

Deep-learning-based face recognition for worker access control management in hazardous areas

Chun-myoung Noh¹ · Su-bong Lee² · Jae-chul Lee[†]

(Received March 31, 2021 : Revised April 27, 2021 : Accepted May 20, 2021)

Abstract: Face recognition (FR) technology, which combines computer vision and artificial intelligence, has recently attracted significant attention as a means of identification. Among biometric technologies, FR technology is used in various fields because it does not require physical contact and is hygienic and convenient. Generally, FR processes use imaging equipment to extract facial feature data representing human faces. One can recognize faces by matching the extracted data to facial feature data stored in a database. In this study, we compared the performances of existing deep-learning-based face detection algorithms (i.e., dlib and the single-shot multi-box detector Mobilnet V2) and FR algorithms (i.e., visual geometry groups and ResNet), and developed new FR algorithms, which are crucial for worker access control systems in hazardous regions. To analyze field applicability, we attempted to implement FR algorithms with high prediction accuracy in various scenarios (e.g., subjects wearing helmets, protective glasses, or both). We applied regularization to improve the performance of the implemented algorithms. Additionally, related data were collected and analyzed to recognize the number of people wearing masks. The results of recognizing the number of people wearing masks were obtained. These results will support future research on safety issues in the manufacturing industry and the use of face and image recognition techniques.

Keywords: Access control, Computer vision, Deep learning, Face recognition

1. Introduction

Computer vision is a technology that extracts meaningful information by recognizing photographs (still images) or videos using computers. Recently, computer vision combined with deep learning has been applied in various industries, such as autonomous driving and/or navigation, industrial robots, and face recognition (FR). In particular, computer vision (i.e., image recognition) technology incorporating big data in the manufacturing industry has become a core component of the Fourth Industrial Revolution. This technology is used to reduce defect rates through defect detection and its application is expanding.

Computer vision technology that interprets characters, biometric information (e.g., faces or fingerprints), license plates, etc. has been widely applied in smart cities, smart factories, and traffic safety management systems, where safety and control are critical. In particular, FR is a biometric technology used for identity authentication in military, financial, and public security areas. FR identifies people using input images and the process is divided into face detection, face landmark detection, face feature extraction, and FR (Figures 1 and 2).



Figure 1: Software flow for FR

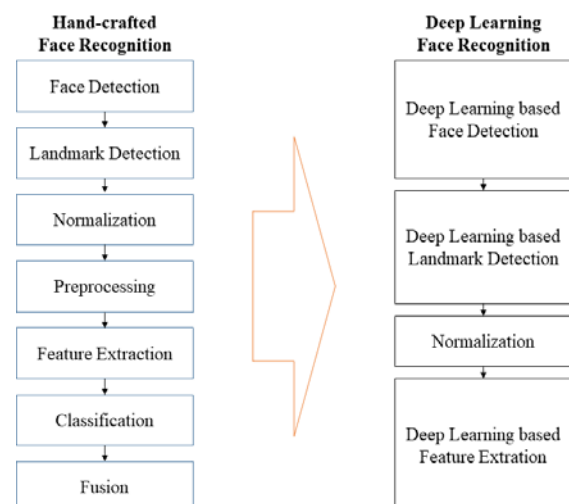


Figure 2: Changes in recognition technology

[†] Corresponding Author (ORCID: <http://orcid.org/0000-0002-1699-7568>): Professor, Department of Ocean System Engineering, Gyeongsang National University, Cheondaehukhi-Gil 38, Tongyeong, Gyeongnam, 53064, Korea, E-mail: j.c.lee@gnu.ac.kr, Tel: +82-772-9195

1 Ph. D. Candidate, Department of Ocean System Engineering, Gyeongsang National University, E-mail: n941114@gmail.com, Tel: +82-772-9199

2 Researcher, R&D, ADIA Lab, E-mail: subong.lee@adialab.com

This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0>), which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

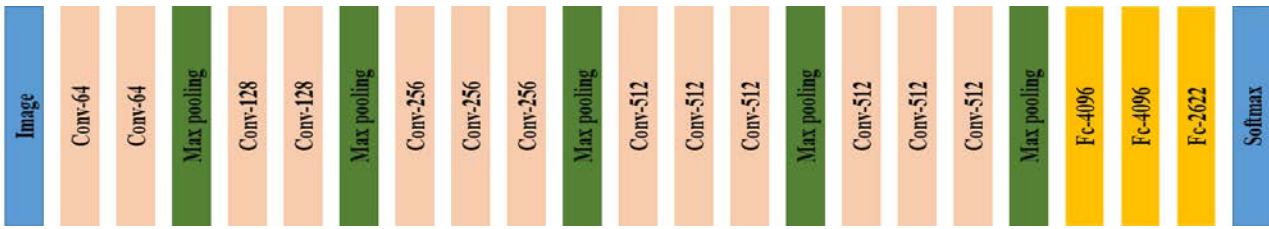


Figure 3: VGGFace deep network architecture

In the past, research on FR was conducted by using hand-crafted features such as HOG [1], local binary patterns [2], and Gabor [3] to extract useful features from images. However, based on recent advancements in deep learning, many face detection algorithms (e.g., AlexNet [4], ImageNet [5], You Only Look Once (YOLO) [6], Faster R-CNN [7], and RefineNet [8]), based on deep learning algorithms such as convolutional neural networks (CNNs) and FR algorithms have been developed. In this study, we aimed to develop an FR algorithm to support the development of smart access systems and access control management systems for hazardous areas. To this end, we compared the performances of conventional deep-learning-based face detection algorithms (i.e., dlib [9] and single-shot multibox detector (SSD) [10]) and FR algorithms (i.e., visual geometry group (VGG) [11] and ResNet [12]). Additionally, we aimed to implement an FR algorithm that achieves high prediction accuracy in various scenarios (e.g., target wearing a helmet, protective glasses, or both).

In Section 2, we describe the deep-learning-based FR structure and an FR algorithm based on DeepFace. We also review literature related to this study and describe techniques for improving algorithm performance. In Section 3, we propose a deep-learning-based FR algorithm that is suitable for industrial sites and examine the effects of normalization on facial recognition. In Section 4, we conclude this paper with suggestions for future research.

2. FR Algorithm

2.1 Deep-learning-based FR structure

In this section, we briefly describe associated algorithms before discussing the algorithms considered in this study. We also describe representative algorithm performance improvement techniques.

2.1.1. VGGFace

One structure that emerged following the emergence of DeepFace is the VGGFace [15] (or DeepFR) deep network structure proposed by the VGG at Oxford University. VGGFace utilizes

the VGG face dataset (a large dataset for FR created through Internet searches) and trains a deep network structure consisting of 15 convolutional layers (Figure 3). The VGG not only provided a VGGFace training model, but also trained it using a relatively simple 3×3 convolution filter, similar to the VGG structure used in ImageNet image recognition challenges. As a result, VGGFace achieved 98.95% accuracy on the LFW dataset, which is approximately 1% better than DeepFace. Additionally, various deep network structures have been proposed for FR, including DeepID [16], DeepID2 [17], DeepID2+ [18], and DeepID3 [19], to improve performance.

2.2 Theories related to DeepFace-based FR

2.2.1 Object detection algorithms

Here, we discuss an object detection algorithm based on a CNN. There are various algorithms ranging from regions with CNNs (R-CNN) [20], which applies a CNN to object detection, to SSD, which has recently exhibited high detection speed. We will briefly describe the CNN-based SSD algorithm and Dlib library used for object detection.

2.2.1.1 Dlib library

Dlib is a modern C++ toolkit that contains machine learning algorithms and tools for developing complex software in C++ to solve real-world problems [9].



Figure 4: Visualizing 68 facial landmark coordinates from the iBUG 300-W dataset

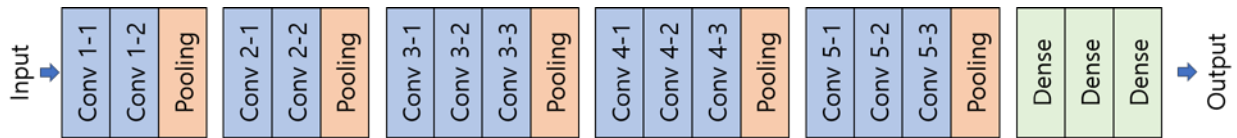


Figure 6: VGG network structure

A pre-trained face landmark detector in the dlib library [9] was used to estimate the position of 68 (x, y) coordinates that map to the structure of a face. Figure 4 presents the indexes of the 68 coordinates.

This annotation is part of the 68 point iBUG 300-W dataset on which the dlib face landmark predictor was trained. There are additional types of facial landmark detectors, including a 194 point detector, which can be trained on the HELEN dataset [21].

Regardless of the dataset used, one can train a shape predictor for input training data by using the dlib framework. This can be useful when one wishes to train a face landmark detector or custom shape predictor.

2.2.1.2 SSD

SSD [10] recognizes objects using feature maps of various sizes without separately training a region proposal network (RPN) to generate candidate regions (Figure 5). The feature maps obtained from a CNN model [22] are reduced in size as the convolutional layers progress, as shown in Figure 5. The SSD recognizes objects by using all feature maps extracted during this process for the inference process. Large feature maps extracted at shallow depths can detect small objects, whereas small feature maps extracted at deeper depths can detect large objects. SSD improves training speed (compared to the faster region-based CNN (Faster R-CNN)) [7] by eliminating the RPN and it can recognize objects more accurately than the YOLO model [6] by using feature maps of various sizes. Experiments on the PASCAL VOC 2007 dataset revealed that SSD achieved a mean average

precision (mAP) approximately 3% higher than that of the Faster R-CNN and achieved a faster detection speed than YOLO by processing 22 images per second.

2.2.2 Object detection

CNN models for object recognition have evolved into deeper structures since the AlexNet model [4] first implemented a deep CNN structure. In this section, we briefly describe the VGG [11] and ResNet [12] models applied to FR in this study.

2.2.2.1 VGG

The VGG [11] was proposed based on research on performance changes observed as a function of CNN layer depth. It uses the same setup for each model by performing integration five times and setting all filter sizes to three to equalize all conditions, except for the layer depth in the model structure. We conducted experiments using five models with depths ranging from 11 to 19. It was determined that the VGG model could serve as a larger filter if it was used repeatedly after setting the filter size to three. Many recent models have used this filter size. Although this model came in second at the ILSVRC-2014 competition, it is still attracting significant attention based on its excellent performance and simple structure, which is presented in Figure 5.

2.2.2.2 ResNet

CNNs for image object recognition have been improved by implementing deeper network structures. However, increasing the number of layers to hundreds or thousands leads to

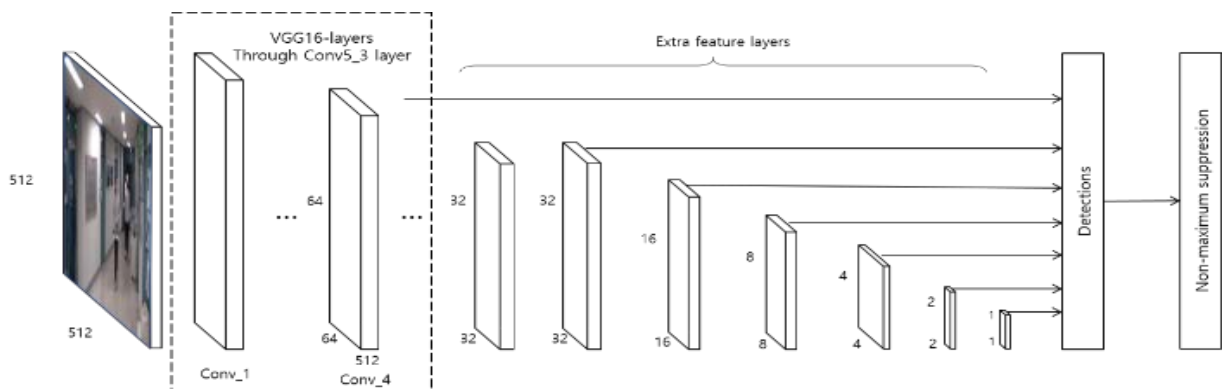


Figure 5: SSD object detection network architecture [10]

inaccuracy. ResNet [12] applied a method called residual learning [12] to solve this problem. In residual learning (Figure 7), a specific layer learns not only an output, but is also trained to respond sensitively to small changes by learning the differences between the inputs and outputs. Learning the differences between inputs and outputs is accomplished through addition only, which maintains computational efficiency because no additional parameters are required. ResNet is a model that applies the concept of residual learning to a 34-layer VGG model. Through our experiments, we confirmed that the accuracy decreased for the same VGG model when residual learning was not applied and the number of layers was increased from 18 to 34. The accuracy increased with an increase in the layer count of the VGG model (i.e., ResNet model) to which residual learning was applied.

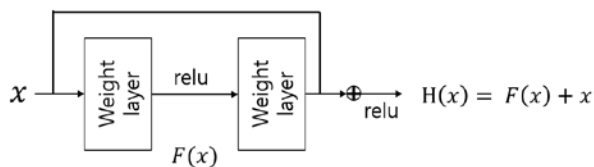


Figure 7: Residual Block

2.3 Performance comparisons of object recognition models

In Sections 2.2.1 and 2.2.2, we discussed object detection and object recognition algorithms based on CNN models. Each model and method have advantages and disadvantages in terms of accuracy and detection speed. Huang *et al.* compared these methods through experiments under the same conditions. We confirmed the experimental results based on training time and accuracy. We conducted experiments on typical object recognition methods, such as the Faster R-CNN, R-FCN, and SSD. The R-FCN and SSD required relatively little training time, but had low accuracy, whereas the Faster R-CNN required more training time, but had higher accuracy. The object recognition performance of different CNN models can be confirmed based on the data provided by Huang *et al.* ResNet-101 and Inception ResNet V2 achieved the highest accuracy and SSD exhibited little variation in accuracy according to the CNN model.

2.4 Public datasets for verifying FR technology

Various large datasets have been used to verify deep-learning-based FR learning and performance. Among them, only CelebFaces, DeepFace (Facebook), and FaceNet (Google) are suitable for deep network training with large datasets, but their application is generally limited because they are non-public datasets. In

contrast, the VGGFace dataset is a public dataset and its two versions (i.e., VGGFace and VGGFace2) are available for deep network training. Various challenges have been held to verify the performance of FR technologies using a large training dataset. Additionally, wild validation datasets including various changes have been released. Among them, the most widely used datasets are LFW, YouTube Face [23], IJB [24]-[26], and MegaFace [27][28]. In this section, we describe the LFW dataset used for algorithm verification.

2.5 Transfer learning

We implemented FR using deep learning algorithms by applying transfer learning, which utilizes pre-trained models. Transfer learning utilizes pre-trained models to solve unknown problems. It is utilized to solve problems that are similar to a pre-trained model when pre-trained models with good performance are available. Extensive resources are required to construct the large datasets that are required for training new models, as described in Section 2.3. Therefore, if pre-trained models with good learning performance are adopted, physical resources and the associated training time and cost can be saved.

Fine-tuning is a process that is required to perform transfer learning. Fine-tuning is performed based on the similarity between a pre-trained model and new dataset used for training, and it can be classified into three main approaches.

Figure 8 presents summaries to provide an understanding of these approaches. The first approach is employed when the size of the training dataset is extremely large, but there is little similarity between the training dataset and pre-trained model. The second approach involves training only a few layers of the model. This approach is employed when the size of the training dataset is extremely large and its similarity with the pre-trained model is high, and when the size of the training dataset is small and its similarity with the pre-trained model is low. The training process can be performed under any conditions in the former case. However, in the latter case, overfitting may occur based on the small size of the dataset if training is performed on several layers. Similarly, training may not be effective if only a few layers are trained. Therefore, the size of the dataset must be enhanced through data augmentation and one must train an appropriate number of layers. Finally, the third approach involves training only the final layer of the model, which is called the classifier layer. This approach is employed when the size of the training dataset is small, but its similarity with the pre-trained model is high.

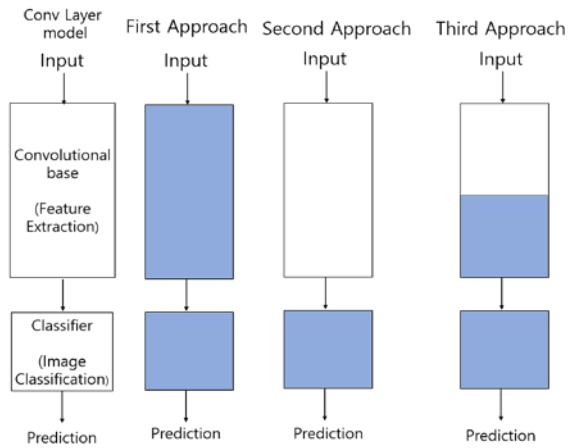


Figure 8: Diagrams of three approaches to transfer learning

3. Deep-learning-based FR Algorithm (DeepFace)

In this study, we analyzed the performance of two face detection algorithms (dlib and SSD-MobilenetV2) and two FR algorithms (VGG and ResNet). We generated four models (dlib-VGG, dlib-ResNet, SSD-VGG, and SSD-ResNet), which were trained on the “Face Detection Dataset and Benchmark” (FDDDB) dataset [29] for face extraction. We compared the performances of the face extraction algorithms on 1,000 images from the LFW dataset [14] (Section 3.2). By utilizing the algorithm with the best recognition rate, we attempted to perform FR in various scenarios (i.e., when subjects were wearing a helmet or protective glasses) to confirm its applicability to worker access control management systems in hazardous areas (Section 3.3).

Figure 9 presents the research method adopted in this study.

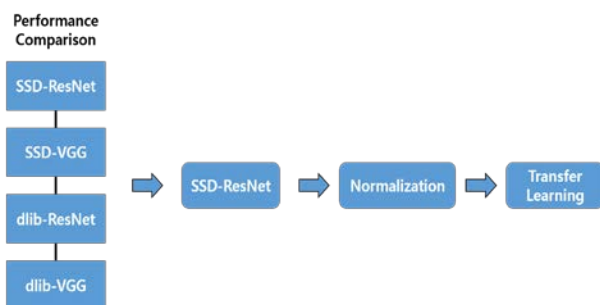


Figure 9: Research method

3.1 Results on benchmark datasets

3.1.1 Training datasets

The FDDDB of face regions was designed to study the problem of unconstrained face detection. This dataset contains annotations for 5,171 faces in a set of 2,845 images taken from the faces in the wild dataset [29].

3.1.2 Testing datasets

To analyze the recognition rates of the FR algorithms, we used public LFW data. The LFW dataset, which was released in 2009, contains 13,233 images of 5,749 celebrities from the internet. Compared to other FR datasets (e.g., FERET and MultiPIE) obtained by shooting in restricted environments, this dataset has been more widely used to verify the performance of FR technologies because it includes changes in lighting, facial expressions, and poses that appear in everyday life. Because the LFW dataset contains 2.31 images per person on average and there are no separate galleries or verification images, it is mainly used to verify the performance of face verification technologies, rather than face identification technologies. The accuracy achieved on the LFW dataset in recent FR research is approximately 99.73% [14], which is known to be saturated by high-quality images (Table 1).

Table 1: FR performances of various FR methods (%)

Dataset	Face Recognition Algorithms	Face Verification
LFW	DeepFace	97.4
	VGGFace	98.9
	SphereFace	99.4
	FaceNet	99.6
	CosFace	99.7

3.2 Performance comparisons of FR algorithms

Recall and precision must be considered simultaneously to evaluate the performance of face extraction algorithms. Recall indicates how well the object to be extracted is detected without being left out and precision indicates the accuracy of the detected results, meaning the number of actual target objects included in the detection results. We can define the precision and recall of the recognition algorithms as follows.

Precision refers to the ratio of correct detections among all detection results. Precision can be expressed by the following equation:

$$Precision = \frac{TP}{TP + FP} = \frac{TP}{all\ detection} \tag{1}$$

Here, TP means true positive and refers to “correct detections,” whereas FP means false positive and refers to “incorrect detections.” In other words, precision characterizes the ratio of correctly detected objects among all of the objects detected by an algorithm.

Recall is the ratio of correctly detected objects among all objects that should be detected. Recall can be expressed by the following equation:

$$Recall = \frac{TP}{TP+FN} = \frac{TP}{all\ groundtruths} \quad (2)$$

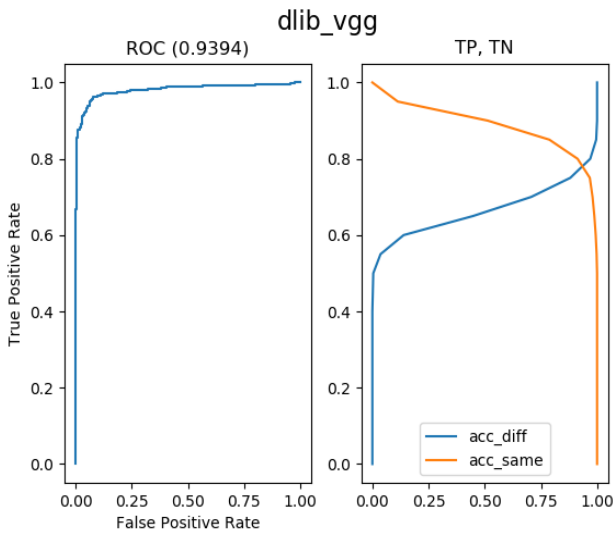


Figure 10: Results of the dlib-VGG model

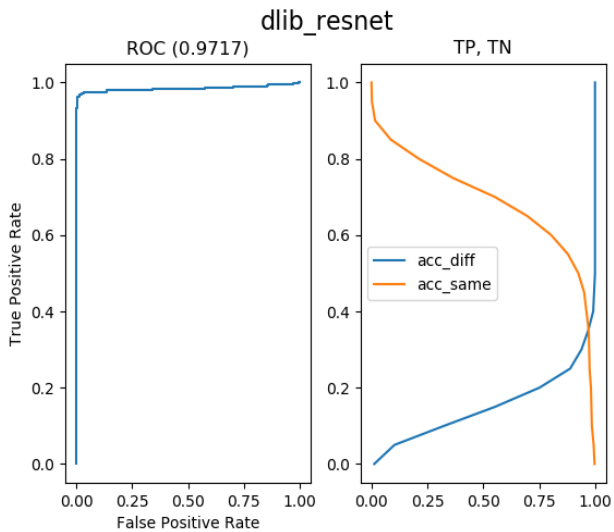


Figure 11: Results of the dlib-ResNet model

In general, regarding the performance of object recognition algorithms, it is not appropriate to express the overall performance of an algorithm as a single value because recall and precision are values that change dynamically depending on the parameter adjustment of the algorithm. Therefore, we use a precision-recall graph to analyze performance changes in terms of precision and recall to evaluate the performance of an object recognition

algorithm. However, although a precision-recall graph has the advantage of representing the overall performance of an algorithm, it is inconvenient to compare the performances of two different algorithms quantitatively.

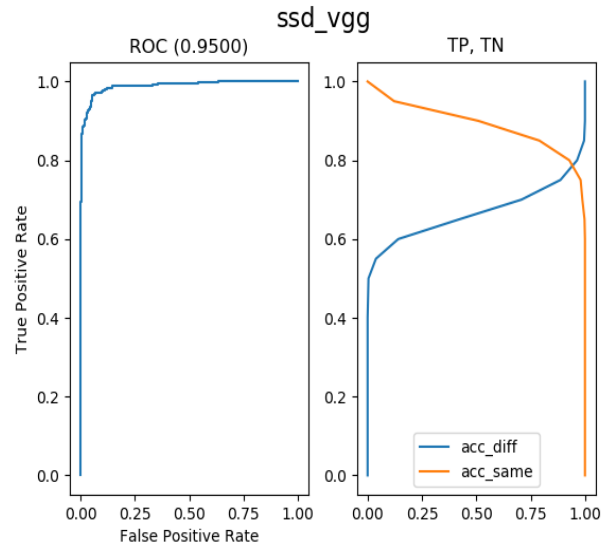


Figure 12: Results of the SSD-VGG model

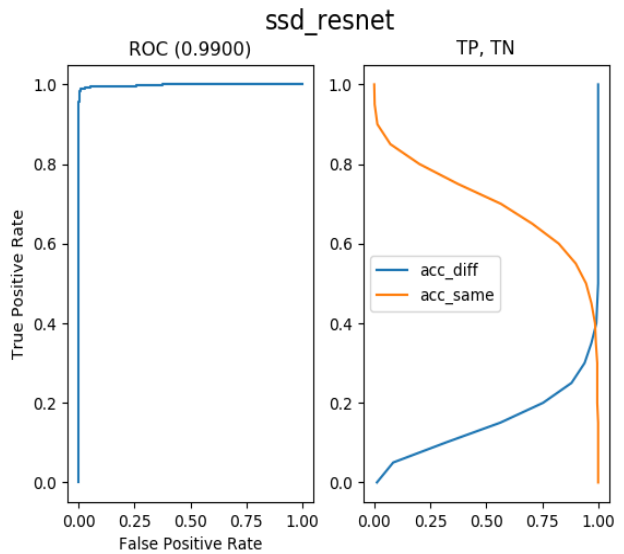


Figure 13: Results of the SSD-ResNet model

Therefore, we use the receiver operating characteristic (ROC) curve as a criterion for evaluating the performance of FR algorithms. The ROC curve is calculated as the area under the graph line in the precision-recall graph. A higher value of the area under the ROC curve indicates that an algorithm performs better. Figures 10 to 13 present the results of training under the same conditions for the quantitative performance comparison of FR algorithms. The closer a value is to one, the better the performance.

One can see results of 0.9394 for Dlib-VGG, 0.9717 for Dlib-ResNet, 0.9500 for SSD-VG, and 0.9900 for SSD-ResNet. We compare the results in **Figure 14** in one plot. SSD-ResNet performs the best with a value of 0.9900.

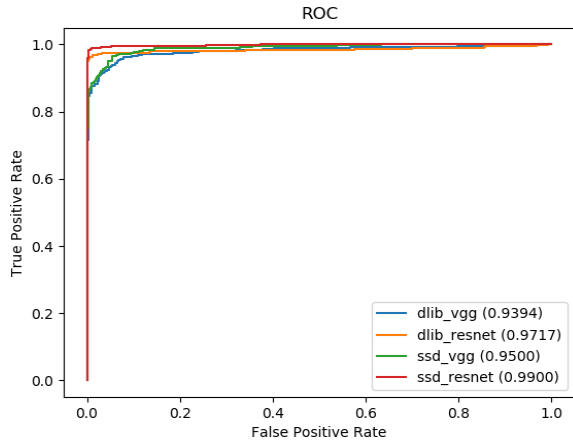


Figure 14: ROC curve (AP)

3.3 Application of DeepFace-based FR algorithms

We evaluated the accuracy of FR by selecting the SSD-ResNet model as an FR algorithm and then applying it to actual data (e.g., one-person recognition, multiple-people recognition). Here, by considering the working environment, we analyzed the

applicability of the selected in the field by analyzing its accuracy based on data captured in various scenarios (e.g., wearing a helmet, protective glasses, or both).

3.3.1 Analysis of the FR rate of DeepFace (SSD-ResNet model)

We captured three photographs of each individual as data for training the SSD-ResNet model (**Figure 13**). The best-performing SSD-ResNet algorithm was selected to perform FR. To verify the face extraction and FR rates of the SSD-ResNet model, we compared the FR rates in cases with only one person and cases with multiple people in a photograph (testing data, **Tables 3 and 4**).

The analysis of these results confirmed good performance for face extraction and FR with one person or multiple people (**Tables 3 and 4**). In each result (**Tables 3 and 4**), the left side of the rectangular box presents the accuracy of face extraction and the right side shows the FR rate. Here, the cosine similarity displayed on the right side of the rectangular box indicates the similarity of two vectors and can be obtained by calculating the cosine angle between the two vectors. If two vectors are exactly the same, then the value is one. If the angle between the two vectors is 90° , then the value is zero. If the vectors are opposite (angle of 180°), then the value is -1 . In other words, the cosine similarity has a value between -1 and 1 . As it approaches one, the similarity increases.

Table 2: Training data

Name	Pictures		
Bae J. S.			
Jang J.K.			



Table 3: Testing data (Individuals)



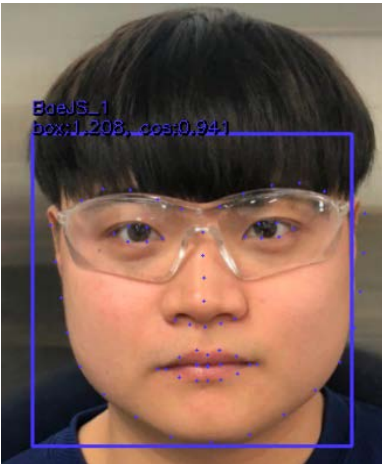

Image	Name (Cosine similarity, Improvement rate)	Image	Name (Cosine similarity, Improvement rate)
	Bae J. S. 0.832		Wearing Helmet Bae J. S. 0.724
Image	Name (Cosine similarity, Improvement rate)	Image	Name (Cosine similarity, Improvement rate)
	Wearing Protect Glasses Bae J. S. 0.941		Wearing Helmet + Protect Glasses Bae J. S. 0.870

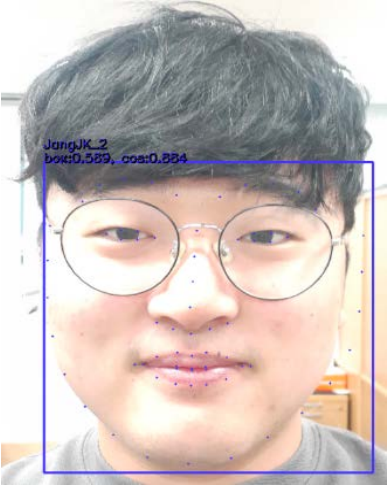
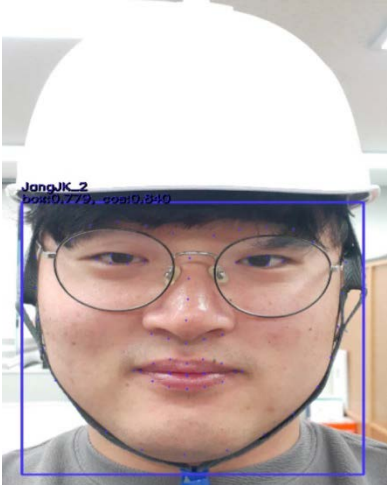




Image	Name (Cosine similarity, Improvement rate)	Image	Name (Cosine similarity, Improvement rate)
	Jang J. K. 0.884		Wearing Helmet Jang J. K. 0.840
	Wearing Protect Glasses Jang J. K. 0.794		Wearing Helmet + Protect Glasses Jang J. K. 0.811
	Kim K. K. 0.841		Wearing Helmet Kim K. K. 0.863

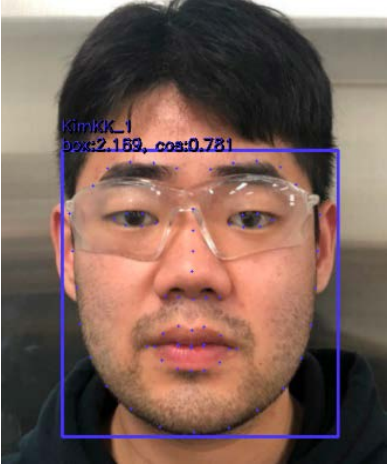
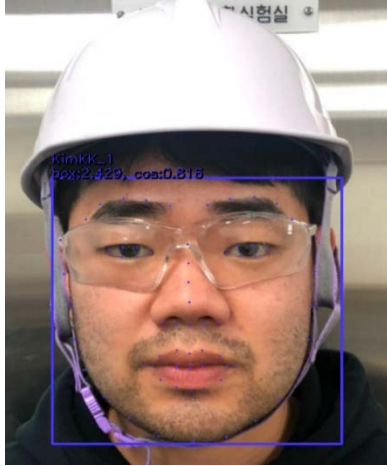
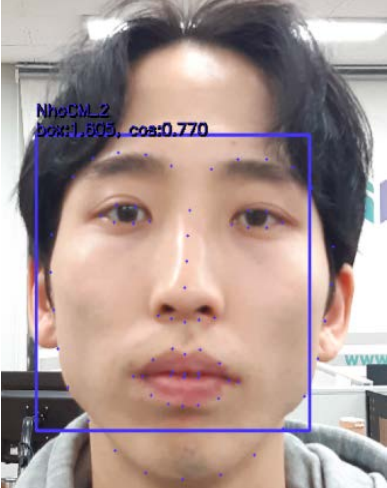
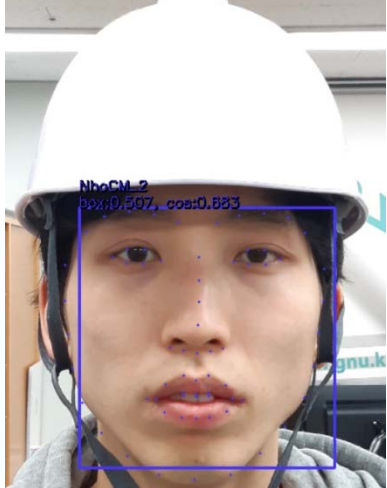
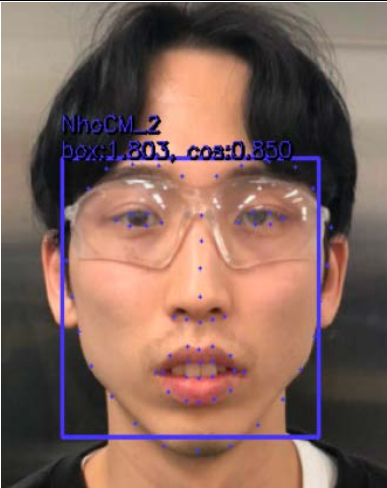

Image	Name (Cosine similarity, Improvement rate)	Image	Name (Cosine similarity, Improvement rate)
	Wearing Protect Glasses Kim K. K. 0.781		Wearing Helmet + Protect Glasses Kim K. K. 0.816
	Noh C. M. 0.770		Wearing Helmet Noh C. M. 0.683
	Wearing Protect Glasses Noh C. M. 0.850		Wearing Helmet + Protect Glasses Noh C. M. 0.778

Table 4: Testing data (Groups)

Name (Cosine Similarity)			
Group_1		Group_2	
Wearing helmet + Protect glasses		Wearing helmet	
Jang J. K. (0.792)		Kim K. K. (0.733)	
Wearing helmet		Wearing helmet + Protect glasses	
Bae J. S. (0.863)		Noh C. M. (0.725)	
Group_3			
Wearing helmet + Protect glasses		-	
Jang J.K. (0.756)		Kim K.K. (0.840)	
Wearing helmet		Wearing helmet	
		Bae J.S. (0.824)	

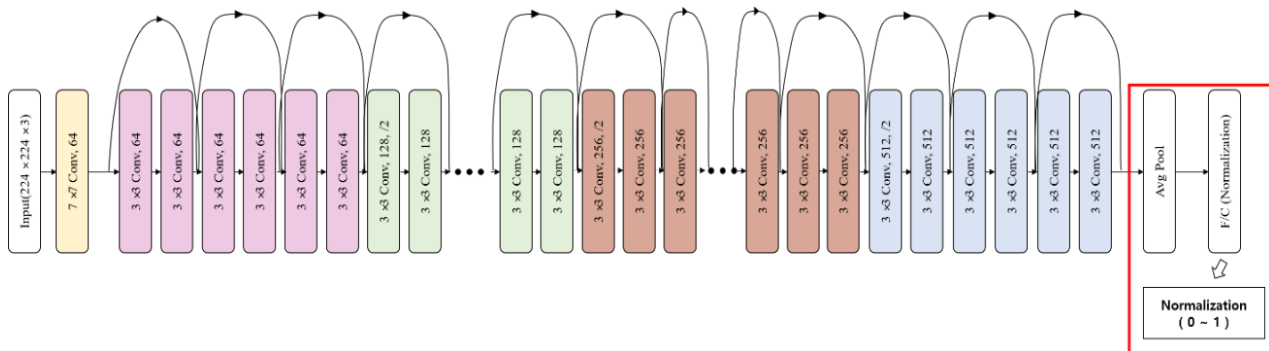


Figure 15: ResNet-SSD (Normalization) architecture

3.3.2 Improvement of the FR rate of DeepFace (SSD-ResNet model)

As shown in the results in **Tables 3** and **4**, the cosine similarity for recognizing faces is above the minimum value. These results may appear promising because we set the cosine similarity value to 0.5 for FR based on the training data. However, compared to the human eye recognition accuracy of 94.90% [30], the FR accuracy of the proposed algorithm is low.

For the safety of workers in confined spaces and hazardous

workplaces, the accuracy rate should be close to that of human eye recognition. Therefore, in this study, we analyzed the effects of normalization on the FR rate by applying normalization (**Figure 15**) to the fully connected layer, which is the last layer of the object recognition algorithm (ResNet) in the SSD-ResNet model. Here, normalization refers to normalizing feature values (MinMaxScaler function) to a range of zero to one as inputs for the fully connected layer.

Table 5: Changes in FR rate with normalization (Individuals)

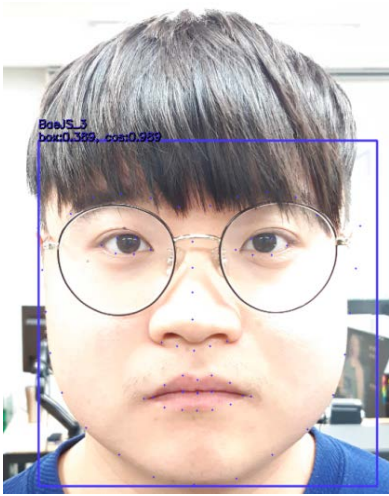



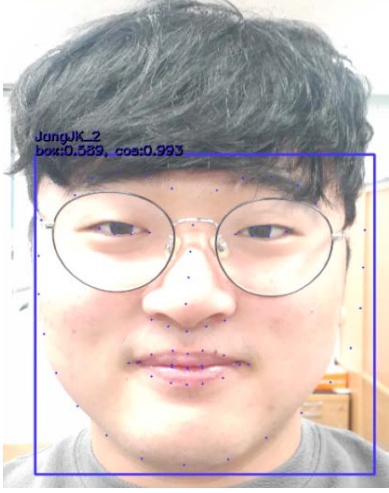

Image	Name (Cosine similarity, Improvement rate)	Image	Name (Cosine similarity, Improvement rate)
	Bae J. S. 0.989, 16%		Wearing Helmet Bae J. S. 0.981, 26%
	Wearing Protect Glasses Bae J. S. 0.996, 6%		Name (Cosine similarity, Improvement rate) Wearing Helmet + Protect Glasses Bae J. S. 0.992, 12%
	Name (Cosine similarity, Improvement rate) Jang J.K. 0.993, 11%		Name (Cosine similarity, Improvement rate) Wearing Helmet Jang J.K. 0.990, 15%



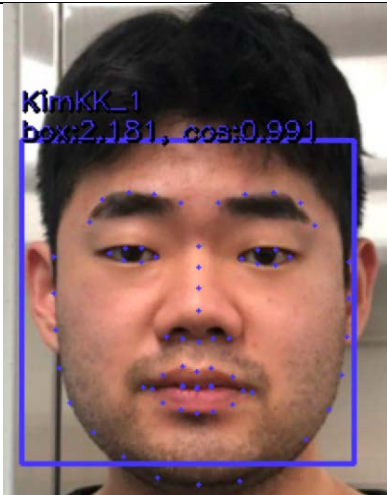

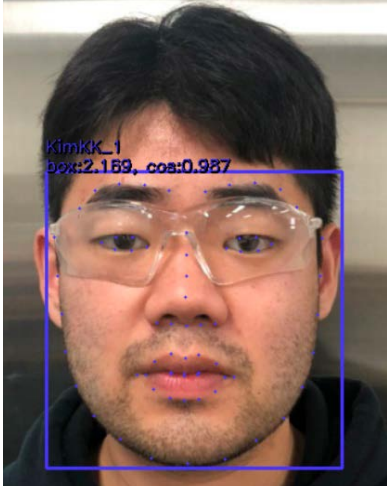

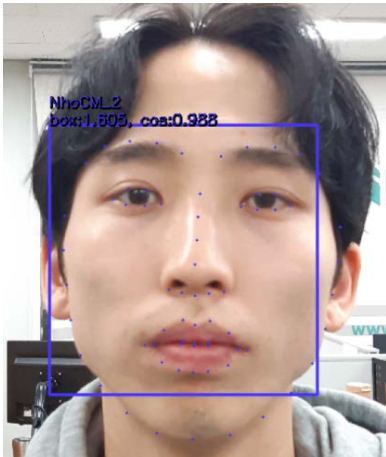

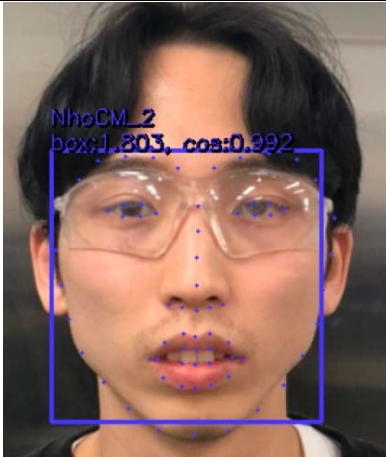

Image	Name (Cosine similarity, Improvement rate)	Image	Name (Cosine similarity, Improvement rate)
	Wearing Protect Glasses		Wearing Helmet + Protect Glasses
	Jang J.K. 0.987, 20%		Jang J.K. 0.987, 18%
Image	Name (Cosine similarity, Improvement rate)	Image	Name (Cosine similarity, Improvement rate)
	Kim K.K. 0.991, 15%		Wearing Helmet
			Kim K.K. 0.992, 13%
Image	Name (Cosine similarity, Improvement rate)	Image	Name (Cosine similarity, Improvement rate)
	Wearing Protect Glasses		Wearing Helmet + Protect Glasses
	Kim K.K. 0.987, 21%		Kim K.K. 0.990, 18%

Image	Name (Cosine similarity, Improvement rate)	Image	Name (Cosine similarity, Improvement rate)
	Noh C.M. 0.988, 22%		Wearing Helmet Noh C.M. 0.984, 31%
Image	Name (Cosine similarity, Improvement rate)	Image	Name (Cosine similarity, Improvement rate)
	Wearing Protect Glasses Noh C.M. 0.992, 14%		Wearing Helmet + Protect Glasses Noh C.M. 0.988, 21%

Interestingly, the results reveal that the FR rate is approximately 18% higher than that before applying the normalization process. This improvement in the FR rate was calculated using the following equation:

$$Improvement\ Rate = \frac{ABS(Before\ Normalize - After\ Normalize) \times 100}{After\ Normalize} \quad (3)$$



The average recognition rate of the algorithm applied with the normalization technique proposed in this paper is approximately 98%, which is higher than the human eye recognition accuracy of 94.90%.

3.4 Analysis and summary of results

The FR results obtained using the deep-learning-based FR algorithm are listed in **Tables 3** and **4**. It was confirmed that FR

using the SSD-ResNet model is sufficiently accurate (**Tables 3 and 4**). However, compared to the human eye recognition accuracy of 94.90%, the FR rate of the proposed algorithm was low. However, it was clear that the FR rate was improved by applying normalization to the FR algorithm (ResNet) (**Tables 5 and 6**). It was confirmed that the algorithm using the proposed normalization process has an excellent recognition rate that is competitive with human eye recognition accuracy. Additionally, before normalization, the SSD-ResNet model had an FR rate of approximately 68% to 94% (**Tables 3 and 4**), which was significantly affected by the quality of photographs. However, after normalization, the recognition rate of the algorithm ranged from 98% to 99% (**Tables 5 and 6**). The proposed normalization process can be used as one method to improve the accuracy of object detection. However, there is a drawback to the proposed algorithm. As shown in Table 8, the proposed algorithm often cannot recognize a face when the subject is wearing a face mask. To perform FR

Table 6: Changes in FR rate with normalization (Groups)

Name (Cosine Similarity, Improvement rate)			
Group_1		Group_2	
			
Wearing helmet + Protect glasses	Wearing helmet	Wearing helmet + Protect glasses	Wearing helmet
Jang J.K. 0.985, 20%	Bae J.S. 0.991, 13%	Kim K.K. 0.986, 26%	Noh C.M. 0.986, 26%
Group_3			
			
Wearing helmet + Protect glasses	-	Wearing helmet	
Jang J.K. 0.983, 23%	Kim K.K. 0.991, 15%	Bae J.S. 0.989, 17%	

in the manufacturing industry, there is a need for ongoing research to develop technology that can recognize faces in various scenarios.

3.5 Model retraining through fine tuning

As shown in **Table 7**, there were cases where subjects wearing a face mask could not be recognized by the proposed algorithm. To address this issue, we constructed an image dataset containing approximately 2,400 individuals wearing a face mask, as shown in **Table 8** [31]. Subsequently, we applied fine tuning to the face recognition algorithm described in Section 3.3 and trained the model using the constructed dataset.

The face recognition results obtained from fine tuning using the additional constructed dataset are presented in **Figure 16**. One can see that both the subjects wearing face masks and those not wearing face masks were accurately recognized. Furthermore, the results for those wearing face masks yielded a cosine

similarity of 0.995, indicating a high recognition accuracy with a low error rate, similar to the subjects not wearing masks.

Table 7: Areas for improvement in the proposed FR algorithm



	
The result of recognizing a person wearing a mask.	
	
The result of not recognizing the person wearing the mask.	

Table 8: Examples from the mask dataset**Figure 16:** Results of fine tuning

4. Conclusion

We developed an FR algorithm that is expected to play a key role in access control systems used for recognizing workers in confined and hazardous workplaces. We compared and analyzed the performances of conventional deep-learning-based face detection algorithms (dlib, SSD-Mobilenet V2) and FR algorithms (VGG, ResNet), and attempted to implement an FR algorithm with high prediction accuracy considering various scenarios (e.g., with subjects wearing a helmet, protective glasses, or both).

We selected the SSD-ResNet model (AP: 0.99) with the highest AP, which is a criterion for evaluating the performance of face recognition algorithms. The AP of the proposed algorithm was found to be in the range of 0.683–0.863. Compared to the recognition accuracy of human eyes (94.90%), we concluded that applying the proposed algorithm in real industrial sites would be insufficient. To resolve this issue, we attempted to increase recognition performance by applying normalized feature values (MinMaxScaler function) ranging from zero to one as inputs for the final layer (fully connected layer) of the face and object recognition algorithm (ResNet). As a result, we confirmed an average recognition rate increase of 18%. However, some limitations of the proposed algorithm were noted. For example,

subjects in images could not be recognized when they were wearing face masks.

To address this issue, we employed the transfer learning technique. We generated additional relevant data for training and fine tuning an existing pre-trained model for FR. Consequently, we successfully performed FR for subjects wearing face masks.

In the manufacturing industry, it is necessary to develop algorithms that can perform FR, regardless of whether a person is wearing a face mask or other safety equipment to protect themselves from the hazards posed by working environments. Accordingly, we plan to conduct additional studies on other FR algorithms that can be applied in different scenarios, as well as on intelligent image-recognition-based “Smart H.S.E.” (Health, Safety, Environment) systems.

Acknowledgement

This research was supported by Korea Institute for Advancement of Technology(KIAT) grant funded by the Korea Government(MOTIE) (N P0001968, The Competency Development Program for Industry Specialist).

Author Contributions

Conceptualization, J. C. Lee; Methodology, J. C. Lee and C. M. Noh; Software, S. B. Lee and C. M. Noh; Formal Analysis, J. C. Lee and C. M. Noh; Investigation, C. M. Noh and S. B. Lee; Data Curation C. M. Noh and S. B. Lee; Writing-Original Draft Preparation, C. M. Noh and S. B. Lee; Writing-Review & Editing, J. C. Lee; Visualization, S. B. Lee; Supervision, J. C. Lee; Project Administration, J. C. Lee; Funding Acquisition, J. C. Lee.

References

- [1] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” *International Conference on Computer Vision & Pattern Recognition (CVPR '05)*, vol. 1, pp. 886-893, 2005.
- [2] T. Ahonen, A. Hadid, and M. Pietikainen, “Face description with local binary patterns: Application to face recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 2037-2041, 2006.
- [3] M. Yang and L. Zhang, “Gabor feature based sparse representation for face recognition with Gabor occlusion dictionary,” *Proceeding of European Conference on Computer Vision*, pp. 448-461, 2010.

- [4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communication of the ACM*, vol. 60, no. 6, pp. 84-90, 2017.
- [5] J. Deng, W. Dong, R. Socher, L. -J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248-255, 2009.
- [6] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *Proceeding of the IEEE conference on computer vision and pattern recognition*, pp. 779-788, 2016.
- [7] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *Neural Information Processing System*, pp. 91-99, 2015. Available: <https://arxiv.org/abs/1506.01497>.
- [8] L. Guosheng, A. Milan, C. Shen, and I. Reid, "RefineNet: Multi-path refinement networks for high-resolution semantic segmentation," *Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR)*, pp. 1925-1934, 2017.
- [9] D. King, *Dlib C++ Library*, <https://dlib.net>, Accessed November 15, 2020.
- [10] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, *et al.*, "SSD: Single shot multibox detector," vol. 9905, pp. 21-37, 2015.
- [11] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *International Conference on Learning Representations*, pp. 1-14, 2014. <https://arxiv.org/abs/1409.1556>.
- [12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770-778, 2016.
- [13] Y. Taigman, M. Yang, M. Ranzato, and Wolf, "DeepFace: Closing the gap to human-level performance in face verification," *Proceeding of Conference on Computer Vision and Pattern Recognition*, pp. 1701-1708, 2014.
- [14] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller, *Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments*, 07-49, University of Massachusetts, United States of America, 2007.
- [15] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," *Proceedings of British Machine Vision Conference*, pp. 41.1-41.12, 2015.
- [16] Y. Sun, X. Wang, and X. Tang, "Deep learning face Representation from predicting 10,000 classes," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1891-1898, 2014.
- [17] Y. Sun, Y. Chen, X. Wang, and X. Tang, "Deep learning face representation by joint identification-verification," *Neural Information Processing System*, pp. 1988-1996, 2014. Available: <https://arxiv.org/abs/1406.4773>.
- [18] Y. Sun, X. Wang, and X. Tang, "Deeply learned face representations are sparse, selective, and robust," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2892-2900, 2015. arXiv:1412.1265, 2014.
- [19] Y. Sun, L. Ding, X. Wang, and X. Tang, "Deepid3: Face recognition with very deep neural networks," *Computing Research Repository*, arXiv preprint, 2015. Available: <https://arxiv.org/abs/1502.00873>.
- [20] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580-587, 2014.
- [21] V. Le, J. Brandt, Z. Lin, L. Bourdev, and T. S. Huang, "Interactive facial feature localization," *2012 European Conference on Computer Vision*, pp. 679-692, 2012.
- [22] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural computation*, vol. 1, no. 4, pp. 541-551, 1989.
- [23] L. Wolf, T. Hanssner, and I. Maoz, "Face Recognition in Unconstrained Videos with Matched Background Similarity," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 529-534, 2011.
- [24] B. F. Klare, *et al.*, "Pushing the frontiers of unconstrained face detection and recognition: IARPA Janus benchmark A," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1931-1939, 2015.
- [25] C. Whitelam, *et al.*, "IARPA Janus benchmark-B face dataset," *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 592-600, 2017.
- [26] B. Maxe, *et al.*, "IARPA Janus Benchmark-C: Face Dataset and Protocol," *International Conference on Biometrics*, pp. 158-165, 2018.
- [27] I. Kemelmacher-Shlizerman, *et al.*, "The MegaFace benchmark: 1 million faces for recognition at scale," *IEEE Conference on Computer Vision Pattern Recognition*, pp. 4873-4882, 2016.

- [28] A. Nech and I. Kemelmacher-Shlizerman, "Level playing field for million scale face recognition," 2017 IEEE Conference on Computer Vision and Pattern Recognition, pp. 3406-3415, 2017.
- [29] V. Jain and E. Learned-Miller, Fddb: A Benchmark for Face Detection in Unconstrained Settings, UM-CS-2010-009, Dept. of Computer Science, University of Massachusetts, United States of America, 2010.
- [30] S. H. Lee, "Recent Artificial Intelligence (AI) Development Trends and Future Evolution Directions," LG Economic Research Institute, vol. 10, no. 10, pp. 2, 2017.
- [31] W. Zhongyuan, W. Guangcheng, H. Baojin, X. Zhangyang, H. Qi, W. Hao, Yi Peng, K. Jiang, W. Nanxi, P. Yingjiao, P, *et al.*, "Masked face recognition dataset and application," arXiv preprint, 2020. Available: <https://arxiv.org/abs/2003.09093>.