Scalable Mobile Visual Classification by Kernel Preserving Projection over High-Dimensional Features

Yu-Chuan Su, Tzu-Hsuan Chiu, Yin-Hsi Kuo, Chun-Yen Yeh, Winston H. Hsu

***Abstract*—Scalable mobile visual classification – classifying images/videos in a large semantic space on mobile devices in real time – is an emerging problem as observing the paradigm shift towards mobile platforms and the explosive growth of visual data. Though seeing the advances in detecting thousands of concepts in the servers, the scalability is handicapped in mobile devices due to the severe resource constraints within. However, certain emerging applications require such scalable visual classification with prompt response for detecting local contexts (e.g., Google Glass) or ensuring user satisfaction. In this work, we point out the ignored challenges for scalable mobile visual classification and provide a feasible solution. To overcome the limitations of mobile visual classification, we propose an unsupervised linear dimension reduction algorithm – kernel preserving projection (KPP), which approximates the kernel matrix of high dimensional features with low dimensional linear embedding. We further introduce sparsity to the projection matrix to ensure its compliance with mobile computing (with merely 12% non-zero entries). By inspecting the similarity of linear dimension reduction with low-rank linear distance metric and Taylor expansion of RBF kernel, we justified the feasibility for the proposed KPP method over high-dimensional features. Experimental results on three public datasets confirm that the proposed method outperforms existing dimension reduction methods. What is even more, we can greatly reduce the storage consumption and efficiently compute the classification results on the mobile devices.**

***Index Terms*—Mobile Image Classification, Dimension Reduction, Distance Metric Learning, Manifold Learning**

I. Introduction

With the explosive growth of web images and videos, the semantic understanding for such big visual data is in dire needs; and the growth happens not only in the scale of data, but also in the number of concepts (or categories) to be detected [1]–[4]. Motivations for increasing the semantic space of visual recognition system come from research interests as well as real application needs; for example, since human can recognize tens of thousands of concepts from images and categorize them accordingly, an ideal automatic photo annotation system should also be able to do so. The scalable classification methods (e.g., linear classifier [5]) have been shown effective for recognizing high dimensional features in large semantic space [6]–[8], and by leveraging the distributed servers, we are now able to detect thousands of concepts in real time.

Meanwhile, the rapid development of mobile technologies has induced the paradigm shift from personal computer (PC) to mobile devices. On nowadays mobile devices, high quality camera becomes a basic component; and combining with the rich contexts in mobile devices, the camera can enable many proactive and smart applications such as remote healthcare, lifelog, automatic photo annotation. Many of these applications rely on *mobile visual recognition* – labeling or recognizing the semantic meaning (i.e., categories, scenes, tags) of images or videos on the mobile devices.

Although large scale visual recognition has been enabled by many promising technologies [6]–[10], most of them are designed for servers with abundant computing resources. However, the computing resources on the mobiles are more restricted. One of the most severe resource constraint is the *storage limit*, where the storage of current mobile devices are in dozens of GBs, which is much smaller than regular servers. Unfortunately, many state-of-the-art visual recognition systems rely on high dimensional features [6], which result in complicated models that cannot be stored on mobile devices. To overcome the limits, we propose a new linear dimension reduction entailed by a sparse projection matrix that aims at preserving classification performance. The resultant classification models and projection matrix can fit in mobile devices and enable scalable mobile visual recognition on devices directly.

The primary contributions of this paper include:

- We address the importance and requirements of *scalable mobile visual recognition* and propose a feasible system design. While there exist works aiming at either mobile or scalable visual recognition, the challenge of combining the two has not been investigated.
- We propose a new dimension reduction method, *Kernel Preserving Projection* (KPP), especially designed for mobile visual classification. KPP preserves the classification performance of low dimensional linear classifiers; furthermore, the projection matrix is sparse and can easily

Y.-C. Su is with the Department of Computer Science and Information Engineering, National Taiwan University, Taipei 106, Taiwan (e-mail:ycsu@cmlab.csie.ntu.edu.tw).

T.-H. Chiu is with the Department of Computer Science and Information Engineering, National Taiwan University, Taipei 106, Taiwan (e-mail:d98944011@ntu.edu.tw).

Y.-H. Kuo is with the Graduate Institute of Networking and Multimedia, National Taiwan University (e-mail: kuonini@cmlab.csie.ntu.edu.tw).

C.-Y. Yeh is with the Department of Computer Science and Information Engineering, National Taiwan University, Taipei 106, Taiwan (e-mail:chunyen0702@gmail.com).

W. H. Hsu is with the Graduate Institute of Networking and Multimedia and the Department of Computer Science and Information Engineering, National Taiwan University, Taipei 10617, Taiwan (e-mail: winston@csie.ntu.edu.tw). Prof. Hsu is the contact person.

fit in current mobile devices.
- We justify the feasibility of KPP by Taylor expansion of kernel function. Experimental results over three public datasets also confirm that it outperforms existing dimension reduction techniques in classification.

The remaining of this paper is organized as follows. In section II, we describe related work. In section III, we discuss the issues for mobile computing and describe our system design to fulfill the requirements. In section IV, we propose a new linear dimension reduction especially compliant with mobile computing. We show the experimental results in section V. Discussions and conclusions are in section VI.

II. Related Work

A. Scalable visual recognition

Large scale visual recognition is a very active research topic in recent years [2], [6], [11], [12]. The introduction of ImageNet dataset [1], which contains more than 20,000 concepts, has enabled such studies. The importance of scalable visual recognition has been addressed by multiple research groups [2], [3], [11], and many interesting properties and challenges for large scale visual recognition have been reported.

Two common components for state-of-the-art large scale visual recognition systems are high dimensional image features and linear support vector machine (SVM) [6], [7]. High dimensional image features formed by pooling local features are introduced to improve the discriminability, while linear classifiers ensure the efficiency and tractability of both training and testing [5]. The most popular pooling methods include bag-of-feature (BoF) methods [13] with spatial pyramid (SPM) [14] and residual based methods such as vector of locally aggregated descriptors (VLAD) [15]. All these features are high dimensional and lead to complicated model which requires large storage and may not fit in mobile devices. Even with linear classifiers, the model may still grow too large when the size of semantic space increases, which limits the scalability of mobile visual recognition system.

One obvious challenge for large scale visual recognition is how to train the classifiers efficiently, which leads to the popularity of linear classifier such as linear SVM. To further improve the learning speed, many systems adopt primal space solver and use stochastic gradient descend for learning [6]. Recently, efforts have also been made to improve the testing efficiency. In [16], the authors speed up the classification efficiency given a big trained model which may contain millions of classifiers by hashing both classifiers and features to Hamming space and use the Hamming distance to approximate the original classifier. By replacing inner product on high dimensional floating point vector with compact hamming code, the classification process can be 20 to 200 times faster than using original classifiers, and the size of the classifier can also be significantly reduced. Although prediction with trained model is usually much more efficient than training the model, these improvements help to reduce the resource requirement for testing, which is especially important in limited resource environment such as mobile devices.

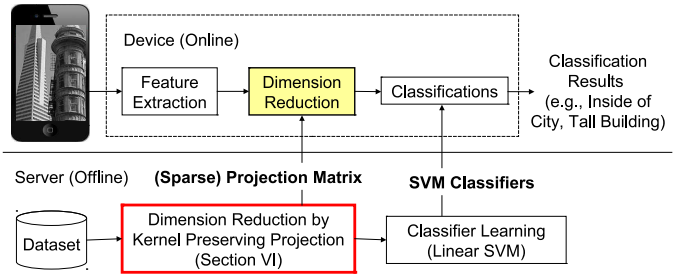


Fig. 1: System overview for mobile visual classification. The photo (or video) is recorded by the camera on the mobile device. A high-dimensional feature is computed by the device, then the feature dimension is reduced by a sparse projection matrix learned offline. The classifiers are learned in the reduced dimension space. Multiple classifications are thus performed efficiently on the device using the low-dimensional linear support vector machine (SVM). The key for mobile visual classification is whether the projection matrix preserves the classification accuracy and is feasible in more compact representation (i.e., sparsity).

B. Mobile visual search

Mobile visual search has been widely studied in the past few years [17], [18]. The most popular framework of mobile visual search system is to compute image features on the mobile device and send the features to server, where the server performs retrieval and returns the result. With the computing power of current mobile devices, the feature computation can be done within 2 seconds [17]; in fact, in our preliminary implementation, it takes only around 0.35 seconds to pool SURF local features and generate the VLAD signature (by sparse projection matrix) on iPhone 5. However, due to the network bandwidth limitation, transmitting (local or aggregated high-dimensional) features over wireless network can be very slow and the main challenge of mobile visual search is to reduce the traffic between devices and servers. The feature size reduction is mainly through hashing the features to generate representative (binary) signatures [9], [18], [19]. Although the success in mobile visual search brings light to the possibility of scalable mobile visual recognition systems, directly porting existing large scale visual classification system on mobile devices is infeasible; see section III for more detailed discussions.

An important lesson learned from mobile visual search system is that the ground truth may growth and change over time due to the appearance change of concepts, where the information may be captured through newly contributed content or user feedback [20]. Therefore, a system that can adapt to updated training set is highly desired for real mobile application, which raises the need for distance metric and recognition model that can be updated easily in online system.

C. Dimension reduction

Dimension reduction is a common preprocessing step before performing classification. It reduces the storage and memory requirement of training data as well as the resultant model by

reducing the input space. It may also improve performance by rejecting noisy features before training to avoid overfitting.

While there exists many dimension reduction algorithms, not all of them fit in the context of mobile computing. In particular, the most popular dimension reduction method in mobile visual search systems is by linear hashing, such as random projection (RP) [9], random maximum margin hashing (RMMH) [21]. These linear hashing methods require binarization after linear projection as the last step, while binarization introduces quantization error which is ignored in most hashing method yet may introduce significant degradation in performance, especially in eigenvector based projection. Iterative quantization (ITQ) [22] is designed to reduce quantization error of binary code by applying an additional rotation after linear projection. Since linear hashing is essentially a linear projection with binarization of the projected vector, it can be computed very efficiently for unseen data, which is important in mobile applications where real-time response is desired.

Among all of the hashing algorithms, the semi-supervised Sequential Projection Learning for Hashing (SPLH) [19] achieves the state-of-the-art performance by utilizing the (supplemental) data pair information. A sparse version of SPLH is also proposed to better fit into mobile devices [18], where the sparsity reduces the storage requirement and computational cost of the projection. It is worth noting that when applying SPLH to a classification problem, the class label can be used for the label matrix $\mathbf{S}$ and lead to a supervised algorithm.

Besides dimension reduction, a feature compression method – product quantization (PQ) [10] – is also adopted in large scale visual recognition. Although both PQ and linear hashing reduce the training data size and enable the learning process over big data, PQ does not reduce the number of parameters to learn nor reduce the model size, because the classifier is learned in the original input space.

D. Distance metric learning

Distance metric learning is an active research area in machine learning [23]–[26]. The goal of distance metric learning is to find the optimal distance metric $d(i, j)$ between data points i and j, where the criteria of "optimal" is application dependent.

Based on the form of $d(i, j)$ and the learning process, distance metric can be categorized into either linear or nonlinear and supervised or unsupervised. There is a direct link between linear projection and linear distance metric: since a linear distance metric can be formulated as

$$d(\mathbf{x}, \mathbf{y}) = (\mathbf{x} - \mathbf{y})^T \mathbf{M} (\mathbf{x} - \mathbf{y}), \tag{1}$$

with $\mathbf{M}$ being positive semidefinite and can be decomposed into $\mathbf{M} = \mathbf{L}^T \mathbf{L}$, a linear distance metric is equivalent to a feature transform with linear projection $\phi(x) = \mathbf{L}\mathbf{x}$ such that

$$d(\mathbf{x}, \mathbf{y}) = (\mathbf{L}\mathbf{x} - \mathbf{L}\mathbf{y})^T (\mathbf{L}\mathbf{x} - \mathbf{L}\mathbf{y}). \tag{2}$$

And the inner product in the transformed feature space becomes $\phi(\mathbf{x})\phi(\mathbf{y}) = \mathbf{x}^T \mathbf{L}^T \mathbf{L}\mathbf{y}$.

Many unsupervised distance metric learning algorithms are essentially designed for dimension reduction, such as principal component analysis (PCA), multidimensional scaling (MDS) and locality linear embedding (LLE) [27]. Kandola *et al.* argue the theoretic benefit of linear distance metric is that it can capture the correlation between different dimensions by the off-diagonal terms [28]. For high dimensional features, the correlation between different dimensions should be weak and which results in a sparse distance metric [24], [25]. In [26], the authors further argued that the eigenvalues of distance metric $\mathbf{M}$ should also be sparse, which leads to a low-rank distance metric that performs dimension reduction by nature.

Despite the close connection of linear distance metric and linear hashing, the benefit of distance metric is ignored in previous works on linear hashing for mobile visual search. In this paper, we consider the theoretical benefit of distance metric and design a new linear dimension reduction accordingly. The resultant dimension reduction is closely related to MDS, which aims to learn a low dimensional embedding that preserves pair-wise distance. Please see section IV for details.

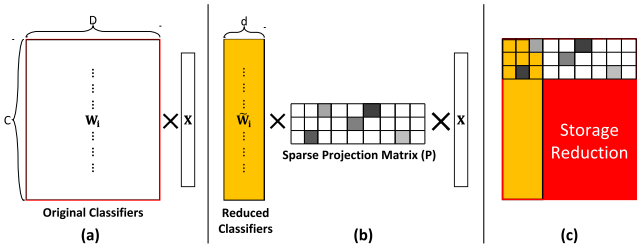


Fig. 2: The storage reduction by linear dimension reduction. Given D-dimensional feature and C-categories for classification (e.g., by linear SVM), original classifiers require storage of CD real values as in (a); performing dimension reduction with a projection matrix with d-dimension output as shown in (b) requires $(D+C)d$ storage, if we further consider sparsity of the projection matrix and let r be the ratio of non-zero matrix element ($r \approx 12\%$ in our experiment), the storage reduction becomes $CD - (rD + C)d$ as illustrated in (c). The reduction also corresponds to computation relatively. (Best seen in color)

III. Scalable Mobile Visual Classification

In this section, we describe our system design for scalable mobile visual classification system. We first discuss the important issues of mobile computing that have to be carefully handled in mobile visual classification system. We then describe our system design and explain how the design fulfills those requirements.

A. Issues of mobile computing

Two of the most important issues for mobile computing is the resource constraints and the requirement for instant response. Resource constraints include storage, computing power, network bandwidth, etc. The most significant one is storage, which restricts the amount of models that can be stored on the devices and limits the scalability of visual classification systems. Other constraints such as computing power limits the use of complicated models for classification, and

the network bandwidth forbids applications that require bulks network transmission. The requirement for instant response comes from the fact that long response time will significantly degrade user experiences. Ensuring the prompt response is even more difficult with the hardware constraints on mobiles. Therefore, a careful system design and algorithm optimization is important.

B. Goal and system overview

Seeing the issues and requirements in mobile visual classification, we preliminarily investigate the feasibility for performing visual classification purely on the mobile. There are two reasons to eliminate the dependency on wireless network; first, the reliability and coverage of wireless network is not satisfactory in many places; second, the network delay will degrade user experience [17]. We summarize the proposed system in fig. 1.

To ensure both classification accuracy and efficiency, we propose to adopt linear SVM with high dimensional visual feature, following many state-of-the-art large scale visual recognition systems [6], [7]. Nonlinear SVM requires the storage of multiple support vectors and the calculations of kernel function over all support vectors on classification, which is not suitable for mobiles in both storage and computation efficiency. The same concerns hold for nearest neighbor classifiers.

Although being more space efficient, the high dimensional linear SVMs still require huge storage which limits the scalability of semantic space. To overcome the limit, we "compress" the classifiers by reducing the input space with linear projection before classification, as illustrated in fig. 2. We also impose a sparse constraint on the projection matrix to reduce the storage overhead of projection matrix, otherwise the projection matrix may not fit in the memory of mobile devices (e.g., 200k to 512 dimension projection matrix takes 780MB). By reducing the size of both classifiers and projection matrix, we improve the scalability of native mobile visual classification systems. Note that because the size of classifiers and projection matrix is equal to the number of floating point operations when performing classification, reduction of storage also corresponds to reduction of computation. The design also improves updatability of the system by reducing the overhead for updating classification models and projection matrix, which is highly desirable or even necessary for real mobile applications.

There are three requirements for the mobile-classification-compliant dimension reduction. First, it has to be computation efficient. Second, it has to preserve the classification performance. Finally, the storage consumption should be small. To fulfill these requirements, we design a new linear dimension reduction algorithm – KPP, which will be described in details in the next section.

IV. Dimension Reduction by Kernel Preserving Projection (KPP)

Because KPP is a linear dimension reduction and can be computed efficiently, the resource limitation of mobile is addressed by nature. The further objectives of the new projection matrix learning algorithm are: (1) preserving the classification accuracy of the original features and (2) reducing the storage overhead of projection matrix.

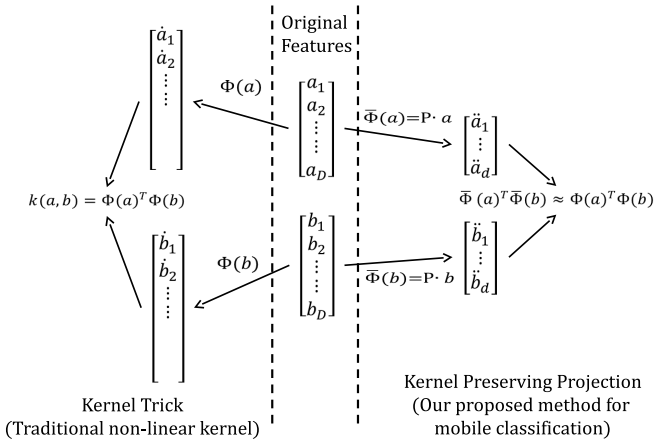


Fig. 3: Illustration for Kernel preserving projection (KPP). Conventionally, a kernel function is the inner product after performing feature transformation to a higher or even infinite-dimensional space (Left). Our proposal, KPP, goes another way (Right). It is a linear feature transformation by projection that "reduces" the dimension of the original features. The inner product of the signatures generated by KPP approximates the original kernel. The justification of approximation is in section IV-D.

The symbol is defined as follows. $\mathbf{X} \in \mathbb{R}^{D\times N}$ denotes the dataset containing N instances, with the column vector $\mathbf{x}_i$ denotes data point i in D dimensional space. $\mathbf{K}$ denotes the kernel matrix, with $\mathbf{K}_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$, where $k(,)$ is the kernel function.

A. Projection learning to approximate kernel matrix

Our primary goal is to find a linear feature transformation (by projection) where the classification performance of resultant feature is similar to the original ones. The goal is similar to that of feature map methods [29], which try to find a explicit feature transformation $\bar{\Phi}(x)$ where

$$k(x_i, x_j) \approx \bar{\Phi}(x_i) \cdot \bar{\Phi}(x_j). \tag{3}$$

Because kernel functions determine the input space of SVM (in dual form) by implicit feature transformation, the feature transformation $\bar{\Phi}(x)$ will yield a SVM similar to that of kernel function $k(,)$ and thus similar performance.

Traditionally, a feature map $\bar{\Phi}$ increases the feature dimension and therefore limits the scalability of data and feature dimension. Gavves *et al.* proposed [30] a feature selection and weighting method for additive kernels by learning the weights of feature dimensions such that the kernel matrix of resultant low dimensional features approximates the original kernel. Although the method improves scalability, it does not consider cross dimension correlations and applies only to additive kernel, while our method applies to general kernels.

Motivated by the feature map methods and the applicability in unseen images in mobile classification, we aim to learn a

projection matrix $\mathbf{P} \in \mathbb{R}^{d \times D}$ such that the resultant kernel matrix of low dimension signature approximates the kernel of the original features

$$\begin{aligned} \mathbf{K} &\approx (\mathbf{PX})^T(\mathbf{PX}) \\ &= \mathbf{X}^T\mathbf{P}^T\mathbf{PX}, \end{aligned} \tag{4}$$

as illustrated in fig. 3. Preliminarily we try to derive the projection matrix as

$$\mathbf{P}^* = \arg\min_{\mathbf{P}} \|\mathbf{K} - \mathbf{X}^T\mathbf{P}^T\mathbf{PX}\|_F, \tag{5}$$

where $\|\cdot\|_F$ denotes the Frobenius norm. The formulation is similar to multidimensional scaling, where the pair-wise distance equals to the kernel function $k(x_i, x_j)$.

B. Information-theoretic-based regularization

To avoid the projection matrix $\mathbf{P}$ from overfitting, we introduce a regularization to eq. 5 that maximizes the variance of pair-wise similarities of training data. In other words, we want the distribution of similarities spreads as wide as possible, similar to the equal partition objective in hashing algorithm like SPLH and RMMH. From an information theory point of view, if the probability distribution of random variable X is a normal distribution, its entropy is a function of variance

$$H(X) = \frac{1}{2} ln(2\pi e \sigma_X^2). \tag{6}$$

Therefore, maximizing variance is to maximize the entropy.

Assuming the data distribution is zero mean, maximizing kernel values variance of signatures will lead to

$$\mathbf{P}^* = \arg\max_{\mathbf{P}} \|\mathbf{X}^T\mathbf{P}^T\mathbf{PX}\|_F^2. \tag{7}$$

Combining with the original objective function, the resultant objective function can be formulated as:

$$\begin{aligned} \mathbf{P}^* = \arg\min_{\mathbf{P}} &\|\mathbf{K} - \mathbf{X}^T\mathbf{P}^T\mathbf{PX}\|_F \\ &-\lambda\|\mathbf{X}^T\mathbf{P}^T\mathbf{PX}\|_F^2. \end{aligned} \tag{8}$$

C. Sparse projection matrix

The last requirement for the projection matrix is to minimize the storage overhead. Given the input and output dimension of the projection matrix, storage reduction can be achieved by introducing the sparse constraint on the projection matrix (i.e., increase the number of zero entries). To add sparsity constraint on the projection matrix, we introduce an L_1 penalty to the objective function. Therefore, the final objective function becomes

$$\begin{aligned} \mathbf{P}^* = \arg\min_{\mathbf{P}} &\|\mathbf{K} - \mathbf{X}^T\mathbf{P}^T\mathbf{PX}\|_F \\ &- \lambda\|\mathbf{X}^T\mathbf{P}^T\mathbf{PX}\|_F^2 + \eta\|\mathbf{P}\|_1. \end{aligned} \tag{9}$$

Despite the practical necessity for sparse projection matrix, since it is also a distance metric and the input features are high dimensional, the matrix should be sparse as argued in [24], [26]. Our experimental results also show that even a very sparse projection matrix (i.e., 12% non-zero entries) can have competitive performance for mobile visual classification.

D. Learning cross dimension correlations through RBF kernel

In eq. 9, the target kernel $\mathbf{K}$ can be any kernels, and we have to specify it before learning. Because linear projection is very similar to distance metric as mentioned in section II-D, the target kernel should consider the theoretical benefits of distance metric. In particular, as argued in [24], [25], [28], distance metric captures the correlation between different dimensions; therefore, the target kernel should also contain cross dimension correlations. Among the popular kernels for visual recognition (e.g., linear, χ^2, intersection, RBF), RBF is the one that captures cross dimension correlations as explained in the next paragraph, and our experiments also show that RBF kernel has better classification accuracy (cf. section V-B). Therefore, we choose RBF as the target kernel.

Conventionally, the justification for RBF kernel is that it introduces an infinite dimensional feature transformation, but for high dimensional features, the contribution of high order feature transformation is actually very small. Consider the Taylor expansion of RBF kernel with the feature vectors normalized to unit length:

$$\begin{aligned} e^{-\gamma\|\mathbf{x}-\mathbf{y}\|^2} &= e^{-\gamma(2-2\mathbf{x}\cdot\mathbf{y})} \\ &= e^{-2\gamma}e^{2\gamma\sum_i \mathbf{x}_i\mathbf{y}_i} \\ &= e^{-2\gamma}\sum_n \frac{(2\gamma)^n}{n!}(\sum_i \mathbf{x}_i\mathbf{y}_i)^n. \end{aligned} \tag{10}$$

Bingham [9] shows that in very high-dimensional space, two random vectors $\mathbf{x}$, $\mathbf{y}$ are sufficiently close to be orthogonal, or more precisely speaking, given $\mathbf{x}$, $\mathbf{y} \in \mathbb{R}^D$,

$$\mathbf{x}^T\mathbf{y} \approx 0 \tag{11}$$

when $D \gg 0$. So eq. 10 is dominated by the leading terms

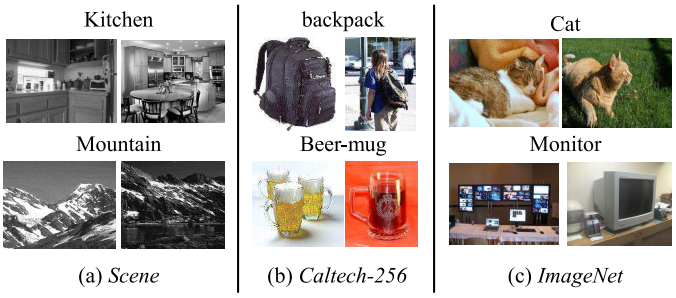
$$e^{-\gamma\|\mathbf{x}-\mathbf{y}\|^2} \approx c_0 + c_1\sum_i \mathbf{x}_i\mathbf{y}_i + c_2\sum_{i,j}\mathbf{x}_i\mathbf{x}_j\mathbf{y}_i\mathbf{y}_j. \tag{12}$$

Notice the first two terms are the same as the linear kernel, so the actual benefit of RBF kernel over linear kernel comes from the third term, which introduces the correlations between different dimensions. In other words, RBF kernel outperforms linear kernel because it considers the correlation between dimensions for high dimensional features, which is the same as linear distance metric. Therefore, we can approximate the correlation introduced by RBF kernel using the projection matrix $\mathbf{P}$ as discussed in section II-D.

E. Optimization solver

The final step is to solve the optimization problem in eq. 9. Note that the problem is not convex, so the initial guess of $\mathbf{P}$ affects the results. Instead of using the standard procedure for non-convex optimization problem by starting from several random initial guesses and selecting the one with the optimal result, we choose the initial guess as follows.

Because the main goal of our objective function is to find a projection matrix that preserves kernel values, which is embedded in the first term in eq. 9, we choose $\mathbf{P}_0$ that optimizes the first term. The projection matrix is guaranteed



(a) *Scene* (b) *Caltech-256* (c) *ImageNet*

Fig. 4: Example images from the three datasets.

to be optimal when $\mathbf{K} = \mathbf{X}^T\mathbf{P}^T\mathbf{P}\mathbf{X}$, which leads to an approximate solution for $\mathbf{P}_0^T\mathbf{P}_0$:

$$\mathbf{P}_0^T\mathbf{P}_0 \approx (\mathbf{X}^T)^\dagger\mathbf{K}\mathbf{X}^\dagger, \tag{13}$$

where $\mathbf{X}^\dagger$ represents the Moore-Penrose pseudoinverse. An approximated $\mathbf{P}_0$ is then computed by eigenvalue decomposition. Experimental results show that the initial guess of $\mathbf{P}_0$ is fairly successful.

Starting from the initial guess, we solve the optimization problem using the *L1General* solver [31]. The solver is a general solver for optimization problem with weighted L_1 regularization. We choose the active-set variant of projected scale sub-gradient algorithm.

V. Experiment

A. Experimental setup

Dataset. We evaluate the proposed method on three widely used datasets: *Scene* [14], *Caltech-256* [32] and *ImageNet* [1] datasets. *Scene* dataset contains 4,485 images with 15 categories, each category consists of images from a scene, such as "coast" or "office," and the number of images for each category ranges from 210 to 410. *Caltech-256* dataset contains 30,607 images in 256 object categories, and each category contains at least 80 images. We subsample 20 categories for evaluation. For *ImageNet* dataset, 19 categories from ImageNet 2011 Fall Release with the same categories of PASCAL Visual Object Classes 2007 Challenge [33] were selected, following the same protocol in [7] (we do not find the synset of "potted plant"). We discard images with resolution smaller than 500×300, which results in a total of 19,886 images. See fig. 4 for example images from the datasets.

Feature. We use VLAD [15] as the image feature, which has been adopted in mobile visual search system [18] for its performance and computation efficiency. Note that the proposed method is general and can be extended to other high-dimensional features. We use only single local feature, Dense SIFT, throughout the experiments. For dense sampling, 20×20 patches with overlapping windows shifted by 5 pixels are used. The codebooks are learned using hierarchical k-means with 16 centers for *ImageNet* and *Scene*. The resultant features are 2,048 dimensions. For *Caltech-256*, 64 centers with one level of SPM is used, which results in 40,960 dimensions of feature. Follow [7], the VLAD vector is first power normalized and then L_2 normalized, with $\alpha = 0.5$ for power normalization.

Evaluation criteria. The performance of the algorithm is evaluated by the classification accuracy on the test set. Ten folds of experiments are performed, and the average classification accuracies are reported. For *Scene* dataset, the images are first divided into ten subsets with one subset for testing and others for training in each fold of experiment. For *Caltech-256*, we follow the general experiment protocol and randomly sampled 30 images in each category for training in each fold. For *ImageNet*, we randomly split the dataset into training and test set with each category being equally partitioned. The parameters for SVM are chosen using 5-fold cross validation on training set, using grid search over the range of $2^{-10} \sim 2^{10}$.

Compared methods. For performance comparisons, several popular linear dimension reduction methods are also evaluated. In particular, we evaluate unsupervised Iterative Quantization on Principal Component Analysis (ITQ) [22] and supervised Sequential Projection Learning for Hashing (SPLH) on *Scene* dataset. Because of the similarity between KPP and MDS, we also compare our method with two embedding methods, Neighbor Preserving Embedding (NPE) and Locality Preserving Projection (LPP). Note we only compare with linear embedding methods because our primary goal is to handle unseen data efficiently. For *Caltech-256*, we evaluate two unsupervised hashing methods that shows promising performance – Random Maximum Margin Hash (RMMH) [21] and Spherical Hashing (SPHH) [34]; we use only linear RMMH because our goal is a linear projection as described in section III-B. For *ImageNet*, we select SPLH and SPHH which performs better on *Scene* and *Caltech-256* respectively. Note that we do not binarize the signature because it does not reduce the storage overhead of classifiers which are always real-valued.

For our proposed KPP, two variants are evaluated. We first adopt only maximum variance regularization (KPP-MV) as in eq. 8. We then adopt both maximum variance and L_1 regularization (KPP-L1MV) by optimizing eq. 9. All classifications are performed with linear SVM using *LIBLINEAR* [35], except the performance evaluation on different SVM kernels which used *LIBSVM* [36].

Parameter Selection. The proposed methods KPP-MV and KPP-L1MV has three meta parameters. γ determines the target kernel matrix and therefore determined the target distance metric; λ determines the weight of regularization, and η determines the sparsity of the projection matrix. We empirically set γ to 0.5 in all experiments, which is based on our experience that RBF kernel with $\gamma \sim 0.5$ usually has reasonable performance. To fit our low storage requirement, the sparsity control parameter η is set so as less than 20% of the matrix elements are non-zero when the target dimension is 512. In practice, η is set to $4e^{-5}$ for *Scene*, $1e^{-4}$ for *Caltech256* and *ImageNet*. For λ, we perform grid search on 3 values (0.75, 0.5, 0.25) with *Scene* dataset and set it to 0.5 for all dataset.

In summary, we fixed the target distance metric and select the sparsity control parameter based on the system constraint. The only parameter that is selected based on performance is the regularization control parameter, which is determined using only one dataset and applied to others to simulate real

TABLE I: Classification accuracy of different kernels using 2,048 dimension VLAD feature on *Scene* dataset. RBF kernel has the best performance over other kernels that are widely adopted in visual classification.

Kernel	Linear	RBF	χ^2
Accuracy	75.23	76.64	74.56

application scenario and avoid overfitting. For the experiments on *Caltech256* and *ImageNet* datasets, the proposed KPP has only one changeable parameter and is determined purely based on density of the projection matrix. Therefore, the results should be able to reflect the performance in the scenario of real applications.

For SPLH and RMMH, each of them has a single meta parameter, where η in SPLH controls the weight of supervised and regularization terms and M in RMMH controls the number of training samples for each hash function. We perform grid search on the meta parameter using 512 output dimensions on each dataset respectively and choose the best parameter. For SPHH, we use the default tolerance value $\epsilon_m = 0.1$ and $\epsilon_s = 0.15$ in the original authors' implementation.

B. Results

We first compare the performance of different kernels on *Scene* dataset. The classification accuracy using 2,048 dimension features is listed in table I, which are on par with those baselines reported in [37]. Note that for χ^2 kernel, we shift the original features to be all positive because the kernel is designed to work with positive feature value. While χ^2 kernel is widely adopted in image classification, it does not perform well with VLAD. And despite the promising performance of linear kernel [5], [6], RBF kernel slightly outperforms linear kernel (76.64% vs. 75.23%).

Next we compare the performance of different dimension reduction methods, as shown in fig. 5. The results on *Scene* are in fig. 5a, the proposed KPP can achieve similar or even better performance than the supervised SPLH. This is because while SPLH learns the pairwise similarity which does not guarantee separability, KPP learns a separable data distribution and improves the classification performance directly. Also note that KPP-MV performs better than the original feature with linear SVM using only 512 dimension signature. The reason for this superior performance is that KPP captures the inter dimensions correlation, and it optimizes all dimensions simultaneously, thus generates more informative signatures.

The experiments on *Caltech-256* are in fig. 5b, where KPP outperforms two state-of-the-art unsupervised hashing algorithms, SPHH and RMMH. The results on *ImageNet* also confirm the superior performance of KPP, as in fig. 5c. ITQ does not perform well in our experiments, because it is designed to reduce quantization error in binary codes while we use real value in our experiment setting. In fact, ITQ performs nearly identical with PCA on *Scene* dataset. Although NPE and LPP perform well in low dimension, their performances degrade as dimension increases, so their optimal performances are still poor even if we have enough resources for higher dimension signatures. Note KPP is the only algorithm that generates a sparse projection matrix, which is important for mobile computing.

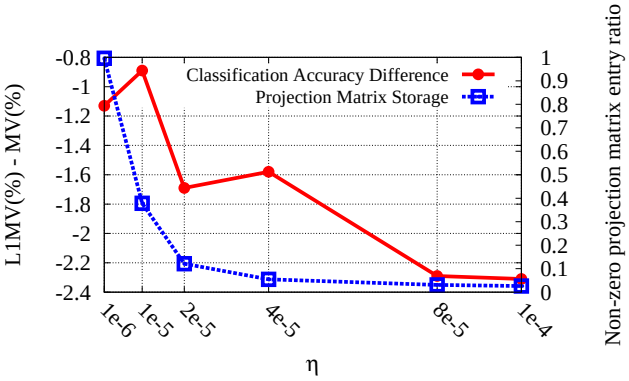


Fig. 6: The classification accuracy degradation and the storage reduction on *Scene* dataset under different η in eq. 9. The projection is from 2,048 dimensions to 512 dimensions. The classification accuracy (red curve) shows the difference between KPP-MV and KPP-L1MV (incurred by the sparsity). For storage reduction (blue curve), we calculate the ratio of non-zero elements between KPP-L1MV and KPP-MV. The classification accuracy decreases as η increases, but the decrease is very marginal with significant storage reduction.

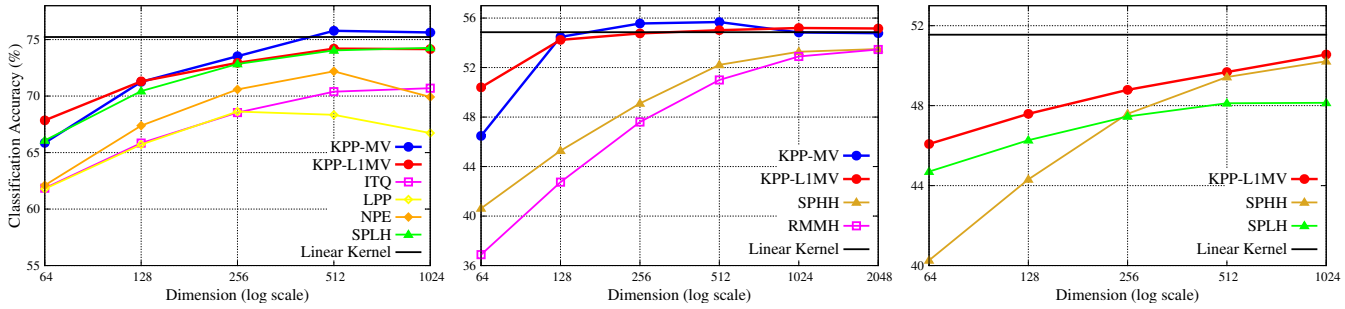
C. The effect of sparse projection matrix

In this section, we examine the effect of sparse constraint to the KPP algorithm. Because the sparsity of projection matrix is induced by the L_1 penalty term, which is controlled by its coefficient η in eq. 9, we evaluate the performance under different η values. As shown in fig. 6, increasing the weight of L_1 penalty reduces the size of projection matrix with the performance slightly degrades as well. But the performance degradation is moderate, and even using a sparse projection matrix achieves comparable performance with existing hashing methods such as SPLH.

VI. Discussions and Conclusions

In this paper, we address the emerging challenge of scalable mobile visual classification. Although scalable visual classification has been addressed in previous works, and applications based on mobile visual recognition are gaining attention for practical applications, the technical challenges of combining the two have not been discussed. Our analysis shows the intrinsic limitations of mobile visual classification and the drawbacks of applying existing techniques on the problem.

Based on the observed issues of mobile computing, we propose a purely native system for visual classification. To enable scalable visual classification of native mobile systems, we further develop a novel linear dimension reduction algorithm, KPP, that extends multidimensional scaling based on feature map methods, which also ensure the classification performance. The space efficient system design not only makes a native system possible but also reduces the model update overhead, which is important for real applications.



(a) *Scene* (b) *Caltech-256* (c) *ImageNet*

Fig. 5: Classification results on three widely used public datasets. (a) For *Scene* dataset, the dimension of original feature is 2,048. The performance of KPP outperforms the state-of-the-art supervised SPLH. (b) For *Caltech-256* dataset, the original feature dimension is 40,960, which leads to a huge covariance matrix that cannot fit in memory. Therefore, we cannot include SPLH in comparison; instead we introduce two state-of-the-art unsupervised hashing. KPP outperforms other methods even with fewer dimensions. (c) For *ImageNet* dataset, the original feature dimension is 2,048, and we include only SPLH and SPHH which performs best in *Scene* and *Caltech-256* respectively (except KPP). The result confirms the superior performance of KPP. (Best seen in color)

The performance of KPP benefits from the correlations between dimensions since it is a linear distance metric, as discussed in [24]–[26], [28]. Because the exact correlation is unknown, we try to learn it through approximating RBF kernel matrix, as discussed in section IV-D. Our experimental results on three popular datasets (*Scene*, *Caltech-256* and *ImageNet*) show that it outperforms the state-of-the-art linear hashing algorithms widely adopted in mobile visual search, both supervised and unsupervised, and the dimension reduction algorithm is compliant to mobile computation framework.

Although KPP is designed to reduce the size of classifiers with sparse projection matrix, because the kernel matrix can also be considered as a similarity matrix, KPP can also reduce the computational complexity of data similarity. In fact, the amount of data reduction also corresponds to the amount of computation reduction. Therefore, KPP can be extend to methods which require efficient similarity estimation. For example, in graph-based methods [38], the graph is represented by one or multiple similarity matrices, where each matrix element correspond to the similarity between two data instances. KPP can accelerate the similarity computation. Combining with the iterative graph learning algorithm, it may improve the learning efficiency of graph-based methods.

Note that although we use RBF-kernel for similarity in this work, the similarity may well be replaced by other distance metric; it is even possible to incorporate label information into the distance metric, so the unsupervised learning can be extended to supervised learning. However, unlike the case of RBF-kernel, there is no theoretical guarantee for arbitrary distance metric, and their performances require further verification.

In this work, we propose a purely native mobile visual classification system to avoid the dependency on wireless network. The correlations introduced by RBF kernel are, however, not exact, and a more precise correlation should further improve the performance. Therefore, we would like to improve the learning process of KPP in the future, such as a better method to learn the correlation between dimensions; another desired improvement is to speed up the offline learning process of projection matrix.

ACKNOWLEDGMENT

This work was supported in part by grants from the National Science Council of Taiwan, under Contracts NSC 101-2628-E-002-027-MY2, Excellent Research Projects of National Taiwan University, 102R7762, and MediaTek Inc.

REFERENCES

[1] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and F.-F. Li, "Imagenet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[2] J. Deng, A. C. Berg, K. Li, and F.-F. Li, "What does classifying more than 10,000 image categories tell us?" in *Proc. of the 11th European Conf. Computer Vision*, 2010, pp. 71–84.

[3] Z. Ma, F. Nie, Y. Yang, J. R. R. Uijlings, and N. Sebe, "Web image annotation via subspace-sparsity collaborated feature selection," *IEEE Trans. Multimedia*, vol. 14, no. 4, pp. 1021–1030, 2012.

[4] J. R. R. Uijlings, A. Smeulders, and R. Scha, "Real-time visual concept classification," *IEEE Trans. Multimedia*, vol. 12, no. 7, pp. 665–681, 2010.

[5] G.-X. Yuan, C.-H. Ho, and C.-J. Lin, "Recent advances of large-scale linear classification," *Proc. IEEE*, vol. 100, no. 9, pp. 2584–2603, 2012.

[6] A. Berg, J. Deng, and F.-F. Li, "Large scale visual recognition challenge 2010." [Online]. Available: http://www.image-net.org/challenges/LSVRC/2010/index

[7] F. Perronnin, J. Sánchez, and T. Mensnik, "Improving the fisher kernel for large-scale image classification," in *Proc. of the 11th European Conf. Computer Vision*, 2010, pp. 143–156.

[8] J. Wang and Y. Gong, "Discovering image semantics in codebook derivative space," *IEEE Trans. Multimedia*, vol. 14, no. 4, pp. 986–994, 2012.

[9] E. Bingham and H. Mannila, "Random projection in dimensionality reduction: applications to image and text data," in *Proc. 7th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, 2001, pp. 245–250.

[10] J. Sánchez and F. Perronnin, "High-dimensional signature compression for large-scale image classification," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2011, pp. 1665–1672.

[11] A. Torralba, R. Fergus, and W. Freeman, "80 million tiny images: A large data set for nonparametric object and scene recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 11, pp. 1958–1970, Nov. 2008.

[12] S. Zhu, C.-W. Ngo, and Y.-G. Jiang, "Sampling and ontologically pooling web images for visual concept learning," *IEEE Trans. Multimedia*, vol. 14, no. 4, pp. 1068–1078, 2012.

[13] J. Sivic and A. Zisserman, "Video google: a text retrieval approach to object matching in videos," in *Proc. 9th IEEE Int. Conf. Computer Vision*, 2003, pp. 1470–1477.

[14] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2006, pp. 2169–2178.

[15] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2010, pp. 3304–3311.

[16] S. Litayem, A. Joly, and N. Boujemaa, "Hash-based support vector machines approximation for large scale prediction," in *Proc. British Machine Vision Conference*, 2012.

[17] B. Girod, V. Chandrasekhar, D. Chen, N.-M. Cheung, R. Grzeszczuk, Y. Reznik, G. Takacs, S. Tsai, and R. Vedantham, "Mobile visual search," *IEEE Signal Processing Mag.*, vol. 28, no. 4, pp. 61–76, Jul. 2011.

[18] G.-L. Wu, Y.-H. Kuo, T.-H. Chiu, W. H. Hsu, and L. Xie, "Scalable mobile video retrieval with sparse projection learning and pseudo label mining," *IEEE Multimedia*, vol. 20, no. 3, pp. 47–57, 2013.

[19] J. Wang and S.-F. Chang, "Sequential projection learning for hashing with compact codes," in *Proc. 27th Int. Conf. Machine Learning*, 2010, pp. 1127–1134.

[20] H. Goëau, P. Bonnet, A. Joly, V. Bakić, J. Barbe, I. Yahiaoui, S. Selmi, J. Carré, D. Barthélémy, N. Boujemaa, J.-F. Molino, G. Duché, and A. Péronnet, "Pl@ntnet mobile app," in *Proc. 21st ACM Int. Conf. on Multimedia*, 2013.

[21] A. Joly and O. Buisson, "Random maximum margin hashing," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2011, pp. 873–880.

[22] Y. Gong and S. Lazebnik, "Iterative quantization: A procrustean approach to learning binary codes," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2011.

[23] K. Q. Weinberger and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," *J. Mach. Learn. Res.*, vol. 10, pp. 207–244, Jun. 2009.

[24] G.-J. Qi, J.-H. Tang, Z.-J. Zha, T.-S. Chua, and H.-J. Zhang, "An efficient sparse metric learning in high-dimensional space via l1-penalized log-determinant regularization," in *Proc. 26th Int. Conf. Machine Learning*, 2009, pp. 841–848.

[25] W. Liu, S.-Q. Ma, D.-C. Tao, J.-Z. Liu, and P. Liu, "Semi-supervised sparse metric learning using alternating linearization optimization," in *Proc. 16th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, 2010, pp. 1139–1148.

[26] Y. Hong, Q.-N. Li, J.-Y. Jiang, and Z.-W. Tu, "Learning a mixture of sparse distance metrics for classification and dimensionality reduction," in *Proc. 13th IEEE Int. Conf. Computer Vision*, 2011, pp. 906–913.

[27] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, Dec. 2000.

[28] J. Kandola, J. Shawe-Taylor, and N. Cristianini, "Learning semantic similarity," in *Proc. Conf. Advances in Neural Information Processing Systems*, 2003, pp. 657–664.

[29] S. Maji and A. C. Berg, "Max-margin additive classifiers for detection," in *Proc. 11th IEEE Int. Conf. Computer Vision*, 2009, pp. 40–47.

[30] E. Gavves, C. G. M. Snoek, and A. W. M. Smeulders, "Convex reduction of high-dimensional kernels for visual classification," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2012, pp. 3610–3617.

[31] M. Schmidt, "Graphical model structure learning with l1-regularization," Ph.D. dissertation, Univ. British Columbia, 2010.

[32] G. Griffin *et al.*, "Caltech-256 object category dataset," California Institute of Technology, Tech. Rep., 2007. [Online]. Available: http://authors.library.caltech.edu/7694

[33] M. Everingham *et al.*, "The pascal visual object classes challenge 2007 (voc2007) results," 2007.

[34] J.-P. Heo, Y. Lee, J. He, S.-F. Chang, and S.-E. Yoon, "Spherical hashing," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2012, pp. 2957–2964.

[35] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "Liblinear: A library for large linear classification," *J. Mach. Learn. Res.*, vol. 9, pp. 1871–1874, Jun. 2008.

[36] C.-C. Chang and C.-J. Lin, "Libsvm: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, May 2011.

[37] M.-Q. Xu, X. Zhou, Z. Li, B.-Q. Dai, and T. Huang, "Extended hierarchical gaussianization for scene classification," in *Proc. 17th IEEE Conf. Image Processing*, 2010, pp. 1837–1840.

[38] M. Wang, X.-S. Hua, R. Hong, J. Tang, G.-J. Qi, and Y. Song, "Unified video annotation via multigraph learning," *IEEE Trans. Cir. and Sys. for Video Technol.*, vol. 19, no. 5, pp. 733–746, May 2009.

Yu-Chuan Su received B.S. from the Department of Physics and Computer Science at the National Taiwan University. He is pursuing the M.S. degree in the Computer Science and Information Engineering Department of the National Taiwan University. His research interests in computer vision and machine learning focus on visual semantic understanding and multimedia content analysis.

Tzu-Hsuan Chiu is a doctoral student in the Graduate Institute of Networking and Multimedia at the National Taiwan University. His research interests include multimedia content analysis and machine learning. Chiu received his M.S. in computer science and information engineering from the National Taiwan University.

Yin-Hsi Kuo is a doctoral student in the Graduate Institute of Networking and Multimedia at the National Taiwan University. Her research interests include multimedia content analysis and image retrieval. Kuo has an MS in computer science and information engineering from the National Taiwan University.

Chun-Yen Yeh received the B.S. degree from the Department of Computer Science, National Chiao Tung University, Taiwan, in 2012. He is currently a M.S. student in Graduate Institute of Computer Science and Information Engineer at National Taiwan University. His research focuses on multimedia retrieval and analysis on portable device.

Winston H. Hsu (M07VSM12) received the Ph.D. degree in electrical engineering from Columbia University, New York, NY, USA. He has been an Associate Professor in the Graduate Institute of Networking and Multimedia, National Taiwan University, Taipei, Taiwan, since February 2007. Prior to this, he was in the multimedia software industry for years. He is also with the Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan. His research interests include multimedia content analysis, image/video indexing and retrieval, machine learning, and mining over large scale databases. Dr. Hsu serves in the Editorial Board for the IEEE Multimedia Magazine.