processes

MDPI

Article

Face Verification Based on Deep Learning for Person Tracking in Hazardous Goods Factories

Xixian Huang [1], Xiongjun Zeng [1], Qingxiang Wu [1,2,*], Yu Lu [2], Xi Huang [1] and Hua Zheng [1]

[1] Key Laboratory of OptoElectronic Science and Technology for Medicine of Ministry of Education, College of Photonic and Electronic Engineering, Fujian Normal University, Fuzhou 350108, China; hxixian@163.com (X.H.); xjzeng1118@163.com (X.Z.); huangxi@fjnu.edu.cn (X.H.); hzheng@fjnu.edu.cn (H.Z.)
[2] Concord University College, Fujian Normal University, Fuzhou 350117, China; media@fjnu.edu.cn
* Correspondence: qxwu@fjnu.edu.cn

Abstract: Person tracking in hazardous goods factories can provide a significant improvement in security and safety. This article proposes a face verification model which can be used to record travel paths for staff or related persons in the factory. As face images are captured from the dynamic crowd at entrance–exit gates of workshops, face verification is challenged by polymorphic faces, poor illumination and changing of a person's pose. To adapt to this situation, a new face verification model is proposed, which is composed of two advanced deep learning neural network models. Firstly, MTCNN (Multi-Task Cascaded Convolutional Neural Network) is used to construct a face detector. Based on the SphereFace-20 network model, we have reconstructed a convolutional network architecture with the embedded Batch Normalization elements and the optimized network parameters. The new model, which is called the MDCNN, is used to extract efficient face features. A set of specific processing algorithms is used in the model to process polymorphic face images. The multi-view faces and various types of face images are used to train the models. The experimental results have demonstrated that the proposed model outperforms most existing methods on benchmark datasets such as the Labeled Faces in the Wild (LFW) and YouTube Face (YTF) datasets without multi-view (accuracy is 99.38% and 94.30%, respectively) and the CNBC/FERET datasets with multi-view (accuracy is 94.69%).

Keywords: hazardous goods factory; face verification; person tracking; deep learning

Citation: Huang, X.; Zeng, X.; Wu, Q.; Lu, Y.; Huang, X.; Zheng, H. Face Verification Based on Deep Learning for Person Tracking in Hazardous Goods Factories. *Processes* **2022**, *10*, 380. https://doi.org/10.3390/pr10020380

Academic Editors: Yo-Ping Huang, Yue-Shan Chang and Hung-Chi Chu

Received: 20 December 2021
Accepted: 10 February 2022
Published: 17 February 2022

1. Introduction

In a hazardous goods factory, the high density of numerous workshops and workers can easily lead to mass death, casualties and domino effects of any accidents [1]. The whole life cycle process of hazardous goods includes multiple links such as production, operation, storage, transportation, use and disposal. Therefore, enterprise safety production requires vigorous promotion of the intelligent management of personnel on the job and position traceability [2]. In existing factories, especially chemical plants, steel mills and other places with high risk coefficient, accidents are often accompanied by heavy casualties. To improve the safety of hazardous goods production, it is currently necessary to use an automated hazard management system that integrates the information management system, software system and technical means of data collection and transmission based on local computer networks [3]. A human's unsafe action or unsafe conditions are the direct causes of accidents, and so these must be focused on in safety management [4]. Therefore, it is very important to monitor and manage the staff's movements in such factories. However, the existing technology is not adequate enough to monitor the staff's movements in the factory, and the monitoring of the staff's actions is not accurate enough. In this paper, the features of the face are used to track individuals by means of a person tracking system with a set of gate terminal devices based on face verification, as shown in Figure 1. P_{in} denotes the face image features captured from the person who enters a workshop or factory. P_{out}

denotes the face image features captured from the person who leaves a workshop or factory. P_w denotes the face image features of all persons in a workshop. P_{w1}, P_{w2}, . . . , P_{wn}, and P_F change with time. For example, $P_{w1}(t + 1) = P_{w1}(t) + P_{in1}(t) - P_{out1}(t)$. P_{w1}, P_{w2}, . . . , P_{wn}, and P_F are reflected positions for all staff or persons in the factory. If P_{in} and P_{out} are recorded for each gate when a person goes through a gate, the person's travel path will be recorded.

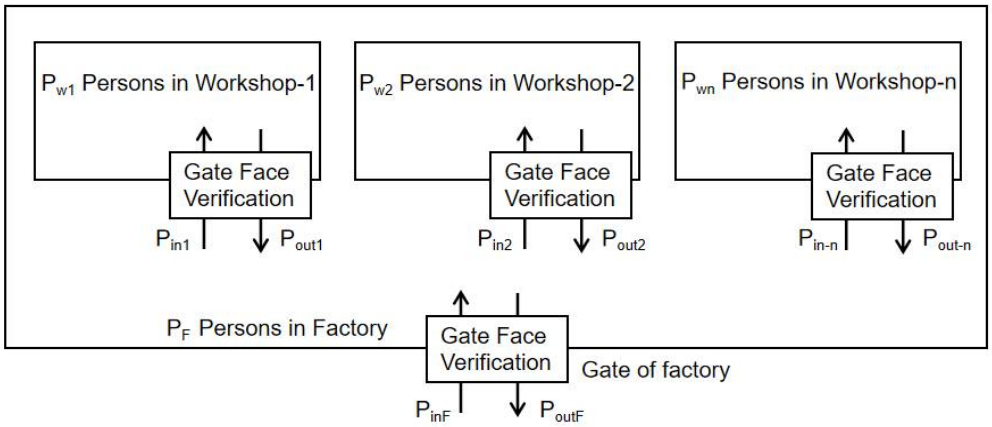

Figure 1. The person tracking system in a hazardous goods factory using gate face verification model.

In order to improve the safety of hazardous goods factories and standardize the behavior of staff and related individuals, the system mentioned above can be applied to personnel tracking and positioning. To implement this system, the key task is to design the gate face verification model for the terminal device at all the gates.

2. Materials and Methods

Since there are some problems in face verification, there are a few personnel tracking and positioning systems based on the face verification model. In this paper, we design a new face verification model using a set of deep learning models, MTCNN and MDCNN, to manage with face images and recognize faces so that the personal tracking system can capture the persons and record their travel paths in the factory.

Face recognition or verification has been one of the most active research areas for over two decades. With the continuous optimization of technology, a large number of face recognition models and algorithms have been proposed for various scenarios in the literature [5–14]. In this paper, we present a comprehensive architectural model called the gate face verification model to deal with polymorphic face inputs such as the frontal face, multi-view faces, dynamic faces, and multiple faces in one image.

Most existing face recognition models are based on front face images. There are a few models that can handle polymorphic face inputs. Currently, typical methods include the Kernel Grassmann Discriminant Analysis (KGDA) [15], based on spectral algorithm, local radon binary pattern (LRBP) for face sketch [16], and Pairwise Constrained Component Analysis (PCCA) for learning distance metrics from sparse pairwise similarity/dissimilarity constraints in high dimensional input space [17]. The results show that these algorithms cannot work well for face recognition in inputs of polymorphic faces due to the flexibility problems (see the results Section 4—Results of Experiments). As the emergence of neural network learning booms, researchers have put forward a great number of excellent algorithms of deep convolutional neural networks (DCNN) to optimize the metric learning [12,13] and construct network models and frameworks [18] to represent more effective features [13,16,19]. It has been shown that a DCNN model can not only extract high-level features by combining low-level features, but also learn a compact and discriminative feature representation when the size of the training data is sufficiently large.

Recently, the method of deep hypersphere embedding for face recognition (SphereFace) [20] has attracted increasing attention. In [20], in order to have smaller

maximal intra-class distance than the minimal inter-class distance under a suitably chosen metric space for face features, the angular softmax (A-Softmax), Additive Angular Margin Loss [6], and Angular Sparsemax [10] loss were proposed. Unlike other loss functions, such loss can enable DCNN to learn angularly discriminative features of face images, and they have achieved excellent performance on two real-world datasets: Labeled Faces in the Wild (LFW) [21] and YouTube Faces (YTF) [22]. DCNN can learn high-level feature representation more effectively as the neural network becomes deeper. However, some of the aforementioned face recognition algorithms are based on single frontal face (without multi-view). They are not good enough for the polymorphic faces captured from a dynamic crowd at gates.

In this work, we propose a new face verification model in which optimal convolutional layers are used in the convolutional neural network to adapt to multiple scenarios with polymorphic inputs such as real-photo, multi-view faces, dynamic faces and multiple faces in one image. One major challenge is high performance of face verification for so many different types of inputs. The key task of this paper is to find a solution against this challenge.

In the gate face verification model, as shown in Figure 2, two pipelines of processes are used. The first pipeline is to extract the face image from the person who enters the workshop/factory, in which the face image is extracted by means of the deep learning model MTCNN [23]. Then, a set of algorithms is processed to perform a normalization of face images. The second pipeline is to perform face feature extraction for face verification of the person who leaves the workshop/factory. The feature extraction is performed by the proposed MDCNN. The principle of the face verification is based on face similarity matching in the main facial features [13,19,20].

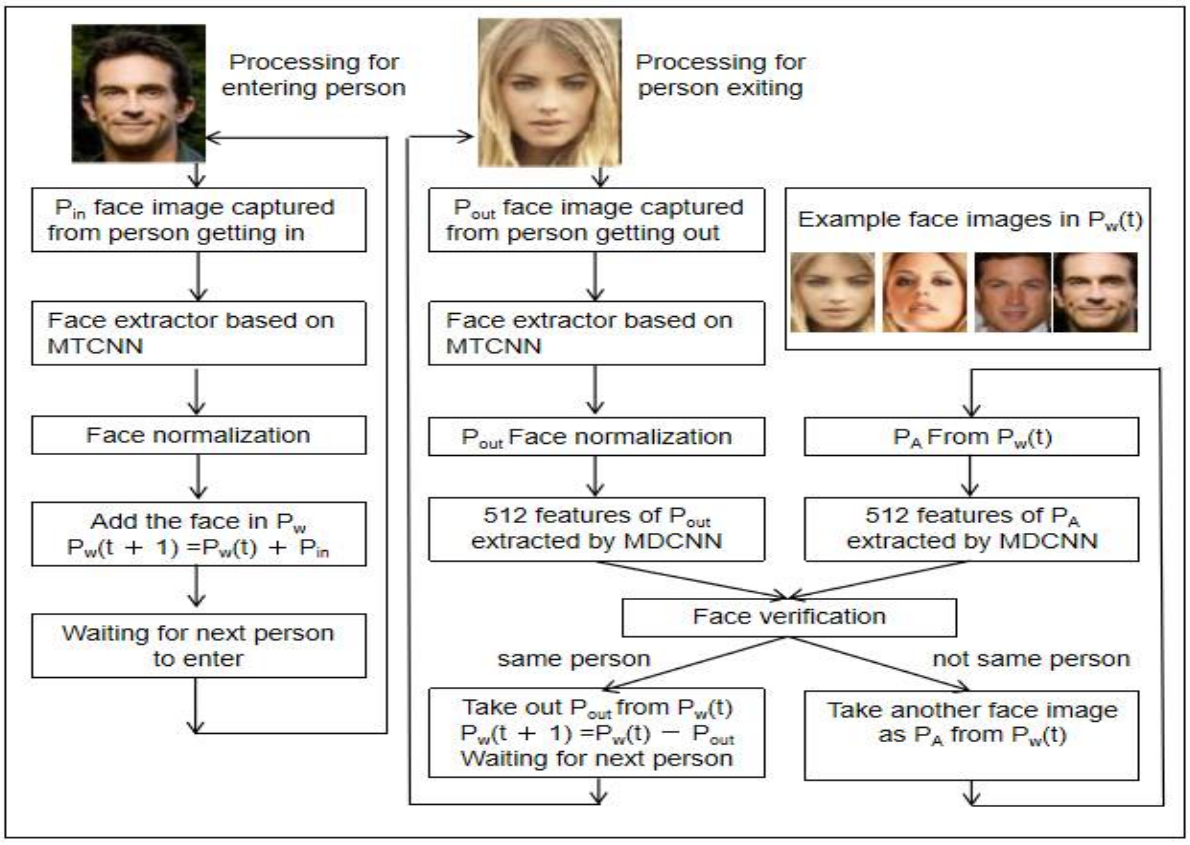

Figure 2. The gate face verification model. The MTCNN and the MDCNN are trained by different types of face images.

For multi-view faces, the following facts will be encountered in the face images: (1) The change in facial posture and angle will lead to the nonlinear change in two-dimensional shape and the change in local position information. (2) The face with multi-view has less face detail features and more interference factors. Thus, how to extract the main feature points in an angled facial view and learn the characteristics of angle discrimination for the model as much as possible is an essential step. For example, Wang et al. proposed the Kernel Grassmann Discriminant Analysis (KGDA) [15] based on a spectral algorithm. In order to solve the computational complexity of the spectral algorithm, Lin et al. proposed the multi-reproducing Kernel Hilbert space (Multi-RKHS) [14] via the analysis of the reproducing kernel, and Wu et al. [24] proposed the regression algorithm to refine vector representations. However, these algorithms are not sensitive enough for angular features and have no processing mechanism for lost details of the negative side in the lateral face. Based on SphereFace-20 [20], the model of DCNN is reconstructed and optimized to 27 layers, which is called the Modified SphereFace-20, and it is used to improve the performance. In the gate face verification model, the A-Softmax [20] and face normalization algorithms are used to enhance the proposed model in the angular separability of features and make up for lost details, respectively. Notice that the face extraction before normalization is implemented by the multi-task cascaded convolutional networks (MTCNN) [23].

For the face verification, there are some algorithms such as the Pairwise-constrained Multiple Metric Learning (PMML) [12] and Discriminative Deep Metric Learning (DDML) [25]. The experimental results show that it is difficult for them to achieve better performance due to incomplete feature extraction, lower learning efficiency, and insufficient generalization ability for current network models. For the characteristics of different types of facial features or multi-view face images, the deeper network is required. As the network model becomes deeper, the network becomes more and more difficult to train. Furthermore, training DCNN is complicated by the fact that the input distribution in each layer changes during the training process. If the above problems can be effectively solved, the network can be optimized to deal with multi-view faces. Inspired by internal covariate shift [26], Batch Normalization (BN) [11,26] is embedded between convolution and activation for each layer in the Modified SphereFace-20 so that stable data distribution can be achieved during the training process and the new network is designed (see the details in the Section 3).

Our major contributions in this paper are as follows:

(1) Based on SphereFace-20, we have reconstructed the architecture and embedded Batch Normalization for each layer to form a new network with 27 layers, which is called MDCNN, to solve the problems of feature extraction for polymorphic faces.

(2) The face normalization module is designed in the gate face verification model for the handling of different types of faces using a set of specific algorithms.

(3) A new gate face verification model, as shown in Figure 2, is designed with the advanced deep learning models and the proposed MDCNN in this paper, and the models are trained with different training datasets. It not only achieves competitive results on several datasets: Labeled Faces in the Wild (LFW) [27], YouTube Faces (YTF) [20], and other public databases, but also obtains better performance on our multi-faces dataset and in real scenarios.

3. Algorithms in the Gate Face Verification Model

In this model, face detection is first performed to localize the face region in the captured image using MTCNN [23], and the face image is normalized using a set of algorithms proposed in this paper. Then MDCNN is proposed to extract facial features. Finally, face verification is implemented to obtain the similarity between a pair of face images. The gate face verification model is illustrated in Figure 2. The details and the related algorithms are presented as follows.

3.1. Face Extractor Based on MTCNN

In this process, face images are detected with a CNN model called multi-task cascade convolutional neural networks (MTCNN) [23], which is trained with polymorphic faces by a transfer learning to better adopt different face images. The face extractor based on MTCNN consists of three networks, i.e., P-Net, R-Net, and O-Net, as illustrated in Figure 3. For each input image, it is resized to different scales to generate an image pyramid to obtain face images and non-maximum suppression which is used to achieve better face boxes. Through three networks, the face box with landmarks is obtained. For more details, please refer to [23].

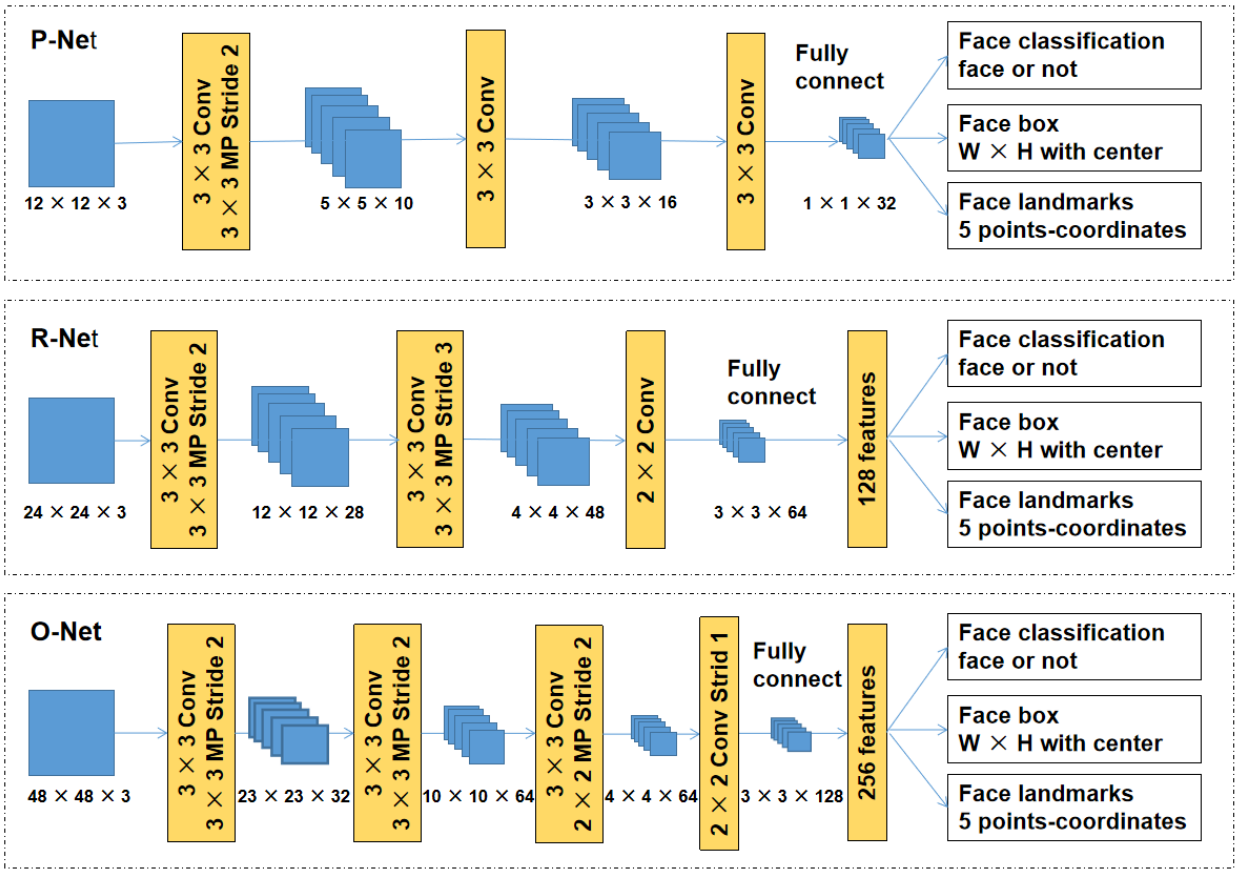

Figure 3. P-Net, R-Net, and O-Net in MTCNN.

In Figure 4, it is shown that the results are achieved by the face detector with the MTCNN algorithm. The bounding box regression, non-maximum suppression and real-time face alignment are used to refine face regions. It can be seen that the extracted face boxes are not the normalized faces with different sizes, directions, and lateral faces. Methods to normalize the faces are addressed in the next section.

3.2. Module of Face Image Normalization

The face detector can extract the face box and five points of face landmarks. In many practical applications, the extracted faces are different type faces such as multi-view faces, different emotion faces, dynamic pose faces, and various direction faces. In this module, a set of algorithms is used to reconstruct the face image to a normalized face image. The task can be regarded as a transformation from a different face image to a normal face image. The formal transformation M_A can be expressed as follows.

$$M_A = f_d(M) \tag{1}$$

where $f_d(.)$ is the transformation function and M is the input face image extracted by the face detector. This transformation can be implemented using a set of combinations of mathematical transformations such as rotation transformation, affine transformation, perspective transformation, etc. For simplicity and high speed of the performance, only the following two operations are used in this module.

(1) Normalization of face image size

Let M represent the face image cropped from the input image using the face detector. In the experiments of this paper, the face box is normalized to 112×96.

(2) Lateral face processing

Since lateral faces often appear in the captured images, the feature information of faces is not always symmetrically distributed. The feature information is very different from front face information and the lateral faces on the other side. Therefore, face images should be reconstructed to a formal face image. Based on the face symmetry studied, the lateral face image M performs a transformation to recover the feature information of the other half of a lateral face. The half-face information in the positive side is denoted by a matrix R, and R is reconstructed from a half of the lateral face according to the center landmark point of the M matrix. The half face at the opposite side is denoted by a matrix U, which can be reconstructed with the k-order transformation matrix A. That is, the characteristic information of the opposite half face is expressed as follows:

$$U = R \cdot A \tag{2}$$

where A is a $k \times k$ matrix as follows:

$$A = \begin{bmatrix} 0 & 0 & \dots & 0 & 1 \\ 0 & 0 & \dots & 1 & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 1 & \dots & 0 & 0 \\ 1 & 0 & \dots & 0 & 0 \end{bmatrix} \tag{3}$$

where k is the size of the normalized face M. The normalized face image can be obtained with the fusion of R and U as follows:

$$M_A = R \circledcirc U \tag{4}$$

where $\circledcirc$ is the fusion operation for merging two half face matrices by means of a face topological model. The example result is shown in Figure 5.

Figure 4. Sample experiment results using MTCNN.

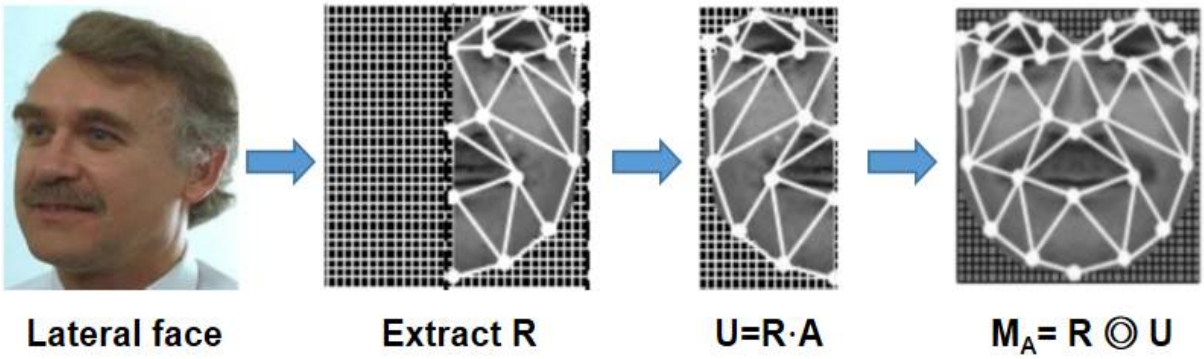

Figure 5. Fusion of two half faces.

3.3. The Proposed MDCNN for Feature Extraction

In the gate face verification model, the MDCNN model is used to extract efficient features from normalized face images. The SphereFace-20 [17] can be used to conduct this task, but it is not good enough due to unstable distribution of data transferred between the convolutional layers. Our studies have found that Batch Normalization is embedded in each of the convolutional layers; the data distribution can be improved so that better results can be achieved. Therefore, the structure with Batch Normalization as shown in Figure 6a is embedded in each layer between the convolution operation and activation operation.

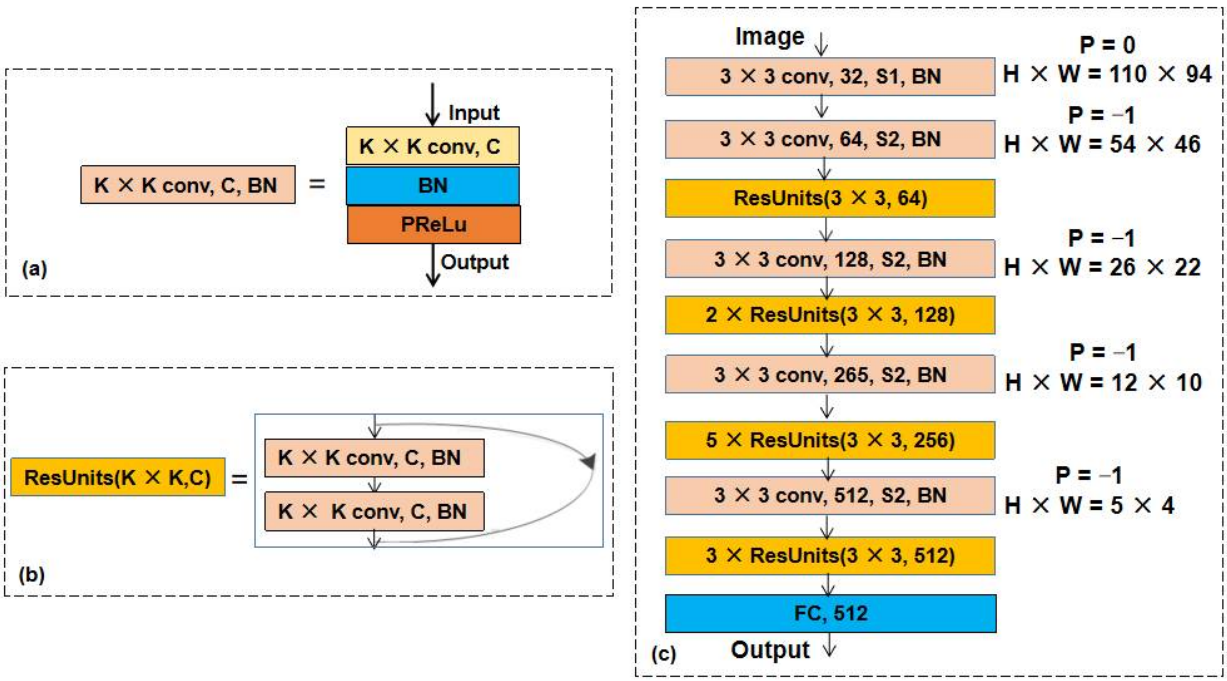

Figure 6. MDCNN for extraction of face features. (**a**) Convolutional layer with Batch Normalization. (**b**) Structure of ResUnits. (**c**) Over all architecture of MDCNN. Where "K" means kernel size, "C" means channels, "H" means the height of image, "W" means the width of image, and "P" means padding. "3 × 3 conv, 64" denotes the convolution layers with 64 filters of size 3 × 3, S2 denotes stride 2, and "FC" is fully connected layer.

ResUnits, as shown in Figure 6b, has also been designed to construct the new network architecture in Figure 6c, which is called MDCNN (Modified SphereFace-20 with the Batch Normalization). Compared with SphereFace-20 [20], the MDCNN has been extended by seven more layers to extract the more detailed features. In this model, the angular softmax loss is used so that the MDCNN can extract more discriminative features. After

the normalized face M_A is obtained and has been normalized, the face image M_A is fed to the MDCNN model for feature extraction.

Facial feature extraction is critical for face recognition, especially for multi-view faces, and it becomes a difficult problem due to the angular deflection view and loss of details. Since the new model has more layers to retain details of features, Batch Normalization is used to deal with data distribution, and with the lateral faces reconstructed, the MDCNN can improve results for different face images.

In the MDCNN, the main task of BN is to transform data to those with zero mean and unit variance for each layer and optimize model parameters to represent the data. Assume that the expectation and variance of the data after convolution are $E[x]$ and $Var[x]$, respectively. In order to normalize to zero mean and unit variance, the universal measure is expressed as follows:

$$\hat{x} = (x - E[x])/\sqrt{Var[x]} \tag{5}$$

where $\hat{x}$ is output data after normalization. However, this method has an extreme influence on the nonlinear functions; it constrains activation functions to the linear regime of the nonlinearity [26]. When BN [11,26] is adopted after each convolution and before activation, a pair of learning parameters γ and β are introduced into the network. In the process of training, the original distribution of data is recovered using the following expressions:

$$S = BN_{\gamma,\beta}(x) = \gamma\hat{x} + \beta \tag{6}$$

$$\gamma = \sqrt{Var[x]} \tag{7}$$

$$\beta = E[x] \tag{8}$$

where S is output data after joining BN. Thus, the distribution of characteristics is learned and it is easier for the nonlinear functions to stay in their non-saturated regimes. After that, the input of each layer has a stable distribution.

Therefore, the output of each complete convolutional layer (as shown in Figure 6a) can be formulated as:

$$S_j = BN\gamma,\ \beta(x_j) = \gamma\hat{x}_j + \beta \tag{9}$$

$$y_j = \psi(S_j)(j = 1,\ 2,\ \ldots,\ 27) \tag{10}$$

where S_j is the output of the BN layer j, $\psi(\bullet)$ refers to the activation function of PReLu [28] and is the output of complete convolutional layer j.

In a fully connected layer, all feature maps are feature-fused to generate a 512-dimensional nonlinear vector y_j of facial features, which is extracted from the fusion of the face feature from the normalized image M_A which has considered the lateral face features in R and U.

3.4. Face Verification or Recognition

In this module, the main task is to use metric learning to discriminate the similarity or learn a distance function. Most recent methods are based on Mahalanobis distance learning [25] and Euclidean distance [29]. The former learns a matrix for a distance metric, but it exaggerates the role of small variables. The latter is the distance between two points in N-dimension Euclidean space, however, it is not compatible with A-Softmax loss. In the gate face verification model, metric learning that has ability to learn angular distribution to calculate similarity is required. For these issues, the cosine distance can provide reliable metric learning for classification. Strictly speaking, the cosine distance is not a distance but a similarity, which is based on the vector direction to determine the vector similarity and related to the relative size of each dimension of the vector.

In the gate face verification model, the metric learning of cosine distance is employed to obtain the similarity for a pair of faces. The module, shown in Figure 7, is called face verification, and it can be computed as follows:

$$d(\text{similarity}) = \frac{\sum_{v=1}^{K} \partial_v \eta_v}{\sqrt{\sum_{v=1}^{K} \partial_v{}^2} \sqrt{\sum_{v=1}^{K} \eta_v{}^2}} \tag{11}$$

where K = 512, ∂_v is a 512-dimensional facial feature vector from the face images of persons in a workshop, and η_v is a 512-dimensional facial feature vector of the test face image from persons who leave the workshop.

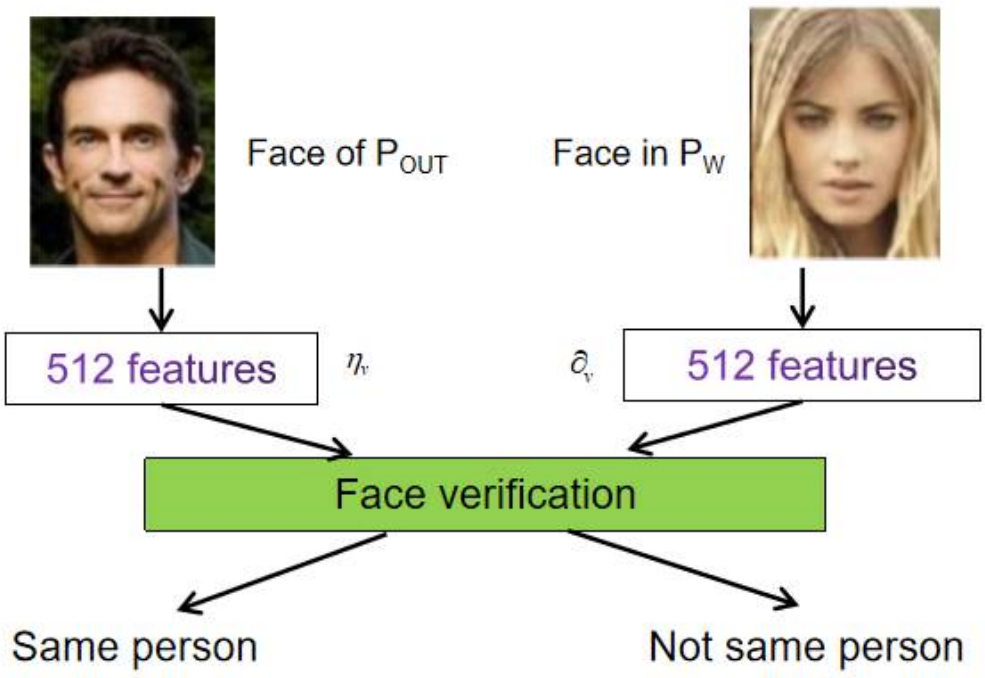

Figure 7. Module of face verification.

3.5. Training of the Model

In gate face verification model, the training is performed separately in different modules. In the module of face verification, if metric learning methods are used, the corresponding training has been performed. In this paper, the cosine distance is used to verify faces so that only a threshold T has been determined in the training process. According to the training data, the threshold T has determined. The similarity is calculated using cosine distance. The similarity of two faces is compared with the threshold T to determine whether they belong to the same person or not.

3.5.1. Training of MTCNN

The MTCNN is trained in three stages. Firstly, a labeled face image training set with two classes (face/nonface) is used to train the P-net to obtain coarse face boxes and landmarks. Secondly, based on coarse face boxes, the R-net is trained to obtain more refined face boxes and landmarks. Finally, based on refined face boxes, the O-net is trained to achieve accurate face boxes and five points of landmarks. The trained MTCNN is used for extraction of face boxes and face landmarks from input face images.

3.5.2. Training of MDCNN

The MDCNN is trained using the training set labeled with the face box and person identification class instead of only labels for face/nonface. The face boxes are extracted by the trained MTCNN, and the extracted face matrix is used to train the MDCNN. In order to enhance discernibility, the loss related to verification can be considered in the loss function in the training process.

4. Results of Experiments

4.1. Details of Training Process

In order to adapt to different face images, about one million face images are used to train MTCNN and MDCNN models. In the experiment, the training dataset contains multi-view face images from the internet and the publicly available databases CASIA-WebFace. In order to adapt to multi-view faces, the face images are horizontally flipped for data augmentation to extract multi-view features. Notice that the residual units [30] are also used in the architecture of the MDCNN model. The model starts with a base learning rate of 0.1 and is trained on a server of Model SerpurMicro E5-2678V3 in parallel on two GPU cards (Model: Nvidia GTX 1080Ti made by ASUS in Shenzhen, China) with the batch size of 128. In the process of training, the mini-batch Stochastic Gradient Descent (SGD) [31] and the activation function of PReLu [32] are employed to improve convergence. The training is finished at 30 K iterations. The experimental results have demonstrated that the training speed increases by two times after combining with BN. In the process of testing, the threshold of face similarity T is set to 0.70.

4.2. Evaluation for Face Recognition with Multi-View

In this part, the dataset contains 3000 pairs of face images with multi-view from CNBC and FERET databases, while CNBC contains 1000 pairs and FERET contains 2000 pairs. The angle of frontal faces is defined as 0°. Each face view group contains 750 pairs of faces, respectively, in frontal, 15°, 30°, and 45°. Some samples are shown in Figure 8.

Figure 8. Some samples with multi-view.

To evaluate the performance of the gate face verification model for face detection (FD) and face verification (FV) with multi-view, the gate face verification model is compared with the existing methods [14,15,22,24] and the results are illustrated in Table 1, which shows that our performance outperforms most of the existing algorithms and achieves 98.13% and 94.69% for face detection and verification, respectively.

Table 1. Comparison of our method and other methods with multi-view.

Method	Dataset	FD (%)	FV (%)
KGDA [15]	FERET	58	N/A
Multi-RKHS [14]	FERET	77	N/A
Wu et al. [24]	Private	90.10	N/A
MTCNN [23]	AFLW	93.10	N/A
Ours	CNBC	96.62	90.58
Ours	FERET	98.91	96.78
Ours	CNBC + FERET	98.13	94.69

Since many current methods encounter problems for lateral faces in the face verification in factory gates, we have enhanced the proposed model in dealing with lateral faces so that the model is working well in our prototype system. More details are shown in Section 4.5. As shown in Table 1, our model is also superior to other related methods.

4.3. Evaluation for Face Recognition Benchmark Dataset without Multi-View

4.3.1. Performance on LFW Dataset

The LFW dataset [21] contains 13,233 faces with small angle deflection, multi-expression, different illumination conditions, and different types of skin. For the sake of following the standard evaluation protocol, 6000 pairs of faces are randomly taken out from LFW, which contains 3000 pairs of positive faces, 3000 pairs of negative faces, and is divided into 10 groups. The final results are obtained from ten-fold cross validation.

To evaluate the performance of the proposed method on the LFW dataset, the gate face verification model is compared against most excellent methods [12–14,17–20,25]. The results are given in Table 2, which shows that the proposed algorithm outperforms most of the existing algorithms for face recognition and maintains a high-level accuracy. It is noticed that the accuracy of the MDCNN is slightly lower than FaceNet [19] and SphereFace-64 [20], the reason being that the former is trained with more than 20 million data and while the latter is 64-layers larger than the gate face verification model, it uses more memory and more complicated architecture than the proposed model.

Table 2. Comparison of our method and other methods on the LFW.

Method	Models	Accuracy (%)
PMML [12]	1	76.40
STRFD + PMML [13]	1	89.35
DDML [25]	1	87.83
DDML (combined) [25]	6	90.68
DeepFace [13]	3	97.35
DeepID2+ [18]	1	98.70
PCCA (SIFT) [17]	1	83.80
SphereFace-20 [20]	1	99.26
SphereFace-36 [20]	1	99.35
SphereFace-64 [20]	1	99.42
ArcFace (5.1 M) [6]	1	99.83
AMsoftmax [7]	1	99.17
CosFace (5 M) [8]	1	99.73
Ours (1 M)	1	99.38

4.3.2. Performance on YTF Dataset

The YTF [19] dataset consists of 3425 videos of more than 1500 different identities. In experiments, 5000 pairs of face videos are chosen from YTF and divided into 10 groups (each group contains 250 pairs of positive samples and 250 pairs of negative samples) to keep the standard evaluation protocol. The method of evaluating results is similar to LFW.

For video-based face recognition, the gate face verification model achieves 94.30% accuracy on the YTF dataset and maintains a high running speed. The results of the proposed model compared with most excellent methods [17–20,25] are shown in Table 3.

Table 3. Comparison of our method and other methods on the YTF.

Method	Models	Accuracy (%)
LBP + DDML [22]	1	81.26
STRFD + PMML [13]	1	79.48
DDML [25]	1	82.30
DDML (combined) [25]	6	82.34
DeepFace [13]	1	91.40
DeepID2+ [18]	1	93.20
SphereFace-20 [20]	1	94.10
SphereFace-36 [20]	1	94.30
SphereFace-64 [20]	1	95.00
ArcFace (5.1 M) [6]	1	98.02
CosFace (5 M) [7]	1	97.60
Ours (1 M)	1	94.30

As shown in Tables 2 and 3, SphereFace-64, ArcFace, and CosFace display a slight improvement over our model, but the proposed MDCNN has a small network scale of 27 layers, instead of that of SphereFace-64 which is on a large network scale of 64 layers, while ArcFace and CosFace are on a large network scale of 50 or 100 layers. The proposed model is superior to these in efficient implementation of face verification with an economic and small device.

4.4. Evaluation for Face Recognition with Multi-Face

Aiming at the performance of multi-face recognition in one image for the gate face verification model, we built a multi-face dataset, which contains 500 face images from 25 individuals with different poses, expressions, illuminations, and distances. The images were selected from several public image datasets (i.e., LFW, FERET, CelebA). Each image was labeled by us manually. The images were divided into four groups with 125 images, which contain one face, two faces, three faces, and four faces, respectively. If the maximum similarity is greater than the threshold of face similarity (0.70), this will demonstrate that the pair of faces belongs to the same identity. Table 4 shows that the gate face verification model achieves 95.60% accuracy and maintains high speed for multi-face recognition.

Table 4. Recognition details for multi-face in one image.

Face (s)	Number	Recognition	Accuracy (%)	Time (ms)
1	125	123	98.40	83
2	125	118	94.40	92
3	125	118	94.40	101
4	125	119	95.20	99
Total	500	481	95.60	

4.5. Prototype System and Test of Real Scenario at Factory Gates

In order to apply the proposed model in a factory for tracking a person, we have designed a prototype system with gate face verification devices, as shown in Figure 9. The gate face verification device is shown in Figure 9a, which is composed of the simplified PC system with Intel Core i7 CPU, NVIDIA GTX1050 card, Gigabit Ethernet port, WIFI, and 4G/5G modules, (As the proposed model has less scale network, it is possible to implement in an ARM system to form a smaller and more economic device, or it can be implemented in 5G edge computing equipment. These are in a plan for our further study). Two cameras are connected to the device. One is used to capture the face images at the entrance and

the other is used to capture face images at the exit. The face verification devices can be installed at the gates of the factory/workshop which can be connected to a network through Ethernet cable, WiFi, or 4G/5G mobile network. The architecture of the network is shown in Figure 9b. The server in the network can be used to manage the related information, which can include factory management information.

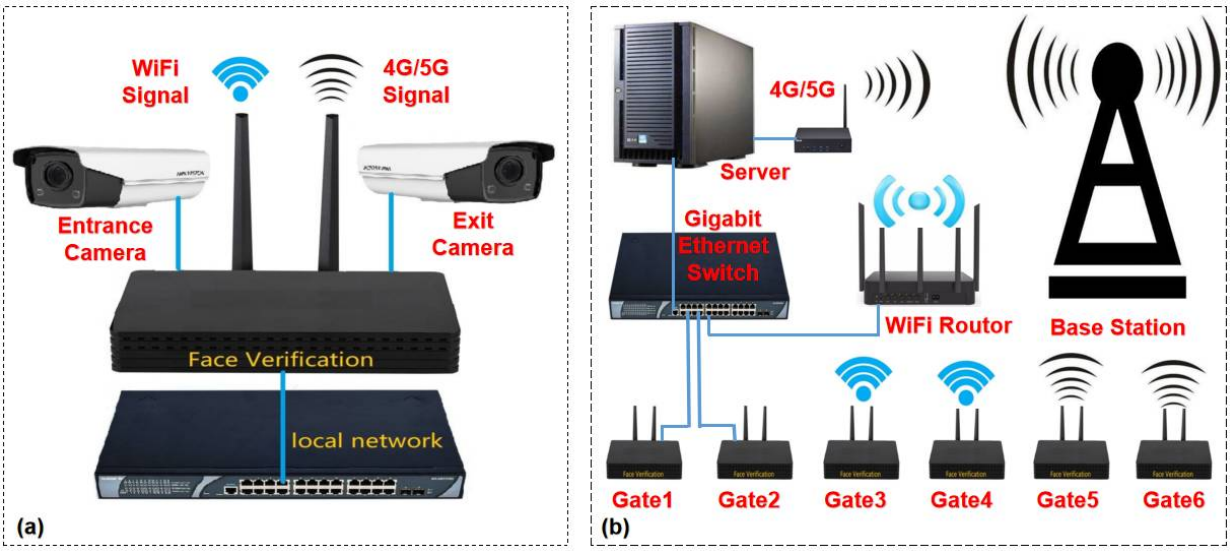

Figure 9. Personal tracking system with face verification devices. (**a**) Face verification device. (**b**) Network architecture of the prototype system.

In order to test the proposed model in a real scenario at factory gates, we have designed a test interface for gate face verification devices, which is used to capture face images from the factory entrance and the exit. The interface is shown in Figure 10. When a person enters the factory, the face can be captured as shown in the left box of the interface. It should be noted that the device/system has not stored the picture/image of the person. The system has only stored the extracted features, which consist of a vector with 512 values represented for example P_{in} in Section 1. The values cannot be reversed to image/picture, so the system does not store private personal images and does not result in any ethical issues. As the features are captured in the entrance and updated every day, the system is very robust against some changes to a person's face. In order to test the proposed model against the lateral face and multi-faces in one image, the camera was installed far away from the exit gate, and the device still works very well. An example is shown in Figure 10. The image captured at the factory exit shows that the person is far away from the camera and in lateral face in a multi-face image. The person can be correctly verified with similarity 0.91. The device can detect multiple faces, as shown in Figure 11; the maximal face in an image is selected for verification in the device. Figure 12 shows how a very similar person can be identified correctly.

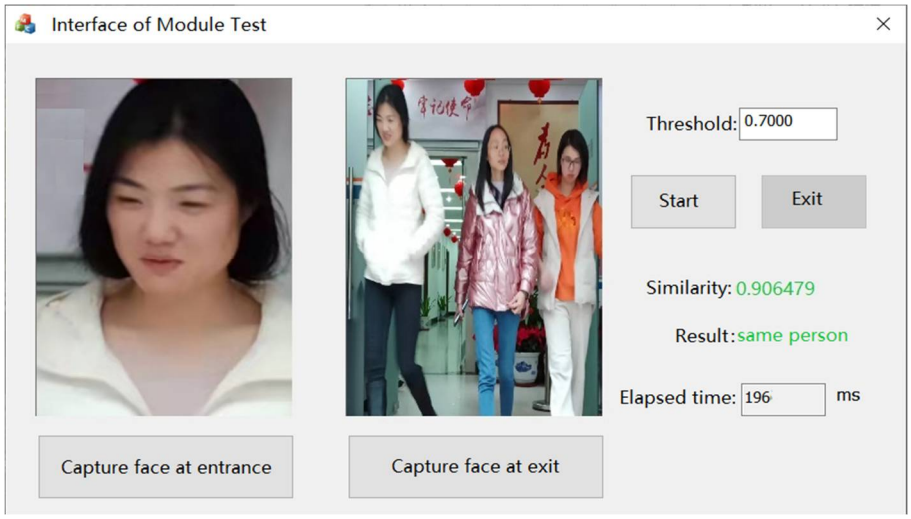

Figure 10. The same person is captured at the entrance and the exit.

Figure 11. Multiple faces are extracted.

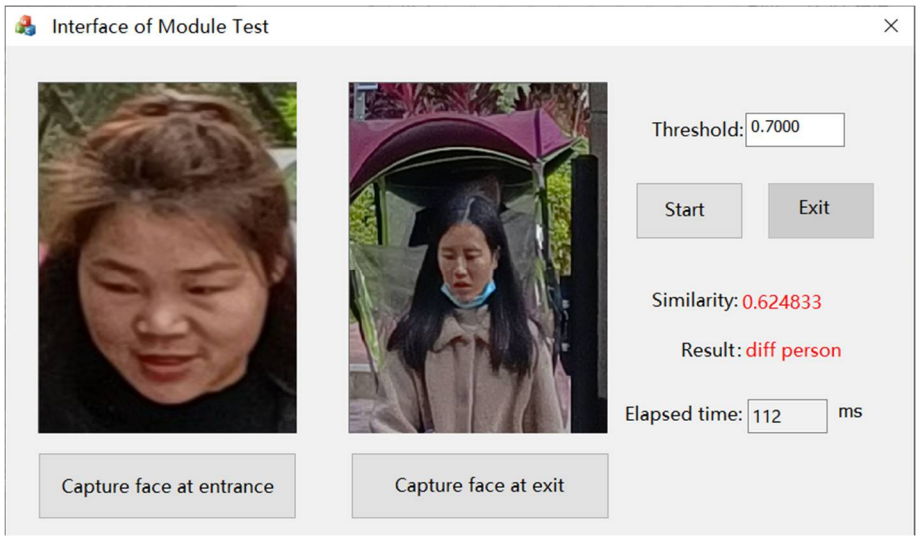

Figure 12. Test of different persons captured at the entrance and the exit.

5. Conclusions

In this paper, the gate face verification model was proposed to adapt to polymorphic inputs (multi-view faces, dynamic faces, and multiple faces in one image) for face verification. Experimental results demonstrated that the gate face verification model not only outperforms most existing algorithms across several databases, including LFW, YTF, and other public databases for multi-view face recognition, but also has better processing power for multi-face recognition on our multi-face databases. The two main contributions for performance improvement are the combination of the Modified SphereFace-20 and Batch Normalization to construct the proposed architecture—MDCNN model, and based on the model, a personal tracking system for a hazardous goods factory is proposed for application at workshop and factory gates.

Author Contributions: Conceptualization, Q.W. and X.H. (Xixian Huang); methodology, X.Z. and X.H. (Xixian Huang); software, X.Z. and X.H. (Xixian Huang); validation, Y.L., X.H. (Xi Huang) and H.Z.; formal analysis, X.H. (Xixian Huang); investigation, X.H. (Xi Huang); resources, Y.L.; data curation, H.Z.; writing—original draft preparation, X.Z. and X.H. (Xixian Huang); writing—review and editing, Q.W.; visualization, X.H. (Xixian Huang); supervision, Q.W.; project administration, X.H. (Xi Huang); funding acquisition, X.H. (Xixian Huang). All authors have read and agreed to the published version of the manuscript.

Funding: The authors gratefully acknowledge the support from Fujian Provincial Engineering Technology Research Center of Photoelectric Sensing Application, the National Natural Science Foundation of China (Grant No. 82171991), the fund from Special Funds of the Central Government Guiding Local Science and Technology Development (2020L3008), Natural Science Foundation of Fujian Province(2019J01271) and the Project on the Integration of Industry and Education of Fujian Province (2021H6026).

Institutional Review Board Statement: Only benchmark datasets are used in this paper. There are no any ethic issues.

Informed Consent Statement: The benchmark datasets are downloaded from public resources.

Data Availability Statement: The benchmark datasets can be found from the related references in text of the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Li, Y.; Wang, H. Quantitative Area Risk Assessment and Safety Planning on Chemical Industry Parks. In Proceedings of the International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering, Chengdu, China, 15–18 July 2013; pp. 413–419.
2. Fang, L.; Liang, J.; Jiang, L.; Wang, E. Design and Development of the AI-assisted Safety System for Hazardous Plant. In Proceedings of the 13th International Congress on BioMedical Engineering and Informatics, Chengdu, China, 17–19 October 2020; pp. 60–65.
3. Pavlenko, E.N.; Pavlenko, A.E.; Dolzhikova, M.V. Safety Management Problems of Chemical Plants. In Proceedings of the International Multi-Conference on Industrial Engineering and Modern Technologies (FarEastCon), Vladivostok, Russia, 6–9 October 2020; pp. 768–774.
4. Ma, Y.; Chang, D. Study on Safety Production Management Improvement of Small and Medium Sized Chemical Enterprises. In Proceedings of the International Conference on Logistics, Informatics and Service Sciences (LISS), Sydney, NSW, Australia, 24–27 July 2016; pp. 978–983.
5. Cevikap, H.; Triggs, B. Face Recognition Based on Image Sets. In Proceedings of the 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), San Francisco, CA, USA, 13–18 June 2010; pp. 2567–2573.
6. Deng, J.; Guo, J.; Yang, J.; Xue, N.; Cotsia, I.; Zafeiriou, S.P. ArcFace: Additive Angular Margin Loss for Deep Face Recognition. In Proceedings of the IEEE Transactions on Pattern Analysis and Machine Intelligence, Long Beach, CA, USA, 15–20 June 2019. [CrossRef]
7. Wang, F.; Cheng, J.; Liu, W.; Liu, H. Additive Margin Softmax for Face Verification. *IEEE Signal Process. Lett.* **2018**, *25*, 926–930. [CrossRef]
8. Wang, H.; Wang, Y.; Zhou, Z.; Ji, X.; Gong, D.; Zhou, J.; Li, Z.; Liu, W. CosFace: Large Margin Cosine Loss for Deep Face Recognition. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 5265–5274. [CrossRef]

9. Hsu, G.J.; Wu, H.; Yap, M.H. A Comprehensive Study on Loss Functions for Cross-Factor Face Recognition. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, WA, USA, 14–19 June 2020; pp. 3604–3611. [CrossRef]
10. Chan, C.H.; Kittler, J. Angular Sparsemax for Face Recognition. In Proceedings of the 25th International Conference on Pattern Recognition (ICPR), Milan, Italy, 10–15 January 2021; pp. 10473–10479. [CrossRef]
11. Ahmed, N.K.; Hemayed, E.E.; Fayek, M.B. Hybrid Siamese Network for Unconstrained Face Verification and Clustering under Limited Resources. *Big Data Cogn. Comput.* **2020**, *4*, 19. [CrossRef]
12. Cui, Z.; Li, W.; Xu, D.; Shan, S.G.; Chen, X. Fusing Robuts Face Region Descriptors Via Multiple Learning for Face Recognition in The Wild. In Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Portland, OR, USA, 23–28 June 2013; pp. 3554–3561.
13. Taigman, Y.; Yang, M.; Ranzato, M.A.; Wolf, L. DeepFace: Closing the Gap to Human-Level Performance in Face Verification. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 23–28 June 2014; pp. 1701–1708.
14. Lin, S.; Gong, Z.H.; Han, Z.H.; Shi, H.B. Multi-angle face recognition algorithm based on multi-RKHS. *Acta Photonica Sin.* **2013**, *42*, 1436–1441.
15. Wang, T.S.; Shi, P.F. Kernel grassmannian distances and discriminant analysis for face recognition from image sets. *Pattern Recognit. Lett.* **2009**, *30*, 1161–1165. [CrossRef]
16. Galoogahi, H.K.; Sim, T. Face Sketch Recognition by Local Radon Binary Pattern: LRBP. In Proceedings of the 2012 IEEE International Conference on Image Processing (ICIP), Orlando, FL, USA, 30 September–3 October 2012; pp. 1837–1840.
17. Mignon, A.; Jurie, F. Pcca: A New Approach for Distance Learning from Sparse Pairwise Constraints. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 16–21 June 2012; pp. 2666–2672.
18. Sun, Y.; Wang, X.; Tang, X. Deeply Learned Face Representations Are Sparse, Selective, and Robust. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 2892–2900.
19. Schroff, F.; Kalenichenko, D.; Philbin, J. Facenet: A Unified Embedding for Face Recognition and Clustering. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 815–823.
20. Liu, W.; Wen, Y.; Yu, Z.; Li, M.; Raj, B.; Song, L. SphereFace: Deep Hypersphere Embedding for Face Recognition. In Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6738–6746.
21. Huang, G.; Mattar, M.; Berg, T.; Miller, E. Labeled Faces in The Wild: A Database Forstudying Face Recognition in Unconstrained Environments. In Proceedings of the Workshop on Faces in'Real-Life'Images: Detection, Alignment, and Recognition, Marseille, France, 16–18 October 2008.
22. Wolf, L.; Hassner, T.; Maoz, I. Face Recognition in Unconstrained Videos with Matched Background Similarity. In Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Colorado Springs, CO, USA, 20–25 June 2011; pp. 529–534.
23. Zhang, K.; Zhang, Z.; Li, Z.; Qiao, Y. Joint face detection and alignment using multi-task cascaded convolutional networks. *IEEE Signal Process. Lett.* **2016**, *23*, 1499–1503. [CrossRef]
24. Wu, D.; Wu, X.; Qin, H. Research on multi-view face recognition with regression algorithm. *Tech. Acoust.* **2015**, *34*, 172–175.
25. Hu, J.L.; Lu, J.W.; Tan, Y.P. Discriminative Deep Metric Learning for Face Verification in The Wild. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 23–28 June 2014; pp. 1875–1882.
26. Ioffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In Proceedings of the International Conference on Machine Learning (ICML), Lille, France, 6–11 July 2015.
27. Wang, X.; Tang, X. Face photo-sketch synthesis and recognition. *IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI)* **2009**, *31*, 1955–1967. [CrossRef] [PubMed]
28. Patil, S.; Shubhangi, D.C. Froensic Sketch Based Face Recognition Using Geometrical Face Model. In Proceedings of the 2017 International Conference for Convergence in Technology, Mumbai, India, 7–9 April 2017; pp. 450–456.
29. Hu, Y.; Mian, A.; Owens, R. Sparse Approximated Nearest Points for Image Set Classification. In Proceedings of the 2011 Conference on Computer Vision and Pattern Recognition (CVPR), Colorado Springs, CO, USA, 20–25 June 2011; pp. 121–128.
30. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
31. Liu, J.; Bae, S.; Park, H.J.; Li, L.; Yoon, S.B.; Yi, J. Face photo-sketch recognition based on joint dictionary learning. In Proceedings of the 14th Iapr International Conference on Machine Vision Applications (MVA), Tokyo, Japan, 18–22 May 2015; pp. 77–80.
32. He, K.; Zhang, X.; Ren, S.; Sun, J.L. Delving Deep into Rectifiers: Surpassing Human-Level Performance on Imagenet Classification. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 1026–1034.