Hindawi
Mathematical Problems in Engineering
Volume 2021, Article ID 5510027, 12 pages
https://doi.org/10.1155/2021/5510027

Research Article

Unrestricted Face Recognition Algorithm Based on Transfer Learning on Self-Pickup Cabinet

Zhixue Liang

School of Computer and Software, Nanyang Institute of Technology, Nanyang 473000, China

Correspondence should be addressed to Zhixue Liang; 3161015@nyist.edu.cn

Received 25 February 2021; Revised 31 March 2021; Accepted 31 March 2021; Published 15 April 2021

Academic Editor: Yandong He

In the contactless delivery scenario, the self-pickup cabinet is an important terminal delivery device, and face recognition is one of the efficient ways to achieve contactless access express delivery. In order to effectively recognize face images under unrestricted environments, an unrestricted face recognition algorithm based on transfer learning is proposed in this study. First, the region extraction network of the faster RCNN algorithm is improved to improve the recognition speed of the algorithm. Then, the first transfer learning is applied between the large ImageNet dataset and the face image dataset under restricted conditions. The second transfer learning is applied between face image under restricted conditions and unrestricted face image datasets. Finally, the unrestricted face image is processed by the image enhancement algorithm to increase its similarity with the restricted face image, so that the second transfer learning can be carried out effectively. Experimental results show that the proposed algorithm has better recognition rate and recognition speed on the CASIA-WebFace dataset, FLW dataset, and MegaFace dataset.

1. Introduction

At present, the global epidemic prevention and control has become normal, so it is necessary to develop effective prevention and control measures. In the field of logistics distribution, terminal contactless distribution has become the focus of the public [1]. At present, the main research technology of terminal contactless matching is face recognition technology. In the contactless scene, the face images are often interfered by unrestricted environmental factors such as light, occlusion, and expression. For the face image affected by the unrestricted environment, the quality is poor, so it is difficult to recognize, and the recognition accuracy is low [2]. Therefore, the study of the fast face image recognition algorithm in unconstrained environment is of great significance for the field of terminal contactless distribution.

In computer vision, face recognition is one of the most important research directions. Face recognition can be used in digital cameras, access control systems, identity recognition network applications and entertainment applications, and other fields. With the rapid development of artificial intelligence and image analysis technology, face recognition technology has been widely used in many fields [3]. However, face images collected under an unrestricted environment are subjects to the mixed interference of illumination, occlusion, expression, and other factors, so that the face recognition accuracy is greatly reduced. At the same time, because the original image exists in high dimensional data, the speed of face recognition is also affected. Therefore, it is of great significance to study the fast face image recognition algorithm in unrestricted environment.

The face recognition method can be divided into the traditional method and deep neural network method. The traditional methods are mainly composed of the geometric feature-based method, local feature analysis method, and eigenface method. In recent years, with the rapid development of deep learning theory, the deep neural network has become the most widely used algorithm in face recognition. The first deep neural network model that attracted attention in face recognition was the Facebook's DeepFace [4] model in 2014. The DeepFace model is the first study to use deep learning to approach human performance in face recognition. It achieves an average accuracy of 97.35% on the LFW dataset, approaching the human limit of 97.5%. DeepFace

extracts highly compressed facial features, and models facial features with the 3D model. The model assumes that the feature parts of all faces are fixed at the pixel level, and the faces are aligned by region-based affine transformation. Finally, a deep neural network is used as a feature extractor to extract the face features from the image. This model contains a very large number of parameters due to the use of the full connection layer. Subsequently, Professor Tang et al. developed a series of new deep neural network structures of DeepID. DeepID1 [5] is a small network consisting of four convolutional layers and two fully connected layers. The output DeepID feature of the penultimate layer contains 160 dimensions. The higher the number of layers is, the larger the receptive field is in the features of the convolutional neural network. So the connection mode takes into account both the local features of the face and the global features of the whole face. DeepID2 [6] added the loss function of face verification on the basis of DeepID1 and learned the discriminant features of faces through two objective functions. In the final loss function, the importance of face classification and face verification is adjusted by changing the weight of the two loss functions. The final loss function is composed of the weighted sum of the two loss functions of face classification and face verification. The subsequent DeepID2+ [7] network continues to change the network structure on the basis of DeepID2, from 160 dimensions to 512 dimensions. The second is that a lot of structural analysis has been performed, and it turns out that neurons at the higher levels are more sensitive to faces. DeepID2+ achieves results that exceed human performance on LFW datasets and is robust to appropriate face occlusion. Compared with the DeepID2+ model, the latest DeepID3 model further deepens the number of layers and achieves better results.

In recent years, it has become a trend to train models to complete face recognition based on the deep neural network using larger labeled datasets. For example, Google used 200 million face images containing 8 million different individuals in FaceNet [8]. However, the cost of collecting and tagging such datasets is enormous, and training them through deep neural networks requires better hardware support. Larger and larger datasets are used to train better models, but this is not a good development direction. For example, when the face verification accuracy on the LFW dataset increased from 99.47% to 99.77%, the number of trained images increased from 200,000 to 1.2 million. However, in most realistic scenarios, only a small amount of data can be used, and how to learn rich knowledge from these limited data is a problem to be solved. There is growing interest in the research and development of technologies in different fields, such as domain adaptive and transfer learning [9], and the personalized search model achieved amazing results [10]. In a study [11], a transfer learning algorithm combining a large number of source domain samples with a relatively small number of target domain data is proposed. In a study [12], a deep transfer metric learning method for cross-domain visual recognition is proposed by transferring recognition knowledge from the labeled source domain to the unlabeled target domain. Therefore, the best way to guide the face representation learning of a few samples through deep learning can be knowledge transfer or domain adaptation. In other words, you can learn some preknowledge from other large databases and then fine-tune that knowledge in your target domain.

Therefore, compared with traditional machine learning methods, deep learning methods have great advantages in the field of feature extraction. In other words, the convolutional neural network algorithm can automatically extract the features of the image content layer by layer without any prior knowledge [13]. When the number of samples is large enough and the number of network layers is large enough, the data can be fully excavated to extract excellent features with resolution. However, deep learning is driven by data, and it is difficult to extract features with generalization ability when the amount of data is insufficient. In the unrestricted environment, the number of face images is small and the acquisition cost is high, so the data scale is not enough to support the training of the network.

In this study, the transfer learning method is used to solve the problem of insufficient number of face images in unrestricted environment. First, the faster RCNN algorithm is improved to improve the recognition speed of the algorithm. Then, the network parameters trained by the ImageNet large-scale dataset [14] are used for initialization by using one transfer learning. Second, the network parameters are tuned through the face images under restricted conditions with relatively sufficient data volume, and the face images under restricted conditions are trained to be able to recognize. Finally, the face image is enhanced under unrestricted conditions, such as attitude alignment, illumination brightness enhancement, and angle rotation. Therefore, it can increase its similarity with the face image under the restricted condition. After secondary transfer learning of the previously obtained restricted face image recognition network, the network that can accurately recognize the unrestricted face image is obtained.

The rest of this article is organized as follows. The second section introduces the face recognition model of the improved faster RCNN algorithm. The third section introduces the enhancement method of unrestricted face image. The fourth section gives the transfer learning method and experimental results. The fifth section is the conclusion of this study.

2. Materials and Methods

2.1. Face Recognition Model with the Improved Faster RCNN Algorithm. Faster RCNN is an improvement object detection framework over the existing RCNN algorithm framework. RCNN combines CNN with a regional candidate box. On this basis, a faster RCNN algorithm appears [15]. The basic idea of these algorithms is to divide the original image into different candidate boxes. The convolutional neural network CNN is used as a feature extractor, and a feature vector is extracted from the candidate box. Then, a classifier is trained to classify the feature vectors. Finally, the target detection framework consists of three parts as shown in Figure 1.

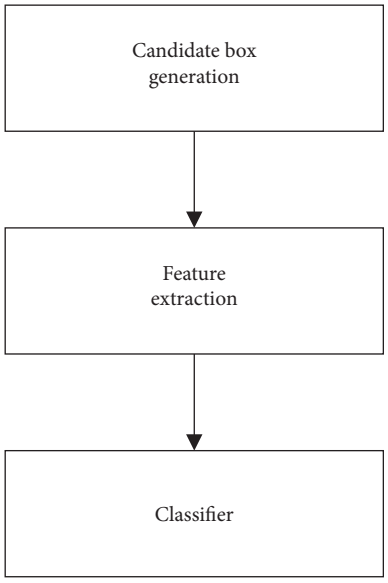


Figure 1: The target detection framework.

The faster RCNN proposes a region proposal network (RPN) to generate candidate boxes based on the convolutional neural network. In this network, the faster RCNN is still used as the detector. RPN and faster RCNN actually share a convolutional layer to extract features; thus, RPN and faster RCNN combine to form a single unified a faster RCNN network. As shown in Figure 2, it can be roughly divided into four parts: convolutional trunk network, RPN micronetwork, faster RCNN detector, and multitask loss.

In order to speed up the detection process of the model, the convolutional backbone network and the multitask loss function are adjusted; at the same time, the RPN micronetwork of faster RCNN is improved in this study.

2.2. The Adjusted Convolution Backbone. The convolutional layer in faster RCNN borrows the convolutional layer architecture of the classical classification model and its pretrained weight. The pretrained model of the classification task was applied to a similar detection task, and the weight of the classification model was directly adjusted, which greatly reduced the training amount of the model.

As shown in Figure 3, the convolutional layer part of the VGG-16 classification model [16] is adopted by the convolutional backbone network in this study. The feature map is not pooled before output, which changes slightly. The step length of all convolution operations is 1, and the boundary filling is 1.The width and height of convolution kernel is 3×3, which ensures that the width and height of the image remain unchanged before and after convolution. The pooling layer adopts the maximum pooling of 2×2 with a step size of 2. The pooling layer does not affect the number of channels in the image. However, after each pooling, the width and height of the image will be halved. The number of channels for convolution is 64, 128, 256, 512, and so on. The number of channels represents the number of feature images extracted by convolution. After each convolutional layer, the rectified

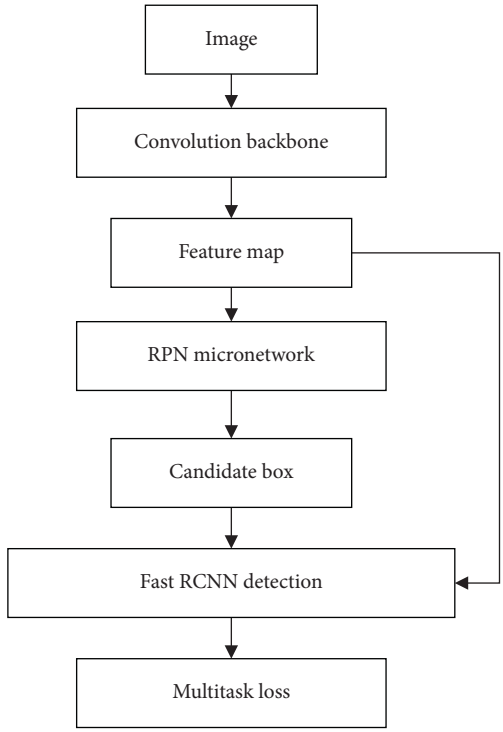


Figure 2: The algorithm framework of faster RCNN.

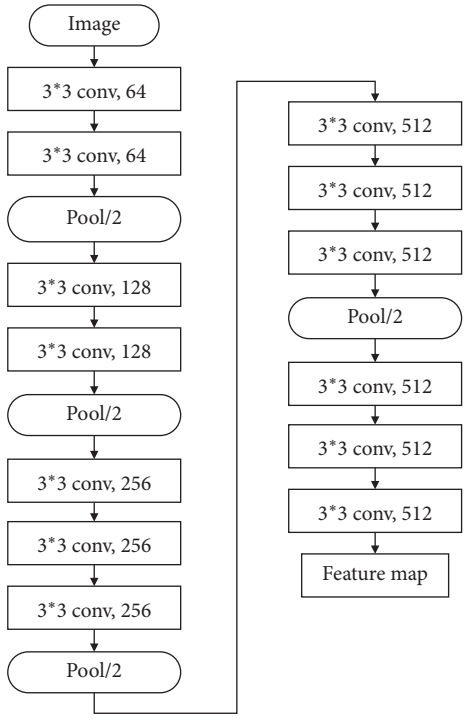


Figure 3: The convolution backbone of faster RCNN.

linear unit (ReLU) activation function performs nonlinear transformation, which does not affect the width and height of the feature and the number of channels. Therefore, after the input image is convolved with 13 layers and pooled with 4 layers, the width and height of the output feature map

obtained will become 1/16 of the original image, and the number of channels will change from RGB 3 channels to 512.

2.3. Improved RPN Micronetwork. The RPN micronetwork adopts the sliding window mode to generate 9 anchors in the input image for each point on the feature map. As shown in Figure 4, it is the anchor point corresponding to the center point of the feature map on the input image. The outer black box is the original image of 800×600 pixels. The inside, the middle, and the outside of the three thick and thin boxes, respectively, represent the size of 128, 256, and 512. In each scale, there are three situations of aspect ratio of 1 : 2, 1 : 1, and 2 : 1, so each sliding window corresponds to 9 anchors.

In the original anchor, 128, 256, and 512 are set in order to ensure that the target object can adapt to various scales. As for anchor settings with three scales and three proportions, this is equivalent to a point in the feature map that can correspond to 9 regions in the original image perception field. Each area corresponds to an anchor. With supervised learning parameter training, the model can adjust the parameters, so that the calculated feature map can correspond to the object in the original picture. Smaller scales can capture small differences between objects, which allow different classes of objects to be distinguished. The larger scale can ensure that the original image is covered, that is, all the receptive fields, so that the original image will not miss the undetected objects.

The RPN network structure is shown in Figure 5. For a given input image, the convolutional layer generates the convolutional feature map. The RPN micronetwork slides a small window of 3×3 on the feature map after convolution. Each window maps to a 256-dimensional eigenvector, which is then fed into two branch networks: Cls classification network and Reg regression network. Here, the original 512-dimension eigenvector is improved to a 256-dimension eigenvector, which accelerates the detection speed.

Cls classifier classifies the feature vectors of window mapping. It predicts a foreground probability and background probability for each anchor, so there will be $2 \times 9 = 18$ probability values, which are represented by 18 neurons. The Reg regression performs regression on the eigenvectors of the window map. It predicts the center point coordinates and offset of width and height of each anchor, which is represented by (t_x, t_y, t_w, t_h). So there are $4 \times 9 = 36$ offsets, represented by 36 neurons. Note that the processing of the feature map is carried out in a sliding window mode, so these processes can be realized by convolution operation.

2.4. Multitasking Loss. For RPN training, the multitask loss is adopted in this study to combine the cross entropy loss of the classifier with the Smooth_{L1} loss of the regression. In order to get the multitask loss, suppose that its classification loss is $L_{\text{cls}}(p_i, p_i^*)$ and regression loss is $L_{reg}(t_i, t_i^*)$, then the multitask loss $L(\{p_i\}, \{t_i\})$ of all samples is calculated as follows:

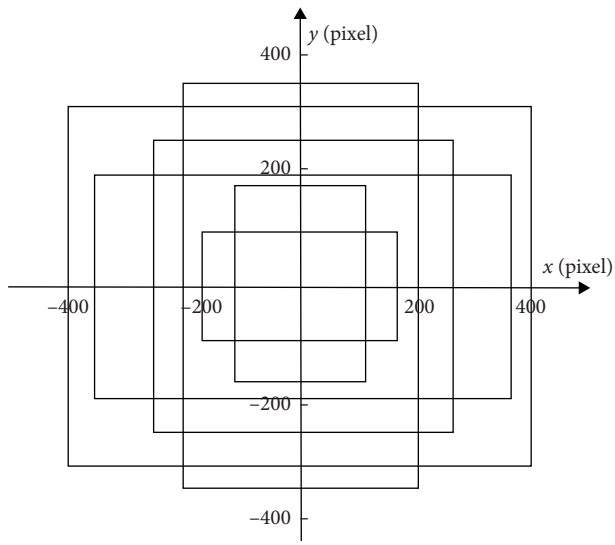


Figure 4: Anchor corresponding to the center point of the feature map.

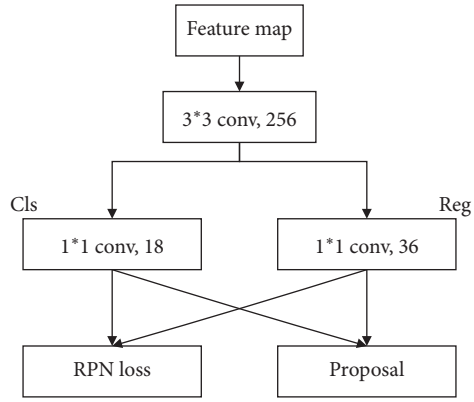


Figure 5: The RPN micronetwork.

$$\frac{1}{N_{\text{cls}}} \sum_i L_{cls} + \lambda \frac{1}{N_{\text{reg}}} \sum_i p_i^* L_{\text{reg}}, \tag{1}$$

where N_{cls} and N_{reg} are the standardized terms, and λ is the tradeoff coefficient. Consider the classification loss of a single sample

$$L_{cls}(p, p^*) = -\log\ p_{p^*}, \tag{2}$$

where p^* is the category tag corresponding to anchor, and p is the prediction probability of its corresponding category tag.

In the multitask loss, only the regression loss of the anchor marked as positive is calculated, and the regression loss function is considered separately.

$$L_{\text{reg}}(t_i, t_i^*) = \text{Smooth}_{L1}(t_i - t_i^*), \tag{3}$$

where

$$\text{Smooth}_{L_1}(x) = \begin{cases} 0.5x^2, & \|x\| < 1, \\ \|x\| - 0.5, & \text{others}, \end{cases} \tag{4}$$

where t_i and t_i^* are represented by a source of four groups. In order to express the simplicity of above, the subscript i is removed. And only the regression under a single sample is considered for the predicted offset t_i of anchor and the true offset t_i^* of ground truth for anchor.

$$\begin{aligned} t &= \left(t_x, t_y, t_w, t_h\right), \\ t^* &= \left(t_x^*, t_y^*, t_w^*, t_h^*\right), \end{aligned} \tag{5}$$

where

$$\begin{aligned} t_x &= \frac{(x - x_a)}{w_a}, \\ t_x^* &= \frac{(x^* - x_a)}{w_a}, \\ t_y &= \frac{(y - y_a)}{h_a}, \\ t_y^* &= \frac{(y^* - y_a)}{h_a}, \\ t_w &= \log\left(\frac{w}{w_a}\right), \\ t_w^* &= \log\left(\frac{w^*}{w_a}\right), \\ t_h &= \log\left(\frac{h}{h_a}\right), \\ t_h^* &= \log\left(\frac{h^*}{h_a}\right). \end{aligned} \tag{6}$$

The SGD method [17] is used in the RPN network to optimize the multitask loss function to minimize the loss function $L(\{p_i\}, \{t_i\})$. In the optimization process, the model adjusts the parameters to find a local optimal solution. During the test, RPN is used to predict the category probability of each anchor and the regression offset of the anchor marked as positive. The candidate box of regression offset correction is obtained by using the nonmaximal suppression method from the output of the RPN micronetwork.

3. The Enhancement Method of Unrestricted Face Image

3.1. Face Posture Alignment through Image Rotation. Because of the complicated attitude problem, the image in nature brings great challenge to the key point positioning. Therefore, face image posture clustering is needed, and then, different categories of images are trained. For an image I, the goal of face alignment is to learn a nonlinear mapping function D from features to key points. Due to the large difference in attitude, D's learning process is complex, so D is divided into several simple subtasks $\{D_1, D_2, \ldots, D_n\}$. In this way, in each subtask D_k, faces have similar postures, which simplifies the learning of D.

Because of the diversity of posture, affine transformation is used to adjust face pose before clustering. Affine transformation matrix M is given in equation (7). Affine transformation only needs two sets of three-point coordinates to obtain the matrix M. The three coordinates are the coordinates of the two eyes and the middle position of the mouth. For each image, one is the coordinates $(x, y, 1)^T$ in the original coordinate system, and the other is the coordinates $(u, v, 1)^T$ in the target seat system. Notice that the position of the eyes in the target coordinate system is on the same horizontal line. Once the transformation matrix $M$is calculated, it can be used to affine transform the entire image. The result is shown in Figure 6. The first row is before the affine transformation, and the second row is after the transformation. There are only three kinds of corrected facial posture: positive face, left face, and right face.

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} a_1 b_1 c_1 \\ a_2 b_2 c_2 \\ 000 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}. \tag{7}$$

Considering that no real labels about poses are provided in the dataset, the K-means unsupervised clustering algorithm [18] is used to realize pose clustering. Then, a better initial position is provided closer to the true position for all samples in each class. The adaptive SDM model is used to extract discriminative features, and each category is trained separately to obtain three different training models. Since the key point position is corrected by affine transformation, the final output key point position needs to be converted to the source coordinate system by an inverse transformation. As shown in formula (8), $x_t'$is the position coordinate in the coordinate system after affine transformation, and x_t is the coordinate of the key point position in the source coordinate system.

$$x_t = M^{-1} x_t'. \tag{8}$$

3.2. The Alignment of Face Images. In face alignment processing, a regression function is learned to predict the position increment between the current position and the real position. Considering that the regression function is a complex nonlinear mapping function, a linear regression method is used by SDM instead of complex nonlinear regression to predict the position. The objective function is as follows:

$$\min f(x_0 + \Delta x) = \left\| h\left(d\left(x_0 + \Delta x\right)\right) - \Phi_* \right\|_2^2. \tag{9}$$

Suppose that a picture has m pixels $d \in R^{m \times 1}$, and $d(x) \in R^{p \times 1}$ is the p key points on the picture. $x_0 \in R^{p \times 2}$ represents the initial position, and h is a nonlinear feature extraction function. In the experiment of this study, the

Figure 6: Face posture before and after affine transformation.

HOG feature is used. $\Phi_* = h(d(x_*))$ represents the feature extracted based on the real position. For each sample, there is an initial position x_0. According to Newton's gradient descent criterion, it is only needed to iterate on formula (7) repeatedly to obtain a sequence of Δx, $\{\Delta x_1, \Delta x_2, \ldots, \Delta x_k\}$. And after each iteration, $x_k = x_{k-1} + \Delta x_k$ is corrected. After several iterations, $x_k$will converge to the optimal position x_*.

Taylor's expansion was carried out on equation (9); then, Δx is derived. Let the derivative be 0; then, equation (10) is obtained as follows:

$$\Delta x = -H^{-1} J_f = -2H^{-1} J_h^T (\Phi_0 - \Phi_*). \tag{10}$$

Let $R_0 = -2H^{-1}J_h^T$, $b_0 = 2H^{-1}J_h^T\Phi_*$, and the first iteration can be expressed as follows:

$$\Delta x_1 = R_0\Phi_0 + b_0, \tag{11}$$

where R_0 is seen as the direction of decline. A series of descending directions R_k and $b_k$need to be calculated and expressed as equation (12). The features extracted at each stage constitute a set$\Phi = \{\Phi_1, \Phi_2, \ldots, \Phi_k\}$.

$$\Delta x_k = R_{k-1}\Phi_{k-1} + b_{k-1}. \tag{12}$$

The adaptive feature extraction is embodied in Φ_k. As shown in Figure 7, here are five key points as an example. The red dot represents the position obtained at each stage, the green dot represents the real position, and the red circle represents the radius r of the feature extraction frame. Figure 7(a) shows the transformation trend of the radius r of the SDM model. It can be seen that the size of r is constant. This will extract useless features that affect the positioning of key points.

Face alignment is a process from coarse to fine. The size of the radius r of the feature extraction frame is related to the position increment Δx generated in each stage.

When Δx in the training sample is widely distributed, it is more inclined to use large r to extract features. Follow the rule from coarse to fine and adaptively change the size of r to obtain discriminative features. As shown in Figure 7(b), in the initial stage, the obtained position x_k is far from the real position x_*, and Δx is widely distributed. The use of large feature boxes near key points to extract more useful information is conducive to handling large differences in face shape and ensuring robustness. As the stage increases, the distance between x_k and x_*becomes smaller and smaller, and the use of a gradually reduced feature extraction frame can effectively obtain discriminative features. Especially in the later stages, a small feature extraction frame can reduce noise and ensure accuracy. Equation (13) expresses the acquisition process of the radius r_k of the adaptive feature extraction frame, and x_k^{ij} represents the position of the j^{th} key point of the i^{th} sample in the k stage.

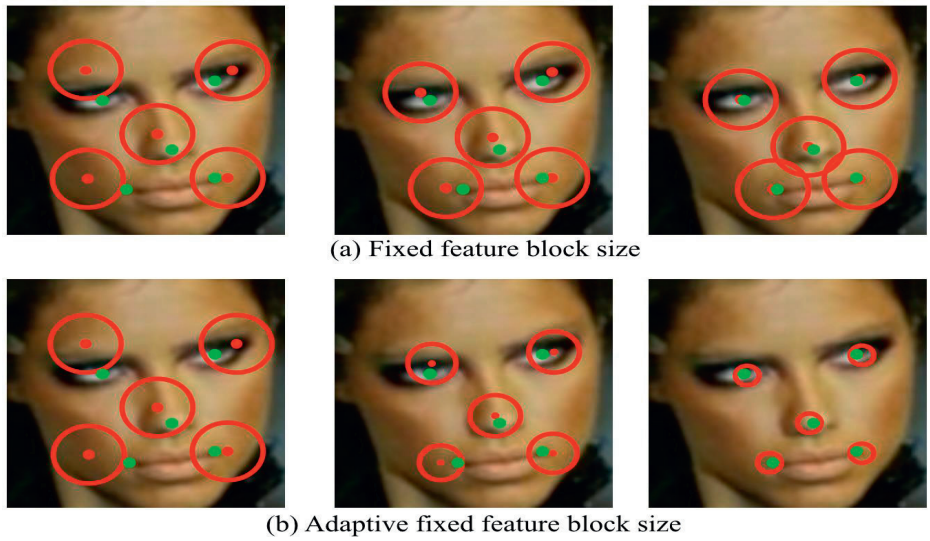


FIGURE 7: Trend of feature extraction block size with the number of stage. (a) Fixed feature block size. (b) Adaptive fixed feature block size.

$$r_k = \max\left(\left\|x_k^{ij} - x_*^{ij}\right\|\right), \quad j = 1, 2, \ldots, p, i = 1, 2, \ldots, N. \tag{13}$$

Although r_k is gradually decreasing, the strategy is tough, and it does not take into account the distribution of position increments Δx generated at each stage of the training sample. In the experiment of this study, the radius r_k of the feature extraction frame is adaptively obtained according to the Δx produced in each stage. At each stage, each sample will produce Δx with a dimension of $p \times 2$. Calculate the distance between the current position of each key point and the real position to obtain P distances. N samples will produce $N \times p$ distances. The maximum distance is selected among $N \times p$ distances, which is regarded as the size of the feature extraction frame r of each key point of all samples at this stage. The reason for the largest selection is to extract useful features around the real key points. In this way, the size of the feature extraction frame selected at each stage fully considers the distribution of the current position and the true position of the sample. As the stage increases, it will gradually decrease, and the extracted features can be extracted at the real position to the greatest extent, and the interference of redundant features is also reduced.

By obtaining the radius r of the adaptive feature extraction frame, the discriminative feature $\Phi = \{\Phi_1, \Phi_2, \ldots, \Phi_k\}$ is obtained. The values of R_k and b_k can be calculated by minimizing the difference between the current position increment and the actual position increment, which is shown as follows:

$$\arg\min_{R_k, b_k} \sum_{d^i} \left\|\Delta x_*^i - R_k \Phi_k - b_k\right\|. \tag{14}$$

This equation is a typical linear least squares problem, and an analytical solution can be obtained. Then, according to formula (12), the position increment Δx_k of the k^{th} stage can be obtained. Then, the key point position x_k of the k^{th} stage can be obtained. After the iteration is completed, R_k and b_k obtained at each stage can be saved.

In a test sample, the attitude of the face image is first determined, and the corresponding initial position x_0 is given. Then, a series of R_k and b_k obtained in the training stage are used to predict the position of key points.

3.3. Enhancement of the Face Image. It is difficult and costly to obtain sufficient samples of unrestricted face images, and it is difficult to train a satisfactory model based on the number of existing samples. When encountering the problem of insufficient data volume, the common method is to expand the dataset by random cropping, color conversion, and other methods. Although this method can improve the recognition accuracy of the network, the improvement is limited. In this study, the method of transfer learning is adopted with the aid of the daytime aerial photography dataset with relatively sufficient data volume. Fine-tune the daytime vehicle recognition network model through nighttime data to realize the recognition of nighttime targets.

Since restricted face images and unrestricted face images have great similarity, the features extracted by the network are also very similar. Transfer learning takes advantage of this similarity to transplant the restricted face training model to the unrestricted face recognition network. Then, we use the unrestricted face data to enhance the image to make the algorithm more suitable for the recognition of unrestricted face images. The higher the similarity of two objects connected by transfer learning is, the more conducive to transfer learning is. By comparing restricted face data with nonrestricted face data, the main difference lies in the interference of illumination, occlusion, expression, and other factors. Therefore, in order to improve the similarity between restricted face data and unrestricted face data, it can be processed from multiple perspectives such as illumination, occlusion, and expression to improve the similarity. In this study, illumination enhancement is selected for image processing to

improve the similarity between restricted and unrestricted face data. In the algorithm of illumination enhancement, the Retinex algorithm is adopted in this study because it can weaken the influence of light on the object in the image and restore the original color, edge, and other information of the object.

In Retinex theory, images are thought to be composed of incident and reflected light [19]. The basic idea of the image enhancement method is to remove the influence of the illuminating light and retain the reflection properties of the object itself. The Frankle-McCann Retinex iterative algorithm is used in this study. The iterative piecewise linearization based on spiral structure and compares paths to estimate illumination is adopted in this algorithm. The spiral structure path is that the pixel correction result at point (0, 0) will be jointly determined by the pixel value of the inflection point of the path. The number of selected reference points is moderate. The closer to the target point, the more intensive the sampling, the better the result. The color images are used in this study, so the three channels of RGB are processed separately. Finally, the three channels are merged and output.

(1) The early stage of the data conversion

In order to reduce the amount of calculation in subsequent calculations, the pixel value of the original image is converted from the integer domain to the logarithmic domain. To avoid negative values, the original image is added with 1 to the pixel value as a whole, which is expressed as follows:

$$s(x, y) = \log[1 + S(x, y)]. \tag{15}$$

Then, the constant matrix r can be initialized. The constant value takes the average value of the original image pixels, and the size is the same as the original image.

(2) Comparison and correction between pixels

For an image with a pixel size of $m \times n$, the coordinate change between the two comparison points at the furthest distance from the target point is expressed as follows:

$$D = 2^p, \tag{16}$$

where

$$P = \text{fix}[1b \min(m, n) - 1], \tag{17}$$

where fix is the rounding function. Then, the distance between the two comparison points in each next step is shortened to half of the previous step, and the direction is rotated clockwise, which is shown as follows:

$$D = -\frac{D}{2}. \tag{18}$$

And the direction is rotated until the interval of the comparison points is less than 1.

Assuming that $r_n(x, y)$ is the result of the previous iteration, $r_{n+1}(x, y)$ is the result of this iteration. If $D > 0$, then

$$\begin{cases} r_{n+1}(x + D, y) = r_n(x, y) + s(x + D, y) - s(x, y), \\ r_{n+1}(x, y + D) = r_n(x, y) + s(x, y + D) - s(x, y). \end{cases} \tag{19}$$

Otherwise, if $D < 0$, then

$$\begin{cases} r_{n+1}(x, y) = r_n(x - D, y) + s(x, y) - s(x - D, y), \\ r_{n+1}(x, y) = r_n(x, y - D) + s(x, y) - s(x, y - D). \end{cases} \tag{20}$$

(3) Image output display

After the iterative operation, the gray value of the reflected image is often a floating point number, which needs to be linearly converted to an effective gray value:

$$R(x, y) = \frac{r_{n+1}(x, y) - r_{(n+1)\min}(x, y)}{r_{(n+1)\max}(x, y) - r_{(n+1)\min}(x, y)} \times 255, \tag{21}$$

where $r_{(n+1)\max}(x, y)$ and $r_{(n+1)\min}(x, y)$ are the maximum and minimum values of the iteration result $r_{n+1}(x, y)$, respectively, and $R(x, y)$ is the final enhancement result.

4. Results and Discussion

4.1. Method of Transfer Learning. In order to effectively recognize face images in an unrestricted environment, an unrestricted face recognition algorithm based on transfer learning is proposed. The region extraction network of the faster RCNN algorithm is improved to improve the recognition speed of the algorithm. In order to improve the detection accuracy, a transfer learning method is adopted, in which network parameters trained by large-scale datasets are used for initialization. Then, the network parameters trained by the face dataset under unrestricted conditions are fine-tuned. The specific steps are expressed as follows.

Step 1. Large ImageNet dataset and face image dataset are transferred for the first time. The improved faster RCNN network in this study is trained by using the large ImageNet dataset and face image dataset. The parameters obtained from the training are used for network initialization.

Step 2. The secondary transfer learning of face image datasets under unconstrained conditions is carried out. The network parameters trained by the face dataset can be fine-tuned under unrestricted conditions.

Step 3. The initialized network is used to train the RPN and generate ROI

Step 4. According to the ROI obtained in Step 3, the source domain dataset is used to conduct classification and regression training for the initialized network

Step 5 . The network obtained in Step 4 is used to train RPN, adjust only the network layer parameters specific to RPN, and generate ROI

Step 6 . The generated ROI training network was used for classification and regression, and the shared convolutional layer parameters were kept fixed. So far, the training of the faster RCNN network for the target domain data detection model is completed.

4.2. Restricted Face Image Recognition Experiment. In order to verify the effectiveness of the face recognition algorithm proposed in this study, the recognition experiment is conducted on the CASIA-WebFace face dataset [16], LFW dataset [20], and MegaFace dataset [21] under unrestricted conditions.

4.2.1. CASIA-WebFace Dataset. CASIA-WEBFACE is one of the most important large-scale datasets in the field of face recognition. It contains more than 494,000 face images with labels of 10,575 people, and the size of its training set is only 0.49 MB. In this study, face images belonging to the same person as LFW and MegaFace were first removed from the dataset. A total of 122,875 face images of 2580 people were selected from the rest of the dataset, and these images were divided into training set, verification set, and test set according to the ratio of 7:2:1. The face image in the training set is preprocessed. Obtain the largest face region in the face image and remove the interference outside the face region. Key points were set in the image, and affine transformation was carried out according to the nose and eyes, so as to make the eyes flush and the nose centered. Then, the face image is further processed to make its size as 112×112. Among them, the face image of the training set is shown in Figure 8.

4.2.2. Recognition Results of Restricted Face Images. The network model trained by ImageNet is used to initialize the network parameters of the algorithm in this study, and the restricted face image set in the CASIA-WebFace dataset is used to fine-tune the algorithm network in this study, and the target recognition of the restricted face image based on one transfer learning is completed. In this experiment, the recognition effects of different networks are compared on CASIA-WebFace. The results are given in Table 1.

By observing the recognition results in Table 1, it can be seen that the network model in this study can accurately identify restricted face images, and the proposed algorithm has a faster recognition speed.

4.3. Unrestricted Face Image Recognition Experiment. In order to verify the effectiveness of the proposed algorithm for unrestricted face image recognition, this experiment was tested on LFW and MegaFace datasets under unrestricted conditions. In the model training of CASIA-WebFace datasets, face images belonging to the same person as those in LFW and MegaFace datasets have been removed.

The low-level convolution layer of the network model is used to extract shallow features such as edges, colors, and textures, which has little influence on different datasets. In this study, the parameters of the low 3-layer convolutional network are fixed for the trained constrained face image recognition model. According to the dataset of unrestricted face images, only the parameters of the deeper network are fine-tuned.

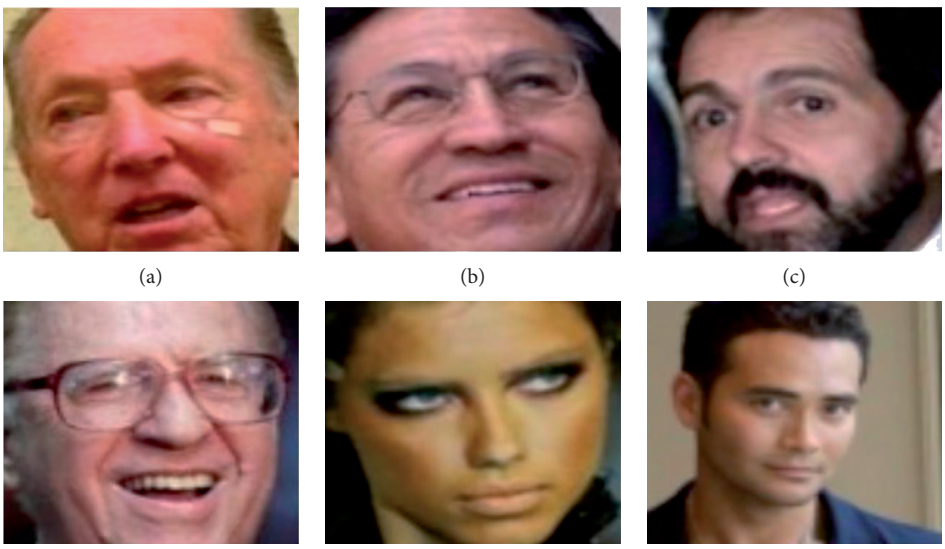
4.3.1. LFW Dataset Experiment. The LFW dataset contains more than 13,000 facial pictures of 5749 people in the natural environment. In the natural environment, human faces are often affected by illumination, expression, and occlusion, which bring great challenges to recognition. In the experiment, a View2 test set containing 6000 pairs of faces was used. In this dataset, it contains a total of 10 folds, and each fold contains 300 pairs of matched and mismatched faces. The random sample of the LFW dataset is shown in Figure 9.

Comparison tests were performed on unrestricted datasets, and the results are shown in Table 2. Through experimental comparison in Table 2, it can be found that the algorithm proposed in this study achieves a quite good recognition effect in the FLW dataset, which is 3.20% higher than Ouamane in the same small dataset. Even in the big data training set, the recognition efficiency is higher than that of DeepFace and DDML. The accuracy is similar to that of Face Net with large training data, which shows that the algorithm framework in this study has a good recognition rate. And the recognition speed of the algorithm in this study has a very good application value for actual engineering applications because of other algorithms.

4.3.2. MegaFace Dataset Experiment. MegaFace is a public face test dataset with millions of interference items added. MegaFace dataset includes multiple application scenarios such as face verification, face training, and face confirmation.

MegaFace specifies that a training set below 0.5 MB is a small dataset, while a training set above 0.5 MB is a large dataset. However, the network proposed in this study is trained and evaluated under a small dataset. The random sample of MegaFace dataset is shown in Figure 10.

As can be seen from the experimental comparison in Table 3, the recognition rate of the algorithm proposed in this study is 2.10% higher than that of the Ouamane algorithm in the same test results. At the same time, the algorithm in this study exceeds the recognition rate of FaceNet in large-scale data, which is enough to show that the proposed algorithm has good robustness under unconstrained conditions. At the same time, the proposed algorithm has a faster recognition speed.



(a) (b) (c)

(d) (e) (f)

Figure 8: The random sample of the CASIA-WebFace dataset.

Table 1: Comparison of recognition rate on the CASIA-WebFace dataset.

Methods	Training set size/MB	Test of time/s	Recognition rate/%
FaceNet [8]	199.8	0.61	99.18
DeepFace [22]	3.99	0.29	97.27
DDML [23]	2.53	0.38	98.81
Ouamane [24]	0.68	0.52	98.99
DeepID2 [6]	0.27	0.32	98.87
Faster RCNN [15]	0.49	0.185	97.42
Proposed	0.49	0.049	99.52

(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 9: The random sample of the LFW dataset.

Table 2: Comparison of recognition rate on the LFW dataset.

Methods	Training set size/MB	Test of time/s	Recognition rate
FaceNet [8]	199.8	0.62	98.02
DeepFace [22]	3.99	0.29	92.29
DDML [23]	2.53	0.39	93.73
Ouamane [24]	0.68	0.58	94.99
DeepID2 [6]	0.27	0.35	94.82
Faster RCNN [15]	0.49	0.188	91.45
Proposed	0.49	0.051	98.19

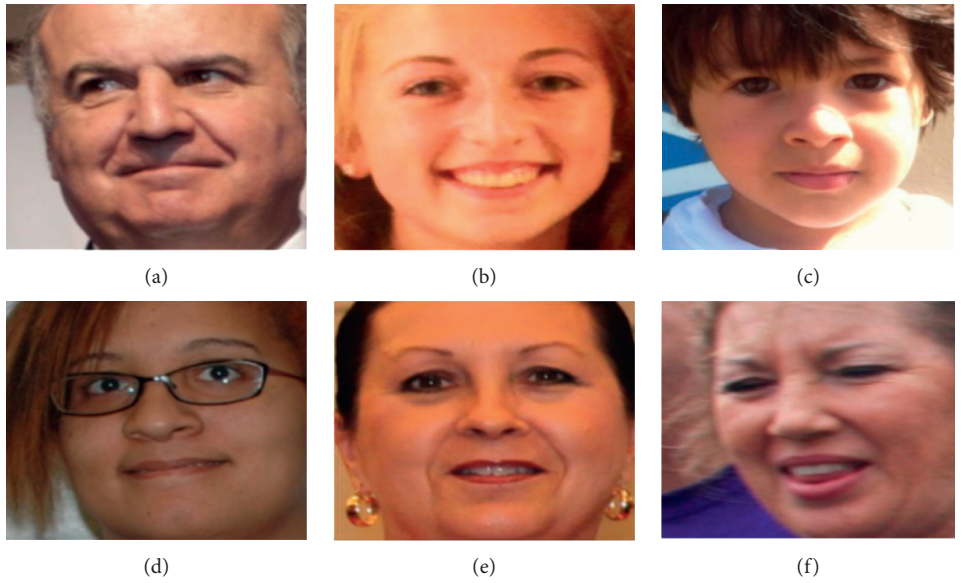


Figure 10: The random sample of the MegaFace dataset.

Table 3: Comparison of recognition rate on the MegaFace dataset.

Methods	Training set size/MB	Test of time/s	Recognition rate
FaceNet [8]	199.8	0.61	91.01
DeepFace [22]	3.99	0.29	86.25
DDML [23]	2.53	0.38	87.76
Ouamane [24]	0.68	0.57	90.10
DeepID2 [6]	0.27	0.34	88.85
Faster RCNN [15]	0.49	0.188	85.42
Proposed	0.49	0.050	92.20

5. Conclusions

In this study, the first transfer learning is completed from the large-scale dataset ImageNet to medium-scale restricted face image set, and the effective recognition of restricted face image set is realized. Then, the secondary transfer learning from the medium-scale restricted face image set to the small-scale unrestricted face image set is completed. The image enhancement methods, such as pose alignment, illumination brightness enhancement, and angle rotation, are applied to the unrestricted face image set to facilitate the smooth transfer learning. Experimental results show that the proposed algorithm has high identification accuracy and can meet the requirements of rapid detection and has the engineering application value for the field of terminal contactless distribution.

Data Availability

The source code of this algorithm cannot be provided directly because of the programmer's reason.

Conflicts of Interest

The author declares that there are no conflicts of interest.

References

[1] C. Cao and G. Wang, "Evaluation of intelligent speech technology in epidemic prevention: take iflytek input software in Chinese and Japanese recognition as an example," *Journal of Physics: Conference Series*, vol. 1631, no. 1, 2020.

[2] L. Hui-Ying and S. Yu-Guo, "Research on unconditioned face recognition based on residual network," *Computer Engineering & Software*, vol. 40, no. 11, pp. 143–147, 2019.

[3] S. V. Dharsini, B. Balaji, K. S. K. Hari et al., "Music recommendation system based on facial emotion recognition," *Journal of Computational and Theoretical Nanoscience*, vol. 17, no. 4, pp. 1662–1665, 2020.

[4] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: closing the gap to human-level performance in face

verification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1701–1708, Columbus, OH, USA, June 2014.

[5] Yi Sun, X. Wang, and X. Tang, "Deep learning face representation from predicting 10,000 classes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, USA, June 2014.

[6] Y. Chen, Y. Chen, X. Wang et al., "Deep Learning Face Representation by Joint Identification- verification," in *Proceedings of the International Conference On Neural Information Processing Systems*, MIT Press, Cambridge; MA, USA, June 2014.

[7] Yi Sun, X. Wang, and X. Tang, "Deeply learned face representations are sparse, selective, and robust," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, USA, June 2015.

[8] Y. Zhao, AiP. Yu, and D.T. Xu, "Person recognition based on FaceNet under simulated prosthetic vision," *Journal of Physics: Conference Series*, vol. 1437, no. 1, 2020.

[9] Q. Wang, G. Michau, and O. Fink, "Domain adaptive transfer learning for fault diagnosis," in *Proceedings of the 2019 Prognostics And System Health Management Conference*, Qingdao, China, October 2019.

[10] Y. Song, H. Wang, and X. He, "Adapting deep ranknet for personalized search," in *Proceedings of the 7th ACM International Conference on Web Search and Data Mining*, pp. 83–92, ACM, New York, NY, USA, February 2014.

[11] X. Cao, "A practical transfer learning algorithm for face verification," in *Proceedings of the IEEE International Conference on Computer Vision*, Sydney, Australia, April 2013.

[12] J. Hu, J. Lu, and Y.-P. Tan, "Deep transfer metric learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 325–333, Seattle, WA, USA, June 2015.

[13] A. G. Guo, A. H. Wang, A. Y. Yan et al., "A fast face detection method via convolutional neural network," *Neurocomputing*, vol. 395, pp. 128–137, 2020.

[14] F. Cen and G. Wang, "Boosting occluded image classification via subspace decomposition-based estimation of deep features," *IEEE Transactions on Cybernetics*, vol. 99, pp. 1–14, 2019.

[15] R. Girshick, "Fast R-CNN," *Computer Science*, vol. 1, 2015.

[16] T. Kaur and T. K. Gandhi, "Automated brain image classification based on VGG-16 and transfer learning," in *Proceedings of the 2019 International Conference On Information Technology (ICIT)*, IEEE, Bhubaneswar, India, August 2019.

[17] Y. Koga, H. Miyazaki, and R. Shibasaki, "A CNN-based method of vehicle detection from aerial images using hard example mining," *Remote Sensing*, vol. 10, no. 1, p. 124, 2018.

[18] W. U. Yaqin and W. Xiaodong, "Hybrid differential evolution *K*-means unsupervised clustering algorithm in big data mining," Chongqing University of Technology (Natural Science), Chongqing, China, 2019.

[19] W. Chen, L. Wang, Y. Zhang et al., "Anti-disturbance grabbing of underwater robot based on retinex image enhancement," in *Proceedings of the 2019 Chinese Automation Congress (CAC)*, IEEE, Hangzhou, China, February 2020.

[20] P. A. Deshmukh, "Optimal face retrieval from LFW dataset," *IJARCCE*, vol. 6, no. 3, pp. 452–455, 2017.

[21] Z. Wang, K. He, Y. Fu et al., "Multi-task deep neural network for joint face recognition and facial attribute prediction," in *Proceedings of the ACM on international Conference on multimedia retrieval*, pp. 365–374, ACM, Ottawa, ON, Canada, June 2017.

[22] Y. Taigman, M. Yang, M. Ranzato et al., "DeepFace: closing the gap to human-level performance in face verification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1701–1708, IEEE Computer Society, Columbus, DC, USA, June 2014.

[23] J. Hu, J. Lu, and Y. P. Tan, "Discriminative deep metric learning for face verification in the wild," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1875–1882, Washington, DC, USA, June 2014.

[24] A. Ouamane, "Side-information based exponential discriminant analysis for face verification in the wild," in *Proceedings of the 2015 11th IEEE international conference and workshops on automatic face and gesture recognition (FG)*, IEEE, Ljubljana, Slovenia, May 2015.