Discriminative Codeword Selection for Image Representation

Lijun Zhang[1]
zljzju@zju.edu.cn

Chun Chen[1]
chenc@zju.edu.cn

Jiajun Bu[1]
bjj@zju.edu.cn

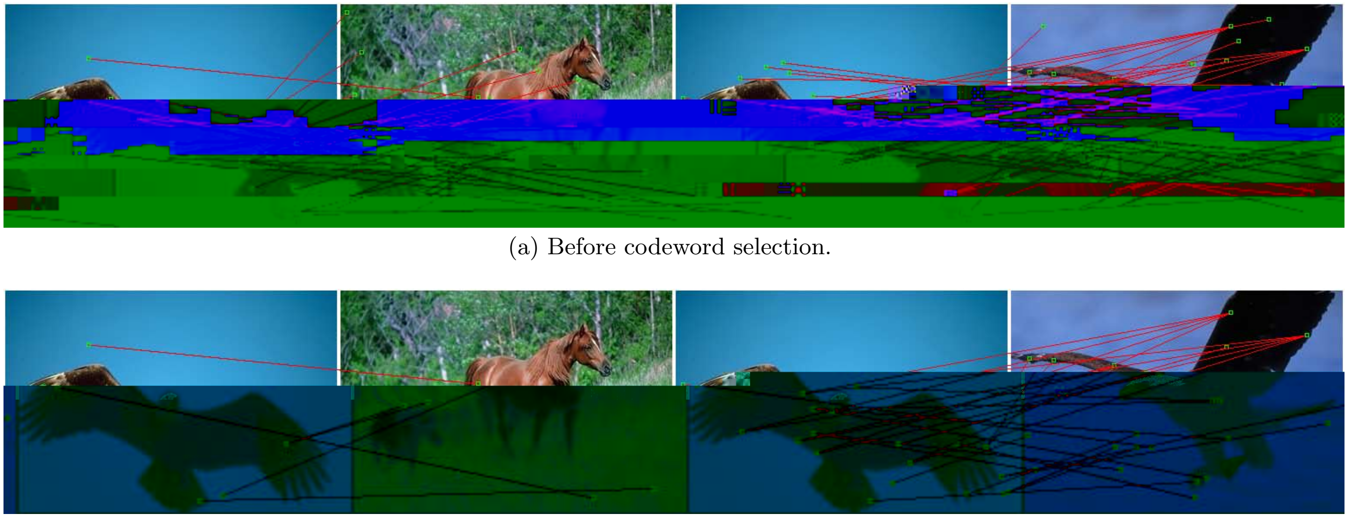
Zhengguang Chen[1]
cerror@zju.edu.cn

Shulong Tan[1]
laos1984@zju.edu.cn

Xiaofei He[2]
xiaofeihe@cad.zju.edu.cn

[1]Zhejiang Key Laboratory of Service Robot, College of Computer Science, Zhejiang University, Hangzhou, China
[2]State Key Lab of CAD&CG, College of Computer Science, Zhejiang University, Hangzhou, China

Bag of features (BoF) representation has attracted an in-



(a) Before codeword selection.

trices. We use $\mathrm{Tr}(\cdot)$ to denote the trace of a matrix, and $\|\cdot\|_F$ to denote the Frobenius norm of a matrix. $\mathrm{Diag}(\cdot)$ denotes a diagonal matrix formed from its vector argument, and $\mathrm{diag}(\cdot)$ denotes a column vector consisting of the diagonal elements of its matrix argument. Let $\succeq$ denote the associated generalized inequality of the positive semidefinite cone: $A \succeq B$ means $A - B$ is a positive semidefinite matrix. Script capital letters (e.g. $\mathcal{C}$) are used to denote ordinary sets.

In this section, we give a brief review of the existing codeword selection algorithms. Since many codeword selection algorithms are based on the feature selection techniques, we begin with a discussion of feature selection.

In real applications, dimensionality reduction techniques [9, 20, 21, 30, 34] are widely used to deal with the *curse of dimensionality* [13]. Among various methods, feature selection reduces the dimensionality by choosing a subset of revelent features for compact representation [12]. Two types of feature selection techniques have been studied: supervised and unsupervised.

The typical approach for supervised feature selection is to evaluate the correlation between features and labels to determine their relevance. Pearson correlation, Fisher score, Kolmogorov-Smirnov test and Information Gain [10] are several popular methods. More advanced supervised techniques leverage some supervised learning models to select the most useful features. Linear regression based feature selection [35] and Support Vector Machine (SVM) based feature selection [28] have received a lot of attention in recent years. For example, in the Enhanced Biologically Inspired Model [15], SVM and AdaBoost are combined to select the effective features.

Due to the lack of labels, unsupervised feature selection is much harder. Existing unsupervised feature selection techniques can be classified into two categories. The first category exploits the geometrical structure of the data space to guide the selection [3, 14, 25]. The typical algorithms in this category include maximum variance, unsupervised feature selection for PCA [3] and Laplacian score [14]. Maximum variance selects features with the largest variances and unsupervised feature selection for PCA selects a subset of features that can best reconstruct other features. Different from these two methods, Laplacian score [14] selects features that best preserve the local geometrical structure. The second category of unsupervised feature selection techniques aims to maximize some clustering performance [2, 6, 40]. For example, $Q - \alpha$ [40] measures the cluster coherence by analyzing the spectral properties of the affinity matrix. A remarkable property of this algorithm is that it always yields sparse solutions.

The goal of codeword selection is to remove the redundancy and noise in the codebook, which is usually constructed by using an clustering algorithm. Since each codeword corresponds to one feature in the frequency histogram, feature selection techniques can be used for codeword selection.

In [18], three feature selection methods: mutual information (MI), odds ratio (OR) and linear SVM weights (LSVM) are used to select the most informative codewords. The criterion of information gain (IG) is used in [29] to select the codewords that are most informative about specific location. As more images can be utilized, the retrieval performance of city-scale location recognition is significantly improved. An entropy-based minimum description length (MDL) criterion is proposed in [19] for simultaneous classification and codeword selection.

In [37], a boosting feature selection approach is proposed to select the most discriminative codewords from a multi-resolution codebook. The key idea is to associate each weak classifier with a codeword, and the selection of codeword can be achieved by the selection of the weak classifier. Codeword selection is formulated as a multi-subset search problem in [11], and a novel region selection algorithm is proposed to identify region types that are frequently found in a particular class of scenes but rarely exist in other classes, and also consistently occur together in the same class of scenes. The work in [24] introduces one online codeword selection algorithm based on the dual-gradient descent approach. Side information in the form of pairwise constraints (*must-link* and *must-not link*) is required for this algorithm. A subset of codewords is selected such that the distance computed using them satisfies the given pairwise constraints. The work in [44] considers finding the Descriptive Visual Words (DVWs) and Descriptive Visual Phrases (DVPs) for each image category.

Let $\mathcal{I} = \{\mathcal{I}_1, \cdots, \mathcal{I}_m\}$ be the given set of m images, whichI

We consider fitting a multi-output linear function $f(H) = HW + \mathbf{1}_m\mathbf{b}^T$ to model the relationship between H and Y. In this linear function, $\mathbf{1}_m$ is a m-dimensional vector of all ones, $W \in \mathbb{R}^{k\times r}$ is the coefficient matrix, and $\mathbf{b} \in \mathbb{R}^r$ is the intercept. Following ridge regression [13], fitting this function can be mathematically formulated as

$$\min_{W,\mathbf{b}} \|Y - HW - \mathbf{1}_m\mathbf{b}^T\|_F^2 + \alpha\|W\|_F^2 \tag{1}$$

where $\|\cdot\|_F$ denotes the matrix Frobenius norm, and $\alpha \geq 0$ is the trade-off parameter for the regularizer $\|W\|_F^2$.

Taking the first order partial derivatives of Eq. (1) with respective to W, $\mathbf{b}$ and requiring them to be zero, we get the optimal W^* and $\mathbf{b}^*$:

$$W^* = (H^T\Pi H + \alpha I)^{-1}H^T\Pi Y \tag{2}$$

$$\mathbf{b}^* = \frac{1}{m}\left(Y^T - (W^*)^T H^T\right)\mathbf{1}_m \tag{3}$$

where I is the identity matrix and $\Pi = I - \frac{1}{m}\mathbf{1}_m\mathbf{1}_m^T$ is the centering matrix. To simplify the presentation, we assume that the data has zero mean, so that we have

$$\Pi H = H \tag{4}$$

Substituting the values of W^* and b^* into Eq. (1), we obtain the *fitting error* of the estimated linear function [1]:

$$\begin{aligned}
&J(Y,H)\\
=&\|Y - HW^* - \mathbf{1}_m(\mathbf{b}^*)^T\|_F^2 + \alpha\|W^*\|_F^2\\
=&\left\|Y - HW^* - \frac{\mathbf{1}_m\mathbf{1}_m^T}{m}(Y - HW^*)\right\|_F^2 + \alpha\|W^*\|_F^2\\
=&\|\Pi(Y - HW^*)\|_F^2 + \alpha\|W^*\|_F^2\\
=&\|\Pi\big(I - H(H^TH + \alpha I)^{-1}H^T\big)Y\|_F^2\\
&+ \alpha\|(H^TH + \alpha I)^{-1}H^TY\|_F^2\\
=&\operatorname{Tr}\left(Y^T\big(\Pi - H(H^TH + \alpha I)^{-1}H^T\big)^2Y\right)\\
&+ \alpha\operatorname{Tr}\left(Y^TH(H^TH + \alpha I)^{-2}H^TY\right)\\
=&\operatorname{Tr}\left(Y^T\big(\Pi - H(H^TH + \alpha I)^{-1}H^T\big)Y\right)
\end{aligned} \tag{5}$$

In the above derivation, we have used the fact that the centering matrix is idempotent, that is, $\Pi = \Pi^k$ for $k = 1, 2, \cdots$. Following the Woodbury-Morrison formula [33], Eq. (5) can be simplified as [1]:

$$\begin{aligned}
&\operatorname{Tr}\left(Y^T\big(\Pi - H(H^TH + \alpha I)^{-1}H^T\big)Y\right)\\
=&\operatorname{Tr}\left(Y^T\Pi\big(I - H(H^TH + \alpha I)^{-1}H^T\big)\Pi Y\right)\\
=&\operatorname{Tr}\left(Y^T\Pi(I + \frac{1}{\alpha}HH^T)^{-1}\Pi Y\right)\\
=&\alpha\operatorname{Tr}\left(Y^T\Pi(\alpha I + HH^T)^{-1}\Pi Y\right)
\end{aligned} \tag{6}$$

As can be seen, the fitting error $J(Y,H)$ contains Y and H as the variables. Then, it is natural to require that a good indicator matrix Y and the sub-matrix H lead to minimal $J(Y,H)$. In other words, we are looking for a feature subset $\mathcal{H}$, such that if the data is represented by these features, the performance of discriminative clustering is the best.

In the following, we give a mathematical formulation of our codeword selection problem. The constraint that Y is a $m \times$ indicator matrix is equivalent to the following two constraints:

$$Y \in \{0,1\}^{m\times r},\ Y\mathbf{1}_r = \mathbf{1}_m \tag{7}$$

By introducing a -dimensional vector $\boldsymbol{\lambda} = [\lambda_1, \cdots, \lambda_n]^T \in \{0,1\}^n$, where λ_i indicates whether or not feature $\mathbf{f}_i$ is chosen, we have

$$H^TH = \sum_{i=1}^{k}\mathbf{h}_i\mathbf{h}_i^T = \sum_{i=1}^{n}\lambda_i\mathbf{f}_i\mathbf{f}_i^T \tag{8}$$

To ensure that k features are selected, the following constraints should be added

$$\mathbf{1}_n^T\boldsymbol{\lambda} = k \tag{9}$$

Then, our codeword selection problem is formally stated below:

Definition 1. Discriminative Codeword Selection (DCS):

$$\begin{aligned}
\min_{Y,\boldsymbol{\lambda}}\quad & \operatorname{Tr}\left(Y^T\Pi(\textstyle\sum_{i=1}^n\lambda_i\mathbf{f}_i\mathbf{f}_i^T + \alpha I)^{-1}\Pi Y\right)\\
\text{s.t.}\quad & Y \in \{0,1\}^{m\times r},\ Y\mathbf{1}_r = \mathbf{1}_m\\
& \boldsymbol{\lambda} \in \{0,1\}^n,\ \mathbf{1}_n^T\boldsymbol{\lambda} = k
\end{aligned} \tag{10}$$

The problem (10) is difficult to solve due to its combinatorial nature. In this section, we develop a sequential algorithm to find a sub-optimal solution.

Let $(Y^*, \boldsymbol{\lambda}^*)$ be the optimal solution of the problem (10). Initially, we solve the standard discriminative clustering problem [1] with all the features selected. The resulting indicator matrix E can be used as a good estimation of Y^*. Then, by fixing $Y = E$, we solve the problem (10) to find the k most discriminative features.

Our goal in this step is to find a good estimation E of the optimal indicator matrix Y^*, which can be used to guide the search of the most discriminative features. Without any prior knowledge, one natural choice is to solve the original discriminative clustering problem. Setting $\boldsymbol{\lambda} = \mathbf{1}_n$, the problem (10) becomes

$$\begin{aligned}
\min_{Y}\quad & \operatorname{Tr}\left(Y^T\Pi(\textstyle\sum_{i=1}^n\mathbf{f}_i\mathbf{f}_i^T + \alpha I)^{-1}\Pi Y\right)\\
\text{s.t.}\quad & Y \in \{0,1\}^{m\times r},\ Y\mathbf{1}_r = \mathbf{1}_m
\end{aligned} \tag{11}$$

In the following, we adopt the optimization procedure proposed in [1] to solve the above problem. Instead of computing Y, we introduce the variable $M = YY^T$. Using the fact that $\operatorname{Tr}(AB) = \operatorname{Tr}(BA)$, the objective function in the problem (11) becomes:

$$\operatorname{Tr}\left(\Pi(\sum_{i=1}^{n}\mathbf{f}_i\mathbf{f}_i^T + \alpha I)^{-1}\Pi M\right) \tag{12}$$

Following [1], we replace the constraint that M is the product of a $m \times$ indicator matrix and its transpose with the following constraints:

$$\operatorname{diag}(M) = \mathbf{1}_m,\ M \succeq \frac{1}{\ }\mathbf{1}_m\mathbf{1}_m^T,\ M \geq 0 \tag{13}$$

Define $A = \Pi(\sum_{i=1}^n\mathbf{f}_i\mathbf{f}_i^T + \alpha I)^{-1}\Pi$. We have the following optimization problem:

$$\begin{aligned}
\min_{M}\quad & \operatorname{Tr}(AM)\\
\text{s.t.}\quad & \operatorname{diag}(M) = \mathbf{1}_m,\ M \succeq \tfrac{1}{r}\mathbf{1}_m\mathbf{1}_m^T,\ M \geq 0
\end{aligned} \tag{14}$$

The above problem is a Semidefinite Program (SDP), and can be solved by general purpose interior-point methods [4]. However, directly solving the problem (14) has the complexity of $O(n^7)$, which is too slow for large scale data set. In [1], Bach and Harchaoui have proposed a more efficient approach by solving the following partial dual problem of (14):

$$\begin{array}{ll} \max\limits_{\mathbf{a},\mathbf{b},c,D} & \min\limits_{M} \mathrm{Tr}\left(\big(A + f(\mathbf{a},\mathbf{b},c,D)\big)M\right) - g(\mathbf{a},\mathbf{b},c,D) \\ \mathrm{s.t.} & M \succeq 0,\ \mathrm{Tr}(M) = n \\ & f(\mathbf{a},\mathbf{b},c,D) = \mathrm{Diag}(\mathbf{a}) + \frac{\mathbf{b}\mathbf{b}^T}{2c} - D \\ & g(\mathbf{a},\mathbf{b},c,D) = \mathbf{a}^T\mathbf{1}_m + \mathbf{b}^T\mathbf{1}_m + \frac{cr}{2} \\ & c \geq 0,\ D \geq 0 \end{array} \tag{15}$$

where the variables $\mathbf{a} \in \mathbb{R}^m$, ($\mathbf{b} \in \mathbb{R}^m, c \in \mathbb{R}_+$) and $D \in \mathbb{R}_+^{m\times m}$ are the dual variables of the constraints $\mathrm{diag}(M) = \mathbf{1}_m$, $M \succeq \frac{1}{r}\mathbf{1}_m\mathbf{1}_m^T$ and $M \geq 0$. The problem (15) can be solved more efficiently due to the fact that $\min\limits_{M} \mathrm{Tr}\Big(\big(A + f(\mathbf{a},\mathbf{b},c,D)\big)M\Big)$ can be solved simply through an eigenvalue decomposition.

Denote the optimal solution of (15) by M^*. The discrete indicator matrix E are recovered as follows:

1. Computing the first eigenvectors of M^*, and forming a matrix Z by stacking the eigenvectors in columns;

2. Rescaling the rows of Z to unit norms and then perform K-means to obtain E.

For details, please refer to [1].

After solving the discriminative clustering problem, we obtain the indicator matrix E. Substituting $Y = E$ into the problem (10), we get the following problem:

$$\begin{array}{ll} \min\limits_{\boldsymbol{\lambda}} & \mathrm{Tr}\left(E^T\Pi(\sum_{i=1}^n \lambda_i \mathbf{f}_i\mathbf{f}_i^T + \alpha I)^{-1}\Pi E\right) \\ \mathrm{s.t.} & \boldsymbol{\lambda} \in \{0,1\}^n,\ \mathbf{1}_n^T\boldsymbol{\lambda} = \boldsymbol{k} \end{array} \tag{16}$$

where the value of λ_i indicates whether or not feature $\mathbf{f}_i$ is chosen as the most discriminative one. This problem is still difficult to solve due to the integer constraint $\boldsymbol{\lambda} \in \{0,1\}^n$.

In the following, we introduce an efficient sequential approach to find the $\boldsymbol{k}$ most informative features. For conciseness, we firstly update E by centering its columns:

$$E \leftarrow \Pi E \tag{17}$$

Suppose a set of features $\mathcal{H}_t = \{\mathbf{h}_1, \cdots, \mathbf{h}_t\} \subseteq \mathcal{F}$ have been selected as the most discriminative ones, and define $H_t = [\mathbf{h}_1, \cdots, \mathbf{h}_t]$. The ($+1$)-th feature $\mathbf{h}_{t+1}$ can be found by solving the following problem:

$$\begin{array}{ll} \min\limits_{\mathbf{f}} & \mathrm{Tr}\left(E^T(H_tH_t^T + \mathbf{f}\mathbf{f}^T + \alpha I)^{-1}E\right) \\ \mathrm{s.t.} & \mathbf{f} \in \mathcal{F} \setminus \mathcal{H}_t \end{array} \tag{18}$$

The most expensive calculation in (18) is the matrix inverse $(H_tH_t^T + \mathbf{f}\mathbf{f}^T + \alpha I)^{-1}$, which need be computed for each $\mathbf{f} \in \mathcal{F} \setminus \mathcal{H}_t$. We use the Woodbury-Morrison formula [33] to avoid directly inverting a matrix. Let $P = (H_tH_t^T + \alpha I)^{-1}$, we have

$$\begin{aligned} & (H_tH_t^T + \mathbf{f}\mathbf{f}^T + \alpha I)^{-1} \\ =& (H_tH_t^T + \alpha I)^{-1} \\ & - \frac{(H_tH_t^T + \alpha I)^{-1}\mathbf{f}\mathbf{f}^T(H_tH_t^T + \alpha I)^{-1}}{1 + \mathbf{f}^T(H_tH_t^T + \alpha I)^{-1}\mathbf{f}} \\ =& P - \frac{P\mathbf{f}\mathbf{f}^TP}{1 + \mathbf{f}^TP\mathbf{f}} \end{aligned} \tag{19}$$

Then, the objective function of (18) can be rewritten as

$$\begin{aligned} & \mathrm{Tr}\left(E^T(H_tH_t^T + \mathbf{f}\mathbf{f}^T + \alpha I)^{-1}E\right) \\ =& \mathrm{Tr}\left(E^T\left(P - \frac{P\mathbf{f}\mathbf{f}^TP}{1 + \mathbf{f}^TP\mathbf{f}}\right)E\right) \\ =& \mathrm{Tr}(E^TPE) - \frac{\mathrm{Tr}(E^TP\mathbf{f}\mathbf{f}^TPE)}{1 + \mathbf{f}^TP\mathbf{f}} \\ =& \mathrm{Tr}(E^TPE) - \frac{\mathbf{f}^TPEE^TP\mathbf{f}}{1 + \mathbf{f}^TP\mathbf{f}} \\ =& \mathrm{Tr}(E^TPE) - \frac{\|E^TP\mathbf{f}\|^2}{1 + \mathbf{f}^TP\mathbf{f}} \end{aligned} \tag{20}$$

Notice that $\mathrm{Tr}(E^THE)$ is a constant when selecting the ($+1$)-th feature. The optimization problem (18) can be simplified as

$$\begin{array}{ll} \max\limits_{\mathbf{f}} & \|E^TP\mathbf{f}\|^2/(1 + \mathbf{f}^TP\mathbf{f}) \\ \mathrm{s.t.} & \mathbf{f} \in \mathcal{F} \setminus \mathcal{H}_t \end{array} \tag{21}$$

After we have obtained the ($+1$)-th point $\mathbf{h}_{t+1}$ by solving the problem (21), the matrix P can be updated as

$$P \leftarrow (H_tH_t^T + \mathbf{h}_{t+1}\mathbf{h}_{t+1}^T + \alpha I)^{-1} \tag{22}$$

where the matrix inverse can be computed according to (19).

The above process is repeated until we have selected $\boldsymbol{k}$ features. In the beginning, there are no features selected. Therefore, we set $P = (\alpha I)^{-1} = \frac{1}{\alpha}I$.

In this section, we investigate the use of our proposed codeword selection algorithm for image retrieval and clustering.

(a) Samples images from Corel50

(b) Samples images from Caltech10

Figure 2: Sample images from the Corel50 and Caltech10 image data sets.

database. The number of SIFT descriptor extracted from the Corel50 data set is 1,755,935 and 1000 codewords are generated. By assigning the descriptors to the closest codewords, each image in Corel50 is represented by one 1000-dimensional frequency histogram according to the count of each codeword. For the Caltech10 data set, the number of SIFT descriptor is 555,292 and 500 codewords are generated. Thus, each image in Caltech10 is represented by one 500-dimensional frequency histogram.

In the following, several experiments were performed to show the effectiveness of our proposed DCS for unsupervised codeword selection. These experiments include image retrieval and image clustering. The following three codeword selection algorithms are compared:

- **Discriminative Codeword Selection (DCS)**[3]. The unsupervised codeword selection algorithm introduced in this paper.

- Codeword selection based on the $\boldsymbol{Q-\alpha}$ algorithm [40]. $Q-\alpha$ is a unsupervised feature algorithm which selects features to maximize the cluster coherence.

- Codeword selection based on the **Unsupervised Feature Selection using Feature Similarity (FSFS)** [25]. FSFS[4] uses feature similarity for redundancy reduction.

We also provided the results of the **Baseline** method, which uses the original codebook without codeword selection. We compare our proposed approach with $Q-\alpha$ since both of these two approaches aim at discovering the cluster structure of the image database. We compare with FSFS since it has been shown that FSFS is superior to many existing unsupervised feature selection methods such as correlation coefficients and sometimes even better than supervised feature selection methods such as Relief-F [25].

We perform image retrieval experiments on the Corel50 image database. *Precision* is used to evaluate the effectiveness of different codeword selection algorithms. The precision at top N is defined as the ratio of the relevant images presented to the user in the top N ranked images. Each image in the Corel50 database is used as a query image, and the other images are ranked according to the their Euclidean distances to the query image. For Baseline, the Euclidean distances are computed using the original 1000-dimensional frequency histogram. For $Q-\alpha$, FSFS and DCS, a given number ($\boldsymbol{k} = 100, 200, \cdots, 900$) codewords are selected. Then, the part of the original frequency histogram that corresponds to the selected codewords, is used to describe each image. Thus, after codeword selection, the calculation of Euclidean distances will be much faster. The final precision rate is computed by averaging the results over the 4970 queries.

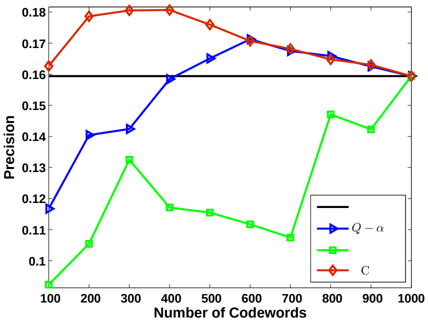
Fig. 3 shows the average precision (at top 20, 40 and 60) versus the number of the selected codewords. As can be seen, our DCS algorithm significantly outperforms the other algorithms in most cases. DCS is very effective in selecting those discriminative visual codewords. With only 100 codewords (selected by DCS), the retrieval performance is almost the same as using all the 1000 codewords. $Q-\alpha$ performs the second best. The accuracy of $Q-\alpha$ is similar to that of DCS when the number of the selected codewords is more than 600. When the number of the selected codewords is more than 400, the accuracy of $Q-\alpha$ is better than Baseline. However, when the number of codewords is smaller than 600, its performance decreases drastically as the number of codewords reduces.

The advantage of DSC and $Q-\alpha$ compared with Baseline validates that codeword selection not only reduces the computational cost, but also has the ability to improve the performance. The performance of FSFS is worse than the Baseline in this experiment. This is probably because FSFS can only remove the redundant codewords, and fails to remove the noisy ones. One common property of $Q-\alpha$ and DCS is that they both aim to maximize the performance of clustering. Thus, the clustering guided codeword selection is more effective for image retrieval. Since DCS is optimized for the discriminative clustering criterion, DCS can select those codewords with higher discriminative power and has higher retrieval performance.

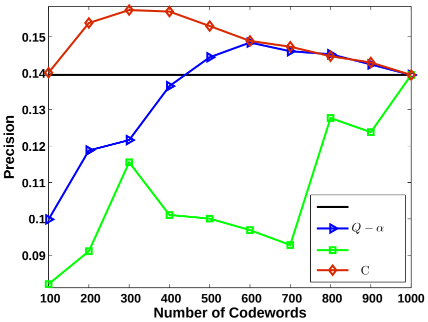
In general, it is appropriate to present 20 images on a screen. Putting more images on a screen may affect the quality of the presented images. Therefore, the precision at top 20 is especially important. Table 1 shows the average precision at top 20 for the 50 categories. $Q-\alpha$, FSFS and DCS are applied to selecting 300 codewords in this table. Considering only the three codewords selection methods, our

[3] The implementation is based on the code for discriminative clustering (http://www.di.ens.fr/˜fbach/diffrac/index.htm).

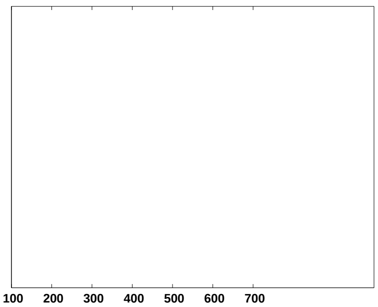
[4] An implementation can be downloaded from http://www.facweb.iitkgp.ernet.in/˜pabitra/paper.html.

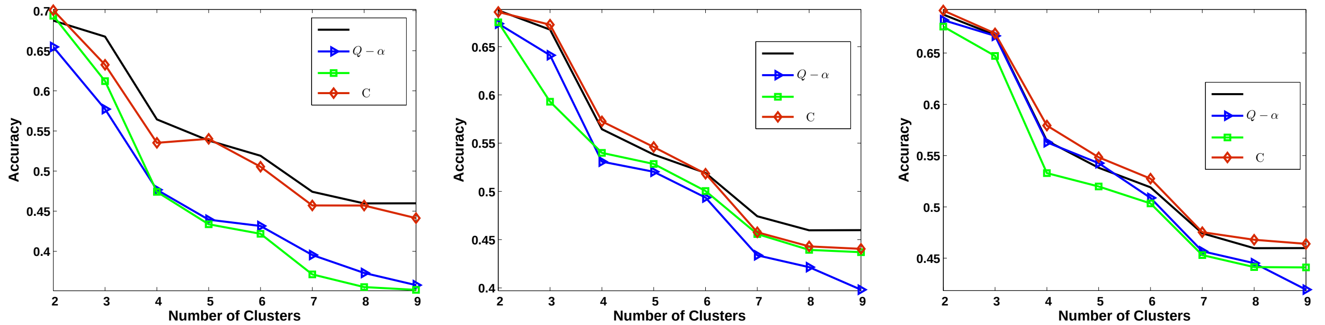


(a) Precision at top 20 ranked images.



(b) Precision at top 40 ranked images.



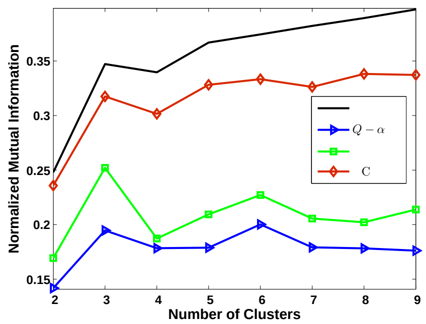


(a) Accuracy with 100 codewords selected.

(b) Accuracy with 200 codewords selected.

(c) Accuracy with 300 codewords selected.

Figure 4: Clustering performance measured in terms of accuracy on the Caltech10 image database. The figures show the average accuracy versus the number of clusters.



(a) Normalized mutual information with 100 codewords selected.

cluster the images using the original 500-dimensional frequency histogram. For $Q - \alpha$, FSFS and DCS, a given number ($k = 100, 200, 300$) codewords are selected. After codeword selection, each image is represented by the part of the original frequency histogram that corresponds to the selected codewords. And the clustering experiments are conducted with this new representation. In the experiments, K-means is used as the clustering algorithm. Because the procedure for solving K-means can only find the local optimum, we ran K-means 10 times with different random starting points and the best result in terms of the objective function of K-means was recorded.

The evaluations were conducted with different number of clusters c, ranging from 2 to 9. At each run of the test, c clusters are randomly selected from the whole database. For each given cluster number c, 10 test runs are conducted, and the average performance was computed over these 10 tests. Fig. 4 shows the average accuracy versus the number of the selected clusters. As can be seen, DCS outperforms the other two codeword selection algorithms in all the cases. With only 200 codewords selected, the accuracy achieved by DCS is better than or comparable to that of Baseline. In terms of accuracy, the performance of $Q - \alpha$ and FSFS is very close. The clustering performance measured by normalized mutual information is shown in Fig. 5. Our DCS still outperforms $Q - \alpha$ and FSFS, and the advantage becomes more obvious. Table 2 shows the detailed clustering results for each algorithm with 200 codewords selected. With all the 500 codewords, the baseline achieves 54.63% in terms of accuracy and 35.56% in terms of normalized mutual information on average. By using only 200 selected codewords, DCS can achieve 54.21% in terms of accuracy (4% relative improvement over FSFS) and 33.88% in terms of normalized mutual information (17.2% relative improvement over FSFS).

In this paper, a novel unsupervised codeword selection algorithm called Discriminative Codeword Selection (DCS) is proposed. DCS uses the performance of discriminative clustering, a recently proposed unsupervised clustering framework, to guide the selection of the most discriminative codewords. As a result, DCS can select those features with most discriminative power. Image retrieval and clustering experiments on two standard image databases show the effectiveness of our proposed approach.

Because the objective function of DCS contains the indicator matrix as a variable, DCS can be easily extend to incorporate the prior knowledge to the indicator matrix. We will investigate this in our future work. More advanced methods for solving the optimization problem will be studied too.

semantic video retrieval. In *Proceedings of the 6th ACM international conference on Image and video retrieval*, pages 494–501, 2007.

[18] F. Jurie and B. Triggs. Creating efficient codebooks for visual recognition. In *Proceedings of the Tenth*