Hierarchical Gaussianization for Image Classification

Xi Zhou [†], Na Cui [‡], Zhen Li [†], Feng Liang [‡], and Thomas S. Huang [†]
[†] Dept. of ECE, University of Illnois at Urbana-Champaign
[‡] Dept. of Statistics, University of Illnois at Urbana-Champaign
{xizhou2, nacui2, zhenli3, liangf}@uiuc.edu, huang@ifp.uiuc.edu

Abstract

In this paper, we propose a new image representation to capture both the appearance and spatial information for image classification applications. First, we model the feature vectors, from the whole corpus, from each image and at each individual patch, in a Bayesian hierarchical framework using mixtures of Gaussians. After such a hierarchical Gaussianization, each image is represented by a Gaussian mixture model (GMM) for its appearance, and several Gaussian maps for its spatial layout. Then we extract the appearance information from the GMM parameters, and the spatial information from global and local statistics over Gaussian maps. Finally, we employ a supervised dimension reduction technique called DAP (discriminant attribute projection) to remove noise directions and to further enhance the discriminating power of our representation. We justify that the traditional histogram representation and the spatial pyramid matching are special cases of our hierarchical Gaussianization. We compare our new representation with other approaches in scene classification, object recognition and face recognition, and our performance ranks among the top in all three tasks.

1. Introduction

Histogram representation, as a description for orderless patch-based features, has been widely used in visual recognition and image retrieval [4, 5]. Despite its popularity, however, histogram representation has some intrinsic limitations. For example, it is sensitive to several factors such as outliers, the choice of bins, and the noise level in the data. Most importantly, encoding high-dimensional feature vectors by a relatively small codebook inclines to large quantization errors and lose of discriminability [21]. Furthermore, histogram representation discards all the spatial configuration of image patches, which is a key attribute for object and scene classification.

Several approaches have been proposed in the literature to overcome these limitations. Soft assignment, which allows each feature vector belonging to multiple histogram bins, have been suggested to capture partial similarity between images [16, 19, 18, 26, 27, 28]. To enhance the discriminating capability of histograms, Farquhar *et al.* [12] and Peronnin *et al.* [16] introduced several ways to construct category-specific histograms, Larlus *et al.* [13] and Yang *et al.* [19] suggested to integrate histogram construction with classifier training, and Moosmann *et al.* [15] proposed to use randomized forests to build discriminative histograms. As a flexible way to model a variety of distributions, GMM emerged as a better alternative to histograms in age estimation, object classification and video event analysis [2, 1, 3]. On the other hand, to alleviate the loss of spatial information in histogram representation, one of the most successful approaches by far is the spatial pyramid matching (SPM) technique proposed by Lazebnik *et al.* [11].

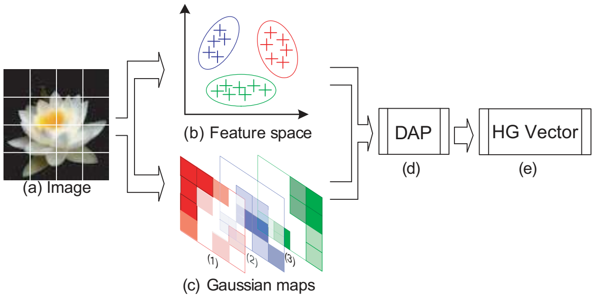


Figure 1. (a) is an input image. (b) shows the patch features in the feature space. Each ”+” denotes a feature vector, whose distribution is approximated by a GMM. (c) shows a set of Gaussian maps, each of which corresponds to one Gaussian component in (b). A supervised dimension reduction algorithm, DAP, is performed in (d) to form the final image representation, hierarchical Gaussianization vector.

In this paper, we propose a new model-based representation for image features, capturing both the appearance and spatial information. First, we adopt a hierarchical GMM for feature vectors at difference levels: the whole corpus, each image and individual patches. We learn the image-specific GMM in a Bayesian framework to allow information shar-

2009 IEEE 12th International Conference on Computer Vision (ICCV)
978-1-4244-4419-9/09/$25.00 ©2009 IEEE

ing across different images and to bridge the universal and individual information retrievals. Given an image-specific GMM, each patch of that image is assigned to a Gaussian component with respect to a posterior probability. All these probabilities constitute a set of so-called *Gaussian maps* over the entire patch grid. After obtaining a GMM and Gaussian maps for each image which we term as a Hierarchical Gaussianization (HG) process, we extract the appearance information from the GMM parameters, and the spatial information from global and local summary statistics over Gaussian maps. Finally, all parameters of the GMM and statistics of the Gaussian maps are concatenated as a super-vector, followed by a supervised dimension reduction to further enhance the discriminating power of the representation. An illustration of this new representation is shown in Figure 1.

The remaining of this paper is arranged as follows. In Section 2, we introduce the new image representation that incorporates both the visual and spatial information. In Section 3, we justify that the histogram representation and the spatial pyramid matching are special cases of the HG representation. In Section 4, we demonstrate the effectiveness of our approach on three image databases. Conclusions are given in Section 5.

2. Hierarchical Gaussianization representation

2.1. GMMs for appearance representation

Let z denotes a p-dimensional feature vector from the I-th image. We model z by a GMM, namely,

$$p(z|\Theta) = \sum_{k=1}^{K} w_k^I \mathcal{N}(z; \mu_k^I, \Sigma_k^I), \qquad (1)$$

where K denotes the total number of Gaussian components, and $(w_k^I, \mu_k^I, \Sigma_k^I)$ are the image-specific weight, mean and covariance matrix of the kth Gaussian component, respectively. For computational efficiency, we restrict the covariance matrices Σ_k^I to be a diagonal matrix Σ_k shared by all images.

The number of model parameters $\Theta = \{w_k^I, \mu_k^I, \Sigma_k\}_{k=1:K, I=1:N}$ increases extensively with respect to N, the number of training images. In practice the size of patches from one image is usually small and thus insufficient for a robust estimate of all parameters. To overcome this problem, we propose a hierarchical Bayesian framework to jointly estimate all the GMM parameters. We model the image-specific GMM parameters w_k^I's and μ_k^I's by conjugate priors:

$$\begin{aligned} (w_1^I, \ldots, w_K^I) &\sim \text{Dir}\,(Tw_1, \ldots, Tw_K), \\ \mu_k^I &\sim \mathcal{N}(\mu_k, \Sigma_k/r),\ k = 1:K. \end{aligned}$$

The prior distribution over the weights w_k^I's is a Dirichlet distribution with parameters $(Tw_1, \ldots Tw_K)$, which can be interpreted as adding total T pseudo-counts with w_k fraction of them from the kth component. The prior distribution for the mean μ_k^I's is a Gaussian centered at a global mean μ_k with a covariance matrix shrunk by a smoothing parameter r. Note that such a prior specification imposes dependence between images. And the rationale behind this is to "borrow" strength across similar images for estimation and therefore overcome the small sample size issue suffered in conventional learning processes.

We estimate the prior mean vector μ_k, prior weights w_k and covariance matrix Σ_k by fitting a global GMM based on the whole corpus, and the remaining parameters by solving the following *Maximum A Posteriori* (MAP) loss,

$$\max_{\Theta} \Big[\ln p(z|\Theta) + \ln p(\Theta) \Big].$$

The MAP estimates can be obtained via an EM algorithm: in the E-step, we compute

$$\begin{aligned} Pr(k|z_i) &= \frac{w_k^I \mathcal{N}(z_i; \mu_k^I, \Sigma_k)}{\sum_{j=1}^{K} w_j^I \mathcal{N}(z_i; \mu_j^I, \Sigma_j)}, && (2) \\ n_k &= \sum_{i=1}^{N} Pr(k|z_i), && (3) \end{aligned}$$

and in the M-step, we update

$$\begin{aligned} \hat{w}_k^I &= \gamma_k n_k / N + (1 - \gamma_k) w_k, && (4) \\ \hat{\mu}_k^I &= \alpha_k m_k + (1 - \alpha_k) \mu_k, && (5) \end{aligned}$$

where

$$m_k = \frac{1}{n_k} \sum_{i=1}^{N} Pr(k|z_i) z_i,$$

$$\alpha_k = n_k/(n_k + r),\ \gamma_k = N/(N + T).$$

If a Gaussian component has a high probabilistic count, n_k, then α_k approaches 1 and the adapted parameters emphasize the new sufficient statistics m_k; otherwise, the adapted parameters are determined by the global model μ_k. The tuning parameters r and T can also affect the MAP adaptation. In general, the larger r and T, the larger the influence of the prior distribution on the adaptation. For example, when r goes to infinity, the MAP adaptation for μ_k^I is fixed at the prior mean, similar for T and w_k^I. In practice we adjust r and T empirically, based on the total number of coordinate patches for each image.

After Gaussinization, we can calculate the similarity between a pair of images via the similarity between two GMMs. A common approach is to summarize the parameters of a GMM as a vector m, and then use some vector metric, such as inner product [2, 1, 3]. Note that $m = f(w, \mu, \Sigma)$ is in general a function involving all parameters of the corresponding GMM. In our experiments, we follow the suggestion in [3] and choose the appearance vector for an image x^I to be

$$m(x^I) = [\sqrt{w_1^I}\Sigma_1^{-\frac{1}{2}}\mu_1^I; \cdots; \sqrt{w_K^I}\Sigma_K^{-\frac{1}{2}}\mu_K^I]. \tag{6}$$

2.2. Gaussian maps for spatial representation

According to equation (2), the feature vector at each patch is again modeled by a mixture of Gaussians with a mixture probability $Pr(k|z_i)$. For a fixed k, all such probabilities $Pr(k|z_i)$ form a map over the patch locations, which we refer to as a *Gaussian map*. While each Gaussian component represents some structure in the feature space, the corresponding Gaussian map shows the geometric location of that structure on an image. For a GMM with K components, we have K Gaussian maps, and we can learn the spatial information of an image by analyzing each of these Gaussian maps.

A natural way to summarize a Gaussian map is to use its mean location or normalized mean location. However, such global summary statistics do not work well for images. In Figure 2, we plot a subset of Gaussian maps for three images from Caltech 101 database that is analyzed in Section 4. It is clear that local information is more important for the discriminant analysis than the global one.

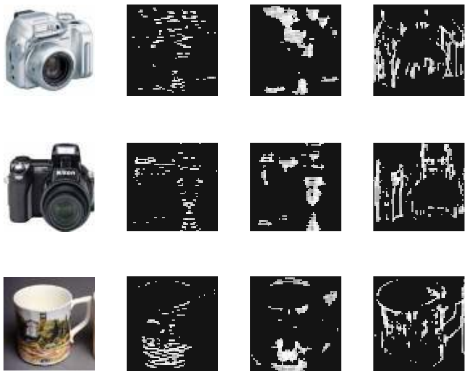


Figure 2. Sample Gaussian Maps of three images from the Caltech 101 dataset.

Therefore we propose to hierarchically split a Gaussian map and extract summary statistics over local regions. Specifically, each of the K Gaussian maps is divided into subregions based on a sequence of increasingly coarser grids; assume there are M subregions in total, then we calculate some summary statistic ν over each of the M regions. As a parallel form to (6), we define $v(x^I)$, a vector expressing spatial information of image x^I as follows,

$$v(x^I) = [\nu_{11}^I; \cdots; \nu_{M1}^I; \nu_{12}^I; \cdots; \nu_{M2}^I; \cdots; \nu_{MK}^I] \tag{7}$$

2.3. Discriminant attribute projection

We concatenate the appearance vector $m(x^I)$ and the spatial vector $\nu(x^I)$ as a supver-vector

$$\phi(x^I) = [m(x^I); \sqrt{\eta}v(x^a)],$$

where η is a tuning parameter balance the information contribution from the two sources. However, directly employing such a high-dimensional vector for image classification may not lead to a good performance, because the super-vector is constructed without considering the inter-category or intra-category relationship.

To enhance the discrimating power of our representation, we propose to project $\phi(x^I)$ to a subspace that depresses the directions with high inter-category variabilities. Let V denote the projection matrix toward the subspace with high inter-category variabilities, that is, $(I - V)\phi(x^I)$ is the discriminant projection we are looking for. We solve V via the following objective function

$$V = \arg\max_{V^TV=I} \sum_{i\neq j} ||V^T\phi(x^i) - V^T\phi(x^j)||^2 W_{ij}, \tag{8}$$

where W_{ij}=1 when x^i and x^j belong to the same category, otherwise $W_{ij} = 0$. Let $\Phi = [\phi(x^1), \phi(x^2), \cdots, \phi(x^N)]$, a matrix with N columns where N is the total number of training images. It can be shown that the optimal solution for V consists of the top eigenvectors corresponding to the largest eigenvalues of matrix $\Phi(D - W)\Phi^T$, where D is a diagonal matrix with $D_{ii} = \sum_{j=1}^N W_{ij}, \forall i$.

Suppose we use the dot product as a similarity measure between super-vectors. After applying discriminant attribute projection (DAP), the similarty between two images, x^a and x^b, is equal to

$$D(x^a, x^b) = \phi(x^a)^T(\mathrm{I} - VV^T)\phi(x^b). \tag{9}$$

That is, the projection toward V, which is irrelevant to the classification, is discarded in the similarity calculation.

In the DAP approach, each eigen-direction is either included or excluded for later analysis. An alternative is to adaptively shrink each directions of the subspace spanned by V: the one with larger eigen-values shrunk less and the one with smaller eigen-values shrunk more. Arrange all the shrinkage factors in a diagonal matrix C, then the similarity metric (9) can be reexpressed as

$$D(x^a, x^b) = \phi(x^a)^T(\mathrm{I} - VCV^T)\phi(x^b). \tag{10}$$

In our experiments, we set $\mathrm{C} = \mathrm{I} - \Lambda^{-1}$, where Λ is a diagonal matrix with eigenvalues of matrix $\Phi(D - W)\Phi^T$.

3. Connection to previous work

3.1. Histogram as a special case of GMMs

It is easy to see that the histogram representation is a special case of GMMs, with only the weights w_k^I being

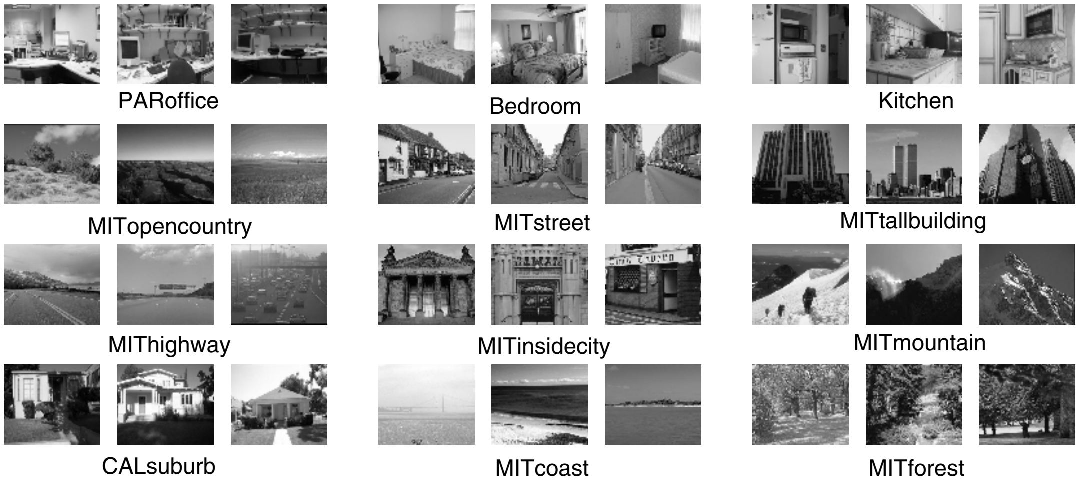


Figure 3. Example images from the scene category database.

adapted: If we set the hyper-parameters $T = 0$ and $r = \infty$, from equations (4, 5), we have all the image-specific GMMs sharing the same mean vectors and covariance matrices, and therefore the only information captured by GMMs is the weight w_k^I which is proportional to the histogram counts.

Here we want to highlight three aspects in which the GMM-based approach extends histograms. First, histograms use the Euclidean distance as the clustering metric in constructing bins, while GMMs use the Mahamalobis distance that takes into account the heterogeneity among features. Second, histograms use a hard decision rule in distributing feature vectors into bins and the resulting data summary is sensitive to noise, while GMMs use a soft decision rule in distributing feature vectors to Gaussian components and the resulting probabilistic summary of the data is more robust. The last and the most important advantage of GMMs over histograms is the gain of information. Histograms summarize the appearance information of an image (i.e., a bag of feature vectors) by the counts in each histogram bin, which correspond to the weights of Gaussian components in the adapted mixture model. In addition to weights, GMMs summarize each image by the adapted mean vectors and covariance matrices, which provide richer information in constructing the super-vector and in calculating similarities between images.

3.2. SPM as a special case of Gaussian maps

To avoid the loss of spatial information with histograms, Lazebnik et al. [11] proposed a successful technique called the spatial pyramid matching (SPM). In SPM, images are repeatedly divided into subregions, similarity measures are repeatedly calculated for each subregions, and their weighted summation forms an overall similarity measure.

Since histogram is a special case of GMMs, SPM corresponds to a hierarchical spatial modeling over a degenerated Gaussian map where the posterior probabilities are either 0 or 1. The special similarity measure used by SPM, the histogram intersection function, corresponds to an intersection function defined over those posterior probabilities. So SPM can be viewed as a special case of Gaussian maps.

4. Experiments

In this section, we report the performance of our image representation on three diverse datasets: fifteen scene categories [10], Caltech101 and CMU PIE face database. We investigate the effectiveness of different aspects of our representation and further compare our results with existing works. All experiments are repeated ten times with different randomly selected training and testing images, and the average of per-class recognition rate is recorded for each run. As we focus on the image representation, we just employ the nearest centroid (NC) classifier in all the experiments. We perform all processing in grayscale, even when color images are available.

4.1. Scene category recognition

The scene database is composed of fifteen scene categories, thirteen provided by Fei-Fei *et al.* in [10] and the other two collected by Lazebnik *et al.* in [11]. Each scene category contains 200 to 400 images. The average size of the images is around 300×250 pixels. This database is one of the most comprehensive scene category databases used in the literature. Example images of different scene categories of this database are illustrated in Figure 3.

Here, the experiment setting is purposely made the same as that in [10] and [11] to guarantee the fairness of performance comparison. Specifically, all experiments are repeated ten times with 100 randomly selected images per class for training and the rest for testing. The 128-dimensional SIFT vector is extracted within a 20 ×20 patch over a grid with spacing of 5 pixels. The dimension of SIFT descriptor is reduced to 64 by Principal Component Analysis (PCA). The GMM contains 512 Gaussian components, while histogram contains 512 bins.

Table 1. Performance comparison on scene category database.

Algorithm	Average accuracy (%)
Histogram [10]	65.2
SPM [11]	81.4
HG	**85.2**

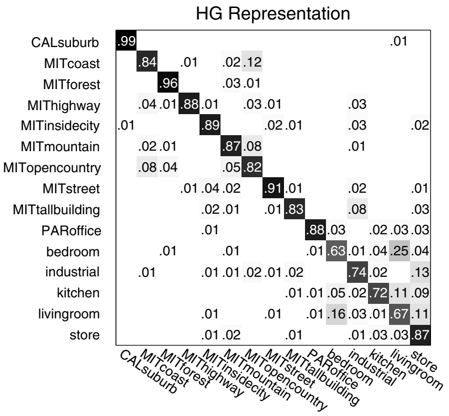
Table 1 compares our approach with several existing systems on the scene classification task. The result in [10] is 65.2%, which is based on histogram representation without any spatial information. In [11], Lazebnik *et al.* introduced spatial pyramid matching (SPM) to incorporate the spatial information with histogram representation and reported an accuracy of 81.4% using SVM with nonlinear histogram intersection kernel. In the experiment, by a simple nearest centroid (NC) classifier, HG representation achieves a superior performance of 85.2% in accuracy. The results are consistent with our analysis in the previous sections: HG is more general than both histogram and SPM.

Table 2. The classification results on scene category database.

Algorithm	Average accuracy (%)
Histogram	41.8
GMM	75.8
GMM+GM	80.4
GMM+DAP	82.1
HG	85.2

Table 2 gives an in-depth analysis into the effectiveness of each aspect of our representation. Here all the results are obtained by nearest centroid (NC) classifier. The table demonstrates the performance when adding the components of our representation one by one. It is evident that the three components, GMM for appearance representation, Gaussian maps (GM) for spatial layout encoding and DAP for discriminant dimension reduction jointly improve the recognition accuracy. Note that [11] reported an accuracy of 74.2% based on histogram representation, which is higher than 41.8% here. This is because [11] employed a nonlinear histogram intersection kernel for SVM. This indicated that the performance of histogram representation is sensitive to choice of kernel metrics, and relies heavily on the classifier.

Figure 4 shows the confusion matrix between the fifteen scene categories for the HG representation.



HG Representation

	CALsuburb	MITcoast	MITforest	MIThighway	MITinsidecity	MITmountain	MITopencountry	MITstreet	MITtallbuilding	PARoffice	bedroom	industrial	kitchen	livingroom	store
CALsuburb	.99													.01	
MITcoast		.84		.01		.02	.12								
MITforest			.96			.03	.01								
MIThighway		.04	.01	.88	.01		.03	.01				.03			
MITinsidecity	.01				.89			.02	.01			.03			.02
MITmountain		.02	.01			.87	.08					.01			
MITopencountry		.08	.04			.05	.82								
MITstreet				.01	.04	.02		.91	.01			.02			.01
MITtallbuilding					.02	.01		.01	.83			.08			.03
PARoffice					.01					.88	.03		.02	.03	.03
bedroom			.01			.01				.01	.63	.01	.04	.25	.04
industrial		.01			.01	.01	.02	.01	.02			.74	.02		.13
kitchen									.01	.01	.05	.02	.72	.11	.09
livingroom					.01			.01		.01	.16	.03	.01	.67	.11
store					.01	.02			.01			.01	.03	.03	.87

Figure 4. Confusion matrix on scene category database for the HG representation. The average classification accuracy is 85.2%. The entry in the i^{th} row and j^{th} column is the percentage of images from class i that were misidentified as class j. For better viewing, please see the pdf file

4.2. Object recognition

Our second set of experiments are conducted on Caltech101 database. This database consists of 101 object classes with high intra-class appearance and shape variability. The number of images in each class vary from 31 to 800, and most images are of medium resolution (about 300 × 300 pixels). This database is one of the most diverse and thoroughly studied databases for object recognition, and significant progress has been made on it for state-of-the-art algorithms. There exist several drawbacks for this database, though. For example, most objects is located at the center of an image with little cluttered background. And many classes are devoid of pose and scale variability. Moreover, the presence of rotation artifacts tend to make some classes (e.g. minaret) much easier to be identified.

In this experiment, the representation step is the same as in scene recognition: first extract SIFT descriptor within a 20 × 20 sliding window, and then learn a 512-mixture GMM and Gaussian map for each image. For experiment setup, we follow the standard procedure, namely we randomly selectly 15 and 30 training images per class and 50 for testing. The recognition rate is then computed as the average of per-class accuracies. Similar to the previous experiments, the entire procedure is repeated ten times, and the average performance and its standard deviation are reported.

Table 3 shows a performance comparison of HG representation with several recently reported methods, all based on a single descriptor. At both training/testing settings, HG representation achieves the best result, i.e., 65.5% for 15 training images and 73.1% for 30 training images. It is worth noting that, most of previous methods used computing-extensive classifiers, such as support vector machine (SVM) and nearest neighbor (NN), or a hybrid of them [11, 23, 24, 25]. Especially for NBNN [21], although it achieved a comparable performance at 15 training samples, it involves finding the nearest patch among all patches in each class, which is extremely time consuming at the testing phase. While it is true that the computational burden of NBNN can be somehow alleviated by approximated-k-nearest-neighbor algorithm, there is still a big issue when the number of labeled samples increase, where this is often the case in many real-world applications. In our framework, however, the classification step becomes trivial after the representation is obtained. We only need to compute the distance from each class centroid (image-to-class distance), of which the cost is constant for a given number of classes.

Table 3. Performance comparison on Caltech101 (single descriptor).

Algorithm	15 Train	30 Train
HS+LS [29]	–	53.9
SPM [11]	56.4	64.6
SVM-KNN [25]	59.1	66.2
GBDist-SVM [23]	59.3	–
GBDist-NN [23]	45.2	–
Griffin SPM [24]	59.0	67.6
LearnDist [22]	63.2	–
ML+CORR [30]	61.0	69.6
NBNN [21]	65.0	70.4
HG	**65.5**	**73.1**

Figure 5. Example images from the CMU PIE database.For each subject, there are 170 near frontal face images under varying pose, illumination, and expression.

4.3. Face recognition

We further investigate the performance of our representation on face recognition. We use the CMU PIE database. This database contains 41,368 face images from 68 individuals. For each individual, face images of varying pose, illumination, and expression are captured by 13 synchronized cameras under 21 flashes. Example images from this database are illustrated in Figure 5. We choose the five near frontal poses (C05, C07, C09, C27, C29) and use all such images under different illuminations, lighting and expressions, which leaves us 170 near frontal face images for each individual. A random subset with $l = \{10, 20, 30\}$ images per individual is taken to form the training set, while the rest of the database was considered to be the testing set. For each given l, we average the results over 10 random splits. All the configuration is the same as in [6].

In the experiments, original images are manually aligned (two eyes are aligned at the same position), cropped, and then re-sized to 32 × 32 pixels, with 256 gray levels per pixel. The patch size is set to be 6 × 6 pixels, and the patches are densely sampled pixel by pixel. As suggested in [14], we extract feature of each patch by 2-D Discrete Cosine Transform (DCT). The GMM contains 256 Gaussian components, while histogram contains 256 bins.

Table 4. Face recognition accuracy (%) on PIE database.

	10 Train	20 Train	30 Train
OLPP [6]	88.6	93.5	95.2
Histogram	68.8 ± 0.69	78.5 ± 0.63	82.2 ± 0.53
GMM+DAP	96.1 ± 0.52	98.9 ± 0.21	99.5 ± 0.08
HG	**96.3 ± 0.45**	**99.1 ± 0.18**	**99.6 ± 0.09**

Table 4 shows the recognition results on PIE database. Note that:

1. The proposed HG representation performed significantly better than orthogonal locality preserving projection (OLPP) proposed in [6], The relative error reduction is more than 65% for any configuration. This indicates that HG representation can well summarize the image informance on well-alignment image classification tasks.

2. The accuracies by histogram representation are much worse than the performance by OLPP. The results indicate that the detail of appearance are quite important for face recognition, while it is difficult to be captured by histogram.

3. Unlike in scene/object recognition task, the spatial layout does not show much benefit in performance for face recognition. This seems reasonable since the differences between face images is mostly due to detailed appearance rather than the spatial configuration.

5. Conclusion

In this paper, we proposed a new model-based image representation, namely Hierarchical Gaussianization (HG), to incorporate both the appearance and spatial information under a hierarchical structure. A supervised dimension reduction technique, DAP, is then proposed to further enhance the discriminating power of the HG representation. Experiments on fifteen scene category database, Caltech101 database and CMU PIE database all demonstrate that the HG representation greatly outperforms previous proposed approaches. Furthermore, the superior performance is achieved by simply adopting NC classifier, which suggests the application of our representation in large-scale recognition tasks.

References

[1] Yan Liu and Florent Perronnin. A Similarity Measure Between Unordered Vector Sets with Application to Image Categorization *CVPR*, 2008.

[2] S. Yan, X. Zhou, M. Liu, M. Hasegawa-Johnson, and T.S. Huang. Regression from patch-kernel *CVPR*, 2008.

[3] X. Zhou, X. Zhuang, S. Yan, S. Chang, M. Hasegawa-Johnson, and T.S. Huang. SIFT-Bag Kernel for Video Event Analysis *ACM MM*, 2008.

[4] B. Schiele and J. Crowley. Recognition without correspondence using multidimensional receptive field histograms. *IJCV*, vol. 36, no. 1, pp. 31-50, 2000 2007.

[5] M. Swain and D. Ballard. Color indexing. *IJCV*, vol. 7, no. 1, pp. 11-32, 1991 2007.

[6] Deng Cai, Xiaofei He, Jiawei Han and Hong-Jiang Zhang. Orthogonal Laplacianfaces for Face Recognition. *IEEE Transactions on Image Processing*, vol. 15, no. 11, pp. 3608-3614, November, 2006.

[7] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, pp. 993-1022, 2003

[8] A. Hatch, S. Kajarekar, and A. Stolcke. Within-Class covariance normalization for svm-based speaker recognition. *ICSLP*, pp. 1471-1474, 2006

[9] J. Hartigan and M. Wang. A k-means clustering algorithm. *Applied Statistics*, vol. 28, pp. 100-108, 1979

[10] L. Fei-Fei and P. Perona. A Bayesian hierarchical model for learning natural scene categories. *CVPR*, 2005

[11] S Lazebnik, C Schmid and J Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. *CVPR*, 2006

[12] J. Farquhar, S. Szedmak, H. Meng, and J. Shawe-Taylor. Improving bag-of-keypoints image categorisation. *Technical report*, 2005

[13] D. Larlus and F. Jurie. Latent mixture vocabularies for object categorization. *British Machine Vision Conference*, 2006

[14] S. Lucey and T. Chen. A GMM Parts Based Face Representation for Improved Verification through Relevance Adaptation. *CVPR*, pp. 855-861, 2004.

[15] F. Moosmann, B. Triggs, and F. Jurie. Randomized clustering forests for building fast and discriminative visual vocabularies. *NIPS*, 2007

[16] F. Perronnin, C. Dance, G. Csurka, and M. Bressian. Adapted vocabularies for generic visual categorization. *European Conference on Computer Vision*, 2006

[17] J. Winn, A. Criminisi, and T. Minka. Object categorization by learned universal visual dictionary. *International Conference on Computer Vision*, 2005.

[18] J. C. van Gemert, J.M. Geusebroek, C.J. Veenman, and A.W.M. Smeulders. Kernel Codebooks for Scene Categorization. *European Conference on Computer Vision*, 2008.

[19] Liu Yang, Rong Jin, Rahul Sukthankar, and Frederic Jurie. Unifying Discriminative Visual Codebook Generation with Classifier Training for Object Category Recognition. *CVPR*, 2008

[20] P.C. Woodland and D. Povey. Large scale discriminative training of hidden Markov models for speech recognition Category Recognition. *Computer Speech & Language*, 2002

[21] O. Boiman, E. Shechtman, and M. Irani. In Defense of Nearest-Neighbor Based Image Classification. *CVPR*, 2008

[22] A. Frome, Y. Singer, F. Sha, and J. Malik. Learning globally consistent local distance functions for shape-based image retrieval and classification. *ICCV*, 2007

[23] M. Varma and D. Ray. Learning the discriminative power-invariance trade-off. *CVPR*, 2007

[24] G. Griffin, A. Holub, and P. Perona. Caltech 256 object category dataset. Technical Report, California Institute of Technology, 2007

[25] H. Zhang, A. Berg, M. Maire, and J. Malik. SVM-KNN: Discriminative nearest neighbor classification for visual category recognition. *CVPR*, 2006

[26] A. Agarwal and B. Triggs. Hyperfeatures-multilevel local coding for visual recognition. *Lecture Notes in Computer Science*, vol. 3951, 2006

[27] T. Tuytelaars and C. Schmid. Vector quantizing feature space with a regular lattice. *ICCV*, 2007

[28] J. Philbin, O. Chum, M. Isard, J. Sivic and A. Zisserman. Lost in quantization: Improving particular object retrieval in large scale image databases. *CVPR*, 2008

[29] J. Zhang, M. Marszalek, S. Lazebnik and C. Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. *IJCV*, vol. 73, pp. 213-238, 2008

[30] P. Jain, B. Kulis and K. Grauman. Fast image search for learned metrics. *CVPR*, 2008