A Cloud-guided Feature Extraction Approach for Image Retrieval in Mobile Edge Computing

Shangguang Wang, *Senior Member, IEEE,* Chuntao Ding, Ning Zhang, *Member, IEEE,* Xiulong Liu, Ao Zhou, *Member, IEEE,* Jiannong Cao, *Fellow, IEEE,* and Xuemin (Sherman) Shen, *Fellow, IEEE*

Abstract—Mobile Edge Computing (MEC) can facilitate various important image retrieval applications for mobile users by offloading partial computation tasks from resource-limited mobile devices to edge servers. However, existing related works suffer from two major limitations. (i) *High network bandwidth cost*: they need to extract lots of image features from the image to be retrieved, and transmit a large amount of the feature data to the cloud. (ii) *Low retrieval accuracy*: they separate the feature extraction processes from the image data set in cloud, thus unable to provide effective features for accurate image retrieval. In this paper, we propose a cloud-guided feature extraction approach for mobile image retrieval. In this approach, the cloud server leverages the relationships among labeled images in the data set to learn a projection matrix $\mathbf{P}$, which satisfies the properties that, if we use two images $\mathbf{x}_i$ and $\mathbf{x}_j$ with the same label to multiply $\mathbf{P}$, the results $\mathbf{P}^T\mathbf{x}_i$ and $\mathbf{P}^T\mathbf{x}_j$ will be quite similar; otherwise, the results will be significantly different. That is, the multiplying result can be interpreted as the features of the corresponding image. The matrix $\mathbf{P}$ is transmitted to the edge server, and is used to multiply the image $\mathbf{x}$ to be retrieved. The result $\mathbf{P}^T\mathbf{x}$, *i.e.*, image features, will be uploaded to the cloud server to find out the label of an image who has the most similar multiplying result. Such a label is regarded as the retrieval result and returned to the mobile user. In our cloud-guided feature extraction approach, fewer but more effective image features can be extracted, which can not only reduce network traffic but also improve retrieval accuracy. We have implemented a prototype system to validate the proposed approach, and conduct extensive experiments to evaluate its performance using a real MEC environment and data set. The experimental results show that the proposed approach reduces the network traffic by nearly 93%, and improves the retrieval accuracy by nearly 6.9%, compared with the state-of-the-art image retrieval approaches in MEC.

Index Terms—Mobile Edge Computing, cloud-guided, feature extraction, image retrieval, edge servers.

1 Introduction

1.1 Motivation & Problem Statement

With the growing popularity of mobile devices, image retrieval approaches can facilitate various promising applications, *e.g.*, object identification for visually impaired people, and facial recognition for authentication [1]–[3] . The most popular solution is based on Mobile Cloud Computing (MCC) [6]–[8], *i.e.*, a mobile user uploads the raw image to be retrieved (or the pre-processed data) to cloud servers, and then gets the retrieval results from cloud servers. However, directly uploading image-related data to remote cloud servers can incur a long network transmission delay. Then, we can use Mobile Edge Computing (MEC) [11], [12] to address the image retrieval problem with a small transmission delay, because mobile users can launch image retrieval request to and get retrieval results from the edge servers, which are much closer to users than cloud servers.

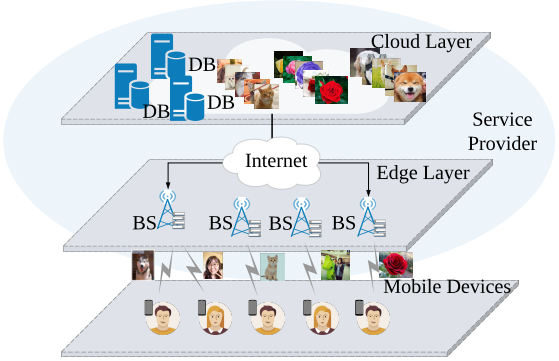


Fig. 1. System architecture of image retrieval in mobile edge computing.

In this paper, we study the problem of image retrieval in the MEC context, which is described as follows. *As illustrated in Fig. 1, the system architecture of MEC consists of three layers of components: the mobile devices (users), the edge servers, and the cloud servers. Mobile devices communicate with the edge servers via LTE, and edge servers are connected to the cloud servers by Internet backbone. A large amount of labeled image data are stored on the cloud servers. From the perspective of mobile users, the edge servers and cloud servers are together regarded as a black box, which is referred to as a service provider. A mobile user uploads an image to the service provider to launch the image retrieval*

- *Shangguang Wang and Ao Zhou are with the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, China. E-mail: sgwang@bupt.edu.cn; aozhou@bupt.edu.cn.*
- *Chuntao Ding is with the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, China and the Department of Computing, The Hong Kong Polytechnic University, Hong Kong, China. E-mail: ctding@bupt.edu.cn.*
- *Xiulong Liu and Jiannong Cao are with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong, China. E-mail: xiulong.liu@polyu.edu.hk; csjcao@comp.polyu.edu.hk.*
- *Ning Zhang is with the Department of Computing Sciences, Texas A&M University-corpus Christi, Corpus Christi, USA. E-mail: ning.zhang@tamucc.edu.*
- *Xuemin (Sherman) Shen is with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Canada. E-mail: sshen@uwaterloo.ca.*

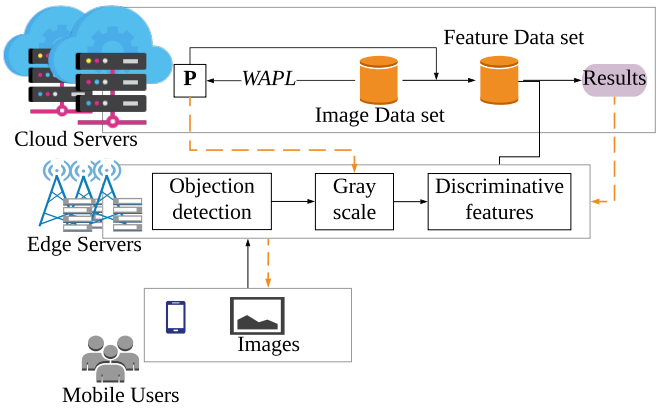


Fig. 2. The cloud-guided feature extraction approach for image retrieval in mobile edge computing.

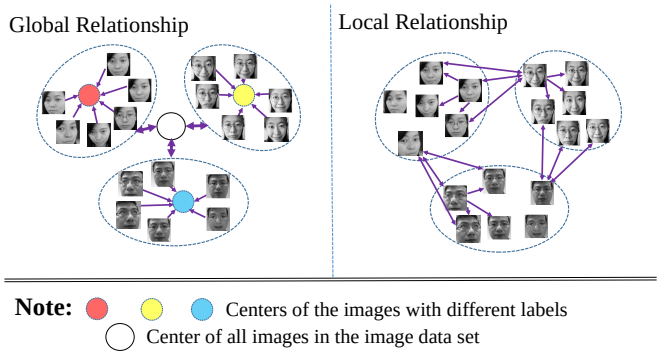


Fig. 3. Illustration of global and local relationships. The global relationship consists of the relationship between the image and the center of all images with the same label, and the relationship between the image centers and the center of all images, *i.e.*, the left sub-figure. The local relationship consists of the relationship between the images with the same label, and the relationship between the images with different labels, *i.e.*, the right sub-figure.

request. Then, the service provider processes and returns the label information of the most similar image to the mobile user.

1.2 Limitations of Prior Art

Existing MEC-based image retrieval approaches offload partial tasks (*e.g.*, extracting features) to edge servers. For example, *Soyata et al.* [10] designed the mobile-cloudlet-cloud architecture to implement a face recognition system to minimize response time by distributing the computation load among cloudlets. *Hu et al.* [15] extract features through the LBP algorithm [9] on the edge server and then upload the features to the remote cloud server. *Liu et al.* [13] pre-processed the captured image on the mobile devices before uploading it to the remote cloud servers. However, these solutions commonly have two major limitations: (i) They need to extract lots of features from the image to be retrieved since they aim to preserve its intrinsic structure. As a result, a large amount of feature data needs to be transmitted from edge servers to cloud servers, increasing the amount of network traffic and network transmission delay. (ii) Their feature extraction processes are isolated from the image data sets in the cloud servers. Thus, the extracted features are not the effective discriminative features, which results in low retrieval accuracy.

1.3 Proposed Approach

In this paper, we propose a cloud-guided feature extraction approach for image retrieval in MEC. As shown in Fig. 2, the image retrieval task is performed collaboratively between the mobile users, the edge servers and the cloud servers.

The image data set is usually stored on the cloud servers, taking into account data security, privacy, and the amount of image data. We first propose an algorithm called Weight-Adaptive Projection matrix Learning algorithm *WAPL* to learn the projection matrix $\mathbf{P}$ using the image data set on cloud servers. Then, the matrix $\mathbf{P}$ is used to extract features from the image data set on cloud servers to generate a low-dimensional feature data set, by using $\mathbf{P}$ to multiply each image data in data set. The multiplying results satisfy that, if we use two images $\mathbf{x}_i$ and $\mathbf{x}_j$ with the same label to multiply $\mathbf{P}$, the results $\mathbf{P}^T\mathbf{x}_i$ and $\mathbf{P}^T\mathbf{x}_j$ will be quite similar; otherwise, the results will be significantly different. That is, the multiplying result can be interpreted as the features of the corresponding image. The matrix $\mathbf{P}$ is transmitted to the edge servers, and is used to multiply the image $\mathbf{x}$ to be retrieved. The result $\mathbf{P}^T\mathbf{x}$, *i.e.*, image features, will be uploaded to the cloud servers to find out the label of an image who has the most similar multiplying result. Such a label is regarded as the retrieval result and returned to the mobile users. Generally, the projection matrix $\mathbf{P}$ can guide to extract discriminative features from the image to be retrieved. Thus, the edge servers just upload a small amount of feature data to the cloud servers. Compared with traditional feature extraction approaches, our cloud-guided feature extraction approach can significantly improve the image retrieval accuracy. In addition, network traffic and network transmission delay can be reduced since fewer feature data needs to be uploaded.

1.4 Challenges and Proposed Solutions

The first challenge is how to guarantee the learned projection matrix $\mathbf{P}$ has the capability of extracting effective discriminative features. The projection matrix $\mathbf{P}$ is very important because both accuracy and response time of the image retrieval mainly depend on the discriminative features extracted using it. Although many algorithms have been proposed to learn the projection matrix $\mathbf{P}$ to extract discriminative features, most of them either consider partial relationships (*e.g.*, global or local relationship) of the original image data set or assign equal weight to global and local relationships (as illustrated in Fig. 3). Different relationships are equally treated, which is not reasonable for most image data sets. As the importance of different relationships can be quite different, their weights need to be carefully decided. To this end, we propose a *WAPL* algorithm for learning projection matrix $\mathbf{P}$, where traditional global and local relationships are divided into four types of dissimilarities. The *WAPL* algorithm not only consists of all types of dissimilarities, but also introduces trade-off parameters α, β and γ to control the weights of them, thus ensuring that the matrix $\mathbf{P}$ has the capability of extracting effective discriminative features.

The second challenge is how to reduce manpower cost involved in determining the dimension of the low-dimensional feature

data sets. The optimal dimensions of the low-dimensional feature data sets of most existing algorithms are estimated empirically, which may require a lot of manpower costs to tune them. To this end, we investigate the relationships between the dimension of the low-dimensional feature space, retrieval accuracy and the number of eigenvalues in different image data sets. Moreover, we prove that the optimal dimension can be evaluated according to the number of positive eigenvalues. Thus, the number of positive eigenvalues can be regarded as the optimal dimension of the low-dimensional feature space, which can avoid tuning it empirically and save considerable manpower cost.

The third challenge is how to meet different requirements of users. In practice, there are different requirements in terms of retrieval accuracy and response time in different scenarios. For example, in the scenario of unmanned obstacle detection, users can accept lower retrieval accuracy but expect to real-time response. For authentication applications, the users care more about the retrieval accuracy than response time. To meet different requirements of users about retrieval accuracy and response time, we develop three interaction strategies between the cloud server and the edge server.

1.5 Novelty and Advantage over Prior Art

The technical novelty of this paper is in proposing a cloud-guided feature extraction approach, containing a new projection matrix learning algorithm. The technical depth of this paper is in learning an effective projection matrix, automatically determining the dimension of low-dimensional feature data set, and meeting various requirements of users. Compared with the state-of-the-art image retrieval approaches in MEC context, the key advantages of the proposed approach are two-fold: (i) In a real MEC environment, experimental results reveal that the proposed approach reduces the network traffic by nearly 93%. (ii) The image retrieval accuracy is improved by nearly 6.9%.

The remainder of this paper is organized as follows. The proposed MEC-based image retrieval approach is presented in Section 2. Section 3 introduces a novel projection matrix algorithm. The interaction strategies between cloud servers and edge servers are introduced in Section 4. In Section 5, we implement a prototype system to evaluate the performance of the approach. Section 6 reviews the related work. Section 7 concludes this paper.

2 The Cloud-guided Feature Extraction Approach

In this section, we will present the architecture of our MEC-based image retrieval system, and then describe the detailed design of the cloud-guided feature extraction approach.

The considered system architecture consists of three layers of components: the mobile users (devices), *e.g.*, smart phones; the edge servers, *e.g.*, base station servers; and the cloud servers, *e.g.*, Alibaba Cloud. In general, the cloud servers are more secure than edge servers. Hence, the large amount of image data sets are stored on the cloud servers. In practice, the image data sets are usually high-dimensional, which contain large amounts of redundant features that not only impair the image retrieval accuracy, but also result in long feature matching time.

As shown in Fig. 2, we propose a projection matrix learning algorithm called *WAPL*, and perform it on the image data sets to learn the projection matrix $\mathbf{P}$. Then, the projection matrix $\mathbf{P}$ is used to extract discriminative features from the image data sets stored on the cloud servers and form low-dimensional feature data sets, *i.e.*, $\mathbf{P}^T\mathbf{X}$. The results $\mathbf{P}^T\mathbf{X}$ satisfy that, if the original images have the same label, their features are compact; otherwise, their features will become separable. In other words, if two images $\mathbf{x}_i$ and $\mathbf{x}_j$ with the same label, the results $\mathbf{P}^T\mathbf{x}_i$ and $\mathbf{P}^T\mathbf{x}_j$ will be quite similar; otherwise, the results will be significantly different. Meanwhile, the projection matrix $\mathbf{P}$ is also transmitted to the edge servers through the Internet backbone. When the mobile users use mobile devices to capture images and launch the image retrieval quests. The mobile users first upload the image data to the edge servers via LTE or WiFi. When the edge servers receive an image retrieved by a user, several image pre-processing operations will be first performed, *e.g.*, executing object detection algorithm to extract object region and remove unrelated regions [36], and converting the image to gray scale image [15]. Then, using the projection matrix $\mathbf{P}$ transmitted from the cloud servers to extract discriminative features from the pre-processed image $\mathbf{x}$, *i.e.*, $\mathbf{P}^T\mathbf{x}$. The edge servers upload the results $\mathbf{P}^T\mathbf{x}$ (*i.e.*, the discriminative feature data) to the cloud servers through the Internet backbone. After the cloud servers receive the image feature data $\mathbf{P}^T\mathbf{x}$ from the edge servers, the feature matching algorithm (*e.g.*, the nearest neighbor classifier [27]) is performed to find the most similar images in data sets. Here, we say two images are similar when the Euclidean distance of their feature data is small.

Finally, the cloud servers transmit the labels (*e.g.*, name, birthplace) of the images in data sets, which are the most similar with the retrieved image, to edge servers, and the edge servers transmit these labels as retrieval results to the mobile users. Note that, edge servers and cloud servers selection has been well studied in previous work, and exceeds the scope of this paper. Hence, we assume that mobile users can always find the most appropriate edge server, and edge servers can also find the most appropriate cloud server. With the development of 5G technology, the transmission delay from mobile devices to edge servers is negligible. Thus, we pay more attention to the communication efficiency between edge servers and cloud servers. In our approach, extracting discriminative features from the pre-processed image and uploading them to the cloud servers can significantly reduce the load on the core network and network transmission delay. Moreover, using the projection matrix to extract discriminative features from the image to be retrieved, the image retrieval accuracy can be significantly improved. Feature matching is conducted in the low-dimensional feature space, which can also reduce the corresponding time.

3 Projection Matrix Learning Algorithm

So far, the unclear issue of the proposed approach is how to develop an algorithm to learn an effective projection matrix. In this section, we will propose a projection matrix learning algorithm, which aims to learn an effective projection matrix to extract discriminative features from the image data set stored on the cloud server and the image to be retrieved. The projection matrix is very important because it determines

TABLE 1
Frequently Used Notations

Symbol	Descriptions
$\mathbf{X}$	an image data set, where $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^N$
$\mathbf{x}_i$	the i-th image
N	the number of images
d	the dimensionality of the images
$\mathbf{Y}$	the corresponding label matrix, where $\mathbf{Y} = \{y_i\}_{i=1}^C$
C	the number of classes
r	the dimension of the low-dimensional feature space
$\mathbf{P}$	the projection matrix, where $\mathbf{P}^T\mathbf{P} = \mathbf{I}$
$\mathbf{I}$	the identity matrix
μ^m	the mean of the images in class m
N_m	the number of images in class m
μ	the mean of all the images
$\mathbf{x}_i^m$	the i-th image in class m
f_{gw}	the global intra-class dissimilarity
f_{gb}	the global inter-class dissimilarity
f_{lw}	the local intra-class dissimilarity
f_{lb}	the local inter-class dissimilarity
$\mathbf{L}_{lw}$	the Laplacian matrix, where $\mathbf{L}_{lw} = \mathbf{D}_{lw} - \mathbf{W}_{lw}$
$\mathbf{L}_{lb}$	the Laplacian matrix, where $\mathbf{L}_{lb} = \mathbf{D}_{lb} - \mathbf{W}_{lb}$
$\mathbf{W}_{lw}$, $\mathbf{W}_{lb}$	the symmetric similarity matrices
$\mathbf{S}_w$, $\mathbf{S}_b$	the intra-class / inter-class scatter matrices

whether the extracted discriminative features are effective. The frequently used notations are summarized in Table 1.

To ensure the learned projection matrix has the capability of extracting effective discriminative features, a novel Weight-Adaptive Projection matrix Learning algorithm *WAPL* is proposed. As illustrated in 3, the *WAPL* algorithm divides the traditional global and local relationships into four types of dissimilarities: global intra-class, global inter-class, local intra-class, and local inter-class dissimilarities. In comparison, these four types of dissimilarities are more granular than traditional global and local relationships. Thus, an effective projection matrix can be learned by incorporating all these types of dissimilarities and reasonably controlling them. Motivated by [23], [24], [34], we first give their quantification.

The global intra-class dissimilarity f_{gw} indicates the relationship between the image $\mathbf{x}_i^m$ and μ^m, which can be quantified as:

$$f_{gw} = \sum_{m=1}^{C}\sum_{i=1}^{N_m}\mathbf{P}^T(\mathbf{x}_i^m - \mu^m)(\mathbf{x}_i^m - \mu^m)^T\mathbf{P} \quad , \tag{1}$$

The global inter-class dissimilarity f_{gb} indicates the relationship between μ^m and μ, which can be quantified as:

$$f_{gb} = \sum_{m=1}^{C} N_m\mathbf{P}^T(\mu^m - \mu)(\mu^m - \mu)^T\mathbf{P} \quad , \tag{2}$$

The local intra-class dissimilarity f_{lw} indicates the pairwise relationship between images with the same label, which can be quantified as:

$$f_{lw} = \sum_{ij}||\mathbf{P}^T\mathbf{x}_i - \mathbf{P}^T\mathbf{x}_j||^2 W_{ij}^{lw} \quad , \tag{3}$$

$$W_{ij}^{lw} = \begin{cases} e^{-\frac{||\mathbf{x}_i - \mathbf{x}_j||^2}{t}}, i \in NS_{k_1}^w(j) \text{ or } j \in NS_{k_1}^w(i) \\ 0, \text{ otherwise} \end{cases} , \tag{4}$$

where $NS_{k_1}^w(i)$ denotes the index set of the k_1 nearest neighbors of the image $\mathbf{x}_i$ with the same label, and t is a constant parameter set according to experinece.

The local inter-class dissimilarity f_{lb} indicates the pairwise relationship between images with different labels, which can be quantified as:

$$f_{lb} = \sum_{ij}||\mathbf{P}^T\mathbf{x}_i - \mathbf{P}^T\mathbf{x}_j||^2 W_{ij}^{lb} \quad , \tag{5}$$

$$W_{ij}^{lb} = \begin{cases} e^{-\frac{||\mathbf{x}_i - \mathbf{x}_j||^2}{t}}, i \in NS_{k_2}^b(j) \text{ or } j \in NS_{k_2}^b(i) \\ 0, \text{ otherwise} \end{cases} , \tag{6}$$

where $NS_{k_2}^b(i)$ denotes the index set of the k_2 nearest neighbors of the image $\mathbf{x}_i$ with different labels.

To improve the retrieval accuracy, it is necessary to minimize f_{gw} and f_{lw}, meanwhile maximizing f_{gb} and f_{lb}. In other words, it is necessary to integrate all types of dissimilarities [24], [34]. However, simple integration of them is not reasonable for most image data sets. Because the importance of different types of dissimilarities can be quite different. To this end, trade-off parameters α, β and γ are introduced to control the importance of them. Most of existing projection matrix learning algorithms use the Fisher criterion [23], [34] to formalize the objective function. Although all types of dissimilarities can be incorporated, it cannot control the weights of them well, and leads to an ineffective projection matrix. To remedy this, the objective function is defined as follows:

$$\begin{aligned} &\max_{\mathbf{P}}[\gamma\beta f_{gb} + \gamma(1-\beta)f_{lb}] - [\alpha(1-\gamma)f_{gw} \\ &\quad + (1-\gamma)(1-\alpha)f_{lw}] \\ &s.t.\ \mathbf{P}^T\mathbf{P} = \mathbf{I} \end{aligned} \quad , \tag{7}$$

where $\alpha, \beta, \gamma \in [0, 1]$ are trade-off parameters that reflect the importance between f_{gw} and f_{lw}; g_{gb} and g_{lb}; αf_{gw}+$(1-\alpha)g_{lw}$ and βf_{gb}+$(1-\beta)f_{lb}$, respectively.

In this way, the weights of all types of dissimilarities can be controlled according to the requirements in different data sets. Moreover, f_{gb} and f_{lb} can be maximized, along with f_{gw} and f_{lw} can be minimized simultaneously. For brevity, Eq. (7) can be rewritten as:

$$\mathbf{P}^* = \arg\max_{\mathbf{P}} tr(\mathbf{P}^T\mathbf{H}\mathbf{P}) \ \ s.t.\ \mathbf{P}^T\mathbf{P} = \mathbf{I} \ , \tag{8}$$

where $\mathbf{H} = \gamma\beta\mathbf{S}_b - (1-\gamma)\alpha\mathbf{S}_w + \mathbf{X}[2\gamma(1-\beta)\mathbf{L}_{lb} - 2(1-\gamma)(1-\alpha)\mathbf{L}_{lw}]\mathbf{X}^T$. Eq. (8) ensures that the projection matrix $\mathbf{P}$ has the capability of extracting effective discriminative features because it incorporates all types of dissimilarities and controls their importance by using trade-off parameters.

The *WAPL* algorithm runs on the cloud servers, the optimal dimension of the low-dimensional feature space should be automatically estimated to avoid a lot of manpower costs. Because different data sets correspond to different optimal dimensions. Moreover, it is impossible to estimate the optimal dimensions empirically on all data sets. Note that, $\mathbf{H}$ is a real symmetric matrix because $\mathbf{S}_b$, $\mathbf{S}_w$, $\mathbf{L}_{lb}$ and $\mathbf{L}_{lw}$ are real symmetric matrices. In addition, it is also non-positive definite and the eigenvalues of $\mathbf{H}$ can be positive, zero, or negative. This motivates us to solve Eq. (8) by utilizing the relationships between the eigenvalues of $\mathbf{H}$, the eigenvectors of $\mathbf{H}$ and $\mathbf{H}$. According to [33], we propose Theorem 1 in the following.

Theorem 1. *The solution $\mathbf{P}^*$ of the objective function in Eq. (8) is composed of eigenvectors $[\mathbf{p}_0, \cdots, \mathbf{p}_{r-1}]$ of $\mathbf{H}$ corresponding to the top r positive eigenvalues, where r is the number of positive eigenvalues of $\mathbf{H}$.*

Proof. The Lagrangian function of problem in Eq. (8) is:

$$\zeta(\mathbf{P}, \mathbf{\Lambda}) = tr(\mathbf{P}^T\mathbf{H}\mathbf{P}) - tr(\mathbf{\Lambda}(\mathbf{P}^T\mathbf{P} - \mathbf{I})) \ , \quad (9)$$

where $\mathbf{\Lambda} = [\lambda_1, \ldots, \lambda_n]$. By calculating its derivative with respect to $\mathbf{P}$ and setting it to zero, we have $\mathbf{H}\mathbf{p}_i = \lambda_i\mathbf{p}_i$. Thus, Eq. (8) can be rewritten as:

$$tr(\mathbf{P}^T\mathbf{H}\mathbf{P}) = \sum_{i=0}^{d-1}\mathbf{p}_i^T\mathbf{H}\mathbf{p}_i = \sum_{i=0}^{d-1}\mathbf{p}_i^T\lambda_i\mathbf{p}_i = \sum_{i=0}^{d-1}\lambda_i. \quad (10)$$

From Eq. (10), in order to maximize $tr(\mathbf{P}^T\mathbf{H}\mathbf{P})$, only the positive eigenvalues should be chosen since zero eigenvalues have no effect on $tr(\mathbf{P}^T\mathbf{H}\mathbf{P})$, and negative eigenvalues are harmful to $tr(\mathbf{P}^T\mathbf{H}\mathbf{P})$. The solution to Eq. (8) must be

$$\mathbf{P}^* = [\mathbf{p}_0, \cdots, \mathbf{p}_{r-1}] \quad (11)$$

Hence, the statements in this theorem is proved. □

The optimal projection matrix $\mathbf{P}^*$ is composed of eigenvectors corresponding to the top r positive eigenvalues according to Theorem 1. Here, the value of r can be estimated, which equals to the number of the positive eigenvalues of $\mathbf{H}$. In other words, the optimal dimension of the low-dimensional feature space can be automatically estimated in our approach according to the number of positive eigenvalues rather than empirically. Therefore, the proposed approach can save a lot of manpower costs.

4 Interaction Strategy

In practice, different interaction strategies between the cloud servers and the edge servers are needed to meet different requirements of users, because both image retrieval accuracy and response time depend on different network transmissions (*i.e.*, the feature data) from the edge servers to the cloud servers. Here, we discuss three scenarios as follows: (i) the scenario chasing high retrieval accuracy more than low response time, *e.g.*, in an authentication system, users could wait even longer but expect ultra-high retrieval accuracy. (ii) the scenario requiring timely response but just decent retrieval accuracy, *e.g.*, in the scenario of unmanned obstacle detection, users may not need to know exactly what the obstacles in front is, but need to be informed real-time that it is an obstacle ahead. (iii) the scenario having high expectation on both retrieval accuracy and response time.

From Theorem 1, the optimal dimension of the low-dimensional feature space can be evaluated according to the number of positive eigenvalues. Thus, when the projection matrix $\mathbf{P}$ consists of all the eigenvectors corresponding to the positive eigenvalues, all discriminative features can be included and the highest retrieval accuracy can be achieved. For the first scenario, the projection matrix $\mathbf{P}$ consists of all the eigenvectors corresponding to the positive eigenvalues. However, for the second scenario, if we directly use the projection matrix suitable for the first scenario, it will incur a huge amount of network traffic, long network transmission delay, and feature matching time, which cannot meet the real-time requirements of users. Thus, we can employ a fraction of leading eigenvectors used the first scenario. Although some discriminative features may be lost, but the network transmission delay and feature matching time can be reduced. For the third scenario, we can use an medium amount of eigenvectors to balance the performance in terms of retrieval accuracy and response time.

Fig. 4. The images cropped from Lab_face data set.

5 Performance Evaluation

In this section, we first evaluate the *WAPL* algorithm on three benchmark data sets. Then, we implement a prototype system to evaluate the proposed approach in practical network environment and with real data set.

5.1 Experiment Setup

The experiment environment consists of three components: mobile device, edge server and cloud server.

- *Mobile Device:* A Huawei honor 8 smart phone is used as the mobile device. This smart phone is equipped with 4 Cortex A72 2.3 GHz, 4 Cortex A53 1.8 GHz, and Android 7.0. It also has a 32 GB internal storage and 4 GB RAM. We implement an APP, "ImagCat", to capture images and upload them to edge servers and cloud servers.
- *Edge Server:* The edge server consists of a base station and an edge server. The base station is based on Open Air Interface, and consists of three components: radio-frequency signal generator, base station server A and base station server B. The radio-frequency signal generator is equipped with USRP-B210. The base station server A is equipped with Intel i7-6700@3.4 GHz CPU and 16 GB RAM running Ubuntu 14.04.3, and used to run eNodeB. The radio-frequency signal generator and the base station A are connected through USB 3.0. The base station server B is equipped with Intel i5-6500@3.2 GHz CPU and 4 GB RAM running Ubuntu 14.04.3, and used to run Home Subscriber Service (HSS), Mobility Management (MME), Serving Gateway (SGW), and PDN Gateway (PGW). The base station servers A and B are connected through LAN. The base station works on Band7 (uplink 2500 MHz-2570 MHz, downlink 2620 MHz-2690 MHz). The edge server is a computer equipped with Intel i5-4590@3.3 GHz CPU and 12 GB RAM. Operations with image pre-processing run on it by using Java to invoke the OpenCV libraries. The mobile device and edge server are connected through LTE base station with upload link speed 1000 KB/s and the down link speed 1.36 MB/s.
- *Cloud Server:* The cloud server is Alibaba Cloud [1], which is equipped with 4 quad-core 2.5 GHz Intel Xeon E5-2682 v4 and 16 GB RAM running Ubuntu 14.04.3 and implements the *WAPL* algorithm and feature matching by Python. The edge server and cloud server are connected via Internet backbone.

5.2 Data Sets

We first evaluate the *WAPL* algorithm on YaleB [31], U-MIST [29], and USPS [30] data sets. Then, we collect a new data set Lab_face and implement a prototype system

1. https://www.alibabacloud.com/

TABLE 2
Description of Benchmark Data Sets

Data set	#Images	#Features	#Classes
YaleB	2414	1024	38
UMIST	574	1024	20
USPS	9298	256	10
Lab_face	420	1024	21

TABLE 3
Comparison of Image Retrieval Approaches

Approach	Edge server	*WAPL*	Edge server with $\mathbf{P}$
MCC_{simple}	No	No	No
MCC_{WAPL}	No	Yes	No
MEC_{simple}	Yes	No	No
MEC_{WAPL}	Yes	Yes	No
Our approach	Yes	Yes	Yes

to evaluate the proposed approach using a real network environment. The detailed information of the benchmark data sets used in the experiments are listed in Table 2, and examples of Lab_face data set are shown in Fig. 4.

5.3 Comparison Algorithms and Approaches

5.3.1 Projection Matrix Learning Algorithms

We compare the *WAPL* algorithm with four state-of-the-art projection matrix learning algorithms: marginal Fisher analysis (MFA) [34], joint global and local-structure discriminant analysis (JGLDA) [35], double adjacency graphs-based discriminant neighborhood embedding (DAG-DNE) [33] and locality adaptive discriminant analysis (LADA) [23].

- MFA [34] was introduced by *Yan et al.* in 2007, which learns the projection matrix by characterizing the intra-class compactness and inter-class separability.
- JGLDA [35] was proposed by *Gao et al.* in 2013, which learns the projection matrix by characterizing both the similarity and diversity of image data.
- DAG-DNE [33] was proposed by *Ding* and *Zhang* in 2015, which learns the projection matrix by preserving the local pairwise relationship between images.
- LADA [23] was proposed by *Li et al.* in 2017, which learns the projection matrix by preserving the local pairwise relationship between samples and solving the problem of making assumptions about data distribution by linear discriminant analysis [24].

5.3.2 Related Image Retrieval Approaches

To evaluate the performance of the proposed approach, we compared it with other four image retrieval approaches. The approaches are described in detail as follows, and the main differences of them are given in Table 3.

- MCC_{simple}: MCC_{simple} is the traditional MCC approach where user first uses mobile device to capture an image. Then, the user uploads it to the remote cloud server for processing, *i.e.*, using LBP algorithm to extract features and matching. Finally, the user receives results from the remote cloud server.
- MCC_{WAPL}: Different from MCC_{simple}, in MCC_{WAPL} approach, the *WAPL* algorithm is used to learn the projection matrix $\mathbf{P}$ and first extract the discriminative features from the image data set. When the cloud servers receive the raw image data to be retrieved from the mobile devices, after pre-processed, the projection matrix is used to extract discriminative features from the pre-processed image. Then, the feature matching is performed in the low-dimensional feature space.
- MEC_{simple}: Different from MCC_{simple}, in MEC_{simple} approach, the user first uploads the raw image data to be retrieved to the edge servers. After the edge servers receives the image data and pre-processing, the features are extracted from the pre-processed image using the LBP algorithm. Then, the edge servers upload the feature data to the cloud servers for feature matching.
- MEC_{WAPL}: Different from MEC_{simple}, the MEC_{WAPL} approach first uses the *WAPL* algorithm to learn the projection matrix $\mathbf{P}$. Then, the matrix $\mathbf{P}$ is used to extract the discriminative features from the image data set on the cloud servers. When the cloud servers receives the feature data from the edge servers, the matrix $\mathbf{P}$ is also used to further extract discriminative features from it. Finally, the feature matching is performed in the low-dimensional feature space.
- Our approach: Different from MEC_{WAPL}, in our approach, we not only use the projection matrix $\mathbf{P}$ to extract discriminative features from the image data set on the cloud servers, but also transmit it to the edge servers to extract discriminative features from the image to be retrieved. Thus, the edge servers only upload discriminative feature data to the cloud servers. Moreover, the feature matching is performing in the low-dimensional feature space.

In this experiment, we compare the proposed approach with MCC_{simple}, MCC_{WAPL}, MEC_{simple} and MEC_{WAPL} approaches on Lab_face data set using a real network environment. We choose five different sizes of images, which are related to the Lab_face data set, to evaluate the network traffic and response time, and the image size order is Image1<Image2<Image3<Image4<Image5. For fair comparison, we set the same value of the nearest-neighbor parameter k_1 and k_2 to construct adjacency graphs for all algorithms. Without prior knowledge, we set k_1=1 and k_2=1. Finally, we use the nearest neighbor classifier [27] to verify the extracted discriminative features.

5.4 Results of the *WAPL* Algorithm

5.4.1 Recognition Accuracy

In this experiment, we compare the *WAPL* algorithm with other state-of-the-art projection matrix learning algorithms. In YaleB, UMIST and USPS data sets, 50% of the images are randomly selected to form the training set, the remaining images are used for testing.

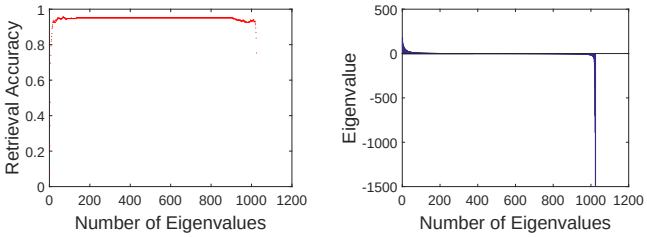
The experiment results are given in Tables 4. The *WAPL* can achieve the highest image retrieval accuracy in all projection matrix learning algorithms for all image data sets under different k values. That is because *WAPL* incorporates four types of dissimilarities and reasonably controls the importance of them in extracting discriminative features.

5.4.2 Relationship between Retrieval Accuracy and the Number of Eigenvalues

In this experiment, we investigate the relationship between the retrieval accuracy and the number of eigenvalues of

TABLE 4
Image Retrieval Accuracy (% ± std)

Data Set	Algorithms	Results		
		k=1	k=3	k=5
YaleB	MFA	87.04±0.33	86.96±0.41	86.94±0.83
	JGLDA	87.08±0.58	86.56±0.83	86.96±0.58
	DAG-DNE	87.54±0.21	87.68±0.66	88.00±0.75
	LADA	88.52±0.25	88.52±0.25	88.52±0.25
	WAPL	**92.36±0.23**	**93.47±0.78**	**91.55±0.37**
UMIST	MFA	97.77±0.82	97.12±0.35	97.30±0.70
	JGLDA	97.65±0.67	97.89±0.32	97.12±0.66
	DAG-DNE	97.89±0.70	97.00±0.21	97.42±0.76
	LADA	97.31±0.43	97.31±0.43	97.31±0.43
	WAPL	**98.99±0.18**	**98.17±0.21**	**98.48±0.24**
USPS	MFA	85.84±0.43	88.41±0.44	89.77±0.97
	JGLDA	85.95±0.65	89.30±0.23	90.92±0.30
	DAG-DNE	92.38±0.64	92.23±0.86	92.51±0.14
	LADA	90.49±0.36	90.49±0.36	90.49±0.36
	WAPL	**95.89±0.46**	**95.24±0.15**	**96.68±0.31**



(a) (b)

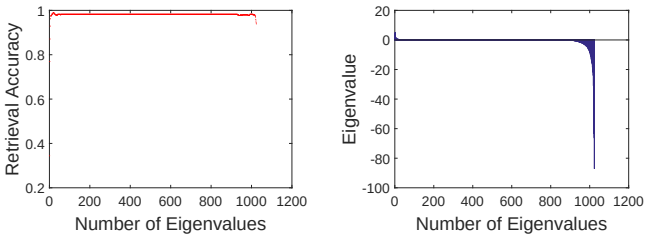
Fig. 5. Relationship between retrieval accuracy, eigenvalue, and the number of eigenvalues on YaleB data set.

WAPL's objection function on three data sets. Figs. 5, 6, and 7 show that the retrieval accuracy of the *WAPL* algorithm rapidly rises with the increase of the number of eigenvectors when the eigenvectors corresponding to positive eigenvalues are chosen. Then, it tends to stabilize when the eigenvectors corresponding to the nearly zero eigenvalue are chosen. Finally, the retrieval accuracy of *WAPL* algorithm decreases when the eigenvector corresponding to the negative eigenvalues are chosen. It manifests that only eigenvectors corresponding to the positive eigenvalues contribute to extracting discriminative features, and the optimal dimension of the low-dimensional feature space can be estimated according to the number of positive eigenvalues. Thus, it can save a lot of manpower costs in extracting discriminative features. In addition, this discovery also helps us design different interaction strategies between the cloud servers and the edge servers to meet different requirements of users in terms of retrieval accuracy and response time. We will discuss it in detail in Section 5.5.4.

5.4.3 Parameters Analysis

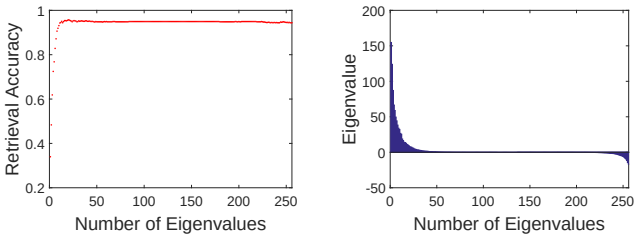
The trade-off parameters α, β and γ can be tuned as follows. Each data set is randomly divided into a training set $\mathbf{X}_{Tr}$ and a test set $\mathbf{X}_{Te}$. The training set $\mathbf{X}_{Tr}$ is also randomly divided into a training set $\mathbf{X}_{Tr1}$ and a validation set $\mathbf{X}_{Va1}$. The training set $\mathbf{X}_{Tr1}$ is used to choose parameters, and the validation set $\mathbf{X}_{Va1}$ is used to validate parameters. α is evaluated by fixing β and γ and varying α from 0 to 1. β and γ are validated in the same way as α. Table 5 shows the corresponding parameter values when the *WAPL* algorithm obtains the highest retrieval accuracy on three data sets.

From Table 5, we observe that, on different data sets, the corresponding parameter values are different when the *WAPL* algorithm achieves the highest retrieval accuracy.



(a) (b)

Fig. 6. Relationship between retrieval accuracy, eigenvalue, and the number of eigenvalues on UMIST data set.



(a) (b)

Fig. 7. Relationship between retrieval accuracy, eigenvalue, and the number of eigenvalues on USPS data set.

This shows that different types of dissimilarities have different importance in extracting discriminative features when dealing with different data sets. Ignoring any of them or ignoring their different importance may undermine the capabilities of the projection matrix. Therefore, it is essential to control the weights of four types of dissimilarities according the characteristics of the data set.

5.5 Results of the Approaches

5.5.1 Network Traffic

As shown in Fig. 8, the proposed approach can reduce network traffic by nearly *93%*, compared with MEC_{simple} and MEC_{WAPL} approaches. The major reason is that, with the cloud-guided feature extraction approach, the edge server only needs to upload fewer and more effective discriminative feature data. However, in MEC_{simple} and MEC_{WAPL} approaches, the edge server uploads more features to preserve the local information rather than the discriminative features to the remote cloud server.

From Fig. 8, the proposed approach can reduce network traffic by nearly 1000 times, compared with MCC_{simple} and MCC_{WAPL} approaches. This is due to that the mobile device uploads the raw image to the cloud server of MCC_{simple} and MCC_{WAPL} approaches. However, in the proposed approach, the edge server uploads the discriminative features to the cloud server. Compared with the raw image, the size of the discriminative feature data is much smaller. Therefore, the proposed approach can significantly reduce network traffic on the core network. Moreover, MCC_{simple} and MCC_{WAPL} approaches have the same network traffic because both of them upload the raw image. MEC_{simple} and MEC_{WAPL} approaches have the same network traffic because they upload the features extracted by LBP algorithm. Moreover, we also find that MEC_{simple} and MEC_{WAPL} approaches can reduce network traffic by more than 17 times, compared to MCC_{simple} and MCC_{WAPL} approaches.

TABLE 5
Highest Accuracy and Corresponding Parameters

Data set	Retrieval Accuracy (%)	α	β	γ
YaleB	95.76	0.9	0.4	0
UMIST	99.69	1	0.5	0
USPS	96.54	0.5	0.5	0.1

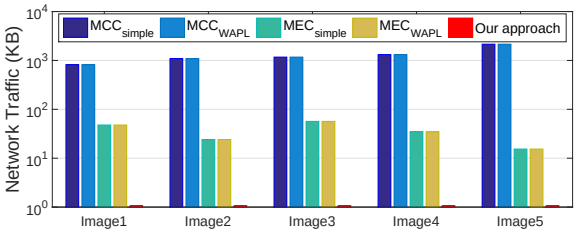


Fig. 8. Network traffic for different approaches.

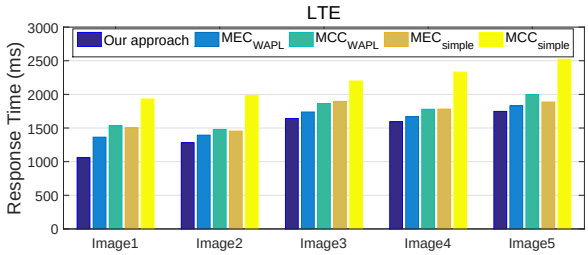


Fig. 9. Response time for different approaches.

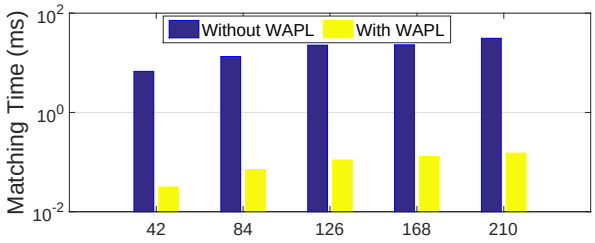


Fig. 10. Matching time for different cases.

5.5.2 Response Time

Fig. 9 shows the response time of five approaches when the mobile device connects the edge server through LTE. It can be seen that the proposed approach can reduce the average response time by up to 35%, compared with the MCC_{simple} approach. The network transmission delay can be reduced by reducing network traffic, and feature matching time can be reduced because the feature matching is performed in the low-dimensional feature space.

As shown in Fig. 9, the response time of MCC_{simple} approach is longer than that of the MEC_{simple} approach, because the features extracted using LBP is smaller than the raw image. MEC_{simple} approach is longer than MEC_{WAPL} approach, because the feature matching is performed in the low-dimensional feature space of MEC_{WAPL} approach.

Fig. 10 shows feature matching time with different sizes of images under two cases (without *WAPL* and with *WAPL*). It can be seen that the feature matching time can be significantly reduced by using the *WAPL* algorithm, because it reduces the number of matching features. The feature matching time can be reduced by 100 times, compared with the case where the *WAPL* algorithm is not used. The major reason is that under the same number of images, the more features are, the longer matching time it takes.

The response time of MCC_{WAPL} approach is longer than MEC_{WAPL} approach, due to the network traffic of MCC_{WAPL} approach is larger than MEC_{WAPL} approach. Under the same bandwidth, the greater the network traffic is, the longer the network transmission delay the mobile users suffer from. More importantly, our approach can get the minimum response time. The major reason is that, with the cloud-guided feature extraction, our approach can extract fewer and more effective discriminative features. Since the discriminative features are fewer, the network transmission delay can be reduced. Moreover, since the dimension of the low-dimensional feature space is low, the feature matching time can be reduced.

In addition, we also evaluate the response time of five approaches when the mobile device connects the edge server through WiFi, where the upload link and download link are set to 9 MB/s. As shown in Fig. 11, the response time of our approach is the shortest in all approaches. From Fig. 11, the response time reduction of our approach could be up to 47%, compared with MCC_{simple} approach. Specifically, when recognizing the Image5, the response time of our approach is 393 ms and the response time of MCC_{simple} approach is 742 ms. The result indicates that our approach can significantly reduce response time with the development of 5G technology, because the transmission delay from mobile devices to edge servers can be negligible. The response time of our approach is shorter than MEC_{WAPL} approach. This result indicates that with the cloud-guided feature extraction approach, fewer and more effective discriminative features can accelerate image retrieval services.

5.5.3 Retrieval Accuracy

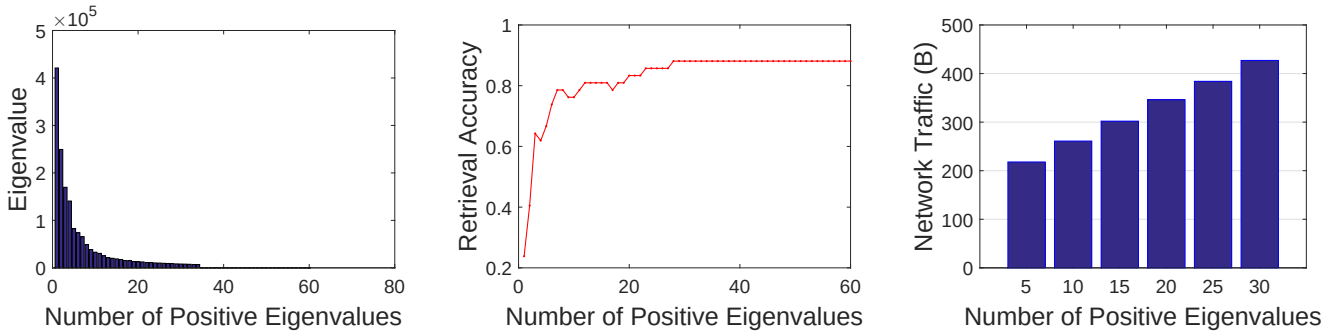
In this experiment, without prior knowledge, we randomly select 90% images from each individual for training, and the remaining are used for testing.

The results are given in Table 6. It can be seen that the retrieval accuracy of the approaches using *WAPL* algorithm is 6.9%, higher than that of the approaches without the *WAPL*. This is because that the projection matrix learned by the *WAPL* algorithm, has the capability of removing redundant features and extracting effective discriminative features from the image data set and the image to be retrieved.

5.5.4 Interaction Strategy

Based on the experimental results of relationship between the retrieval accuracy and the number of eigenvalues, only the projection matrix consists of the eigenvectors corresponding to the positive eigenvalues, which contributes to extract discriminative features. Hence, we need to investigate the relationship between eigenvalue, retrieval accuracy, network traffic and the number of positive eigenvalues. Fig. 12 shows that retrieval accuracy and network traffic rise with the increase of the number of positive eigenvalues. This indicates that the higher the accuracy is, the more network traffic is required.

For scenarios that require higher retrieval accuracy and less stringent response time, we can choose all the number of positive eigenvalues as the dimension of the low-dimensional feature space, the corresponding retrieval ac-



(a) (b) (c)

Fig. 12. Relationship between eigenvalue, retrieval accuracy, network traffic and the number of positive eigenvalues

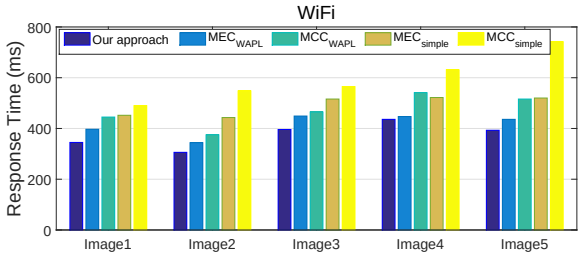


Fig. 11. Response time for different approaches.

TABLE 6
Comparison of Retrieval Accuracy on Lab_face Data Set

Approach	Retrieval Accuracy (%)
MCC_{simple}	81.43%
MCC_{WAPL}	88.33%
MEC_{simple}	81.43%
MEC_{WAPL}	88.33%
Our approach	**88.33%**

curacy is 88.33%, and network traffic is 427 B. For scenarios that require real-time response time but low retrieval accuracy, we can choose $1/3$ the number of positive eigenvalues, the corresponding retrieval accuracy is 76.19%, and network traffic is 261 B. For scenarios that require both retrieval accuracy and response time, $1/2$ the number of positive eigenvalues can be chose, the corresponding retrieval accuracy is 80.95%, and network traffic is 302 B. The higher the retrieval accuracy is, the larger the dimension of the low-dimensional feature space is required, resulting in larger network traffic and longer response time. Therefore, users can choose different interaction strategies in terms of retrieval accuracy and response time.

From the above experimental results, it is demonstrated that the cloud-guided feature extraction can extract fewer and more effective discriminative features to improve image retrieval accuracy and significantly reduce network traffic, as well as meet different requirements of users.

6 Related Work

6.1 Image Retrieval

Image retrieval [25], [26] has been a hot research topic in the computer vision for decades, which aims to retrieval labels of similar images from data sets. In the following, we will discuss two main procedures in an image retrieval system, namely feature extraction, and feature matching.

Feature extraction aims to extract discriminative features from the original high-dimensional data sets. In general, it consists of two steps. The first step is to learn the projection matrix. The second step is to use the projection matrix to extract the discriminative features from the original image data set to form a low-dimensional feature data set. Local binary patterns [9] extracts features to preserve the intrinsic structure of the image. Principle component analysis [28] aims to preserve the global information of the image. However, they do not utilize the label information of the images, so that it is impossible to extract discriminative features that are useful for image retrieval. To remedy this, many feature extraction algorithms using label information to learn projection matrix are proposed, such as [23], [24], [34], [35]. Among them, linear discriminant analysis [24] preserves the global relationship of the image data. Marginal Fisher analysis [34] focuses on local pairwise relationship of the image data. Joint global and local-structure discriminant analysis [35] learns the projection matrix by considering both global and local structures. However, the importance of the two types of structures are viewed equally in dealing with different data sets. In practice, different relationships contribute differently in dealing with different data sets. Ignoring any of them or improperly integrating them may seriously impair the effectiveness of the learned projection matrix. In addition, although the above algorithms are committed to learn an effective projection matrix, the dimensions of the projection matrix are estimated empirically. The inability to automatically determine the dimensions of the projection matrix affects their applications since it requires considerable manpower cost to tune them.

Feature matching aims to design effective classifiers to recognize different images. There are multi-class classifiers, such as the nearest neighbor classifier [27] and support vector machine [32]. Feature matching is the most time-consuming procedure in a real image retrieval system since the image to be retrieved needs to be matched with all the images stored in the image data set, and the images stored in the image data set are high-dimensional.

6.2 Mobile Edge Computing

Mobile Edge Computing (MEC) [12], [21], [22] has recently emerged as a new computing paradigm with proximate access and is a promising complementary to the centralized Mobile Cloud Computing (MCC) [8]. In the MEC paradigm, where a number of small scale servers are placed at the

network edge and they can be reached by nearby mobile users via LTE or WiFi connection. The main idea of MEC is dispersing data storage, process, and applications on devices located at the network edge rather than implementing almost entirely in the remote cloud servers. Compared with MCC, network traffic and network transmission delay of MEC are notably reduced, since the computation and storage resources are much closer to the mobile users.

There are many research work about MEC [4], [5], [10], [13]–[20], [37]. Among them, *Soyata et al.* [10] proposed a mobile-cloudlet-cloud architecture, which aims at performing tasks load among cloud servers to minimize the response time. *Liu et al.* [13] proposed a food recognition system based edge computing service, which preprocessed the captured food image on the mobile devices before uploading it to the remote cloud servers. It can significantly reduce network traffic and network transmission delay. *Hu et al.* [15] proposed a face identification and resolution scheme based on fog computing, which could reduce network traffic by offloading partial process of the image data on fog nodes. All of these schemes benefit from the MEC architecture, by making efficient use of computation and storage resources of the edge servers. However, they do not fully consider the interaction between the cloud servers and the edge servers, which is important for image retrieval applications.

7 Conclusion

In this paper, we have proposed a cloud-guided feature extraction approach for image retrieval in MEC, which aims to improve retrieval accuracy, reduce network traffic and response time, as well as meet different requirements of users. In the proposed approach, a projection matrix learning algorithm is proposed to generate an effective projection matrix, which guides the feature extraction from the image to be retrieved. Thus, fewer and more effective features can be extracted. The edge servers only upload the image feature data to the cloud servers, thus can significantly reduce network traffic and response time. The advantages of the proposed approach have been demonstrated by a prototype system using a real MEC environment.

References

[1] L. Zhu, J. L. Shen, L. Xie, and Z. Y. Cheng, "Unsupervised Visual Hashing with Semantic Assistant for Content-Based Image Retrieval," *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 2, pp. 472-486, 2017.

[2] W. T. Xu, Y. R. Shen, N. Bergmann, and W. Hu, "Sensor-Assisted Multi-View Face Recognition System on Smart Glass," *IEEE Transactions on Mobile Computing*, vol. 17, no. 1, pp. 197-210, 2018.

[3] M. Shahzad, A. X. Liu, and A. Samuel, "Behavior Based Human Authentication on Touch Screen Devices Using Gestures and Signatures," *IEEE Transactions on Mobile Computing*, vol. 16, no. 10, pp. 2726-2741, 2017.

[4] Y. X. Sun, S. Zhou, and J. Xu, "EMM:Energy-Aware Mobility Management for Mobile Edge Computing in Ultra Dense Networks," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 11, pp. 2637-2646, 2017.

[5] X. Ma, S. Zhang, P. Yang, N. Zhang, C. Lin, and X. M. Shen, "Cost-Efficient Resource Provisioning in Cloud Assisted Mobile Edge Computing", In *Proceedings of the IEEE Global Communications Conference*, pp. 1-6, 2017.

[6] J. Zhang, Z. F. Zhang, and H. Guo, "Towards Secure Data Distribution Systems in Mobile Cloud Computing," *IEEE Transactions on Mobile Computing*, vol. 16, no. 11, pp. 3222-3235, 2017.

[7] Y. C. Liu, M. J. Lee, and Y. Y. Zheng, "Adaptive Multi-Resource Allocation for Cloudlet-Based Mobile Cloud Computing System," *IEEE Transactions on Mobile Computing*, vol. 15, no. 10, pp. 2398-2410, 2016.

[8] S. Yang, D. Kwon, H. Yi, Y. Cho, Y. Kwon, and Y. Paek, "Techniques to Minimize State Transfer Costs for Dynamic Execution Offloading in Mobile Cloud Computing," *IEEE Transactions on Mobile Computing*, vol. 13, no. 11, pp. 2648-2660, 2014.

[9] T. Ojala, M. Pietikainen, and D. Harwood, "Performance Evaluation of Texture Measures with Classification Based on Kullback Discrimination of Distributions," In *Proceedings of the International Conference on Pattern Recognition*, pp. 582-585, 1994.

[10] T. Soyata, R. Muraleedharan, C. Funai, M. Kwon, and W. Heinzelman, "Cloud-Vision: Real-time Face Recognition Using a Mobile-Cloudlet-Cloud Acceleration Architecture," In *Proceedings of the IEEE Symposium on Computers and Communications*, pp. 59-66, 2012.

[11] F. Jalali, K. Hinton, R. Ayre, T. Alpcan, R. S. Tucker, "Fog Computing May Help to Save Energy in Cloud Computing," *IEEE Journal on Selected Areas In Communications*, vol. 34, no. 5, pp. 1728-1739, 2016.

[12] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, "Mobile Edge Computing: A Survey," *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 450-465, 2017.

[13] C. Liu, Y. Cao, Y. Luo, G. L. Chen, V. Vokkarane, Y. S. Ma, S. Q. Chen, and P. Hou, "A New Deep Learning-based Food Recognition System for Dietary Assessment on An Edge Computing Service Infrastructure," *IEEE Transactions on Service Computing*, vol. 11, no. 2, pp. 249-261, 2018.

[14] X. Chen, L. Jiao, W. Z. Li, and X. M. Fu, "Efficient Multi-User Computation Offloading for Mobile-Edge Cloud Computing," *IEEE/ACM Transactions on Networking*, vol. 24, no. 5, pp. 2795-2808, 2016.

[15] P. F. Hu, H. S. Ning, T. Qiu, Y. F. Zhang, and X. Luo, "Fog Computing-Based Face Identification and Resolution Scheme in Internet of Things," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 4, pp. 1910-1920, 2017.

[16] Y. Y. Mao, J. Zhang, K. B. Letaief, "Dynamic Computation Offloading for Mobile-Edge Computing with Energy Harvesting Devices," *IEEE Journal on Selected Areas In Communications*, vol. 34, no. 12, pp. 3590-3605, 2016.

[17] Y. Xiao, and M. Krunz, "QoE and Power Efficiency Tradeoff for Fog Computing Networks and Fog Node Cooperation," In *Proceedings of the IEEE International Conference on Computer Communications*, pp. 1-9, 2017.

[18] L. Tong, Y. Li, and W. Gao, "A Hierarchical Edge Cloud Architecture for Mobile Computing," In *Proceedings of the IEEE International Conference on Computer Communications*, pp. 1-9, 2016. 1002-1016, 2017.

[19] A. Ceselli, M. Premoli, and S. Secci, "Mobile Edge Cloud Network Design Optimization," *IEEEACM Transactions on Networking*, vol. 25, no. 3, pp. 1818-1831, 2017.

[20] S. N. Shirazi, A. Gouglidis, A. Farshad, and D. Hutchison, "The Extended Cloud: Review and Analysis of Mobile Edge Computing and Fog From a Security and Resilience Perspective," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 11, pp. 2586-2595, 2017.

[21] Y. Sarikaya, H. Inaltekin, T. Alpcan, and J. S. Evans, "Stability and Dynamic Control of Underlay Mobile Edge Networks," *IEEE Transactions on Mobile Computing*, vol. 17, no. 9, pp. 2195-2208, 2018.

[22] S. Zhang, P. He, K. Suto, P. Yang, L. Zhao, X. M. Shen, "Cooperative Edge Caching in User-Centric Clustered Mobile Networks," *IEEE Transactions on Mobile Computing*, vol. 17, no. 8, pp. 1791-1805, 2018.

[23] X. L. Li, M. L. Chen, F. P. Nie, and Q. Wang, "Locality Adaptive Discriminant Analysis," In *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 19-25, 2017.

[24] R. Haeb-Umbach, and H. Ney, "Linear Discriminant Analysis for Improved Large Vocabulary Continuous Speech Recognition," In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 13-.16, 1992.

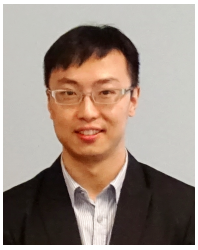
[25] W. G. Zhou, H. Q. Li, J. Sun, and Q. Tian, "Collaborative Index Embedding for Image Retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 5, pp. 1154-1166, 2018.

[26] X. S. Wei, J. H. Luo, J. X. Wu, and Z. H. Zhou, "Selective Convolutional Descriptor Aggregation for Fine-Grained Image Retrieval," *IEEE Transactions on Image Processing*, vol. 26, no. 6, pp. 2868-2881, 2017

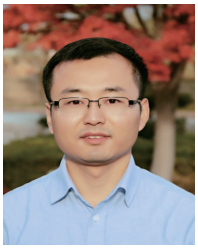
[27] B. L. Li, Q. Lu, S. W. Yu, "An Adaptive K-nearest neighbor text categorization strategy," *ACM Transactions on Asian Language Information Processing*, vol. 3, no. 4, pp. 215-226, 2004.
[28] A. M. Martinez, and A. C. Kak, "PCA versus LDA," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 2, pp. 228-233, 2001.
[29] D. B. Graham, and N. M. Allinson, "Characterising Virtual Eigensignatures for General Purpose Face Recognition," *Face Recognition*, pp. 446-456, 1998.
[30] F. P. Nie, W. Zhu, and X. L. Li, "Unsupervised Large Graph Embedding," In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 2422-2428, 2017.
[31] A. S. Georghiades, P. N. Belhumeur, and D. J. Kriegman, "From Few to Many: Illumination Cone Models for Face Recognition under Variable Lighting and Pose," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 643-660, 2001.
[32] R. E. Fan, K. W. Chang, C. J. Hsieh, X. R. Wang, C. J. Lin,"LIBLINEAR: A Library for Large Linear Classification," *Journal of Machine Learning Research*, vol. 9, pp. 1871-1874, 2008.
[33] C. T. Ding, and L. Zhang, "Double Adjacency Graphs-based Discriminant Neighborhood Embedding," *Pattern Recognition*, vol. 48, no. 5, pp. 1734-1742, 2015.
[34] S. C. Yan, D. Xu, B. Y. Zhang, H. J. Zhang, Q. Yang, and S. Lin, "Graph Embedding and Extensions: A General Framework for Dimensionality Reduction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 1, pp. 40-51, 2007.
[35] Q. X. Gao, J. J. Liu, H. L. Zhang, X. B. Gao, K. Li, "Joint Global and Local Structure Discriminant Analysis," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 4, pp. 626-635, 2013.
[36] P. Viola, and M. Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features," In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1-9, 2001.
[37] U. Drolia, K. Guo, J. Q. Tan, R. Gandhi, and P. Narasimhan, "Cachier: Edge-caching for Recognition Applications," In *Proceedings of the International Conference on Distributed Computing Systems*, pp. 276-286, 2017.

Shangguang Wang received his PhD degree at Beijing University of Posts and Telecommunications in 2011. He is an associate professor at the State Key Laboratory of Networking and Switching Technology (BUPT). He has published more than 100 papers, and played a key role at many international conferences, such as general chair and PC chair. His research interests include service computing, cloud computing, and mobile edge computing. He is a senior member of the IEEE, and the Editor-in-Chief of the International Journal of Web Science.

Chuntao Ding received the B.S. and M.S. degrees from SIAS International University in 2012 and Soochow University in 2015, respectively, both in software engineering. He is currently a Ph.D. candidate at the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications. His research interests include machine learning, mobile edge computing.

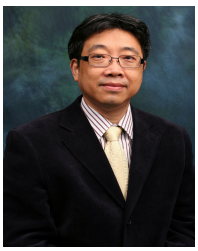


Ning Zhang is an Assistant Professor at Texas A&M University-Corpus Christi, USA. He received the Ph.D degree from University of Waterloo, Canada, in 2015. After that, he was a postdoc research fellow at University of Waterloo and University of Toronto, Canada, respectively. He serves/served as an associate editor of IEEE Access and IET Communication, an area editor of Encyclopedia of Wireless Networks (Springer) and Cambridge Scholars, a guest editor of Wireless Communication and Mobile Computing, International Journal of Distributed Sensor Networks, and Mobile Information System. He also served as the workshop chair for the first IEEE Workshop on Cooperative Edge. He is a recipient of the Best Paper Awards at IEEE Globecom 2014 and IEEE WCSP 2015, respectively. His current research interests include next generation mobile networks, physical layer security, machine learning, and mobile edge computing.



Xiulong Liu is currently a postdoctoral fellow in Department of Computing, Hong Kong Polytechnic University, Hong Kong, China. Before that, he received the B.E. degree and Ph.D. degree from the School of Software Technology and the School of Computer Science and Technology, Dalian University of Technology, China, in 2010 and 2016, respectively. His research interests include RFID systems and wireless sensor networks. He has published more than 30 research papers in prestigious journals and conferences including TON, TMC, TC, TPDS, INFOCOM, etc.

Ao Zhou received the Ph.D. degrees in Beijing University of Posts and Telecommunications, Beijing, China, in 2015. She is currently an Associate Professor with State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications. She has published 20+ research papers. She played a key role at many international conferences. Her research interests include Cloud Computing and Edge Computing.



Jiannong Cao (M'93-SM'05-F'14) received the Ph.D. degree in computer science from Washington State University, Pullman, WA, USA, in 1990. He is currently a Chair Professor of Department of Computing at The Hong Kong Polytechnic University, Hong Kong. He is also the director of the Internet and Mobile Computing Lab in the department and the director of Universitys Research Facility in Big Data Analytics. His research interests include parallel and distributed computing, wireless sensing and networks, pervasive and mobile computing, and big data and cloud computing.

Xuemin (Sherman) Shen (M97, SM02, F09) received the B.Sc. (1982) degree from Dalian Maritime University (China) and the M.Sc. (1987) and Ph.D. degrees (1990) from Rutgers University, New Jersey (USA), all in electrical engineering. He is a Professor and University Research Chair, Department of Electrical and Computer Engineering, University of Waterloo, Canada. He is also the Associate Chair for Graduate Studies. Dr. Shen's research focuses on resource management in interconnected wireless/wired networks, wireless network security, social networks, smart grid, and vehicular ad hoc and sensor networks. He is an elected member of IEEE ComSoc Board of Governor, and the Chair of Distinguished Lecturers Selection Committee. Dr. Shen served as the Technical Program Committee Chair/Co-Chair for IEEE Globecom'16, IEEE Infocom'14, IEEE VTC'10 Fall, the General Chair for ACM Mobihoc'15, the Symposia Chair for IEEE ICC'10, the Tutorial Chair for IEEE VTC'11 Spring and IEEE ICC'08, the Technical Program Committee Chair for IEEE Globecom'07, the General Co-Chair for Chinacom'07 and QShine'06, the Chair for IEEE Communications Society Technical Committee on Wireless Communications, and P2P Communications and Networking. He also serves/served as the Editor-in-Chief for IEEE Network, IEEE Internet of Things Journal, Peer-to-Peer Networking and Application, and IET Communications; a Founding Area Editor for IEEE Transactions on Wireless Communications; an Associate Editor for IEEE Transactions on Vehicular Technology, Computer Networks, and ACM/Wireless Networks, etc.; and the Guest Editor for IEEE JSAC, IEEE Wireless Communications, IEEE Communications Magazine, and ACM Mobile Networks and Applications, etc. Dr. Shen received the Excellent Graduate Supervision Award in 2006, and the Outstanding Performance Award in 2004, 2007 and 2010 from the University of Waterloo, the Premiers Research Excellence Award (PREA) in 2003 from the Province of Ontario, Canada, and the Distinguished Performance Award in 2002 and 2007 from the Faculty of Engineering, University of Waterloo. Dr. Shen is a registered Professional Engineer of Ontario, Canada, an IEEE Fellow, an Engineering Institute of Canada Fellow, a Canadian Academy of Engineering Fellow, a Royal Society of Canada Fellow, and a Distinguished Lecturer of IEEE Vehicular Technology Society and Communications Society.