Journal of Information Hiding and Multimedia Signal Processing ©2020 ISSN 2073-4212
Ubiquitous International Volume **11**, Number **4**, December **2020**

Enhancing the Performance of Manifold Ranking in Image Retrieval using Combined Rank on Low-level Features and Embedded Vectors

Huy Tran Van

Hong Duc University
Thanh Hoa, Vietnam
tranlehuy@hdu.edu.vn

Dzung Pham Thi Kim and Huy Ngo Hoang

Electric Power University of the Vietnam Ministry of Industry and Trade
235 Hoang Quoc Viet, Co Nhue, Tu Liem, Hanoi, Vietnam
zungptk@epu.edu.vn; huynh@epu.edu.vn

Quy Hoang Van

Hong Duc University
Thanh Hoa, Vietnam
hoangvanquy@hdu.edu.vn

Corresponding author: zungptk@epu.edu.vn

Received July 2020; revised September 2020

Abstract. *The effective manifold ranking (EMR), an extended graph-based algorithm, is used quite successfully in content-based image retrieval (CBIR) for large image databases where images are represented by multiple low-level features to describe the color, texture and shape. Moreover, image features and appropriate distance metric are the important factors for researchers to improve the performance of image retrieval. The success of Deep Metric Learning (DML), is based on deep architectures to learn a suitable metric from data and obtain embedded features that are more discriminative. The advantage of low-level features is the ability to quickly recognize differences in color, texture and shape without learning, while embedded features provide a higher discrimination but depend on the pre-trained DML model and a given image object detection algorithm. To increase the performance of the manifold ranking in image retrieval, in this paper we propose the use of a combined rank to take advantage of both low-level features and embedded vectors which represent high-level features derived from the pre-trained DML model. This new rank focuses on embedded vectors when they are not degraded and uses the rank values on low-level features instead if embedded vectors do not provide good retrieval results. Experiments have been conducted to demonstrate the effectiveness of the proposed rank when increasing the quality of EMR.*

Keywords: Content-based image retrieval, EMR, Metric learning, Deep metric learning, Triplet loss.

1. Introduction.

In computer vision, **distance metric learning** is an approach based on a distance metric that aims to establish similarity or dissimilarity between objects. To improve performance, distance metric learning aims to build a good metric over input space by reducing the distance between similar objects but increasing the distance between dissimilar objects [1, 2]. Many studies have shown that distance metric learning significantly enhances accuracy in CBIR [3–5]. **Xing et. al** [1] demonstrated empirically to indicate that using the learned distance metric based on given examples of similar pairs is much more efficient than raw data while clustering with K-means. **Weinberger et. al** [2] improved the performance in kNN classification by using trained distance metrics in which the k-nearest neighbors always belong to the same class and distinguish with examples from different classes by a large margin.

Yang [6] indicated that there exist the connections between **Manifold Learning** and distance metric learning, i.e. the problems in distance metric learning is connected to or the same with those in manifold learning. Related to manifold learning, **Manifold Ranking (MR)** is known as a graph-based model that has been successfully applied to content-based image retrieval (CBIR) using low-level image features [7–10]. To extend its applicable ability to a large database, an **Efficient Manifold Ranking (EMR)** proposed in [7] that aims at building an anchor graph on the database and upgrading construction of adjacency matrix to improve the ranking speed. **Giang, Huy et. al** [8] applied EMR to improve the quality of CBIR systems using low-level image features and Relevance feedback. To raise a higher performance of the manifold ranking algorithm EMR the authors in [9] proposed a method to normalize the values of feature vectors that support finding the weight of each edge in the graph thereby raising the accuracy of the ranking results in K-means clustering. The selection of the anchor points for an EMR graph is also one of important ways to increase the quality of image retrieval. [10] proposed an algorithm for determining the anchor points of the image database by upgrading EMR using a modified FCM clustering. The authors in [11] enhanced the powerful ability of EMR by integrating low-level features into hybrid features or normalized features to calculate better weights. However, these above methods have only mentioned using low-level features for EMR. The EMR algorithm applied to low-level features has been shown to bring a good image retrieval result because these features describe images with color, texture and shape which are normally invariant with rotation and scale. In fact, although there are many kinds of low-level features and baseline machine learning models that have been upgraded and studied for improving image retrieval efficiency [30–32], the retrieval accuracy is generally not very high, does not fluctuate significantly with the different query images, and rarely decreases.

Besides multiple low-level features used in CBIR, high-level features proved more effective because of the semantics they can obtain. In the last years, **the high-level features extracted from CNN** models are considered as a semantic image representation that reveals the strong ability in improving the accuracy of image recognition [12]. In the field of face verification and recognition, Deep Convolution Neural Networks (CNNs) bring significant benefits. The triplet loss used in deep CNNs has been proved to increase the effectiveness and reach the state-of-the-art performance, in which the face images of the same identity should be closer to the face images of the different persons [13, 14]. CNN features are generally immutable with rotation and scale, and they depend on pre-trained models and object detection algorithms [15], so despite the average accuracy in CBIR is high but uneven for different types of query images. In particular, with the case of diverse natural query images, the CNN features, which were derived from the model that had been trained for a specific purpose earlier, may not be appropriate and reduce retrieval quality.

In the last few years, **Deep Metric Learning (DML)**, which is mentioned as a combination of deep learning and metric learning, has exposed its important role in solving learning tasks. While metric learning approaches are related to the linear transformation of the data, DML utilizes deep architectures by obtaining embedded feature similarity through nonlinear subspace learning, from which to develop problem-based solutions that are caused by learning from raw data. DML is based on the similarity relationship among samples using popular networks, such as Siamese and Triplet [16]. However, to improve the performance, sampling strategies also play an important role besides the structure of the network model. For example, Triplet network uses sample triplets, one of which includes an anchor, a positive, and a negative sample to train a network for classification. Learning based on triplet loss ensures that the distance from the anchor to the positive sample is closer than the anchor to the negative sample, thereby improving discriminating power of the samples and increasing the success of network model. **Wang et. al** [17] solved the limitations of sampling triplets by upgrading a selection strategy to find a margin that allows the positive samples to be closer than the negative samples. As a result, input data trained by DML becomes more discriminative and performance is achieved. DML is also applied in many domains to gain better performance, such as offline signature verification [18]. Similar to CNN features, embedded features are also immutable with rotation and scale. CNN features are often extracted from nearby layers to avoid being too bounded to the categories used during training [19], while embedded features are extracted at the last layer and focus on increasing discrimination through semi-supervised learning. Therefore, although the image query accuracy when using embedded features is higher than using the CNN features on the same topic, it is unavoidably degraded in some cases with specific query images.

From the knowledge we gained about the applicability and performance of EMR and DML provided in CBIR, we decide to use high-level image features (i.e. embedded vectors derived from the pre-trained DML model) as input feature database for EMR. However, to solve their limitation, we combined with the use of low-level features in cases when the ranking values on embedded vectors are low. As analyzed above, the advantage of low-level features is the ability to quickly recognize differences in color, texture and shape without learning. Therefore, to increase the performance of the manifold ranking in image retrieval, in this paper, we propose the use of a combined rank to take the advantages of both low-level features (LF) and embedded vectors (EV). The main contributions of our proposal are as follows:

(1) Provide a combination of LF and EV effectively, thereby increasing the accuracy of image retrieval results in CBIR.

(2) Demonstrated experimentally on the dataset VGG_60K. that, the similarity measure on embedded vectors using EMR is more effective than Euclidean distance.

The rest of the paper is organized as follows. Section 2 reviews related words on image retrieval. Sections 3 presents the framework. Section 4 provides the experimental results and the analysis, followed by the conclusions in Section 5.

2. **Related Work.**

2.1. **Deep Metric Learning.**

Deep Metric Learning, which utilizes deep architectures to solve limitations that caused from raw data by obtaining embedded feature similarity through nonlinear subspace learning [16]. The use of image data input which is embedded feature vectors extracted from DML, has been proven to bring effectiveness in classification and clustering. Thanks to the efficient discrimination power, DML is applied in various domains, such as face verification and recognition [13, 20, 21], three-dimensional (3D) modelling [22, 23], offline signature verification [18] etc.

DML consists of three main parts, which are informative input samples, the structure of the network model, and a metric loss function. Informative sample selection plays a very important role to increase the success of DML in classification or clustering. Siamese, Triplet and Quadruple networks are most commonly used to train samples in DML, however, Triplet network is simple but proved to gain high performance in face verification and recognition [13]. There is the reason why in this paper we chose the DML with Triplet network as a pre-trained model for extracting embedded vectors obtaining high-level features. Triplet network trains on raw data using Euclidean distance measure and sample triplets which contains an anchor, a positive, and a negative sample. The most important part of DML models is loss functions which can be represented in many different ways such as Contrastive loss [24], Quadruple loss [25], N-Pair loss [26]. However, a triplet loss function is the one of these that is proved to provide a higher discrimination power for data [13]. The triplet loss function, which aims to pull the anchor point closer to the positive point than the negative point by a fixed margin α, is defined as follows:

$$L_{Triplet} = \max\left(0, \left\|G_W\left(X\right) - G_W\left(X^p\right)\right\|_2 - \left\|G_W\left(X\right) - G_W\left(X^n\right)\right\|_2 + \alpha\right). \tag{1}$$

For X_1, X_2 is a pair of input samples, let $G_w(X_1)$ and $G_w(X_2)$ are generated as a new representation of X_1 and X_2 respectively, then D_w is used to calculate the distance between the two samples.

$$D_W\left(X_1, X_2\right) = \left\|G_W\left(X_1\right) - G_W\left(X_2\right)\right\|_2. \tag{2}$$

The main purpose of learning based on triplet loss is enforcing a margin value to achieve a goal that the distance from the anchor to the positive sample is closer than the anchor to the negative sample, thereby improving discriminating power of the samples and increasing the success of network model. The raw data is initially transformed through triplet networks and then computed distance metric similarity to obtain embedded feature vectors that finally contain more semantics and higher discrimination.

2.2. **The Efficient Manifold Ranking.** An important application of the EMR algorithm is to rank an image database by image queries. In contrast to the standard manifold ranking algorithm, to construct a similar measure between images, the EMR only used anchor image vectors instead of the entire image database. In the EMR algorithm, the adjacent relations of two image vectors are built based on anchor points instead of based on the s-neighbor relationship of each image vector, meaning that E_i is called connected to E_j if $i \neq j$ and there exists a certain common anchor point A_c (different meaning with the anchor mentioned in DML) such that A_c is neighbor of each E_i and E_j.

With each E_i image vector symbol, let us denote by $N_b(i;s)$ is the set of s anchor feature vectors that are closest to E_i (s is an experimental parameter, such as $s = 5$) and

$$d_s = \max_{l \in Nb(i,s)} \left\{d\left(E_i, A_l\right)\right\} \tag{3}$$

, with kernel

$$K\left(t\right) = \frac{3}{4}\left(1 - t^2\right), -1 \leq t \leq 1. \tag{4}$$

The manifold measure is built by solving the following objective function:

$$EMR(r;Q) = \frac{1}{2}\left(\sum_{1 \leq i,j \leq n+1} w_{ij}\left\|\frac{r_i}{\sqrt{D_{ii}}} - \frac{r_j}{\sqrt{D_{jj}}}\right\|^2 + \mu\sum_{i=1}^{n+1}\left\|r_i - r_{0,i}\right\|^2\right) \to \min \tag{5}$$

, where Q is a query image, (For convenience, we assign the image vector of Q to $E_n + 1$),

$$r_{0,n+1} = 1, r_{0,i} = 0, i = \overline{1,n}. \tag{6}$$

$$\mathrm{Z} = \left(z_{\mathrm{ki}}\right)_{1 \leq k \leq C, 1 \leq i \leq n+1}, z_{ki} = \frac{K\left(\frac{d(E_i,A_k)}{d_s}\right)}{\sum\limits_{l \in NB(i,s)} K\left(\frac{d(E_i,A_l)}{d_s}\right)} \forall k \in Nb(i,s), z_{ki} = 0 \forall k \notin Nb(i,s). \tag{7}$$

$$\mathrm{W} = (\mathrm{w_{ij}})_{1 \leq i,j \leq n+1}, W \overset{def}{=} Z^T Z. \tag{8}$$

$$w_{ij} = \sum_{k \in Nb(i,s) \cap Nb(j,s)} z_{ki} * z_{kj}, 1 \leq i, j \leq n+1. \tag{9}$$

$$D_{ii} \overset{def}{=} \sum_{j=1}^{n+1} w_{ij}. \tag{10}$$

As described above, the ranking results obtained by EMR algorithm depend on the selection of the number of anchor points C and the set of anchor points $\{\mathrm{A}_c\}_{c=1}^{C}$.

In CBIR systems, EMR has been prove to bring the performance while using low-level images features [8]. To increase the quality of the manifold ranking algorithm in EMR, the authors in [9] presented a method that allows to find the edge weights of the graph through normalizing the values of feature vectors, and as a result the accuracy of the ranking results in K-means clustering is raised. Although K-means is quite simple, it is an algorithm widely used for clustering and brings high efficiency. In original EMR algorithm, the K-means is used to select clustering centers as anchor points. In a variant of EMR proposed in [10], FCM algorithm has been improved in selecting suitable anchor points for building an anchor graph.

However, the above studies with EMR are limited to deployment on image databases with low-level features. A fact is that the embedded feature vectors, which are extracted from raw data using DML, obtains more semantics and higher discrimination. The samples with more information improve the efficiency in K-means clustering [1], namely increasing the quality of anchor points, thereby upgrading the performance of EMR.

3. **Proposed EMR Ranking Combination.** Image features can be divided into low-level and high-level, where the low-level features contain the characteristics of image, like color, texture, shape etc., and represented in a high-dimensional vector with less semantics but without losing image details, while the high-level features of image can represent more semantics but are limited by the reduction of color and texture details.

For CNN features, [19] has presented a deep analysis and experiments on ImageNet dataset, from which the authors indicated that the two feature representations, fc4096a and fc4096b, which are extracted from the first and the second layers of AlexNet, have a better generalization ability than other features of CNN and bring the high performance. However, CNN features are not widely used because the pre-trained model has classified images in the last layers, which predefined objects to be identified. To improve accuracy, the author in [14] proposed a weighted linear fusion between high- and low-level features, i.e. CNN and SIFT, by integrating their respective ranking scores into average scores. This fusion only aims to take the average scores but does not focus on analyzing the weaknesses of each feature type in order to choose the appropriate rank. Therefore, although the average accuracy increases, there are still very poor results compared to using a single feature type for the same query.

Using high-level features represented by embedded vectors, the image query proved to be remarkably effective with relatively high accuracy [13]. In [28] DML may provides a framework which overcome above challenge and combine two separate modules together, one is learning the color feature, texture feature and another is metric. However, this module combination is quite complex and in general DML also has the same limitation as CNN which is limited in color and texture representation. As an additional note, an embedded vector usually represents the object area that is located from a natural image, however, it is possible that locating objects is unsuccessful, so we assign the value of this embedded vector as undefined (NaN) in our work.

The above limitations of embedded features lead to the fact that when high-level features (i.e. *embedded vectors*) are ineffective, it is possible to use low-level features to replace. This allows us to take advantage of these features, thereby improving image ranking results. We define a query problem in CBIR when the query image Q can or cannot be found a related object area, corresponding to two situations that I_q (the embedded vector of Q) is valued or (NaN). To solve the this problem, we assume that the original dataset contains the images, each of which has a unique embedded vector corresponding to the largest object identified by DML, represented respectively $\{I_1, I_2, I_3, ...\}$. Depending on the value of I_q, we choose the ranking according to low-level featured images $r^*_{lf.v.Q}$ or combine with the ranking of the high-level featured images, i.e. $r^*_i = F(r^*_{lf,Q,i}, r^*_{ev,Q,i})$ where $r^*_{lf.Q.i}$ and $r^*_{ev.Q.i}$ are calculated by EMR. The F function here is a combination function which allows choosing the appropriate rank and eliminating the case of low query results (when using low- or high-level features separately). The following is our algorithm in details.

```
Algorithm 1: CoEMR ($\Omega_0, \Omega_1, T$) (Combination of EMR rankings on low-level features and embedded vectors)
$\Omega_0$: pre-trained model of object detection.
$\Omega_1$: pre-trained model of DML.
$T$: the number of low-level feature sets.
--------------------------------------------------------------------------------
  Input: image dataset $\{I_i\}_{1\leq \mathrm{i}\leq n}$, query image $I_Q$.
  $C$: the number of anchors for EMR
  Output: $r = \{r_i\}_{1\leq i\leq n}, r_i \in [0,1] \forall i = \overline{1,n}$ the combined similarity value of $I_i$ ranked by $Q$.
  Put: $I_{n+1} = I_Q$.
  Step 1 (offline):
  1.1: Calculate low feature vectors $\{I_{t,i}\}_{1\leq t\leq T, 1\leq \mathrm{i}\leq n}$ $of$ $\{I_i\}_{1\leq \mathrm{i}\leq n}$.
  1.2: For each image $I_i$, calculate $\{I_{pre,i}\}_{1\leq \mathrm{i}\leq n}$ by using model $\Omega_0$.
  1.3: For each image $I_{pre,i}$, calculate embedded vectors $\{embv_i\}_{1\leq \mathrm{i}\leq n}$ by using model $\Omega_1$.
  Step 2 (online):
  2.1: Calculate low feature vectors $I_{t,Q}$, $I_{pre,Q}$ by using model $\Omega_0$
  and embedded vectors $embv_Q$ by using model $\Omega_1$ Note, $I_{pre,Q} = I_{pre,n+1} = NaN$
  when the preprocessing is failure and then $embv_Q = embv_{n+1} = NaN$ .
  2.2: Put
          $r_{lf,Q} = \{r_{lf,i}\}_{1\leq i\leq n+1}, r_{lf,i} = 0 \forall i = \overline{1,n}, \mathrm{r_{lf,n+1}} = 1.0$
  and calculate ranking values $r^*_{lf,Q} = \left(r^*_{lf,Q,i}\right)_{1\leq i\leq n}$ based on low-level features
  $\{I_{t,i}\}_{1\leq t\leq T, 1\leq \mathrm{i}\leq n+1}$ by using EMR.
  2.3: Put
          $r_{ev,Q} = \{r_{ev,i}\}_{1\leq i\leq n+1}, r_{ev,i} = 0 \forall i = \overline{1,n}, \mathrm{r_{ev,n+1}} = 1.0$
  and calculate ranking values $r^*_{ev,Q} = \left(r^*_{ev,Q,i}\right)_{1\leq i\leq n}$ based on embedded vectors
  $\{embv_i\}_{1\leq \mathrm{i}\leq n+1}$ by using EMR.
  Step 3: Combine ranking values $r^*_{lf,Q} = \left(r^*_{lf,Q,i}\right)_{1\leq i\leq n}$,
          $r^*_{ev,Q} = \left(r^*_{ev,Q,i}\right)_{1\leq i\leq n}$
  , and obtain $r^* = \{\mathrm{r}^*_{Q,i}\}_{1\leq i\leq n}$, where $r^*_i = F(r^*_{lf,Q,i}, r^*_{ev,Q,i}), 1 \leq i \leq n$..
  Return $r^* = \{\mathrm{r}^*_{Q,i}\}_{1\leq i\leq n}$.
```

4. Experiment result.

4.1. **Image Dataset.** To proceed with the empirical part, we begin by selecting an appropriate dataset to prove the arguments and algorithms proposed in this paper are reasonable. The dataset should be large, complex and unlabeled. We choose face images because the face image belongs to the same type of image object, distinguished through image details, and has many variations compared to the background and posture. In addition, the face image dataset has been tested for the ability of embedded vector, invested in research on triplet loss and other strategies. However, the face recognition accuracy of the trained dataset may be not good when applied on other datasets, the parameters need to be adjusted or the dataset must be retrained.

For the above reasons, we selected the experimental dataset with a total of 60000 images of 500 people taken from the VGGFace2 dataset (test_image) which is a highly complex and unlabeled. The image dataset is divided into 500 equal layers, each containing 20 randomly selected images from the testVGG image set, which has a capacity of 768Mb. This dataset is temporarily named VGG_60K. Image names are assigned by a folder name with a picture number. This dataset is temporarily named VGG_60K (Figure 1) in which the image name is assigned by the folder name with the image number.

Figure 1. Dataset VGG_60K.

4.2. **Feature extraction.** In our experiments, we selected and extracted five global low-level features to describe an image: Color Moments, LBP, Gabor Wavelets Texture, Edge, and GIST. All of these features of the dataset VGG_60K are normalized so that each vector component of each image is within the range [-1, 1] and then are concatenated into one vector with a dimension of $d_{lf} = 809$ (see [9]) In parallel, each image in this dataset is run through the deep learning model Ω_1 with the Multi-task Cascaded Convolutional Networks

(MTCNN) algorithm [29] that is used to detect faces. The obtained result is an image that has been cropped, aligned and normalized with a size of d x d ($d = 160$). The process of selecting an appropriate and unique cropped face image for an ID can be associated with manual filtering. In the next step, this face is put into the model Ω_2 FaceNet using the Triplet loss and a corresponding embedded vector with dimension $d_{eb} = 128$ is obtained. In [10], **Quy, Huy et.al** use only low-level features but the obtained accuracy with EMR is quite high, even in the case of a partially covered human face. In addition, although embedded vectors can represent edges and borders well, not sure about color and texture. Therefore, for a specific problem, a low-level feature combination is needed to provide the best retrieval performance.

4.3. **Evaluation Indicators and Experimental Parameters.** Given a query image-set Q, for each image $q \in Q$, using the similarity metric given by EMR, we choose $N = 120$, is the number of images in a class that have the highest similarity values. The precision value is average ratio between the number of relevant images in the N image that being retrieved by the similarity of each image q. Calling the set of relevant images to the query $q \in Q$ is $\left\{I_{j_1}, I_{j_2}, ..., I_{j_{m_j}}\right\}$, the mAP value for all queries is calculated as follows:

$$mAP = \left(\frac{1}{|Q|}\sum_{j=1}^{|Q|}\frac{m_j}{N}\right) * 100. \tag{11}$$

Most of the previous studies only mentioned the value of mAP, but the experimental process also indicated that although mAP may be high, images with poor retrieval results still exist (Figure 2.a). Therefore, we added two indicators, minP and σP, to determine the minimum accuracy and the uniformity of the retrieval results of all the images in Q. The minP indicator is important for checking and eliminating impaired cases which cause low retrieval results. An improved minP value evaluates the recovery efficiency in queries that give the worst results. A small σP value proves that accuracy is uniform and good for all query images in Q. The indicators are calculated as follows:

$$minP = \min_{1\le j\le |Q|}\left(\frac{m_j}{N}\right) * 100. \tag{12}$$

$$\sigma P = \sigma\left\{\frac{m_j}{N} * 100\right\}_{1\le j\le |Q|}. \tag{13}$$

Experimental parameters. We set the following general parameters for all of our experiments. The coefficient $T = 5$ corresponds to the five selected low-level feature sets (listed in Section 4.2.). The parameter α is adjusted in the range $0.3 - 0.5$ and the number of anchors C for EMR is set with 10000.

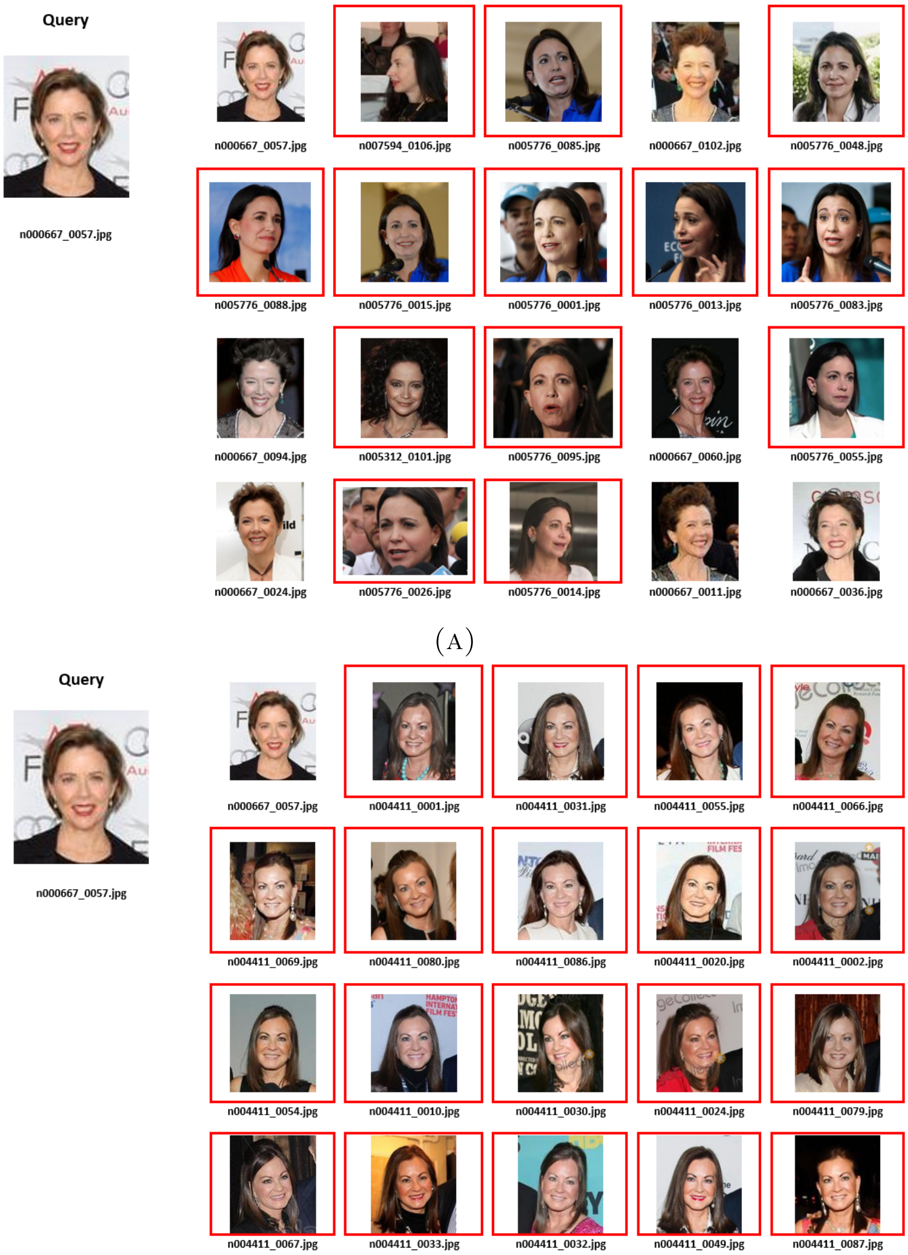
4.4. **Experiment and discussion.** To evaluate the Image retrieval efficiency (IRE) of our proposed algorithm, four experiments were conducted on the selected dataset VGG_60K.

Exp1. IRE with low-level features

In this experiment, we proceed with standard EMR, using K-means clustering, for low-level features of VGG60K. From each layer, 24 images are selected randomly. Similar to the experiments conducted in [9,10], the accuracy obtained from Exp1 is 70.61% and used to compare with our next experiments. With low-level features, the IRE of the retrievals is quite stable because the standard deviation σP of the obtained results is low according to Formula 12 (see Table 1). Therefore, it can be confirmed that the EMR is effective for the image dataset with low-level features.

Exp2. IRE with embedded vectors

With the same VGG_60K image dataset, we divide this experiment into two cases of using Euclidean distance in Exp2.1 and of using standard EMR in Exp2.2 to measure IRE with embedded vectors. An embedded vector is selected corresponding to the query image that contains the largest face in size. Exp 2.1 achieved an accuracy of 83.54%, higher than the case of only using low-level features in Exp1. In Exp 2.2, the accuracy is 88.71% which is 18.1% higher than Exp1. The combined results from the Exp2 indicate that using the embedded features brings good retrieval performance and the image similarity measurement on embedded vectors using EMR is more efficient than using Euclidean distance.



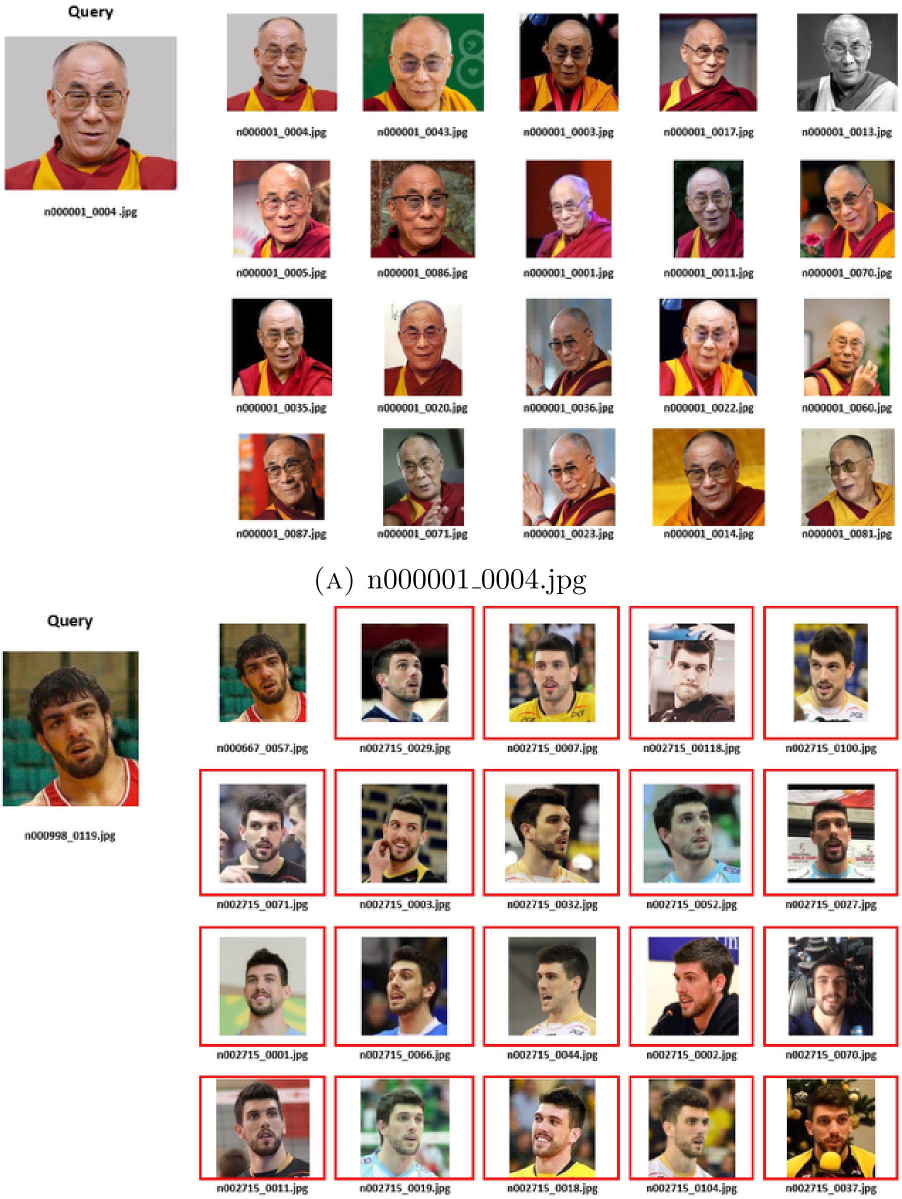
(A)

(B)

Figure 2. Query image n000667_0057.jpg, (A) the case of using Euclidean distance and (B) the case of using EMR.

We also compared the σP and minP indicators of the two cases, using Euclidean distance with embedded vectors and using EMR with embedded vectors (EMR-EV) and low-level features (EMR-LF). As the results in Table 1, σP is higher and minP is lower when using Euclidean distance, which shows the stability of this method is lower and its recovery efficiency is also less than that of EMR. Figure 2 is an example of the low stability of retrieval results when using Euclidean distance.

Reviews. We conducted a number of experiments and found that there are some specific cases for poor retrieval results although the query image has very good embedded vector. Figure 2 is the result of the query image n000667_0057.jpg with two cases of using Euclidean distance and standard EMR. The returned images have very high error rates, even up to 100%. This poor recognition results are due to pre-trained datasets, which are trained on another set of images, and now are used for another dataset, namely VGG_60K



(A) n000001_0004.jpg

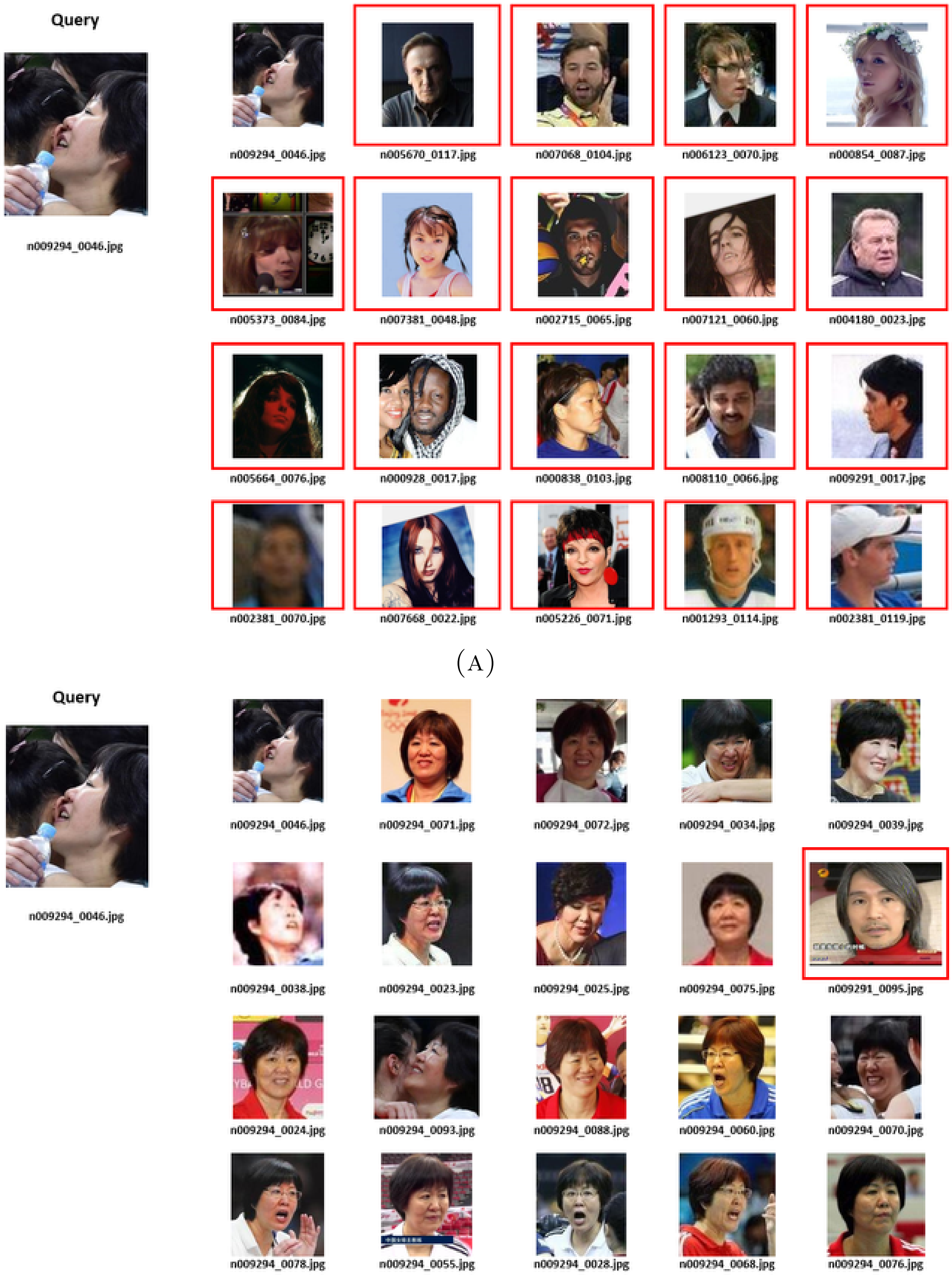
(B) n000998_0119.jpg

Figure 3. Query image n000001_0004.jpg and n000998_0119.jpg, (A) high accuracy and (B) low accuracy.

in this paper. This is a common problem when applying Deep Learning to CBIR because the features are extracted from the model that has been trained on a previous dataset.

Although the average accuracy is high (88.71% in Exp 2.1), the embedded vectors with EMR do not always produce the desired results. This result does not reflect the uniformity of each specific case when there are images that return almost 100% accurate query results but also images that return very poor results, Figure 2 as an example.

More observations, when making queries with n000667_0057.jpg using EMR-LF, the query results returned not bad, accuracy 61.67%. For some complex images such as the objects in the image overlap or a part of the object is obscured, these objects cannot be



(A)

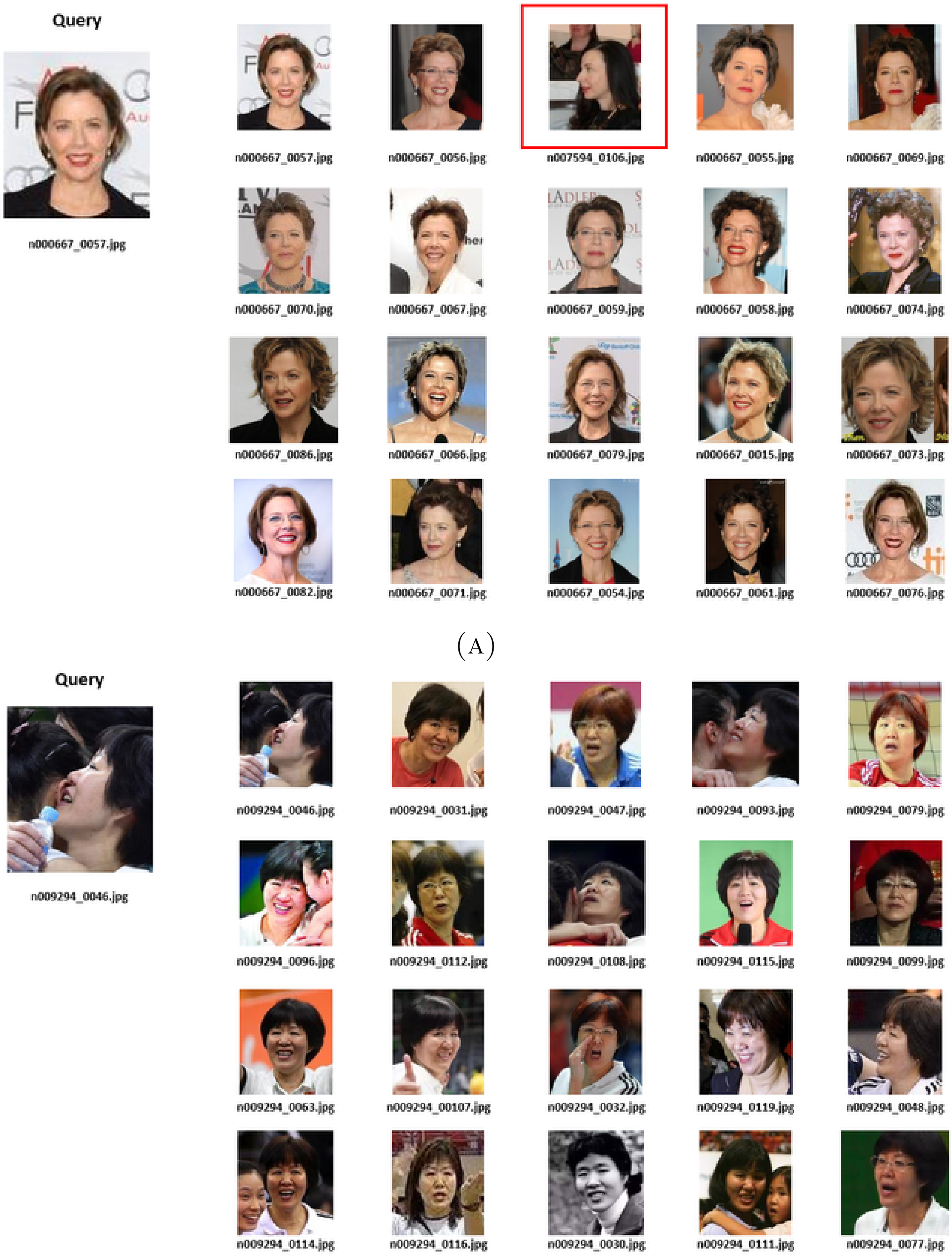
(B)

FIGURE 4. Query image n009294_0046.jpg, the top 20 images obtained (A) EMR-EV and (B) EMR-LF.

cropped or if they can be cropped, the accuracy results are reduced. But when using EMR-LF, the results are much better, as shown in Figure 3.

Figure 4 illustrates the retrival results after applying each EMR-EV and EMR-LF separately for query image n009294_0046.jpg, EMR-EV is inefficient in the case.

With observations in Exp1 and Exp2, we continue to conduct the experiment Exp3 with the cases of embedded vectors and and low-level features combined.



(A)

(B)

Figure 5. The top 20 images obtained by proposed method, (A) Query image n000667_0057.jpg and (B) Query image n009294_0046.jpg.

Exp3. Query effectiveness of combining low-level features and embedded vectors. In this experiment, we used two EMRs to rank respectively for embedded vectors and low-order feature vectors, thereby combining these rankings according to the algorithm proposed in section 3, using the coefficient alpha (α). So we divided this experiment into three small cases and found the best combined results as follows:

Case 1: Conduct combining two rankings from two EMRs according to Formula 14. The obtained accuracy is 89.60%, higher than in Exp2.2.

$$r_c^* = \max\left(r_{ev}, r_{lf}\right). \tag{14}$$

Case 2: Conduct combining two rankings from two EMRs according to Formula 15. The obtained accuracy is 90.61% with α_1=0.5, higher than in Exp2.2

$$r_c^* = \begin{cases} r_{ev}^*, & if \ r_{ev}^* \geq \alpha_1 \ or \ r_{ev}^* \geq r_{lf}^*. \\ r_{lf}^*, & \text{otherwise.} \end{cases} \tag{15}$$

Case 3: Conduct combining two rankings from two EMRs according to Formula 16. The obtained accuracy is 91.7% with α_1=0.3, higher than in Exp2.2.

$$r_c^* = \begin{cases} r_{ev}^*, & if \ r_{ev}^* \geq \alpha_2. \\ r_{lf}^*, & \text{otherwise.} \end{cases} \tag{16}$$

Figure 5 show the efficiency of the combination methods from the rankings of the two EMRs (EMR-EV and EMR-LF) with two query images n000667_0057.jpg and n009294_0046.jpg. The results returned are more accurate.

Table 1. Value indicators of experiments

EXP	Method	mAP	minP	σP
1.	EMR (low-level feature vectors)	70.61%	0.83%	0.155179
2.	Euclidean distance on cropped objects (EV)	83.54%	0.83%	0.179354
3.	EMR on cropped objects (EV)	88.71%	0.85%	0.178524
4.	EMR and combination of r_{ev} and r_{lf}	**91.70%**	**9.71%**	**0.130461**

We summarize the results from the experiments into Table 1 according to our arguments and the cases presented in this paper. The obtained results increased gradually in terms of matching accuracy, which is calculated by mAP, minP and σP, in the following order: EMR applied for low-level feature vectors, Euclidean distance on cropped objects (embedded vectors), EMR on cropped objects (embedded vectors) and EMR and combination of ranks on EV and LF.

5. **Conclusions.**

The combination of low-level feature vectors and embedded vectors in CBIR has proven to be effective by using both color-texture-shape and trained semantic characteristics. This contributes to partially solving the problem of the gap between low-level features and semantics. The combination of low-level features and embedded vectors was presented in paper with two corresponding ranking values calculated separately by the EMR algorithm which were then combined simply into a single ranking. This combination overcome the cases where embedded vectors have a degraded ranking quality of EMR due to the diversity of query images and the immanent limitation of a pre-trained for Deep Metric Learning.

In addition, the paper also demonstrates that for a pre-trained model for Deep Metric Learning, namely FaceNet [13], using the Euclidean distance to measure the similarity between embedded vectors, the results were lower than when using the similarity values obtained by the EMR algorithm. The experimental results have proved the effectiveness of the proposed method. In the next study, we intend to use the EMR algorithm to select triplet sets *(anchor, positive, negative)* for the triplet loss approach in Deep Metric Learning to obtain effective embedded vectors, thereby increasing the efficiency of image queries.

REFERENCES

[1] E. P. Xing, M. I. Jordan, S. J. Russell and A. Y. Ng, Distance metric learning with application to clustering with side-information, *Advances in neural information processing systems*, vol. 27, pp.521-528, 2003.

[2] K. Q. Weinberger, J. Blitzer and L. K. Saul, Distance metric learning for large margin nearest neighbor classification, *Advances in neural information processing systems*, pp.1473-1480, 2006.

[3] S.C.H Hoi, W. Liu and S.F. Chang, Semi-supervised distance metric learning for collaborative image retrieval and clustering, *ACM Transactions on Multimedia Computing, Communications, and Applications*, pp. 1-26, 2010.

[4] C.Jin and S.W. Jin, Content-based image retrieval model based on cost sensitive learning, *Journal of Visual Communication and Image Representation*, pp. 720-728, 2018.

[5] S.Chopra, R. Hadsell and Y. LeCun, Learning a similarity metric discriminatively, with application to face verification, *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 539-546, 2005.

[6] L. Yang, The connection between manifold learning and distance metric learning, *Technical report*, 2007.

[7] B. Xu, J. Bu, C. Chen, C. Wang, D. Cai and X. He, EMR: A scalable graph-based ranking model for content-based image retrieval, *IEEE Transactions on knowledge and data engineering*, vol. 27, no. 1, pp.102-114, 2013.

[8] N. T. Ngo, N. Q. Ngo, N. D. Nguyen and N. H. Ngo, Learning interaction measure with relevance feedback in image retrieval,*Journal of Computer Science and Cybernetics*, vol. 32, no. 2, pp.113-131, 2016.

[9] H. X. Hoang, D. V. Dao, N. H. Ngo, A. Sergey, N. Q. Nguyen and H. V. Hoang, A Novel Non-Gaussian Feature Normalization Method and its Application in Content Based Image Retrieval, *Nonlinear Phenomena in Complex Systems*, vol. 22, no. 1, pp.1-17, 2019.

[10] H. V. Hoang, N. H. Ngo, D. V. Dao, A. Sergey, T. V. Huy, A modified Efficient Manifold Ranking Algorithm for Large Database Image Retrieval, *Nonlinear Phenomena in Complex Systems*, vol. 23, no. 1, pp.79-89, 2020.

[11] Y. Wu, X. Wang and T. Zhang, Crime Scene Shoeprint Retrieval Using Hybrid Features and Neighboring Images, *Information*, vol. 10, no. 2, pp. 45, 2019.

[12] Z. Zhang and Z. J. Zhong, Image Retrieval Based on Fused CNN Features, *DEStech Transactions on Computer Science and Engineering*, no. aics, 2016.

[13] Z. Ming, J. Chazalon, M. M. Luqman, M. Visani and J. C. Burie, Simple triplet loss based on intra/inter-class metric learning for face verification, *2017 IEEE International Conference on Computer Vision Workshops (ICCVW), IEEE*, pp.1656-1664, 2017.

[14] M. Tzelepi and A. Tefas, Deep convolutional learning for content based image retrieval, *Neurocomputing*, vol. 275, pp.2467-2478, 2018.

[15] S. Huang and H. M. Hang, Multi-query image retrieval using CNN and SIFT features, *2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), IEEE*, pp.1026-1034, 2017.

[16] M. Kaya and H. ? Bilge, Deep metric learning: a survey, *Symmetry*, vol. 11, no. 9, pp.1066, 2019.

[17] X. Wang, Y. Hua, E. Kodirov, G. Hu, R. Garnier and N. M. Robertson, Ranked list loss for deep metric learning, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.5207-5216, 2019.

[18] A. Soleimani, B. N. Araabi and K. Fouladi, Deep multitask metric learning for offline signature verification, *Pattern Recognition Letters*, vol. 80, pp.84-90, 2016.

[19] H. Wang, Y. Cai, Y. Zhang, H. Pan, W. Lv and H. Han, Deep learning for image retrieval: What works and what doesn't, *2015 IEEE International Conference on Data Mining Workshop (ICDMW), IEEE* pp.1576-1583, 2015.

[20] J. Hum, J. Lu and Y.P. Tan, Discriminative deep metric learning for face verification in the wild, *Proceedings of the IEEE conference on computer vision and pattern recognition* pp.1875-1882, 2014.

[21] F. Schroff, D. Kalenichenko, and J. Philbin, FaceNet: A unified embedding for face recognition and clustering, *Proceedings of the IEEE conference on computer vision and pattern recognition* pp.815-823, 2015.

[22] G. Dai, J. Xie, F. Zhu and Y. Fang, Deep correlated metric learning for sketch-based 3d shape retrieval, *Thirty-First AAAI Conference on Artificial Intelligence* pp.4002-4008, 2017.

[23] I. Lim, A. Gehre and L. Kobbelt, Identifying style of 3D shapes using deep metric learning, *Computer Graphics Forum* No. 5, pp. 207-215, 2016.

[24] S.Chopra, R. Hadsell and Y. LeCun, Dimensionality reduction by learning an invariant mapping, *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, Vol. 2, pp. 1735-1742, 2006.

[25] J. Ni, J. Liu, C. Zhang, D. Ye and Z. Ma, Fine-grained patient similarity measuring using deep metric learning, *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pp. 1189-1198, 2017.

[26] K. Sohn, Improved deep metric learning with multi-class n-pair loss objective, *Advances in neural information processing systems*, pp. 1857-1865, 2016.

[27] D. Li, X. Chen, Z. Zhang and K. Huang, Learning deep context-aware features over body and latent parts for person re-identification, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 384-393, 2017.

[28] D. Yi, Z. Lei, S. Liao, and S.Z. Li, Deep metric learning for person re-identification, *2014 22nd International Conference on Pattern Recognition*, pp. 34-39, 2014.

[29] K. Zhang, Z. Zhang, Z. Li and Y. Qiao, Joint face detection and alignment using multitask cascaded convolutional networks, *IEEE Signal Processing Letters*, pp. 1499-1503, 2016.

[30] C.-F. Lee, Y.-J. Wang, S.-C. Chu, and J. F. Roddick, An Adaptive Content-Based Image Retrieval Method Exploiting an Affine Invariant Region Based on a VQ-applied Quadtree Robust to Geometric Distortions, *Journal of Network Intelligence*, Vol. 3, No. 3, pp. 214-234, August 2018.

[31] Y.-L. Qiao, Z.-M. Lu, J.-S. Pan and S.-H. Sun, Spline Wavelets Based Texture Features for Image Retrieval, *International Journal of Innovative Computing, Information and Control*, Vol. 2, no. 3, pp. 653-658, 2006.

[32] D. Tian, Support Vector Machine for Content-based Image Retrieval: A Comprehensive Overview, *Journal of Information Hiding and Multimedia Signal Processing*, Vol. 9, No. 6, pp. 1464-1478, November 2018.