Deep learning pre-trained model as feature extraction in facial recognition for identification of electronic identity cards by considering age progressing

M Us gan*, R Ferdiana and I Ardiyanto

Department of Electrical and Information Engineering, Gadjah Mada University, Yogyakarta, Indonesia

*Email: muhammadusgan@mail.ugm.ac.id

Abstract. The usage of the face recognition system is various, one of which is to identify missing people. Cases of missing persons can usually happen to people who do not carry an identity card, such as people with mental disorders. Identity cards have some information, namely name, address, date of birth, and a photo ID of a face. This information has stored at the civil registry office, which is an Indonesian government agency. Therefore to solve this problem, we use a photo ID as a dataset to identify someone's identity. Photo ID as a dataset has age progressing factors in facial recognition systems. Several previous studies have discussed age progressing factors. Previous research used MORPH, CACD, and FGNET datasets. The dataset has several photos of faces consisting of various age levels for the same subject, but it is different from the photo ID that we use. For the training data process, we only use one data, namely a photo ID used to recognize faces at this point. Therefore we use the pre-trained VGGFace2 model, and then, fine-tuning the data during the training process, we use AM-Softmax loss as a loss function. Then the classification is done using SVM. The method we use than compared with the Moustafa method, which also uses a pre-trained model. The results we obtained have a better accuracy performance of 0.9351 compared to the Moustafa method of 0.733

1. Introduction
The facial recognition system is a method used to recognize a person's identity, and it is a topic that has been studied in an unconstrained environment which has made rapid progress in recent studies. However, facial recognition caused by large age variation is still an intriguing subject for study. Face aging variation is one of the facial recognition system problems due to the large amount of intra-personal variation caused by age progression, which changes the face's shape and texture [1].

The identity card is indeed owned by every citizen, especially in the Republic of Indonesia's territory. The identity cards have some information, namely name, address, date of birth, and photo ID, which is stored in the civil registry. The information on the identity card is related to family data. Many missing person cases occurred where they did not carry the identity card, causing them to be stranded on the road or accommodated by the foundation. These cases of missing persons usually last for years. His family must have tried to find his relatives, but it is not easy to find them with the vast
territory of Indonesia. These missing people are usually people who have mental disorders or people who have lost their memories. This study uses ID photos from identity cards as training data to identify someone's identity to overcome this problem. Facial recognition using identity cards is an example of facial recognition caused by age progressing. This is because the photo ID on the identity card has a different age from the current face.

In general, there are two main parts of the facial recognition system, which are feature extraction and classification. The feature extraction method used in the facial recognition system can use a pre-trained model was done by [1, 2, 3]. The pre-trained model [2] used is VGGFace [4] with the VGG- Very-Deep-16 CNN architecture model. The pre-trained model is then fine-tuned to solve the face recognition case caused by age progressing. In their research [2] used the feature fusion to perform a combination of features obtained from several layers of the model VGGFace, namely flatten, fc6, and fc7, with each vector length, is 25088, 4096, 4096. The feature fusion used is Multimodal Discriminant Correlation Analysis (MCDA) [5]. The output of the feature fusion is a feature that is used for classification.

Another method that uses a pre-trained model is [1]. The pre-trained model architecture used is AlexNet. The feature vectors generated from the model are then encode based on their proposed algorithm. However, their method is not suitable for use in the dataset proposed in this study because the method we are proposing uses only one dataset for each subject in the training data process. In contrast, the method they are proposing uses the fusion feature on the same identity for different images, requiring several data for the same identity.

Xiangyu Zhu et al. [3] proposed research using photo ID as a dataset known as the ID vs Spot (IvS) dataset. The research they are doing uses a pre-trained model where they build themselves using the LFW dataset. The method procedure they propose consists of three stages, namely classification (pre-learning), verification (transfer learning), and classification (fine-grained learning). However, what is different from what we are going to do is that our research uses only one data for each subject during the training process, in contrast to [3], which used two-sample data for each subject in the second stage of training data. Based on this description, it is known that the pre-trained model can be used to build facial recognition caused by age progressing.

VGGFace [6] is a pre-trained model used in facial recognition systems to identify electronic identity cards caused by age development. VGGFace uses the ResNet-50 [7] architecture. In addition to the pre-trained model used for feature extraction, a loss function when training data affects the results to be obtained. Therefore, this study uses the Addictive Margin Softmax (AM-Softmax) loss [8]. This loss function minimizes intra-class variation and maximizes inter-class features. This is so important that it produces a discriminating feature. This loss function has been compared with several other loss function approaches, including Softmax [9], Softmax + 75% dropout, Center Loss [10], NormFace [11], and A-Softmax [12]. Therefore, the softmax loss used by the pre-trained VGGFace model is replaced by AM-Softmax loss when fine-tuning. Furthermore, the classification method used is the Support Vector Machine (SVM) [1, 5, 7].

Based on the description above, this study used a pre-trained model from VGGFace2 and then performed fine-tuning. Next, we conduct training data using AM-Softmax loss as a loss function then classify it using SVM. This study's results were compared with [2], who also discussed facial recognition systems due to increasing age. The method they proposed was applied to the dataset we used.
2. Related work

2.1. VGGFACE2

VGGFace2 [6] is also termed a new large-scale face dataset. VGGFace2 is a pre-training dataset trained with a collection of images obtained from Google Image Search, which contains 3.31 million images of celebrities from 9131 subjects, covering various ethnicities such as Chinese and Indians have more number than VGGFace. The gender ratio in the VGGFace2 dataset is 59.3% male and 48.7% female. Furthermore, each subject consists of 90 to 843 images, with an average of 362 images.

The pictures are composed of various poses, age, lighting, ethnicity, and professions. There are three main objectives to be achieved from collecting this dataset are

- It has a large number of both identities and many pictures of each identity
- It has a wide range of poses, ethnicity, and age
- To minimize label noise

The dataset from VGGFace2 is then trained using the ResNet-50 and SE-ResNet-50 (SENET) architectures. The output of this network architecture is a face descriptor, which is then classified using the SVM (Support Vector Machine). The Cosine Similarity is used to represent the similarity between two vectors.

The first test of VGGFace2 is facing probing across poses and face probing across age. This first test uses the VGGFace and MS1M dataset as a comparison. Face probing across poses testing uses 3 conditions, namely front, three-quarter, and profile. Testing for probing across age uses 2 conditions, namely young and mature. Cosine similarity is used to obtain the similarity score. From the test results, it was found that VGGFace2 has better performance than the other two datasets.

The second test is the performance evaluation on the IJB-A, IJB-B, IJB-C dataset. The first thing to do is create a pre-trained model from VGGFace, MS1M, and VGGFace2. The architecture used is ResNet-50. Then evaluate the performance of the pre-trained model using IJB-A, IJB-B, IJB-C. From the results of the trial conducted, VGGFace has better performance than other pre-trained models. This is because VGGFace2 consists of various poses and ages.

2.2. AM-softmax (Additive Margin Softmax)

AM-Softmax loss function development is used for deep face verification, which can be seen as a metric learning problem. AM-Softmax is a learning large-margin face features that minimize intra-class and maximize inter-class features. This is important to do to get good performance. AM-Softmax is not the first approach that optimizes the margin face features. Previously, some approaches modified the softmax loss function, namely Angular softmax (A-Softmax) [12] and Large Margin Softmax (L-Softmax) [14]. The large margin property $\psi(\theta)$ of AM-Softmax [8] is more straightforward than that of A-Softmax and L-Softmax.

$$\psi(\theta) = \cos(\theta - m)$$

The value of $m$ is a number that ranges from 0.35 to 0.4. AM-Softmax normalizes the weight and input features parameters on the last fully connected layer ($f_i$) as done by A-Softmax. Besides, AM-Softmax also introduced a new hyperparameter $s$, which scales the cosine value. In general, the equation of AM-Softmax is as follows.
\[ L_{AMS} = -\frac{1}{N} \sum_{i=1}^{n} \log \left( \frac{e^{s(\cos \theta_{yi}-m)}}{e^{s(\cos \theta_{yi}-m)} + \sum_{j=1: j \neq y}^{c} e^{s \cos \theta_{j}}} \right) \]

\[ = -\frac{1}{N} \sum_{i=1}^{n} \log \left( \frac{e^{s(W_{yi}^{T}f_{i}-m)}}{e^{s(W_{yi}^{T}f_{i}-m)} + \sum_{j=1: j \neq y}^{c} e^{sW_{j}^{T}f_{j}}} \right) \]  \hspace{1cm} (2)

Where the value of s is the scaling factor, the scaling factor set is 30. \( W_{yi}^{T} f_{i} \) is called the logit target. AM-Softmax was then tested using the LFW and MegaFace datasets. Subsequently, an appeal was made against several other types of loss functions, namely Softmax, Softmax + 75% dropout, Center Loss, NormFace, and A-Softmax. The results of this test show that AM-Softmax obtained good performance compared to other loss functions.

2.3. SVM (Support Vector Machine)

SVM is a technique introduced by Vapnik in 1992 as an efficient classification technique for non-linear problems. SVM works by trying to find hyperplanes by maximizing the distance between classes. Hongling Chen[13], in their research, discusses the effect of dimensional reduction on face recognition using the VGG-16 model as feature extraction and SVM is used as a classification. The dimension reduction used is PCA (Principal Component Analysis). The results of feature extraction from the VGG-16 model were then reduced in PCA dimensions and then classified using SVM. The value of the dimensional reduction used was 400. The proposed system was then tested on the CelebA dataset. Their research shows that with the reduction dimension of the vector features on the VGG-16 Net with SVM as a classification, the accuracy increases. Compared to some of the proposed methods such as DeepFace, TL Joint Bayesian, DFD, and several other methods, the proposed method has a better accuracy level.

Moustafa [2] discusses age invariant face recognition. The proposed method uses the VGG-Face model as feature extraction then performs feature fusion using the DCA method. The resulting fusion features are then classified using two methods: the SVM and KNN classifier. This proposed method was then tested on the FGNET and MORPH datasets. The trials conducted showed that the accuracy of SVM was superior to that of the KNN on the FGNET dataset, while on the MORPH dataset, the KNN classifier had better accuracy.

3. Proposed method

This section proposes a method to solve the facial recognition case for electronic ID cards, taking into account age progressing. The proposed method consists of 3 stages, namely image preprocessing, feature extraction, classification. The results of the proposed method were then compared with the method [2]. In general, the proposed method is shown in Figure 1.

3.1. Image preprocessing

The first step is image preprocessing, which is to crop the face area for training data and testing data. The dimensions of the cropped image adopted from VGGFace2 are 224 x 224 pixels where Multitask Cascaded Convolutional Networks (MTCNN) [15] is used to crop the face area. After cropping the face, then changing the image whose shape is RGB is converted to grayscale form. Then perform data augmentation [16] on the training data by rotating the image. The value of the image rotation is 4 ° and -4 °. An example of data augmentation is shown in Figure 2.
Figure 1. ResNet 50 Model where the input feature classifier layer and its layers are normalized

Figure 2. Data augmentation performed on training data is by rotating -4° and 4° in the image

3.2 Fine-tuning and feature extraction

After preparing the data to be used, the next step is to prepare the model and do fine-tuning. The model used is VGGFace2, which uses ResNet-50 architecture. This model has been provided publicly by [6]. Furthermore, the model architecture is modified in the classifier layer, changing the number of output classes to 1 x 283. The output class's value is adjusted to the number of data classes we have, which is 283 identities sourced from primary data and tertiary data.

The next step is to add a normalization function to the weight layer classification parameter and normalize the feature input on the classification layer. Next, change the required gradient parameter for convolution layer one to convolution layer five, namely weight and bias to false, while in the required gradient layer classifier is true. This is done because the model uses model parameters that have been previously trained, while the pre-trained layer parameters are updated during the training process. An overview of the model architecture is shown in Figure 1. The classifier uses a convolution layer based on the VGGFace2 model architecture, while the AM-Softmax loss function uses a fully connected layer. Therefore, the tests are carried out using the original model from VGGFace2 and then comparing it with the modified VGGFace2 architecture in the classifier layer using fully connected layers.

The next procedure is to perform training data, but beforehand it is necessary to prepare some optimizer parameters and loss function. The optimizer used is Stochastic Gradient Descent (SGD). The properties applied to the SGD optimizer are a learning rate of 0.001 and momentum of 0.9. Furthermore, the SGD optimizer parameters are the parameters of the weight and bias of the classifier layer. Next is the loss function used is AM-Softmax[8] with the parameter scale factor (s) 30 and margin (m) 0.4. Then the data training was carried out for 500 epochs.
3.3 Classification

The third step is to perform classification using a Support Vector Machine (SVM). Before using SVM, it is necessary to prepare vector features to fit the SVM. This vector feature results from feature extraction of the layer classifier features from the model that has been training 500 epochs. The SVM classifier property uses a linear kernel function [1, 7]

![Figure 3](image)

Figure 3. The dataset used during the training and testing process. (a) is a primary dataset and (b) is a tertiary dataset

4. Experiment

This section explains the data used and the results of the experiments that have been carried out

4.1. Datasets

We used two types of datasets to test the proposed method, namely the primary dataset for electronic identity cards and the tertiary dataset IvS [3]. The primary dataset is data that has been collected manually through questionnaires, while the tertiary dataset is made available publicly. Both the primary and tertiary datasets that have been collected are then filtered against the usable data. A description of the number of objects between primary data and tertiary data is shown in Table 1. After filtering the data, the next process is image preprocessing, as described in section 3.1. Examples of primary and tertiary datasets are shown in Figure 3.

![Table 1](image)

Table 1. Description of the dataset

| Dataset | Subject | Total Training | Testing |
|---------|---------|----------------|---------|
| Primary | 43      | 43             | 66      |
| Tertiary| 240     | 240            | 381     |
| Total   | 283     | 283            | 447     |

4.2. Experiment setup

This section explains preparation when conducting experiments. Experiments were carried out using primary and tertiary datasets. The primary dataset is obtained either directly or through questionnaires from participants, and the tertiary dataset is the public IvS dataset. The two datasets are then carried out, preprocessing and separating the training data and test data. The dataset structure used is shown in Figure 4. The amount of training data used is one sample per subject. The training data used is a photo with a straight face facing the front with a plain background. An example of training data is a photo of
an electronic identity card. So the scenario is to predict the current photo based on the photo ID card that has been taken in the past.

![Figure 4. The dataset structure is used where the label is the name of the person. An example of a label is 02 Andi Prademon. Furthermore, in the label there is a train and test folder where the folder contains the photos used.](image)

Besides, there also be compared the pre-trained VGGFace2 model on the use of AM-Softmax loss, namely, comparing the original VGGFace2 model with the modified VGGFace2 model in the classifier layer section by replacing it with a fully connected layer. In addition to seeing the accuracy obtained, but also seeing the similarity score obtained between the test data vector feature and predictive data, which is a training data vector feature. To perform this task, we use the public dataset LFW [17] and AFAD [18] as an evaluation in which the dataset is not part of the training model. Cosine similarity is used as a method to measure the similarity score between two vectors. The experiment uses Google Colabs tools with python 3 specifications and takes advantage of the CUDA runtime available on Google Colabs to speed up computation. The framework used is Pytorch.

4.3. Experimental results

The experimental results are obtained after carrying out the steps discussed in section 3. In this initial section, we discussed the comparison between the original VGGFace2 and VGGFace2 architecture, which in the classifier section uses the fully connected layer, or it can be called the comparison of the original VGGFace2 and modified VGGFace2. Both models use AM-Softmax loss as a loss function, which is different only in the classifier part. The model is then carried out by training data on the primary and tertiary datasets. Then evaluated by looking at the accuracy obtained and measuring the similarity score of vector features using the cosine similarity between the test data and the prediction results, which are training data. The evaluation data used is the public dataset from LFW [17] and AFAD [18]. This is done to justify a prediction based on the specified threshold score. A comparison of the VGGFace2 model between the original and modified is shown in Figure 5.
Sometimes we need to justify an object based on its similarity and dissimilarity values. Figure 5 shows a comparison between original VGGFace2 and modified VGGFace2 using primary, tertiary, LFW, and AFAD datasets. The primary and tertiary datasets represent similarity scores. The LFW and AFAD datasets represent dissimilarity scores. The threshold score to be determined in this experiment is 0.5. These results are shown in Table 2 and Table 3.

![Figure 5](image)

**Figure 5.** The similarity score is obtained from the vector feature of the test data on the prediction results (vector features from the training data). (a) is original VGGFace2 and (b) is Modified VGGFace2. Based on the good picture, the original VGGFace2 and Modified VGGFace2 tend to get a similarity score which tends to be greater than 0.5.

Table 2. The similarity score of test data from primary and tertiary datasets with threshold values above 0.5

| Model            | Classifier            | Threshold | Primary | Tertiary |
|------------------|-----------------------|-----------|---------|----------|
| Original VGGFace2 | Convolutional layer   | 0.5       | 0.8906  | 0.9518   |
| Modified VGGFace2| Fully connected layers| 0.5       | 0.8906  | 0.9519   |

Table 3. The dissimilarity score of the test data from the LFW and AFAD datasets with a threshold value below 0.5

| Model            | Classifier            | Threshold | LFW     | AFAD     |
|------------------|-----------------------|-----------|---------|----------|
| Original VGGFace2 | Convolutional layer   | 0.5       | 0.920   | 0.875    |
| Modified VGGFace2| Fully connected layers| 0.5       | 0.9240  | 0.8988   |

Table 2 and Table 3 show the similarity score above 0.5 and dissimilarity score below 0.5. Table 2 shows that modified VGGFace2 is 0.9519 has better value than original VGGFace2 is 0.9518 in the tertiary dataset, and Table 3 shows modified VGGFace2 also has a better value than original VGGFace2, both the LFW and AFAD datasets. Furthermore, the accuracy obtained is shown in Table 4.
Table 4. The accuracy obtained by Original VGGFace2 and Modified VGGFace2

| Metode                  | Dataset             |
|-------------------------|---------------------|
|                         | Primary | Tertiary | Primary + Tertiary |
| Original VGGFace2       | 0.9696  | 0.9265   | 0.9328            |
| Modified VGGFace2       | 0.9696  | 0.9291   | 0.9351            |

A comparison of the accuracy obtained between original and Modified VGGFace2 in Table 4 shows that Modified VGGFace2 is 0.9291 has better accuracy on the tertiary dataset than original VGGFace2 is 0.9265. Based on Table 4, it can be concluded that modifying the classifier into fully connected layers on the classifier layer increases the accuracy obtained and the similarity score.

After determining the model to be used, the next step is to compare the proposed method with the other method [2] to determine the suitable method for the facial recognition system based on the dataset we use. Both our proposed method and [2] method also use a pre-trained model. The experimental results are shown in Table 5.

Table 5. Experimental results

| Metode                               | Dataset             |
|--------------------------------------|---------------------|
|                                      | Primary | Tertiary | Primary + Tertiary |
| ResNet + AM-Softmax Loss             | 0.939   | 0.918    | 0.9217            |
| (proposed method)                    |         |          |                   |
| ResNet + AM-Softmax Loss + SVM       | 0.9696  | 0.9291   | 0.9351            |
| (proposed method)                    |         |          |                   |
| Moustafa [2]                         | 0.65    | 0.748    | 0.733             |

The experimental results in Table 5 show that our proposed method, namely ResNet + AM-Softmax Loss + SVM, shows better accuracy performance than the method [2]. Besides, the fine-tuning result model as feature extraction using AM-Softmax as a loss function and SVM as a classifier obtained better accuracy performance compared to without using SVM as a classifier. Thus our proposed method is suitable for the electronic identity card photo dataset in facial recognition systems.

5. Conclusion

Based on the experimental results that have been stated in section 4, it can be concluded that using the VGGFace2 pre-trained model then adjusts the classifier section, namely the input of the classifier feature, and the classifier layer is normalized. Next, do training data for 500 epochs and AM-Softmax as a loss function. Then the model is used as feature extraction and uses SVM as a classifier. It should be noted that this experiment conducted a comparison between the original VGGFace2 (layer classifier using convolution layer) and modified VGGFace2 (layer classifier using fully connected layer). The comparison of the two shows that modified VGGFace2 has a better accuracy of 0.9351 compared to the original VGGFace2 0.9328. The accuracy performance obtained from modified VGGFace2 and then compared with method [2] obtained an accuracy of 0.733, which also uses a pre-trained model. Based on that explanation, this shows that modified VGGFace2 has better accuracy performance than the method [2] so that it can be used as a facial recognition system method for the electronic photo identity card.
References

[1] Shakeel M S and Lam K M 2019 Deep-Feature Encoding-Based Discriminative Model for Age-Invariant Face Recognition Pattern Recognit. 95 (16) 442–457
[2] Moustafa A A, Elnakib A and Areed N F F 2020 Age-Invariant Face Recognition Based on Deep Features Analysis Signal, Image Video Process. 14 (2) 1027–1034
[3] Zhu X, Liu H, Lei Z, Shi H, Yang F, Yi D, Qi G and Li S Z 2019 Large-Scale Bisample Learning on ID Versus Spot Face Recognition International Journal of Computer Vision 127 (4) 684–700
[4] Parkhi O M, Vedaldi A and Zisserman A 2015 Deep Face Recognition Proceedings of the British Machine Vision Conference (BMVC) 41 1-12
[5] Haghighat M, Abdel-Mottaleb M and Alhalabi W 2016 Discriminant Correlation Analysis: Real-Time Feature Level Fusion for Multimodal Biometric Recognition IEEE Trans. Inf. Forensics Secure 11 (9) 1984–1996
[6] Cao Q, Shen L, Xie W, Parkhi O M and Zisserman A 2018 VGGFace2: A dataset for recognizing faces across pose and age in Proceedings - 13th IEEE International Conference on Automatic Face and Gesture Recognition 67–74
[7] He K, Zhang X, Ren S and Sun J 2016 Deep residual learning for image recognition in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition 770–778
[8] Wang F, Cheng J, Liu W and Liu H 2018 Additive Margin Softmax for Face Verification IEEE Signal Processing Letter 25 (7) 926–930
[9] Sun Y, Wang X and Tang X 2014 Deep learning face representation from predicting 10,000 classes in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition 1891–1898
[10] Wen Y, Zhang K, Li Z and Qiao Y 2016 A discriminative feature learning approach for deep face recognition in Computer Vision -- ECCV 2016 499–515
[11] Wang F, Xiang X, Cheng J and Yuille A L 2017 NormFace: L2 hypersphere embedding for face verification in MM 2017 - Proceedings of the 2017 ACM Multimedia Conference 1041–1049
[12] Liu W, Wen Y, Yu Z, Li M, Raj B and Song L 2017 SphereFace: Deep hypersphere embedding for face recognition in Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017 6738–6746
[13] Chen H and Haoyu C 2019 Face Recognition Algorithm Based on VGG Network Model and SVM Journal Physics Conference Series 1229 (1) 012015
[14] Liu W, Wen Y, Yu Z and Yang M 2016 Large-Margin Softmax Loss for Convolutional Neural Networks in International Conference on Machine Learning 507–516
[15] Zhang K, Zhang Z, Li Z and Qiao Y 2016 Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks IEEE Signal Process. Lett. 23 (10) 1499–1503
[16] Wang X, Wang K and Lian S 2020 A survey on face data augmentation for the training of deep neural networks Neural Comput. Appl. 32 (19) 15503–15531
[17] Huang G B, Ramesh M, Berg T and Learned-Miller E 2008 Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments Workshop on Faces
in ‘Real-Life’ Images: Detection, Alignment, and Recognition 1-14

[18] Niu Z, Zhou M, Wang L, Gao X and Hua G 2016 Ordinal regression with multiple output CNN for age estimation in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition 4920-4928