; 2016:10–14.
3. Du X, Hargreaves C, Sheppard J, Anda F, Sayakkara A, Le-Khac NA, Scanlon M. SoK: Exploring the State of the Art and the Future Potential of Artificial Intelligence in Digital Forensic Investigation. In: *Proceedings of the 15th International Conference on Availability, Reliability and Security*. ARES '20; New York, NY, USA: Association for Computing Machinery. ISBN 9781450388337; 2020:1–10. URL: https://doi.org/10.1145/3407023.3407068. doi:10.1145/3407023.3407068.
4. Rothe R, Timofte R, Gool LV. Deep expectation of real and apparent age from a single image without facial landmarks. *International Journal of Computer Vision* 2018;126(2-4):144–157.
5. Anda F, Lillis D, Le-Khac NA, Scanlon M. Evaluating automated facial age estimation techniques for digital forensics. In: *2018 IEEE Security and Privacy Workshops (SPW)*. IEEE; 2018:129–139.
6. Schroff F, Kalenichenko D, Philbin J. Facenet: A unified embedding for face recognition and clustering. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015:815–823.
7. Anda F, Le-Khac NA, Scanlon M. DeepUAge: Improving Underage Age Estimation Accuracy to Aid CSEM Investigation. *Forensic Science International: Digital Investigation* 2020;32:300921. URL: http://www.sciencedirect.com/science/article/pii/S2666281720300160. doi:https://doi.org/10.1016/j.fsidi.2020.300921.
8. Dixon E. Selfie-FV: Face vectors with age ground-truth. https://github.com/EdwardDixon/selfie-fv; 2020.
9. Geng X, Yin C, Zhou Z. Facial Age Estimation by Learning from Label Distributions. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2013;35(10):2401–2412.
10. Lanitis A, Taylor CJ, Cootes TF. Toward automatic simulation of aging effects on face images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2002;24(4):442–455.
11. Ricanek K, Tesafaye T. Morph: A longitudinal image database of normal adult age-progression. In: *7th International Conference on Automatic Face and Gesture Recognition (FGR06)*. IEEE; 2006:341–345.
12. Chao WL, Liu JZ, Ding JJ. Facial age estimation based on label-sensitive learning and age-oriented regression. *Pattern Recognition* 2013;46(3):628 – 641. doi:https://doi.org/10.1016/j.patcog.2012.09.011.
13. Liu H, Lu J, Feng J, Zhou J. Group-aware deep feature learning for facial age estimation. *Pattern Recognition* 2017;66:82 – 94. doi:https://doi.org/10.1016/j.patcog.2016.10.026.
14. Escalera S, Fabian J, Pardo P, Baró X, Gonzàlez J, Escalante HJ, Misevic D, Steiner U, Guyon I. ChaLearn Looking at People 2015: Apparent Age and Cultural Event Recognition Datasets and Results. In: *2015 IEEE International Conference on Computer Vision Workshop (ICCVW)*. 2015:243–251.
15. Bengio Y, Louradour J, Collobert R, Weston J. Curriculum Learning. In: *Proceedings of the 26th Annual International Conference on Machine Learning*. ICML '09; New York, NY, USA: Association for Computing Machinery. ISBN 9781605585161; 2009:41–48. URL: https://doi.org/10.1145/1553374.1553380. doi:10.1145/1553374.1553380.
16. Antipov G, Baccouche M, Berrani SA, Dugelay JL. Apparent Age Estimation From Face Images Combining General and Children-Specialized Deep Learning Models. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*. 2016:96–104.
17. Anda F, Lillis D, Kanta A, Becker BA, Bou-Harb E, Le-Khac NA, Scanlon M. Improving Borderline Adulthood Facial Age Estimation through Ensemble Learning. In: *Proceedings of the 14th International Conference on Availability, Reliability and Security*. ARES '19; New York, NY, USA: Association for Computing Machinery. ISBN 9781450371643; 2019:1–8. URL: https://doi.org/10.1145/3339252.3341491. doi:10.1145/3339252.3341491.
18. Jekel CF, Haftka RT. Classifying Online Dating Profiles on Tinder using FaceNet Facial Embeddings. *CoRR* 2018;abs/1803.04347. URL: http://arxiv.org/abs/1803.04347.
19. Terhörst P, Huber M, Kolf JN, Damer N, Kirchbuchner F, Kuijper A. Multi-algorithmic fusion for reliable age and gender estimation from face images. In: *2019 22th International Conference on Information Fusion (FUSION)*. IEEE; 2019:1–8.
20. Eidinger E, Enbar R, Hassner T. Age and gender estimation of unfiltered faces. *IEEE Transactions on Information Forensics and Security* 2014;9(12):2170–2179.
21. Swaminathan A, Chaba M, Sharma DK, Chaba Y. Gender Classification using Facial Embeddings: A Novel Approach. *Procedia Computer Science* 2020;167:2634–2642.
22. Zhang Z, Song Y, Qi H. Age progression/regression by conditional adversarial autoencoder. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017:5810–5818.
23. Leong FH. Deep learning of facial embeddings and facial landmark points for the detection of academic emotions. In: *Proceedings of the 5th International Conference on Information and Education Innovations*. 2020:111–116.
24. Shorten C, Khoshgoftaar TM. A survey on image data augmentation for deep learning. *Journal of Big Data* 2019;6(1):60.
25. Lv JJ, Shao XH, Huang JS, Zhou XD, Zhou X. Data augmentation for face recognition. *Neurocomputing* 2017;230:184 – 196. doi:https://doi.org/10.1016/j.neucom.2016.12.025.
26. Liu X, Zou Y, Kuang H, Ma X. Face Image Age Estimation Based on Data Augmentation and Lightweight Convolutional Neural Network. *Symmetry* 2020;12(1):146.
27. Huang GB, Ramesh M, Berg T, Learned-Miller E. Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments. Tech. Rep. 07-49; University of Massachusetts, Amherst; 2007.
28. Bloice MD, Roth PM, Holzinger A. Biomedical image augmentation using Augmentor. *Bioinformatics* 2019;35(21):4522–4524. doi:10.1093/bioinformatics/btz259.
29. Wallhoff F. Facial Expressions and Emotions Database. 2006. URL: http://www-prima.inrialpes.fr/FGnet/html/home.html.
30. Wani MA, Bhat FA, Afzal S, Khan AI. Advances in deep learning; vol. 57. Springer; 2020.
31. Anda F, Becker BA, Lillis D, Le-Khac N, Scanlon M. Assessing the Influencing Factors on the Accuracy of Underage Facial Age Estimation. In: *2020 International Conference on Cyber Security and Protection of Digital Services (Cyber Security)*. 2020:1–8.
32. Masi I, Tran AT, Hassner T, Sahin G, Medioni G. Face-specific data augmentation for unconstrained face recognition. *International Journal of Computer Vision* 2019;127(6-7):642–667.
33. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, et al. Scikit-learn: Machine learning in Python. *the Journal of machine Learning research* 2011;12:2825–2830.
34. King DE. Dlib-ml: A Machine Learning Toolkit. *Journal of Machine Learning Research* 2009;10:1755–1758.
35. Bollé T, Casey E, Jacquet M. The role of evaluations in reaching decisions using automated systems supporting forensic analysis. *Forensic Science International: Digital Investigation* 2020;34:301016. URL: http://www.sciencedirect.com/science/article/pii/S2666281720300755. doi:https://doi.org/10.1016/j.fsidi.2020.301016.
36. Fernández C, Huerta I, Prati A. A Comparative Evaluation of Regression Learning Algorithms for Facial Age Estimation. In: Ji Q, B. Moeslund T, Hua G, Nasrollahi K, eds. *Face and Facial Expression Recognition from Real World Videos*. Cham: Springer International Publishing; 2015:133–144.
37. Liao H, Yan Y, Dai W, Fan P. Age estimation of face images based on

CNN and divide-and-rule strategy. *Mathematical Problems in Engineering* 2018;2018.
38. Kingma DP, Ba J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:14126980* 2014;.
39. Duchi J, Hazan E, Singer Y. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research* 2011;12(7).
40. Bottou L. Large-scale machine learning with stochastic gradient descent. In: *Proceedings of COMPSTAT'2010*. Springer; 2010:177–186.
41. Izmailov P, Podoprikhin D, Garipov T, Vetrov D, Wilson A. Averaging Weights Leads to Wider Optima and Better Generalization. In: *34th Conference on Uncertainty in Artificial Intelligence*; vol. 2. 2018:876–885.
42. neptune.ai . Neptune: experiment management and collaboration tool. 2020. URL: `https://neptune.ai`.
43. Smith LN. Cyclical Learning Rates for Training Neural Networks. In: *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE; 2017:464–472.
44. Maddox WJ, Izmailov P, Garipov T, Vetrov DP, Wilson AG. A Simple Baseline for Bayesian Uncertainty in Deep Learning. In: Wallach H, Larochelle H, Beygelzimer A, d'Alché-Buc F, Fox E, Garnett R, eds. *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc.; 2019:13153–13164.
45. Akiba T, Sano S, Yanase T, Ohta T, Koyama M. Optuna: A Next-generation Hyperparameter Optimization Framework. In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2019:2623–2631.