

Received March 10, 2021, accepted April 5, 2021, date of publication April 9, 2021, date of current version April 19, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3072055

# Joint Face Retrieval System Based On a New Quadruplet Network in Videos of Multi-Camera

GUOYIN REN<sup>1,2</sup>, XIAOQI LU<sup>1,3</sup>, AND YUHAO LI<sup>2</sup>

<sup>1</sup>School of Mechanical Engineering, Inner Mongolia University of Science and Technology, Baotou 014010, China

<sup>2</sup>School of Information Engineering, Inner Mongolia University of Science and Technology, Baotou 014010, China

<sup>3</sup>Inner Mongolia University of Technology, Hohhot 010051, China

Corresponding author: Xiaoqi Lu (lan\_tian1234@hotmail.com)

This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 61179019 and Grant 81571753, and in part by the Youth Innovation Talent Project of Baotou under Grant 0701011904.


**ABSTRACT** At present, a large number of off-line videos is stored in the server of surveillance network. In order to retrieve the target face in these massive videos frames, the face retrieval system is designed. A new Quadruplet Network is constructed by changing the RELU structure of CNN network and training the new Quadruplet Network to acquire the depth features. Join with the online fugitive face picture that launched online to initiate the wanted, with the help of the depth feature contrast to launch the Content-Based Image Retrieval (CBIR). The new Quadruplet Network converges faster than familiar networks such as Alexnet, Googlenet, VGGNet and ResNet. Because of the shared weight design of the network, the retrieval has a high precision, recall and the retrieval rate. Image depth features can be shared quickly online between the cameras. The experimental results show that the proposed method is effective, with an accuracy of 98.74% and a precision of 99.54%, and a frame rate of 28 FPS.

**INDEX TERMS** Quadruplet network, face retrieval, crossing camera, content-based image retrieval.

## I. INTRODUCTION

At present, there are frequent crimes in our society. It is an important technical means for the police to obtain the evidence images from the video surveillance cameras. There are video surveillance cameras everywhere in the streets. The coverage area of the video surveillance cameras is very wide. High definition video surveillance cameras are collecting data all the time, and a large number of offline videos are produced every day. Take the HD camera for example, it produces 50-60 frames per second. The camera can record about 5 million frames of video in 24 hours, so the video data recorded by one camera in one month is very large. If you add up all the camera data in that area, it's much more. For the criminal investigation department, it is obviously inefficient and painful to manually filter suspicious video frame by frame.

Face recognition is always an important technical problem in image recognition. In face recognition technology, the most important work is feature extraction [1]–[3]. Until 2013, face recognition is still in the stage of manual feature extraction. In this stage, we mainly rely on the image processing experts

The associate editor coordinating the review of this manuscript and approving it for publication was Mitra Mirhassani .

to analyze the face carefully, and finally get the face feature, that is, the local descriptor. Face recognition based on deep learning has gradually replaced the traditional artificial face feature extraction method, mainly because the face recognition based on deep learning has high efficiency and high accuracy, which can meet the needs of the criminal investigation department.

Image retrieval technology has been studied for nearly 20 years [4]–[6]. However, there are few content-based face image retrieval systems. The main bottleneck is the low retrieval accuracy and semantic gap. Face retrieval is the combination of face recognition and content-based image retrieval. It searches the target face by matching face features in the existing face image database [7]–[10]. In order to narrow the scope of criminal investigation, criminal investigators need to select a set of image sets that are most similar to the faces of the investigated person, so as to find all evidence associated with the suspect. At present, there are few reports about face retrieval in multi-camera videos, but the demand of criminal investigation department for this technology is becoming more and more urgent [11]–[13].

In recent years, convolutional neural network (CNN) has become an advanced technology in the field of computer vision. The first brilliant work was done by Lu *et al.* [14].

On the Imagenet image Dataset using the AlexNet network, which greatly improved the accuracy of image classification. Since then several CNN models, such as VGGNet, GoogLeNet and Resnet, have been shown to be more accurate in classifying Imagenet images [15]–[17]. Network design is getting deeper and deeper, such as 8 layers AlexNet, 19 layers VGGNet, 22 layers Googlenet and 152 layers ResNet. Scholars have improved the deep learning network and tried to apply it to the field of face feature extraction.

In 2014, Facebook developed DEEPFACE, the first face recognition method based on deep learning network. It performs well on LFW data set, and the recognition accuracy of the network is close to human eyes. The principle of deepface is to complete 3D face alignment by matching the key points of face. DEEPFACE uses Siamese network to extract facial features, which is composed of 5 convolution layers and 2 fully connected layers [18].

In 2015, a research group of the University of Hong Kong proposed a deep recognition face network DEEPID, which trains the face classification network with a large amount of face data to classify everyone's identity. Effective face feature extraction is the premise of identity classification. By training different face regions, the local feature vectors of multiple face regions are obtained, and these local feature vectors are measured by Bayes distance. Finally, the recognition accuracy of LFW data set exceeds the recognition ability of human eyes [19].

In 2016, a team of researchers from the University of Hong Kong, represented by Sun Y., refined the DEEPID method and proposed a new generation DEEPID2 method, which further increases the inter class distance and reduces the intra class distance by adding the supervised signal of the characteristic distance measure to extract better facial features [20]. The team then came up with the DEEPID2+ model, which is based on DEEPID2, in which features are extracted from each convolution layer and monitored using classified and validated signals. By adding supervisory signals at each level, the whole network can be well trained. DEEPID3 [21], relative to DEEPID2+, has a deeper network layer, which makes the face feature deeper and the face recognition effect better. The paper and method of DEEPID series are very important and instructive in deep learning face recognition.

Then Google proposed the FACENET face recognition method in 2015. The FACENET method uses the network structure of GOOGLNET [22]. It is based on the 22 layers INCEPTION model. When the model is trained on LFW (a data set of 200 million faces from 12000 people), the recognition rate is as high as 99.63%. Instead of using supervised classification signals, FACENET uses Triplet Loss [23], which is very different from DEEPID network. Triplet Loss has three input samples, anchor, positive and negative. The distance between the anchor sample and the positive sample is greater than the distance between the anchor sample and the negative sample, so as to train a more effective face feature.

At present, many new deep learning models have been proposed by scholars. Therefore, deep learning-based face

feature extraction has gradually become the mainstream of face recognition and face retrieval. The three breakthrough models of face recognition have many similarities. ① Both Deepface and DEEPID2 use Siamese network, while FACENET uses Triplet network to achieve the highest accuracy in face recognition. ② Both Siamese and Triplet network are metric learning methods. ③ The measurement network based on metric learning is an innovative point worthy of further research and improvement.

Based on the above three problems, this paper proposes a joint face retrieval network based on a new Quadruplet network in videos of multi-camera. A new Quadruplet network is used to learn face depth features from multiple depth network branches, so feature similarity and mutual dissimilarity learning are used to train face features with stronger recognition ability. The characteristics of Quadruplet network are explored, and the results show that the pre-processing methods, such as Batch OHNM, Subspace clustering and Focal Loss, can improve the sample difficulty of quaternion input, the efficiency of finding difficult samples and the balance of positive and negative samples.

In this paper, we used 20 kinds of loss functions to train and evaluate the network in LFW face data set, and explored the most effective set of loss functions through comparative experiments. The effectiveness of each variable in the loss function was evaluated by ablation experiment. Through the design of data channel, the feature synchronization between different cameras was completed. Finally, the system was tested on 6 face retrieval data sets, and a cross camera joint face retrieval system was implemented.

## II. RELATED WORKS

Zhang M., Zhao Y., Sardogan M. *et al.* proposed methods used in image classification based on CNN, and the classification accuracy was improved [24]–[26]. On this basis, Ge W. *et al.* used Siamese Network as shown in Figure 1-(a) to input pairs of the same face images (positive sample pairs) or pairs of different faces (negative sample pairs) for classification training. The above two sample pairs completed the training and minimize the similarity loss function, which minimized the distance between patches of the same class and maximized the distance between patches of different classes, as shown in Figure 1-(a). Although these data set pretreatment methods represented significant improvements over the previous common baseline data sets of PaSC, LFW, PubFig, FERET, AR and YaleB, more researchers had attempted to improve the loss function and network structure to explore the possibility of improvement. For example, Triplet Loss was a widely used measure of learning loss, which improved Siamese network performance by adding a number of network input branches, as shown in Figure 1-(b) [27]. Triplet networks were trained to use Triplet Loss, which required three input images. Unlike Contrast Loss, an input Triplet Loss consisted of a pair of positive and a negative sample. The three images were named as Anchor, Positive and Negative images respectively. Picture *a* and picture *p* were a pair of

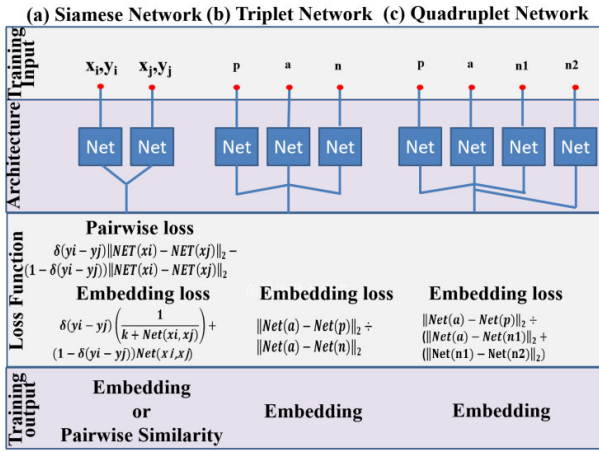


FIGURE 1. Comparison of loss function types and network structure in different model training methods.

positive sample pairs, picture  $a$  and  $n$  were a pair of negative sample pairs. Triplet Loss could shorten the distance between positive sample pairs and pushed away the distance between negative sample pairs. Finally, the same face images were clustered in the feature space. By analogy, Triplet Loss function considered only the relative distance between positive and negative sample pairs, but not the absolute distance between positive sample pairs. Chen W., Chen X. and Zhang J. *et al.* used Quadruplet Network and a new loss function to train the local image descriptor learning model, which could be applied to Siamese and Triplet Network [28], as shown in Figure 1-(c). The Quadruplet Network Loss function produced a feature embedding that minimized the distance variance between Positive images that belonged to the same class and minimized the average distance between sample images that belonged to the same class and maximized the average distance between images of different classes. The Quadruplet Network Loss function added the absolute distance of negative samples to ensure that the Quadruplet Network not only pushed the positive and negative samples away in the feature space, but also kept the positive samples or negative samples close to each other, as show in Figure 1-(c).

TriHard was an improved version of Triplet Loss. Traditional Triplet Loss was to randomly extract three images from training data as input. This method was simple, but most of the random input samples were easy to identify. If a large number of training sample pairs were simple sample pairs, then this was not conducive to network learning a better characterization. A large number of papers had found that using more difficult samples to train the network could improve the network generalization ability, and there were many ways to sample more difficult sample pairs. In this paper, an on-line hard sample sampling method called as Double TriHard Loss based on training batch was proposed. As show in Figure 2-(d), the Euclidean distance of each picture were calculated in TriHard Loss and batch were calculated in the feature space, then selecting the hard positive sample with the furthest (least like) distance and the hard negative sample

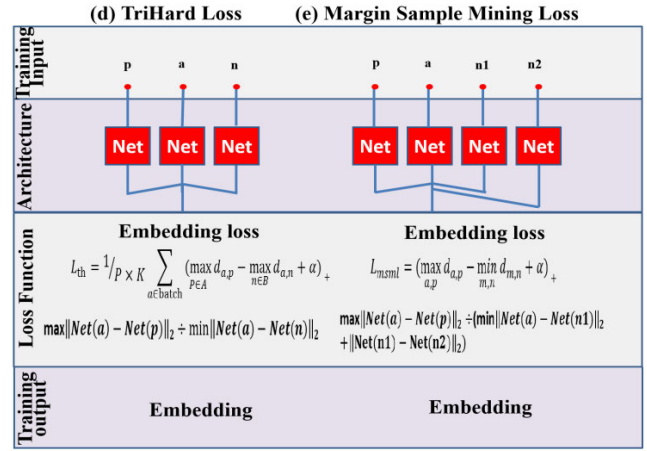


FIGURE 2. Comparison of loss function types and network structure in different hard sample model training methods.

with the nearest (most like) distance to calculated the Triplet Loss [29].

TriHard Loss function were generally more effective than traditional Triplet Loss function. Margin Sample Mining Loss (MSML) was a metric learning method which introduced the idea of hard sample sampling into four input network [30]. Triplet Loss function only considered the relative distance between positive and negative sample pairs. In summary, the TriHard Loss selected a hard Triplet for each image in the batches, while the MSML Loss function selected only the most difficult positive sample pair and the most difficult negative sample pair to form a four tuple input to calculate the loss, as show in Figure 2-(e). So MSML was more difficult to sample than TriHard, and it could be regarded as the upper bound of the distance between positive sample pairs and the lower bound of negative sample pairs. MSML was to push the boundaries of positive and negative sample pairs apart, hence the name Margin Sample Mining Loss. Generally speaking, MSML was a metric learning method which considers both relative distance and absolute distance and introduces the idea of hard sample sampling.

### III. METHODOLOGY

#### A. MARGIN SAMPLE MINING LOSS (MSML)

MSML is a metric learning method which introduces the idea of hard sample sampling.

Triplet Loss only considers the relative distance between positive and negative sample pairs. In order to introduce the absolute distance between positive and negative sample pairs, adding a difficult negative sample to the TriHard Loss constitutes a Quadruplet Loss, and MSML is a form of the Quadruplet Loss. The MSML is also defined as formula (1):

$$L_q = (d_{a,p} - d_{a,n_1} + \alpha)_+ + (d_{a,p} - d_{n_1,n_2} + \beta)_+ \quad (1)$$

If we ignore the effect of the parameter  $\alpha$  and  $\beta$ , we can express the Quadruplet Loss in a more general form formula (2):

$$L'_q = (d_{a,p} - d_{m,n} + \alpha)_+ \quad (2)$$

where  $m$  and  $n$  are a pair of negative sample pairs,  $m$  and  $a$  can be either a pair of positive or negative sample pairs. Then we introduce the hard sample mining idea of TriHard Loss and defined as formula (3):

$$L_{msml} = \left( \max_{a,p} d_{a,p} - \min_{m,n} d_{m,n} + \alpha \right)_+ \quad (3)$$

where  $a, p, m, n$  are all the pictures in the batches,  $a, p$  are the least like positive hard sample pairs in the batches,  $m, n$  are the most like negative hard sample pairs in the batches,  $p, m$  are the most like negative hard sample pairs in the batches. In summary, the TriHard Loss selects a hard Triplet for each image in the batches, while the MSML Loss selects only the most hard positive sample pair and the most hard negative sample pair to calculate the loss. So MSML is more difficult to sample than TriHard. In addition,  $\max_{a,p} d_{a,p}$  can be regarded as the upper bound of the distance between positive sample pairs and  $\min_{m,n} d_{m,n}$  can be regarded as the lower bound of negative sample pairs.

### B. MSML TRAINING STRATEGIES

Online Hard Example Mining(OHEM) is a key technique for training pre-processing strategy, and MSML includes three training pre-processing strategies.

#### 1) MSML with OHEM

$$\sum_{i=1}^{|B|} \left[ \|f(x_i^a) - f(x_i^p)\|^2 - \|f(x_i^a) - f(x_i^n)\|^2 + \alpha \right]_+ \quad (4)$$

$\forall x_i^a, x_i^p, x_i^n \in T$

$$\sum_{i=1}^{|B|} \left[ \|f(x_i^{n1}) - f(x_i^{n2})\|^2 - \|f(x_i^{n1}) - f(x_i^a)\|^2 + \beta \right]_+ \quad (5)$$

$\forall x_i^{n1}, x_i^{n2}, x_i^a \in T$

In the formula (4) and formula (5),  $T$  represents the total sample spaces, but most methods still select the quaternion directly from the total samples, which make it difficult to get the required hard mining, because it does not sufficiently reduce the fuzzy edge area between positive samples and negative samples, this fuzzy area is the confusion area of positive and negative samples, and also the hiding place of hard samples.

Pink is a hard positive, red is an anchor, light green is a hard negative1, and dark green is a hard negative2, as show in Figure 3. All positive and negative sample pairs are selected from the whole sample. It can be seen that most negative sample pairs are far away from positive sample pairs, even though such random negative sample pairs are assigned to a batch, the training strategy fails to achieve the desired goal.

#### 2) MSML with Batch OHNM

The calculation process of Online Hard Negative Mining (OHNM) is defined as formula (6) and formula (7):

$$\sum_{i=1}^{|B|} \left[ \|f(x_i^a) - f(x_i^p)\|^2 - \|f(x_i^a) - f(x_i^n)\|^2 + \alpha \right]_+ \quad (6)$$

$s.t. x_i^n = \arg \min_x \|f(x_i^a) - f(x)\|^2, I(x_i^a) \neq I(x)$

$\forall x_i^a, x_i^p \in T, x \in B$

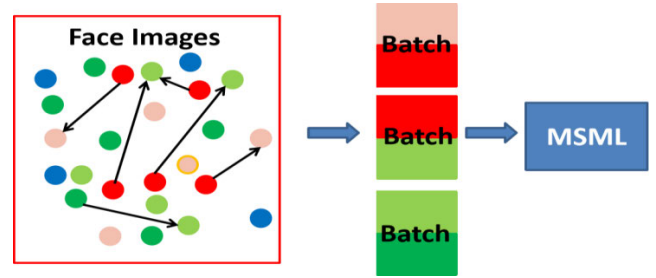


FIGURE 3. Select negative and positive samples from the whole sample.

$$\sum_{i=1}^{|B|} \left[ \|f(x_i^{n1}) - f(x_i^{n2})\|^2 - \|f(x_i^{n1}) - f(x_i^a)\|^2 + \beta \right]_+ \quad (7)$$

$s.t. x_i^a = \arg \min_x \|f(x_i^{n1}) - f(x)\|^2, I(x_i^{n1}) \neq I(x)$

$\forall x_i^{n1}, x_i^{n2} \in T, x \in B$

This method is used to mine as many hard samples as possible in two batches, one batch paired from a hard positive sample and the other batch paired from a hard negative sample. In this case, the negative pair is selected not from the whole sample but from the negative sample, and the positive pair is selected from the positive sample. Compared with selecting matching pairs in the whole sample, negative samples and positive samples are selected in the corresponding sample batches for training. This reduces the scope of the search and may increase the probability of finding hard samples.

As show in Figure 4, it can be seen that both positive and negative sample batches, and then the similarity distance between positive and negative batch pairs is calculated by neural network, and then a quaternion group is formed according to the threshold value of the hard sample pairs. Finally, the network optimization is completed by the loss function.

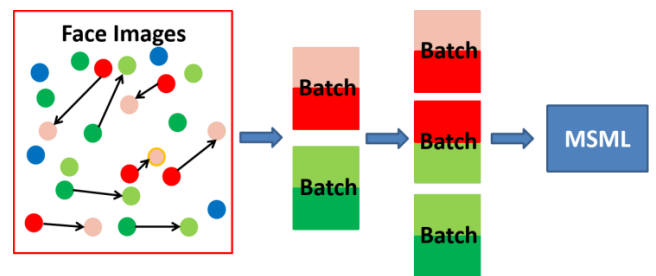


FIGURE 4. Selects negative and positive samples with Batch OHNM.

It is worth noting that the positive and negative sample balance function is used in selecting the positive sample batches and the negative sample batches to prevent the problem of serious imbalance in the proportion of positive and negative samples, as well as in selecting the matching distance. Do not select the closest negative batch pairs as a hard sample, which may lead to poor training. Do not select the closest negative batch pairs. Because the most similar three graph is not conducive to training convergence, because all samples are using

the mold and the class training, the existence of ambiguity of this class easy to make the network into a model collapse.

### 3) MSML with Subspace Clustering

The calculation process of Subspace Clustering is defined as formula (8) and formula (9):

$$\sum_{i=1}^{|B|} \left[ \|f(x_i^a) - f(x_i^p)\|^2 - \|f(x_i^a) - f(x_i^n)\|^2 + \alpha \right]_+ \quad (8)$$

$$s.t. x_i^n = \arg \min_x \|f(x_i^a) - f(x)\|^2, I(x_i^a) \neq I(x)$$

$$\forall x \in B, x_i^a, x_i^p \in T_m, m = 1, \dots, M$$

$$\sum_{i=1}^{|B|} \left[ \|f(x_i^{n1}) - f(x_i^{n2})\|^2 - \|f(x_i^{n1}) - f(x_i^a)\|^2 + \beta \right]_+ \quad (9)$$

$$s.t. x_i^a = \arg \min_x \|f(x_i^{n1}) - f(x)\|^2, I(x_i^{n1}) \neq I(x)$$

$$\forall x \in B, x_i^{n1}, x_i^{n2} \in T_m, m = 1, \dots, M$$

Sun *et al.* [20] consider that it is not appropriate to select two pairs of samples from the batch set of positive or negative samples to form the quaternion batches, and propose to use OHNM to optimize the batches by clustering. Because this positive and negative batches are also selected from the whole batches, it is difficult to match two faces with similar faces when there are too many faces. This also greatly reduces the difficulty of obtaining OHNM samples. A simple idea is to cluster the faces of the same person. Finally, samples that are difficult to distinguish or very similar are found in each cluster space, As show in Figure 5. Although feature extraction and clustering are time-consuming, the search speed in clustering space is greatly improved.

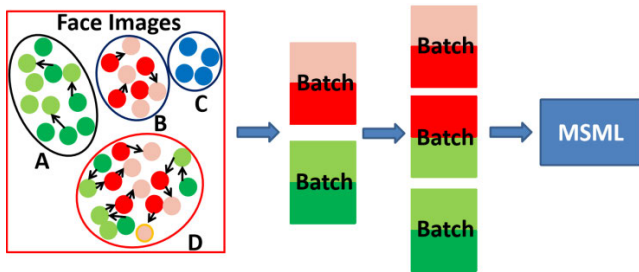


FIGURE 5. Selects negative and positive samples from cluster subspace.

The number of clustering subspaces will determine the success rate of hard sample extraction. If the number of subspaces is too small, it means that there are few clusters, and each cluster contains too many face samples. The partition of this subspace is very similar to the method without clustering, and a large number of dissimilar faces are clustered into a class. If the number of subspaces is too many, the same face is divided into other subclasses, which increases the difficulty of sample mining.

Comparing the above three methods, we can see that method 2 reduces the search space of method 1, and method 3 also reduces the search space of method 2. That further narrows the search space issue. OHNM method is selected from the set of positive samples or negative samples, so the

probability that the hard positive sample pair and hard negative sample pair are selected as quaternion is very small. The Subspace Clustering method is used to cluster all the face samples, and the similar faces are grouped into one group. It is easy to form a quaternion in clustering spaces, and it is more efficient to mine them.

### C. FOCAL LOSS

As mentioned above, subspace clustering can reduce the search range of quaternion samples and improve the generation efficiency of quaternion samples. However, the problem of four tuple hard samples has not been solved only by subspace clustering. As can be seen from Figure 6, face clusters with similar features are formed through subspace clustering. These clusters can be divided into two categories, one is the same person's features constitute a face cluster, and the other is obviously not a person's features, but because they look very similar, they are divided into a person's face cluster. These confusing facial features are the source of hard sample mining. We know that if we select simple samples instead of hard samples as the input of the quaternion, it will not be conducive to the generalization ability of the model, and it will not be able to distinguish easily confused faces. Therefore, we use the Focal Loss function as the screening tool for hard samples. The Focal Loss function can retain the hard sample cluster and eliminate the simple sample cluster. As shown in Figure 6, the Focal Loss function can retain the cluster D and eliminate the clusters A, B and C. At the same time, the Focal Loss function can keep the proportion balance of the positive and negative sample pairs in the D cluster, and there will be no over fitting problem caused by the sample imbalance. The specific principle of Focal Loss function is explained in formula (10) and formula (11) below.

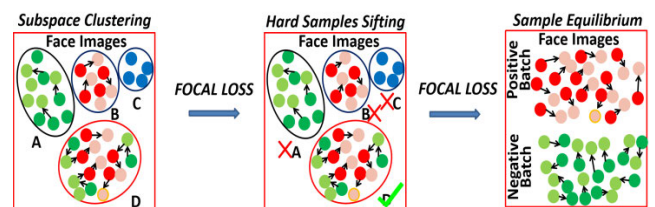


FIGURE 6. Balance of hard negative samples and hard positive samples by Focal Loss.

Focal Loss is intended to solve the problem of a serious imbalance between positive and negative sample proportions [31]. The loss function reduces the weight of a large number of simple negative samples in training, which can also be understood as a hard sample mining.

Focal Loss is a modification of the cross-entropy loss function, starting with a review of the two-category cross-loss defined as formula (10):

$$L = \begin{cases} -\log y', & y = 1 \\ -\log(1 - y'), & y = 0 \end{cases} \quad (10)$$

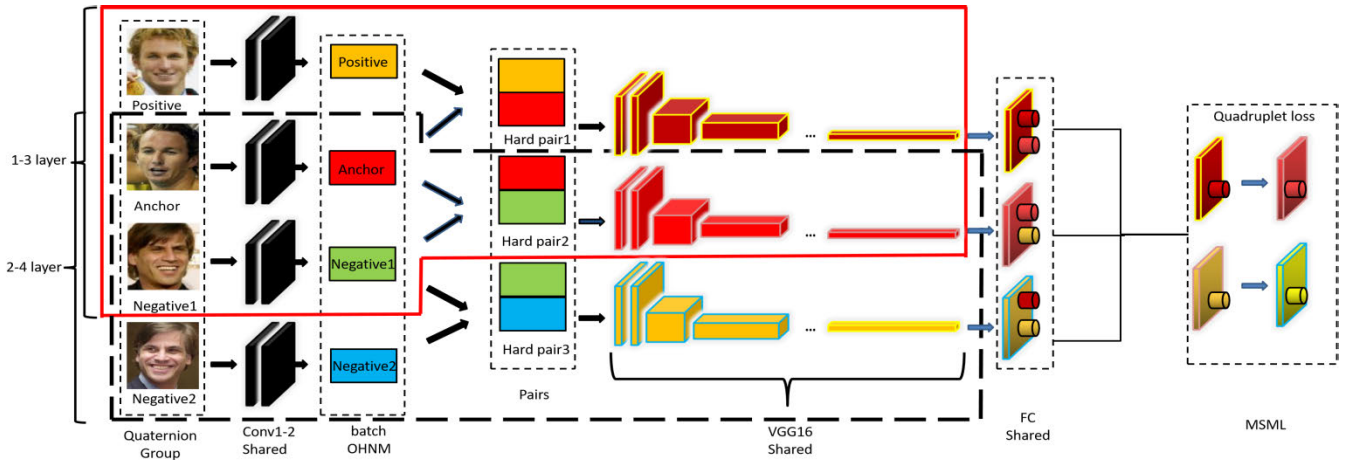


FIGURE 7. Improved MSML model structure diagram.

where  $y'$  is the output of the activation function, so it is between 0 and 1. For positive samples, the larger the output probability, the smaller the loss. For negative samples, the smaller the output probability, the smaller the loss. When  $y^0$  is smaller, the cross-entropy loss function is smaller and the convergence of the positive sample is slower, so it is not suitable to select the hard sample, and the loss function has no parameter to control the proportion of the positive and negative samples, it defined as formula (11).

$$L_{fl} = \begin{cases} -\alpha(1 - y')^\gamma \log y', & y = 1 \\ -(1 - \alpha)y'^\gamma \log(1 - y'), & y = 0 \end{cases} \quad (11)$$

Focal Loss improves the cross-entropy loss function by adding two parameters,  $\alpha$  and  $\gamma$ .  $\alpha$  is the positive and negative sample balance control parameter,  $\gamma$  is the hard sample control parameter.

If  $\alpha$  is constant, where  $\gamma > 0$ , then for a positive sample, if it is close to 1, then the selected sample is a simple sample, so the value of  $-\alpha(1 - y')^\gamma$  will be small, and then the value of the loss function will be smaller. If  $y'$  is close to 0, the sample is hard, the loss is relatively large and the convergence speed is fast. For negative samples, it will choose the most hard sample, that is, if  $y'$  is close to 0.

If  $\gamma$  is constant, where  $\alpha > 0$ , for the positive and negative sample equilibrium control parameter  $\alpha$ , used to balance the proportion of positive and negative sample itself is uneven, the larger the proportion of positive sample  $\alpha$ , the fewer the negative sample. Conversely, the smaller the  $\alpha$ , the smaller the positive sample ratio, and the more negative samples.

#### IV. PROPOSED METHOD

##### A. NEW QUADRUPLER NETWORK

The approach described in this article is based on Margin Sample Mining Loss (MSML).

(1) Network structure: The network used is shown in Figure 7. The images are still a positive, anchor, negative1, negative2 sample. The image features are extracted through

two convolution layers to form three pairs of hard samples. Three pairs of hard samples were trained by the three-layer VGG16 Network to complete the sample classification.

(2) MSML improvement: As can be seen from Figure 7, the four-layer network input can be viewed as two Triplet networks in parallel, the red dotted line part is a Triplet network, the black dotted line part is also a Triplet network. One of the innovations of this article is the improvement of MSML from this perspective.

DTN achieves different goals, that is, the positive sample area becomes larger and the negative sample area becomes larger, until the easily confused area disappears.

The MSML is to input three hard sample pairs, the hard positive sample(the two least-like positive sample pairs) in the first and second layers, and hard positive-negative sample pairs (the two most-like positive-negative sample pairs) in the second and third layers, and hard negative sample pairs (the two most-like negative sample pairs) in the third and fourth layers. The two negative samples use the two most similar faces of different people. In this way, the upper bound of the distance of the positive sample and the lower bound of the negative sample are taken into account, while the relative distance and the absolute distance are considered simultaneously. However, the hard-negative samples are not sufficiently distinguished from the positive-negative samples, and only the most similar spatial distance diversity of the positive-negative samples is considered, but the distance between the most difficult positive-negative samples is not sufficiently separated.

In this part, VGG16 is used to extract facial features in a deeper branch layer. Besides, MSML will be improved from two aspects: the difficulty of selecting negative sample pair is increased. The third layer negative samples select the samples that are most similar to the second layer positive samples, while the fourth layer negative samples select the face samples that are least similar to the third layer negative samples. Thus, The improved MSML model regards as the Double Triplet Networks (DTN). The first three layers

(1-3 layers) constitute two positive samples and one negative sample face Triplet network  $T_1$ , and the last three layers (2-4 layers) constitute two negative samples and one positive sample face Triplet network  $T_2$ . That is, two parallel Triplet networks, as shown in Figure 7.

Each Triplet Networks of DTN achieves different goals, while the  $T_1$ (1-3 layers) and  $T_2$ (2-4 layers) calculate the positive sample distance and the negative sample distance by the TriHard Loss. The design prevents the same human face from being misclassified, instead of being mistaken for other human face. The function of  $T_2$  is to calculate the distance between the negative samples with the farthest TriHard Loss aggregation distance (the most unlikely) and push away the distance from the negative samples with the nearest TriHard Loss aggregation distance (the most likely). This will completely separate the negative sample from the positive sample. This is equivalent to using TriHard twice at the same time. The same kind of sample in the margin of the mold section is fully separated.

### B. MSML WITH SUBSPACE CLUSTERING AND FOCAL LOSS

It is easy to understand that the positive sample batches (from same person's face) and the negative sample batches (from another person's face) are clustered in a subspace, but there may be three unsolvable cases: 1) these samples are likely to be unbalanced between positive and negative samples. 2) the difficulty of selecting positive samples cannot be controlled during the batch process. 3) when the positive sample batches match the negative sample batches, the prediction difficulty of selecting the sample will affect the accuracy of the training results.

Using Focal Loss based on MSML with Subspace Clustering solves all three of these problems. Firstly, the proportion of positive and negative samples can be controlled by adjusting  $\alpha$ . Secondly, control  $y'$  is close to 0, the sample is difficult to sample its loss is relatively large convergence speed is faster. Finally, the sample was chosen to predict the case with difficulty of  $\gamma = 0.9$  in order to get the conclusion through formula (11). The network structure of Figure 7 is a quaternion input, which consists of two pairs of person faces, each pair is from the same person face, and each pair is from a different person face. Our design of the network includes two Triplets, two pairs of face samples are difficult to choose samples that do not look like a person's face. The prediction difficulty is  $\gamma = 0.9$ . In this case, Focal Loss is appropriate.

Therefore, using MSML with Subspace Clustering and Focal Loss can solve the following problems. The Subspace Clustering method can cluster all samples, and the similar faces are grouped into one subspace, which is more efficient. And Focal Loss for all batches in a subspace, those with bad predictions retain most of the cross-entropy losses, and for those with good scores, the cross-entropy losses are substantially reduced. The Focal Loss reduces the sample loss of the hard sample, which has a better relative fraction. At the same time, the model will focus on optimizing the loss of less samples, and the training weight will be increased.

## V. SYSTEM DESIGN

### A. MULTI-CAMERA NETWORK DESIGN

In reality, a face target may appear in several video surveillance areas. If you want to retrieve all the face frames from the surveillance videos in these areas, you need to build a surveillance network with data exchange.

Therefore, it is necessary to realize the interconnection of multi-camera information and data sharing. Multi-camera cooperation can not only increase the monitoring range but also capture the face information from different angles. It is possible for Intelligent Video Surveillance (IVS) to complete feature exchange, if multiple data channels are connected between cameras and features center server for synchronizing face feature information. If facial features can be exchanged between cameras, the similarity of facial features between surveillance cameras can be matched by DTN, so as to realize face retrieval of multiple cameras in smart city.

FTP has three communication modes to realize data synchronization: Pull, Push and Share. Push synchronization: the local data to the remote feature center. Pull synchronization: the remote features to the local. When the feature center receives the features from the local camera, all the features from the local camera are packaged and retransmitted to the local through the data Share synchronization, so as to achieve the purpose of keeping the real-time synchronization with the local features, as shown in Figure 8.

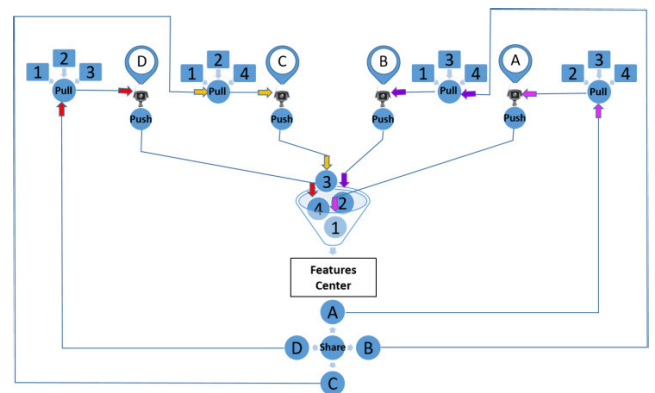


FIGURE 8. Diagram of smart city data channels.

Figure 8 is a schematic diagram of data channels, and combining with Figure 8, people can quickly understand the data channels principle in Figure 9.

In order to more clearly describe the principles of the above three synchronization methods, a 2.5D virtual smart city scene is set here to illustrate the above problems, as shown in Figure 9. There are five main buildings in the virtual scene, which are composed of one features center and four common buildings. In an ideal way, we assume that there is no security camera in the features center, and each of the other four ordinary buildings has a surveillance camera. From Figure 9, we can see that each surveillance camera gives its number with positioning marks, which are A, B, C and D. We can also see that there is just a pedestrian passing by within the camera



FIGURE 9. Data channels stereogram of smart city data channels.

field of view of each building, and the four photographed pedestrians also give their number with positioning marks. Among them, camera A captured pedestrian 1, camera B captured pedestrian 2, camera C captured pedestrian 3, and camera D captured pedestrian 4. When the facial features of pedestrian 1 are captured by camera A, it is first synchronized to the features center through Push. Similarly, the facial features of pedestrian 2, 3 and 4 are respectively pushed to the features center by cameras B, C and D at the first time. After the features center receives these facial features from different cameras, the first thing to do is to package these facial features and send them to each camera at the first time to complete a Share synchronization. After receiving the Share command from the features center, each camera will start Pull synchronization and store it in its own local features library. After completing a synchronization operation of Push, Share and Pull, the features library of each camera will be consistent. The features in the local features library are used to match the captured face features. Therefore, when pedestrian 1 passes through the field of view of B, C and D cameras, pedestrian 1 can be identified and retrieved by the surveillance cameras in each area. Similarly, pedestrians 2 and 3 can also be identified and retrieved by each camera.

Figure 10 is the system architecture diagram, which is divided into three main layers: the first layer is the front-end data acquisition layer, which is used to capture the videos of surveillance camera and storage in disk. The second layer is the GPU features computing layer, which can transform the retrieved image into the deep features of the image for image feature matching. The third layer is the features exchange and storage layer, which shares the facial features of the retrieval target in each region to complete the joint face retrieval.

As shown in Figure 10, three PC in the data acquisition layer initiate a joint face retrieval task for local videos, respectively, the feature extraction layer of the second layer extracts the depth features of the three face retrieval images and pushes them to the feature exchange layer. In the third

layer, the features are shared to the search buffers of the three video surveillance regions by the shared algorithm for the joint retrieval. As you can see from Figure 10, the features of each surveillance area are exactly the same after the data Sharing synchronization and Pull synchronization, with the help of DTN in this paper, face matching and joint retrieval can be completed in different video surveillance areas, and then all the face frames of each person in different areas can be found.

## B. JOINT FACE RETRIEVAL

As show in Figure 11, it is the overall functional structure of the joint face retrieval, from the diagram, we can see that the system is divided into four parts:

### 1. TEST DATA SOURCE

Take LFW test data set and three different video surveillance areas of local videos as the data source, each video is divided into image frames, which are used as the input of DTN network.

### 2. DTN TRAINING

#### ① Face training dataset LFW

Labeled Faces in the Wild LFW is a common test set of face recognition, which provides images of faces from natural scenes in life, so the recognition will be more difficult, especially because of multi-pose, light, expression, age, occlusion and other factors, even the same person's photos are very different. And some photos may have more than one face, for these multi-face images only choose the central coordinates of the face as the target, other areas as background interference. Now, the performance evaluation of LFW database has become an important index of face recognition algorithm.

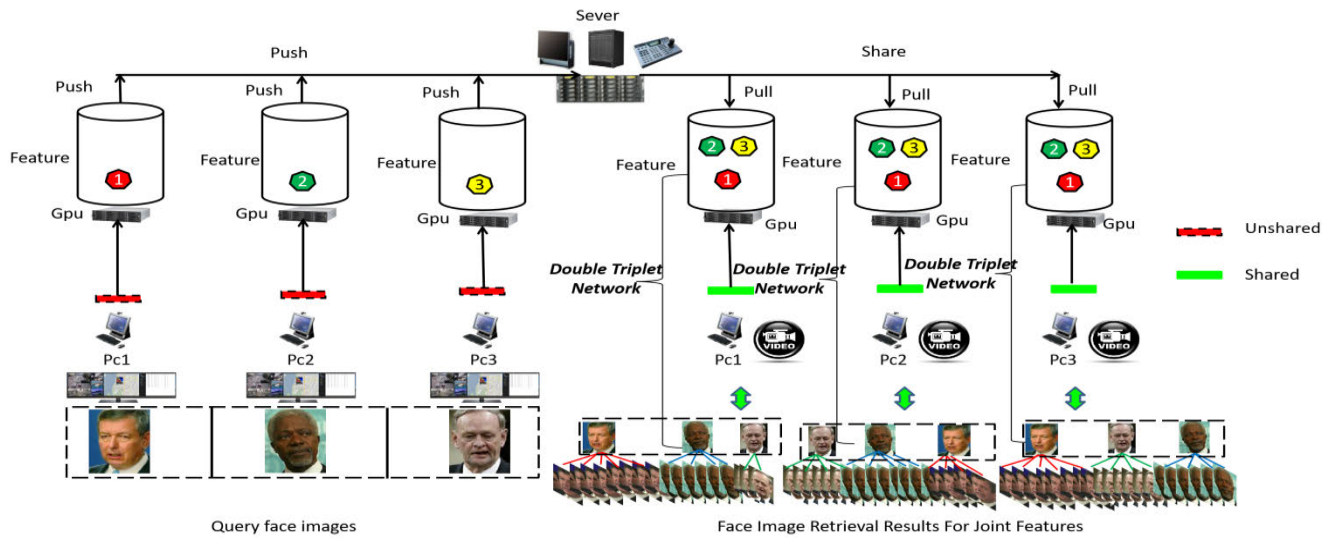
#### ② Image pre-processing

A. Subspace clustering in the online difficult sample sampling method based on training Batch, the Subspace clustering method is used to cluster all the samples of LFW data set, and the similar faces are all clustered into one subspace. It will be easier and more efficient to implement training-based Batch selection in local clustering space.

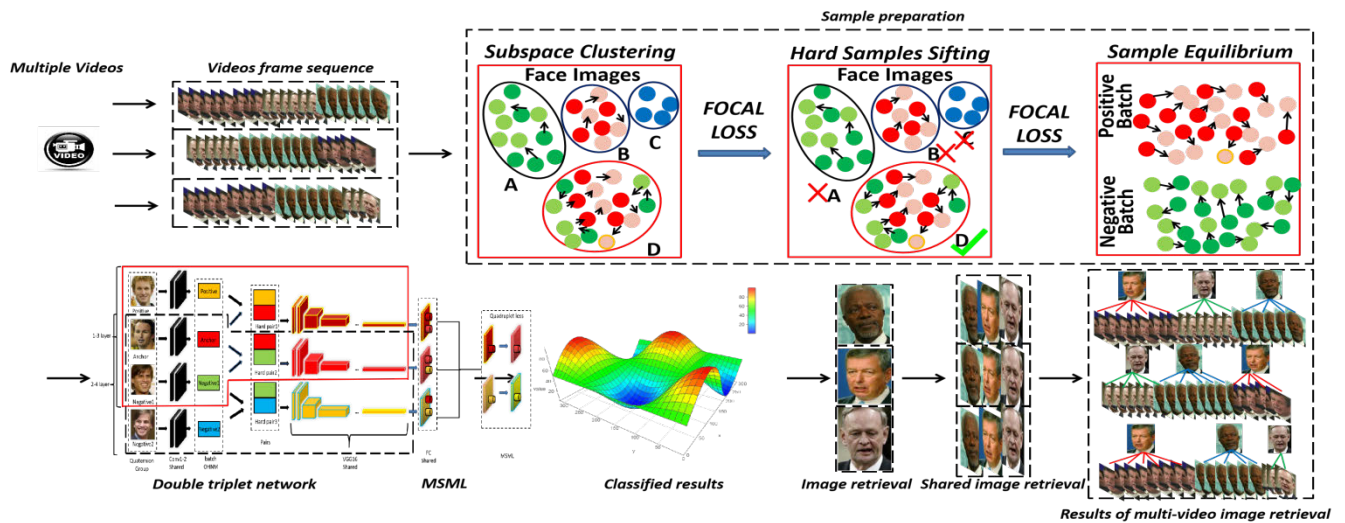
B. Easy sample and hard sample are included in each similar face subspace. Because OHNM is an online screening sample, all subspace containing easy sample should be eliminated. This article uses hard sample sifting of Focal Loss to eliminate easy sample, leave the hard sample subspace, from which the quaternion of the network input for this article are selected. This Quadruplet includes two difficult samples, one batch is a hard positive sample and the other batch is a hard negative sample. In order to prevent leading to poor training, hard negative sample would not select the closest positive batch as the difficult sample. The coefficients are shown in formula (8).

C. The sample equation can select two positive samples and two negative samples in the same subspace, and the Focal Loss can balance the positive and negative samples, which can solve the imbalance of positive and negative samples.





**FIGURE 10.** The retrieval image used in this paper uses one image for each retrieval area, and the three areas take three face images as an example. In fact, each video area can choose more than one image, each region can also select multiple videos to be processed at the same time, and this method is suitable for all the above situations.



**FIGURE 11.** The overall functional structure of the joint face retrieval.

### 3. DTN TRAINING USING MSML

DTN loads the trained weights, and inputs each video frame into DTN through sample pre-processing to match and retrieve all faces.

### 4. COMBINED WITH FACE IMAGE RETRIEVAL

Each video surveillance area transforms the image to be retrieved into a deep feature of the image, which is used for image feature matching, after the feature exchange and the face features of the storage layer are shared to other video surveillance areas, the joint face retrieval is completed.

## VI. EXPERMENTS AND RESULT ANALYSIS

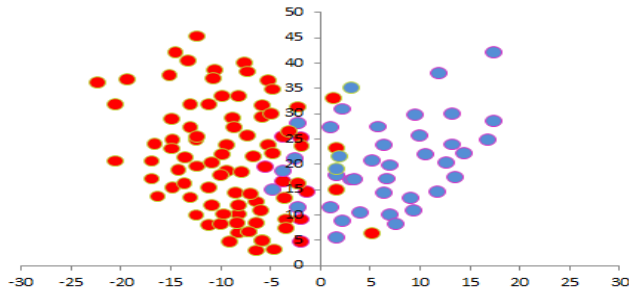
### A. THE FUNCTION OF IMPROVED LOSS

All the positive and negative samples are included in the online batch subspace processed through subspace clustering processes. There are a lot of indistinguishable samples in

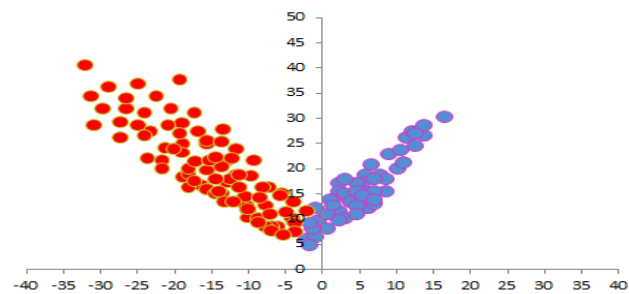
these samples, and they're both positive and negative because the faces are so similar. As shown in Figure 12, the red ball is a negative sample, and the blue ball is a positive sample. It can be seen that the positive and negative samples have some fuzzy areas mixed together, and the number of positive samples is more than that of negative samples.

DTN trained by Focal Loss is very effective in the hard sample sifting process, In Formula 11,  $\gamma = 0.9, y' = 0.1$ , with positive samples getting closer and negative samples getting closer, as shown in Figure 13.

DTN trained by Focal Loss is also effective in balancing positive and negative samples, In Formula 11,  $\alpha = 0.5, \gamma = 0.9, y' = 0.1$  increasing positive samples, decreasing negative samples and balance the proportion of positive and negative samples. At the same time, the mixing degree of positive and negative samples is decreasing, and the distance



**FIGURE 12.** The positive and negative samples have a part of the fuzzy area.



**FIGURE 13.** DTN with subspace clustering makes negative samples and positive samples from discrete to aggregation.

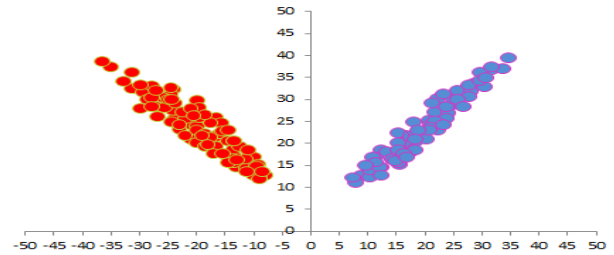
between positive and negative samples is gradually widening, as shown in Figure 14.

## B. COMPARATIVE ANALYSIS OF VALIDITY OF DTN LOSS FUNCTION

**Data Set:** The LFW data set mainly tests the accuracy of face recognition. The database randomly selected 6,000 pairs of faces to form face recognition pairs, of which 3,000 pairs belong to the same person, 3,000 pairs of faces belonging to different people. LFW gives a pair of images and asks if the two images in the test are the same person, then the neural network gives a yes or no answer. The accuracy of face recognition can be obtained by comparing the system answers of face test results with the real answers.

**Evaluation:**  $N$  Images are selected from LFW for testing. If the network judges that  $C$  of the correctly identified  $M$  images is correct, the accuracy is  $C / M$  and the Coverage is  $M / N$ . By changing the similarity threshold, the Precision rate above 0.95 can be adjusted. The LFW is assessed as a percentage of the correct pairs of 6,000 pairs of faces.

**Pre-Processing:** In the training, we use data expansion to increase the number of samples. There are 12K face images in the original LFW samples. In these images, the number of a person's face is only 6K, and the amount of face data cannot fully mine the difficult samples. Therefore, we use cutting and rotation to expand the sample. The same pre-processing is used for all images in the LFW dataset. Given a training image, we first adjust it to  $256 \times 256$ , and then cut it into a  $224 \times 224$  sub image sum and flip it randomly. This operation is carried out 10 times continuously, and the amount of data is expanded to ten times of the original. In particular, we do



**FIGURE 14.** DTN with Focal Loss training can balance positive and negative samples effectively.

not use mean subtraction or image whitening, only change the spatial position of image pixels, because we put a batch normalization layer after the input data to learn the normalization parameters. In the test phase, the size of the training image and the test image is adjusted to  $224 \times 224$  and flipped randomly, and then the average value of the original image and the flipped image is taken as the final representation.

**Network and Training:** In order to learn deep facial features, we use the popular VGG16 network, which is deep enough to deal with our problems. The network used in this paper is trained on Beijing LINKZOT GPU cluster. The cluster consists of four geforce GTX 1080ti GPU graphics cards, with a total of 44GB video memory. In DTN network, the margin parameter of contrast loss is set to 0.6. For the learning rate, we set it to 0.05, which is used to train the classification network from scratch. The weight attenuation is set to 0.005. The training of the two networks ends at a rate of 0.0001, with a maximum of 100000 iterations, which is equivalent to 1000 epochs. In particular, the nesterov accelerated gradient method (NAG) is used for optimization, and its convergence speed is much faster than SGD.

**Parameter Setting:** For the number of subspaces, we set  $M = 5$ ,  $M = 10$ ,  $M = 20$  ( $m$  in formula 9) to ensure that each subspace contains about 15K, 10K, 5K personal faces respectively. Finally, we use Focal Loss to set  $\alpha = 0.5$  ( $\alpha$  in equation 11) in the whole experiment to ensure that the positive and negative samples are input into the network in the same proportion,  $\gamma' = 0.1$ ,  $\gamma = 0.9$  (in equation 11) to ensure that both positive and negative samples are difficult sample pairs, that is to say, the selected sample pairs come from the D region (D face set in Figure 5). The processed input positive samples are from the same person's face, the negative samples are from another person's face, and the positive and negative samples are from different people's faces.

DTN has a great advantage in face recognition, the accuracy of face recognition is 99.51% in LFW face dataset. While FACENET is not surpassed in accuracy, DTN is also well represented. In addition to the design of the network structure, the most important aspect of DTN is the selection of the loss function.

This paper uses 20 kinds of loss functions to train Siamese, Triplet, Quadruplet and DTN networks by comparing which network model's loss function is the best performance. In this paper, the method1-12 training accuracy data from

the Literature [32]. This paper adds a variety of DTN Loss function and then to train from a number of points of view analysis of the optimal results. As you can see from the accuracy of the 20 methods in Table 1,  $\text{acc}(\text{Siamese}) < \text{accn}(\text{Triplet}) < \text{acc}(\text{Quadruplet}) < \text{acc}(\text{TriHard}) < \text{acc}(\text{MSML}) < \text{acc}(\text{DTN})$ , from the network structure, from the comprehensive analysis of the network structure and loss function, we can get some rules from the data in the Table 1:

A. We can find that the accuracy of face recognition is greatly improved with the increase of the network layer by the horizontal contrast between method 1 and method 2.

B. From methods 3 to 7, you can see that using hard sample as input improves Triplet classification accuracy. However, it is not as accurate as softmax when doing subspace clustering instead of batch OHNM. This is mainly because softmax reduces the training difficulty and makes the multi-classification problem more convergent. Softmax encourages larger outputs in the real target category than in other categories. Softmax encourages separation of features between categories, but does not encourage much separation of features. However, the pre-use of subspace clustering on OHNM by the hard batch of Triplet can greatly improve the convergence speed and accuracy. Subspaces can be separated more effectively if softmax is added from method 8-11.

C. In the course of training methods 5,6, 7 using OHNM and subspace clustering, insufficient division of subspace will increase the negative sample interference, so the search efficiency will decline. But too small subspace will separate the similar face images. Therefore, it is more important to choose the right number of subspaces.

D. Comparing method 15-17 with method 5-7, and comparing method 8 with method 13, it is found that DTN is more accurate than TriHard.

E. Method 9-11, and method 8-20 show that Focal Loss is better than softmax for sorting. After analysis, it is concluded that DTN (MSML) + Batch OHNM + Subspace-10 + Focal Loss is the best loss functions, and the face recognition effect is the most accurate.

In the experiment, an intuitive line chart is made for the accuracy performance of the 20 methods in Table 1, as shown in Figure 15. As you can see in the Figure 15, the best performer is FACENET method 12, because of the network structure of FACENET and the sheer size of the data set. In addition, the best performance of method 19 is the network structure and loss function, that is, using DTN, OHNM, batch, subspace and Focal Loss to classify the best results.

### C. COMPARISON OF ROC CURVES BY MULTIPLE METHODS

In this paper, the 20 methods in Table 1 are compared from different angles. Divide the 20 methods into six groups, each with six methods. The ROC curves of True Positive Rate (TPR) and False Positive Rate (FPR) are established respectively in each group. Each model has the adjustment of similarity threshold. The closer the curve is to the upper left corner, the better the classifier is. As shown in Figure 16-21,

TABLE 1. Units for method –N properties.

Method -N (LFW DATASET)	Acc(%)
1-Siamese+contrastive Loss <sup>[32]</sup>	96.21
2- Quadruplet Loss <sup>[32]</sup>	98.85
3-Triplet+Softmax (Baseline) <sup>[32]</sup>	99.36
4-Triplet+Batch OHNM <sup>[32]</sup>	98.98
5-Triplet+Batch OHNM+Subspace-5 <sup>[32]</sup>	99.25
6-Triplet+Batch OHNM+Subspace-10 <sup>[32]</sup>	99.38
7-Triplet+Batch OHNM+Subspace-20 <sup>[32]</sup>	99.33
8-Triplet+Batch OHNM+Softmax <sup>[32]</sup>	99.33
9-Triplet+Batch OHNM+Subspace-5 + Softmax <sup>[32]</sup>	99.38
10-Triplet+Batch OHNM+Subspace-10 + Softmax <sup>[32]</sup>	99.48
11-Triplet+Batch OHNM+Subspace-20 + Softmax <sup>[32]</sup>	99.41
12-Face Net (Triplet+OHNM) <sup>[32]</sup>	99.63
13-Double Triplet Networks (MSML)+Batch OHNM + Softmax	99.37
14-Double Triplet Networks (MSML)+Batch OHNM + Focal Loss	99.38
15-Double Triplet Networks (MSML)+Batch OHNM+Subspace-5	99.28
16-Double Triplet Networks (MSML)+Batch OHNM+Subspace-10	99.39
17-Double Triplet Networks (MSML)+Batch OHNM+Subspace-20	99.35
18-Double Triplet Networks (MSML)+Batch OHNM+Subspace-5 + Focal Loss	99.39
19-Double Triplet Networks (MSML)+Batch OHNM+Subspace-10 + Focal Loss	99.51
20-Double Triplet Networks (MSML)+Batch OHNM+Subspace-20 + Focal Loss	99.43

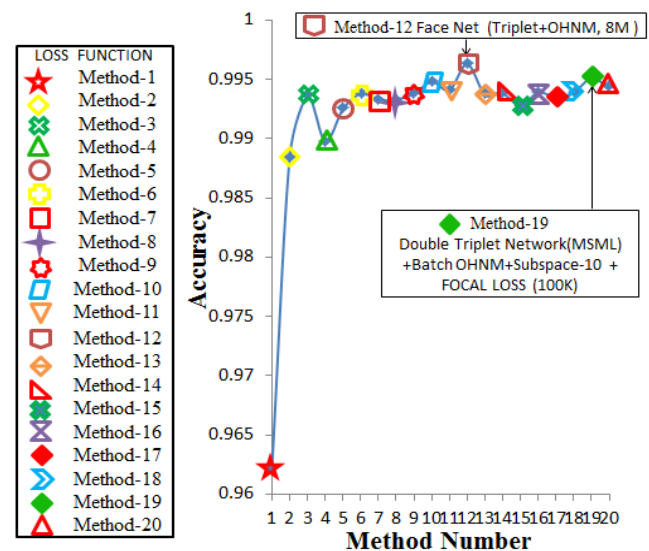


FIGURE 15. A visual line chart of the accuracy performance of the 20 methods.

each ROC curve is composed of six ROC curves. In the same FPR 0.1 case, the red classifiers in each ROC curve yield higher TPR. This indicates that the higher the ROC, the better the classifier.

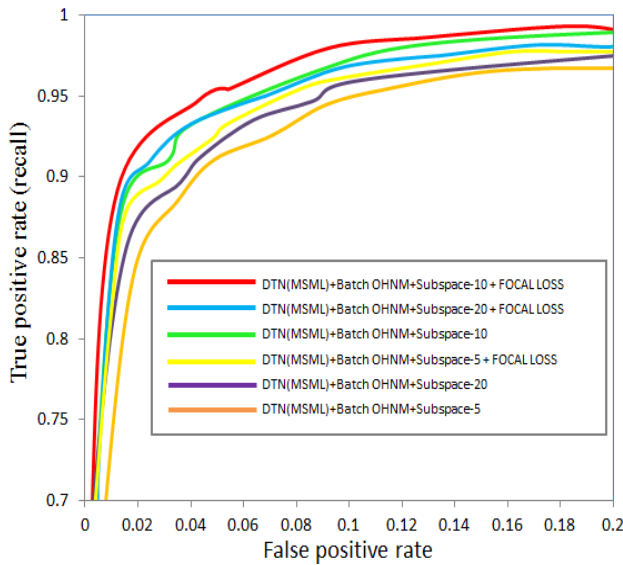


FIGURE 16. ROC for DTN of different Subspace numbers.

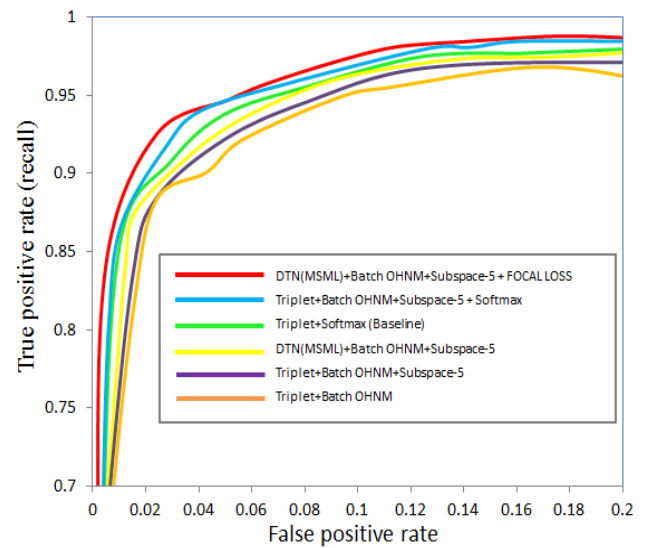


FIGURE 18. ROC for different network of 5 Subspace number.

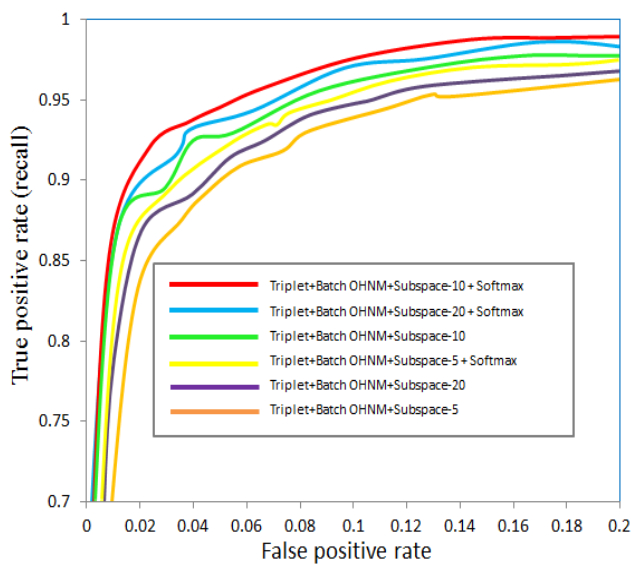


FIGURE 17. ROC for Triplet of different Subspace numbers.

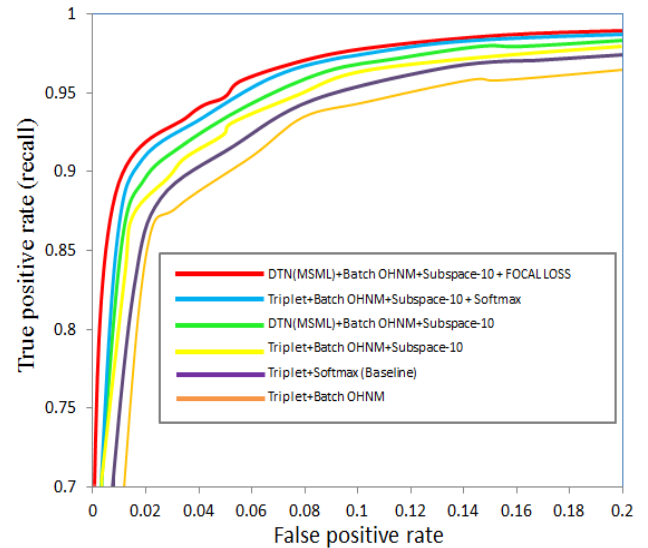


FIGURE 19. ROC for different network of 10 Subspace number.

Two types of methods are used in Figure 16: DTN + MSML + Batch OHNM + Subspace and DTN (MSML) + Batch OHNM + Subspace + Focal Loss. Draw a ROC curve for each of the three subspaces: subspace 5, subspace 10, and subspace 20. As you can see from the Figure 16, adding the Focal Loss function for training improves classification accuracy. Also, the choice of number of subspaces is the key, and a subspace cannot be too more or too less. Too many subspaces tear apart similar faces. The number of subspaces is too small, and many unsimilar faces are divided into similar face subspace, so the traversal efficiency is reduced.

Figure 17 uses two types of methods, Triplet + Batch OHNM + Subspace and Triplet + Batch OHNM + Subspace + Softmax, to plot a ROC curve for each of three kinds

of subspace numbers: subspace = 5, subspace = 10, and subspace = 20. From the graph, it can be found that training with the function of Softmax can improve the classification accuracy of Triplet network. Also, the choice of number of subspaces is critical, not too more or too less. Too many subspaces tear apart similar faces. If the number of subspaces is too less, many unsimilar faces will be divided into similar face sets, which will reduce the traversal efficiency.

Figure 18 fully demonstrates that the choice of the number of subspaces is critical and consistent with the conclusions drawn in Figure 16 and Figure 17.

Figure 19 uses two methods, DTN (MSML) + Batch OHNM + Subspace + Focal Loss and Triplet + Batch OHNM + Subspace-10 + Softmax. Based on these two methods, we compare them by removing Focal Loss and Softmax

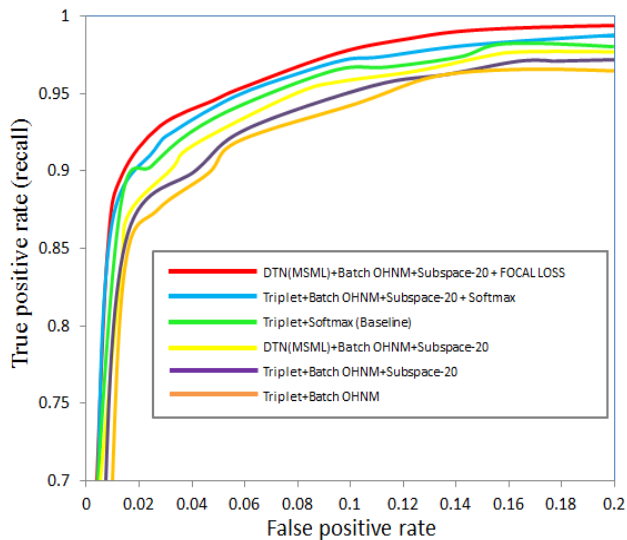


FIGURE 20. ROC for different network of 20 Subspace number.

and then training them. In principle, the sample balance is solved, and the classification of difficult samples is better. On the basis of the above, training by removing subspace spatial clustering separately leads to the conclusion that the setting of a subspace can improve the sample search speed of cal LOSS classification over Softmax classification.

Figure 20. A comparison of several models using a variety of LOSS functions shows that DTN (MSML) + Batch OHNM + Subspace + Focal Loss is the most prominent model and function.

Figure 21. Through the comparison of the best performance models on Facebook, it is shown that the training effect of DTN cannot be compared with that of FACERNET, but the training effect of DTN is improved obviously compared with other models.

**D. VALIDITY OF JOINT FACE RETRIEVAL**

*Dataset:* This article selects methods of 5,6,7,9,10,11,15,16, 17,18,19, 20 in Table 1.

In the data set PASC, LFW, PubFig, FERET, AR and YaleB, this paper use these methods to calculate on the average search precision ARP and the highest matching average search precision ARP performance evaluation.

*Evaluation Methods:* This paper uses Precision, Recall and F-score to evaluate the results of face retrieval. Each image in each dataset is used as a query image to retrieve all the remaining images to record the retrieval performance of the corresponding network. The retrieval precision and the retrieval rate of all data in the dataset are averaged to get the Average Retrieval Precision (ARP) and the Average Retrieval Rate (ARR). The number of Correct Retrieved Images is represented by *C*, the number of Retrieved Images is represented by *M*, and the number of Similar Images is represented by *N*, formula (12), formula (13) and formula (14) give the process

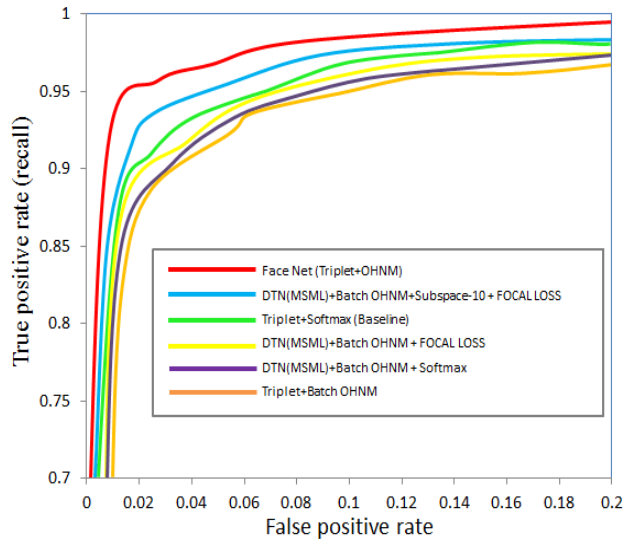


FIGURE 21. ROC comparison between different networks and FACENET.

of calculating *F-score*.

$$Precision = \frac{C}{M} \tag{12}$$

$$Recall = \frac{C}{N} \tag{13}$$

$$F - score = 2 \times \frac{ARP \times ARR}{ARP + ARR} \tag{14}$$

The larger the ARP, ARR, and F-score values, the better the retrieval performance.

Triplet + Batch OHNM + Subspace, Triplet + Batch OHNM + Subspace + Softmax and DTN (MSML) + OHNM + Batch + Subspace + Focal Loss were used to test ARP, ARR, and F-score on PaSC, LFW, PubFig, FERET, AR and YaleB data sets.

Table 2 uses the above three methods to calculate the best matching faces with the highest average retrieval accuracy on each of the above six databases. It can be seen from the Table 2 that the ARP of these three kinds of subspace is the highest on LFW and the lowest on YaleB. This is because the extended YaleB adds a lot of interference factors, such as light, attitude, etc. Among the three methods, the DTN (MSML) + Batch OHNM + Subspace + Focal Loss method performance is the best. In the same way, selecting a subspace = 10 can improve training accuracy.

Table 3 uses the above three kinds of subspace to calculate the average retrieval accuracy of the above six databases, and selects the top five best matching faces in each database to obtain the average retrieval accuracy. It can be seen from the Table 3 that the ARP of these three kinds of subspace is the highest on LFW, and besides the ARP of YaleB, PASC and FERET also have the obvious decreasing trend, which is because some of the top five face samples have great differences in the similarity of face images due to the great changes in light and face angle. Among the three methods, DTN (MSML) + Batch OHNM + Subspace + Focal

**TABLE 2.** Average Retrieval Precision, ARP(%) for topmost match using Triplet+Batch OHNM+Subspace, Triplet+Batch OHNM+Subspace+Softmax and DTN+Batch OHNM+Subspace+Focal Loss over the PaSC, LFW, PubFig, FERET, AR and YaleB.

Dataset	Method 5	Method 6	Method 7	Method 9	Method 10	Method 11	Method 15	Method 16	Method 17	Method 18	Method 19	Method 20
PaSC	94.16	94.25	94.21	94.18	94.36	94.25	94.31	94.43	94.36	95.05	95.32	95.28
LFW	99.22	99.36	99.30	99.30	99.45	99.39	99.26	99.36	99.33	99.35	99.49	99.40
PubFig	98.18	98.30	98.24	98.26	98.40	98.34	98.31	98.34	98.32	98.33	98.44	98.35
FERET	96.12	96.16	96.15	96.23	96.27	96.25	96.27	96.32	96.29	96.41	96.50	96.45
AR	99.76	99.78	99.77	99.82	99.87	99.83	99.82	99.84	99.79	99.85	99.87	99.86
YaleB	87.01	87.09	87.07	87.09	87.12	87.10	87.13	87.17	87.14	87.24	87.27	87.25

**TABLE 3.** ARP(%) for 5 numbers of retrieved images using Triplet+Batch OHNM+Subspace, Triplet+Batch OHNM+Subspace+Softmax and DTN+Batch OHNM+Subspace+Focal Loss over the PaSC, LFW, PubFig, FERET, AR and YaleB.

Dataset	Method 5	Method 6	Method 7	Method 9	Method 10	Method 11	Method 15	Method 16	Method 17	Method 18	Method 19	Method 20
PaSC	88.22	88.26	88.24	88.51	88.58	88.53	88.77	88.79	88.76	89.24	89.27	89.25
LFW	98.36	98.41	98.38	98.46	98.48	98.45	98.50	98.57	98.53	98.57	98.61	98.58
PubFig	95.47	95.54	95.50	96.23	96.33	96.29	96.69	96.73	96.71	97.33	97.54	97.39
FERET	86.04	86.24	86.12	86.54	86.64	86.58	86.82	86.96	86.88	88.45	88.62	88.53
AR	94.73	94.84	94.81	95.27	95.33	95.39	95.45	95.49	94.41	95.56	95.72	95.63
YaleB	78.22	78.37	78.27	78.42	78.48	78.44	78.47	78.49	78.47	78.48	78.52	78.49

**TABLE 4.** ARP(%) for 10 numbers of retrieved images using Triplet+Batch OHNM+Subspace, Triplet+Batch OHNM+Subspace+Softmax and DTN+Batch OHNM+Subspace+Focal Loss over the PaSC, LFW, PubFig, FERET, AR and YaleB.

Dataset	Method 5	Method 6	Method 7	Method 9	Method 10	Method 11	Method 15	Method 16	Method 17	Method 18	Method 19	Method 20
PaSC	84.33	84.38	84.36	84.37	84.43	84.41	84.67	84.74	84.69	84.89	84.98	84.91
LFW	97.41	97.58	97.51	97.54	97.65	97.57	97.64	97.71	97.67	97.72	97.77	97.75
PubFig	94.55	94.61	94.56	94.63	94.71	94.65	94.77	94.84	94.81	95.22	95.41	95.33
FERET	81.02	81.11	81.09	81.14	81.19	81.13	81.12	81.17	81.14	81.24	81.48	81.32
AR	81.72	81.84	81.75	81.87	81.96	81.89	81.95	82.00	82.97	82.25	82.39	82.28
YaleB	70.73	70.83	70.75	70.85	70.95	70.87	71.04	71.18	71.12	71.22	71.39	71.25

**TABLE 5.** Average Retrieval Rate, ARR(%) for 10 numbers of retrieved images using Triplet+Batch OHNM+Subspace, Triplet+Batch OHNM+Subspace+Softmax and DTN+Batch OHNM+Subspace+Focal Loss over the PaSC, LFW, PubFig, FERET, AR and YaleB.

Dataset	Method 5	Method 6	Method 7	Method 9	Method 10	Method 11	Method 15	Method 16	Method 17	Method 18	Method 19	Method 20
PaSC	27.53	27.74	27.63	27.79	27.95	27.83	28.08	28.16	28.09	28.18	28.29	28.23
LFW	29.81	30.12	30.04	30.34	30.41	30.35	30.37	30.43	30.40	31.03	31.17	31.09
PubFig	19.02	19.15	19.10	19.21	19.53	19.43	19.40	19.50	19.44	19.59	19.71	19.65
FERET	31.15	31.27	31.18	31.23	31.36	31.27	31.33	31.39	31.37	31.47	31.55	31.47
AR	30.34	30.43	30.56	30.52	30.65	30.54	30.62	30.66	30.65	30.85	30.95	30.87
YaleB	12.38	12.42	12.40	12.54	12.63	12.55	12.56	12.63	12.58	12.78	12.91	12.80

**TABLE 6.** F-Score(%) for 10 numbers of retrieved images using Triplet+Batch OHNM+Subspace, Triplet+Batch OHNM+Subspace+Softmax and DTN+Batch OHNM+Subspace+Focal Loss over the PaSC, LFW, PubFig, FERET, AR and YaleB..

Dataset	Method 5	Method 6	Method 7	Method 9	Method 10	Method 11	Method 15	Method 16	Method 17	Method 18	Method 19	Method 20
PaSC	43.12	43.25	43.17	43.24	43.42	43.30	43.32	43.37	43.29	43.46	43.61	43.50
LFW	45.45	45.54	45.43	45.49	45.60	45.53	45.46	45.53	45.49	45.62	45.78	45.67
PubFig	28.13	28.26	28.21	28.24	28.38	28.32	28.29	28.36	28.32	28.42	28.56	28.47
FERET	42.35	42.56	42.42	42.61	42.75	42.66	42.73	42.81	42.78	43.01	43.15	43.07
AR	45.35	45.47	45.36	45.51	45.64	45.57	45.54	45.63	45.60	45.79	45.95	45.82
YaleB	18.55	18.67	18.56	18.73	18.84	18.76	18.89	18.96	18.90	19.19	19.34	19.28

Loss performance is the best. In the same way, selecting subspace = 10 can improve the training accuracy.

In Table 4, we choose the top 10 best matching faces in each database to get the average of the retrieval precision. Because the similarity of the face database is low due to the

lack of similar samples or the large difference of similar faces, among the three kinds of subspace, DTN (MSML) + Batch OHNM + Subspace-10 + Focal Loss performance is the best.

In Table 5, the average of the best-fit face retrieval rate ARR, which ranks in the top 10 in each database, is calculated

**TABLE 7.** ARP(%) for 10 numbers of retrieved images using DTN+ Subspace-10+Focal Loss over the PaSC, LFW, PubFig, FERET, AR and YaleB. Influence of Batch OHNM on retrieval accuracy and sample Pre-processing time..

Dataset	Random sample		Batch OHNM	
	Acc(%)	Pre-processing time(s)	Acc(%)	Pre-processing time(s)
PaSC	81.36	12.6	84.98	16.9
LFW	95.78	13.5	97.77	18.3
PubFig	92.26	67.5	95.41	91.2
FERET	79.16	13.2	81.48	17.8
AR	80.55	2.9	82.39	3.9
YaleB	68.74	6.7	71.39	9.1

using the above three kinds of subspace. It can be seen that these three kinds of subspace have the highest ARR on FERET, among the three methods, DTN (MSML) + Batch OHNM + Subspace-10 + Focal Loss performance is the best. In Table 6, the best-matching F-score of the top 10 faces in each database is selected. The F-score score is the highest on LFW and the worst on YaleB. The DTN (MSML) + Batch OHNM + Subspace-10 + Focal Loss performance is the best among the three kinds of subspace. In the evaluation of the above datasets, ARR and F-score are the highest and the best comprehensive performance is obtained by integrating the above evaluation criteria DTN (MSML) + Batch OHNM + Subspace-10 + FOCAL Loss.

## VII. ABLATION EXPERIMENT

Through the above comparative experiments, it is found that the best combination of model loss function is: DTN (MSML) + Batch OHNM + Subspace-10 + Focal Loss. In this section, we will establish the ablation experiment of this model [33]. The purpose of ablation experiment is to explore the role of each variable from only one perspective. Here, DTN (MSML) is used as the basic model, and the control variables mainly include Batch OHNM, Subspace and Focal Loss. We only reduce one variable at a time to observe the change of retrieval accuracy.

### (1) Control variable Batch OHNM

Remove Batch OHNM from the combination of DTN (MSML) + Batch OHNM + Subspace-10 + Focal Loss, and select random sample pairing. From Table 7, we can see that there is no hard sample pairing mechanism of positive sample pair and negative sample pair, which makes DTN model lose the significance of pulling apart the relative distance between positive and negative samples, resulting in a much

lower retrieval rate. After OHNM mechanism, two samples of different modes can be added to DTN model, which makes the model heterogeneous and improves the retrieval accuracy greatly. However, due to the difficulty of sample selection, the pre-processing time is longer than that of random samples.

### (2) Control variable subspace

The subspace is removed from the combination of DTN (MSML) + Batch OHNM + Subspace-10 + Focal Loss, and no clustering subspace is generated. From Table 8, it can be seen that the retrieval accuracy of DTN model without subspace clustering is greatly reduced, and Focal Loss can't find the fuzzy sample set, and then can't balance the proportion of difficult sample pairs. At the same time, there is no subspace clustering, which makes the pairing of difficult samples inefficient and leads to slow pre-processing. After adding subspace clustering mechanism, similar samples can be found quickly, including similar positive samples and similar negative samples. However, from Table 8, too little clustering Subspace Partition will slow down the search speed, such as subspace = 5, while too much clustering Subspace Partition will split many similar sample pairs, such as subspace = 20, subspace = 40, which also destroys the internal relationship of samples. Therefore, it is very important to choose the appropriate number of clustering subspaces. From the experiment, we can see that subspace = 10 is the best subspace partition, and the retrieval accuracy is the highest on each data set.

### (3) Control variable Focal Loss

The Focal Loss is removed from the combination of DTN (MSML) + Batch OHNM + Subspace-10 + Focal Loss, which neither does positive and negative sample equalization nor difficult sample screening. From Table 9, we can see that the input side of DTN model which lacks difficult sample screening is to a large extent select sample pairs from simple sample clustering subspace, which cannot fully play the model's mutual advantage. At the same time, there is no balance between positive and negative samples, which leads to the confusion of positive and negative samples, which leads to the over fitting problem of model training, so the retrieval accuracy has been greatly reduced. The increase of Focal Loss makes it possible to filter difficult samples, and can also separate the over fitting problems brought by fuzzy samples. After adjusting the parameters of Focal Loss function, it is found that when  $\alpha = 0.5$ , i.e. positive and negative sample equalization, the training accuracy is improved with the increase of sample difficulty. However, when the number of

**TABLE 8.** ARP(%) for 10 numbers of retrieved images using DTN+Batch OHNM+Focal Loss over the PaSC, LFW, PubFig, FERET, AR and YaleB. Influence of the number of clustering subspaces on retrieval accuracy..

Dataset	Subspace=0	Subspace=5	Subspace=10	Subspace=20	Subspace=40
PaSC	81.35	82.15	84.98	81.98	81.49
LFW	95.25	97.49	97.77	96.45	95.96
PubFig	92.26	94.67	95.41	94.45	93.49
FERET	79.49	80.95	81.48	80.12	79.89
AR	80.19	81.69	82.39	81.47	81.02
YaleB	68.68	70.66	71.39	70.34	69.48

**TABLE 9.** ARP(%) for 10 numbers of retrieved images using DTN+Batch OHNM+Subspace = 10 over the PaSC, LFW, PubFig, FERET, AR and YaleB. Influence of focal loss and its corresponding parameters on retrieval accuracy..

Acc%	Without Focal Loss	$\alpha=0.5$	$\gamma=0.1$	$\alpha=0.5$	$\gamma=0.5$	$\alpha=0.5$	$\gamma=0.9$	$\alpha=0.1$	$\gamma=0.9$	$\alpha=0.9$	$\gamma=0.9$
<i>PaSC</i>	81.28	81.96		82.89		84.98		83.16		83.24	
<i>LFW</i>	95.17	95.63		96.59		97.77		96.92		96.86	
<i>PubFig</i>	92.05	95.39		93.92		95.41		94.34		94.44	
<i>FERET</i>	79.18	81.32		80.49		81.48		80.79		80.82	
<i>AR</i>	79.98	82.25		81.13		82.39		81.88		81.76	
<i>YaleB</i>	68.56	69.10		70.26		71.39		70.72		70.83	

samples is equal, the training accuracy is improved,  $\gamma = 0.9$ , that is, when the difficulty of samples is the most. The over fitting problem caused by sample imbalance makes the retrieval accuracy decrease (for example, when  $\alpha = 0.1$  and  $\alpha = 0.9$ ).

## VIII. CONCLUSION

In this paper, a new network structure DTN is proposed. By comparing Siamese, Triplet and Quadruplet, we find that the network structure is related to the number of network channels. After the same number of channels, the most important thing is how to design the effective loss function. This paper improves on DTN network. The Network structure is a kind of parallel Triplet structure, with quaternion samples as input, the distance between positive samples and negative samples can be widened, and the positive and negative samples can be completely separated. The loss function designed in this paper is a synthesis of several kinds of loss functions. In order to improve the search speed of the sample OHNM Batch, the subspace clustering method was selected. In the sample pre-processing phase of DTN network, the Focal Loss function is used in the training phase, because of how to solve the problem of increasing the difficulty and balancing the samples. Based on TriHard and MSML, the network loss was compared by 20 methods. Finally, the experimental results showed that DTN (MSML) + OHNM + Batch + Subspace-10 + Focal Loss has higher ARP, ARR and F-score. The effectiveness of each method was verified by ablation experiments. Therefore, the face retrieval method can meet the practical application of face retrieval, and the system takes into account the practical difficulties of remote videos retrieval in practical application. A multi-cameras face retrieval system is designed to solve the problem of multi-cameras feature sharing and joint features, so as to complete multi-cameras face retrieval.

## IX. DISCUSSION

The results of the above ablation experiments show that the role of each variable is indispensable. At the same time, each variable has the parameter adjustment state to achieve the optimal solution. The optimal parameters can be obtained from Table 7, 8 and 9. The optimal parameters are as follows: ① Batch OHNM pre-processing is selected; ② subspace = 10 is the optimal clustering number of subspace clustering;

③  $\gamma' = 0.1$ ,  $\alpha = 0.5$ ,  $\gamma = 0.9$  is the optimal parameter of Focal Loss. So under the optimal parameter setting, this paper selects the best combination of DTN (MSML) + Batch OHNM + Subspace-10 + Focal Loss, which can jointly improve the retrieval accuracy.

Suggestions for future work: (1) There is still room for improvement. Our network application face detection model first cuts the face region, but in the dense crowd, many low pixel undetected faces may also appear in the background. At this time, the extracted face features will be interfered by the low pixel face features. In the future, we can design and use multi branch network framework based on attention consistency to locate the face of the same person in two different images. The feature is extracted to solve the robust invariant representation of cross view matching. By learning to pay attention to the same person's face image region, a strong feature invariant representation is generated. This solves the problem that the attention regions of the same pedestrian face must be consistent in dense faces. (2) This method can be improved to remove background clutter. The face features extracted from the whole image include not only pedestrian face features but also background clutter. Although the learning method of face region detection can accurately extract the local features from the image, the local features still include the interference of the background region. Therefore, eliminating background interference is helpful to further improve the accuracy of face retrieval and recognition. In the future, FCN and mask R-CNN can be used to segment the network to get better face mask images. The segmented face mask and background mask can fully highlight the features of the target face and eliminate the interference of background features on face features.

## REFERENCES

- [1] H. Wu, T. Yokoyama, D. Pramadihanto, and M. Yachida, "Face and facial feature extraction from color image," in *Proc. 2nd Int. Conf. Autom. Face Gesture Recognit.*, Oct. 1996, pp. 345–350.
- [2] Y. Shi, X. Ren, S. Yang, and P. Gong, "A generalized kernel Fisher discriminant framework used for feature extraction and face recognition," in *Proc. 12th Int. Conf. Natural Comput., Fuzzy Syst. Knowl. Discovery (ICNC-FSKD)*, Aug. 2016, pp. 1487–1491.
- [3] Z. Lu, X. Jiang, and A. Kot, "Deep coupled ResNet for low-resolution face recognition," *IEEE Signal Process. Lett.*, vol. 25, no. 4, pp. 526–530, Apr. 2018.
- [4] C. Schmid and R. Mohr, "Local grayvalue invariants for image retrieval," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 5, pp. 530–535, May 1997.



- [5] A. Raza, H. Dawood, H. Dawood, S. Shabbir, R. Mehboob, and A. Banjar, "Correlated primary visual textron histogram features for content base image retrieval," *IEEE Access*, vol. 6, pp. 46595–46616, 2018.
- [6] J. Xiang, N. Zhang, R. Pan, and W. Gao, "Fabric image retrieval system using hierarchical search based on deep convolutional neural network," *IEEE Access*, vol. 7, pp. 35405–35417, 2019.
- [7] T. T. D. Pham, S. Kim, Y. Lu, S.-W. Jung, and C.-S. Won, "Facial action units-based image retrieval for facial expression recognition," *IEEE Access*, vol. 7, pp. 5200–5207, 2019.
- [8] Z. Dong, C. Jing, M. Pei, and Y. Jia, "Deep CNN based binary hash video representations for face retrieval," *Pattern Recognit.*, vol. 81, pp. 357–369, Sep. 2018.
- [9] S. Nagpal, M. Singh, R. Singh, and M. Vatsa, "Regularized deep learning for face recognition with weight variations," *IEEE Access*, vol. 3, pp. 3010–3018, 2015.
- [10] M. A. Abuzneid and A. Mahmood, "Enhanced human face recognition using LBPH descriptor, multi-KNN, and back-propagation neural network," *IEEE Access*, vol. 6, pp. 20641–20651, 2018.
- [11] C.-C. Lin and Y. Hung, "A prior-less method for multi-face tracking in unconstrained videos," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 538–547.
- [12] Y.-G. Lee and J.-N. Hwang, "Facial feature-integrated inter-camera human tracking," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2018, pp. 1638–1642.
- [13] C. Selvarathi and R. Sujatha, "Face tracking algorithm for tracking target in WSN," *Pure Appl. Math.*, vol. 118, pp. 2063–2070, Jan. 2018.
- [14] S. Lu, Z. Lu, and Y.-D. Zhang, "Pathological brain detection based on AlexNet and transfer learning," *J. Comput. Sci.*, vol. 30, pp. 41–47, Jan. 2019.
- [15] K. He, R. Girshick, and P. Dollar, "Rethinking ImageNet pre-training," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 4918–4927.
- [16] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1026–1034.
- [17] D. Marmanis, M. Datcu, T. Esch, and U. Stilla, "Deep learning Earth observation classification using ImageNet pretrained networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 1, pp. 105–109, Jan. 2016.
- [18] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the gap to human-level performance in face verification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1701–1708.
- [19] W. Ouyang, X. Wang, X. Zeng, S. Qiu, P. Luo, Y. Tian, H. Li, S. Yang, Z. Wang, C.-C. Loy, and X. Tang, "DeepID-Net: Deformable deep convolutional neural networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 2403–2412.
- [20] Y. Sun, Y. Chen, X. Wang, and X. Tang, "Deep learning face representation by joint identification-verification," in *Proc. Neural Inf. Process. Syst. (NIPS)*, vol. 27, 2016, pp. 1–9.
- [21] Y. Sun, D. Liang, X. Wang, and X. Tang, "DeepID3: Face recognition with very deep neural networks," 2015, *arXiv:1502.00873*. [Online]. Available: <http://arxiv.org/abs/1502.00873>
- [22] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 815–823.
- [23] X. Dong and J. Shen, "Triplet loss in siamese network for object tracking," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Munich, Germany, Sep. 2018, pp. 459–474.
- [24] M. Zhang, W. Li, and Q. Du, "Diverse region-based CNN for hyperspectral image classification," *IEEE Trans. Image Process.*, vol. 27, no. 6, pp. 2623–2634, Jun. 2018.
- [25] Y. Zhao, K. Hao, H. He, X. Tang, and B. Wei, "A visual long-short-term memory based integrated CNN model for fabric defect image classification," *Neurocomputing*, vol. 380, pp. 259–270, Mar. 2020.
- [26] M. Sardogan, A. Tuncer, and Y. Ozen, "Plant leaf disease detection and classification based on CNN with LVQ algorithm," in *Proc. 3rd Int. Conf. Comput. Sci. Eng. (UBMK)*, Sep. 2018, pp. 1102–1113.
- [27] W. Ge, "Deep metric learning with hierarchical triplet loss," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Munich, Germany, Sep. 2018, pp. 269–285.
- [28] W. Chen, X. Chen, J. Zhang, and K. Huang, "Beyond triplet loss: A deep quadruplet network for person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 403–412.
- [29] X. Lv, C. Zhao, and W. Chen, "A novel hard mining center-triplet loss for person re-identification," in *Proc. Chin. Conf. Pattern Recognit. Comput. Vis. (PRCV)*, Xi'an, China, Nov. 2019, pp. 199–210.
- [30] H. Yao, D. Fu, P. Zhang, M. Li, and Y. Liu, "MSML: A novel multilevel semi-supervised machine learning framework for intrusion detection system," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 1949–1959, Apr. 2019.
- [31] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.
- [32] C. Wang, X. Lan, and X. Zhang, "How to train triplet networks with 100K identities?" in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2017, pp. 1907–1915.
- [33] H. Zhang, Y. Ji, W. Huang, and L. Liu, "Sitcom-star-based clothing retrieval for video advertising: A deep learning framework," *Neural Comput. Appl.*, vol. 31, no. 11, pp. 7361–7380, Nov. 2019.



**GUOYIN REN** was born in Hulunbair, Inner Mongolia Autonomous Region, China, in 1985. He is currently pursuing the Ph.D. degree with the School of Mechanical Engineering, Inner Mongolia University of Science and Technology (IMUST), China. He is also with IMUST. He participates in the Provincial Key Laboratory, as an important position, and participated in a number of National Natural Science Funds. His research interests include pattern recognition and intelligent image processing, and his current research interests include face retrieval, face recognition, and re-identification based deep learning.



**XIAOQI LU** received the M.S. degree in medical information processing from Xi'an Jiaotong University, Xi'an, China, in 1989, and the Ph.D. degree in control theory and control engineering from the University of Science and Technology Beijing, Beijing, China, in 2003. He is currently a Doctoral Supervisor with the School of Mechanical Engineering, Inner Mongolia University of Science and Technology, Hohhot, China. He is also a Professor with the Inner Mongolia University of Technology. His research interests include intelligent image processing, pattern recognizing, and neural networks.



**YUHAO LI** is currently pursuing the master's degree with the School of Information Engineering, Inner Mongolia University of Science and Technology (IMUST), China. He is also with IMUST. He participates in the Provincial Key Laboratory. His research interests include pattern recognition and intelligent image processing, and his current research interest includes face detection based deep learning.