論文 / 著書情報
Article / Book Information

Title	q-Gaussian Mixture Models Based on Non-Extensive Statistics for Image And Video Semantic Indexing
Author	Nakamasa Inoue, Koichi Shinoda
Journal/Book name	ACCV2012, , ,
発行日 / Issue date	2012, 11
DOI	http://dx.doi.org/10.1007/978-3-642-37444-9_39
権利情報 / Copyright	The original publication is available at www.springerlink.com.
Note	このファイルは著者（最終）版です。 This file is author (final) version.

q-Gaussian Mixture Models Based on Non-Extensive Statistics for Image And Video Semantic Indexing

Nakamasa Inoue, Koichi Shinoda

Dept. of Computer Science, Tokyo Institute of Technology, Japan.
inoue@ks.cs.titech.ac.jp, shinoda@cs.titech.ac.jp

Abstract. Gaussian mixture models (GMMs) which extend the bag-of-visual-words (BoW) to a probabilistic framework have been proved to be effective for image and video semantic indexing. Recently, the q-Gaussian distribution, which is derived in the non-extensive statistics, has been shown to be useful for representing patterns in many *complex* systems in physics such as fractals and cosmology. We propose q-Gaussian mixture models (q-GMMs), which are mixture models of q-Gaussian distributions, for image and video semantic indexing. It has a parameter q to control its tail-heaviness. The long-tailed distributions obtained for $q > 1$ are expected to effectively represent complexly correlated data, and hence, to improve robustness against outliers. In our experiments, our proposed method outperformed the BoW method and achieved 49.4% and 10.9% in Mean Average Precision on the PASCAL VOC 2010 dataset and the TRECVID 2010 Semantic Indexing dataset, respectively.

1 Introduction

With the advent of web-sites for sharing multimedia content, a huge amount of image and video data has been made available. To improve image/video search performance, automatic assignment of semantic tags representing objects, events, and scenes is necessary. However, assigning semantic tags is a challenging task owing to the variety of object categories and the wide range of their poses and motion.

Recent research [1–4] and results of the TRECVID Semantic Indexing Task [5] and the Pascal Visual Object Classes Challenge [6] indicate that the bag-of-visual-words (BoW) [1] is an effective representation for images and videos. In the BoW method, each low-level feature (e.g. SIFT [7]) extracted from an image is assigned to a visual word, i.e., a code word obtained by vector quantization (VQ). To improve the performance of BoW, the Gaussian mixture models (GMMs) [8] which extend BoW to a probabilistic framework are often utilized. The GMMs perform better than BoW since they reduce quantization errors occurred in VQ.

Recently, the q-Gaussian distribution, which is derived in the non-extensive statistics, has been shown to be useful for representing patterns in many *complex*

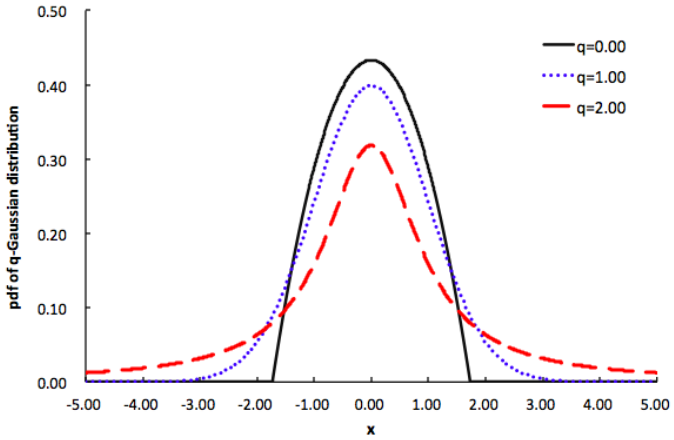

Fig. 1. The q-Gaussian distributions. The (normal) Gaussian distribution is obtained when $q = 1$. The tail of a q-Gaussian distribution is longer than that of a Gaussian distribution when $q > 1$.

systems in physics such as fractals and cosmology. The q-Gaussian distribution is a generalization of the Gaussian distribution and is derived by maximizing Tsallis entropy [9] while the Gaussian distribution is derived by maximizing Boltzmann-Gibbs entropy. The q-Gaussian distribution has a parameter q to control its tail-heaviness as shown in Fig. 1. Many statistical frameworks including the BoW method can be extended to the non-extensive statistics by using the q-Gaussian distribution. The long-tailed distributions obtained for $q > 1$ are expected to effectively represent complexly correlated data, and hence, to improve robustness against outliers. Due to this background, we propose q-Gaussian mixture models (q-GMMs) based on the non-extensive statistics and their application to image and video semantic indexing systems.

This paper is organized as follows. The proposed method with the definition of q-GMMs is described in Sec. 2. Experimental results on PASCAL VOC and TRECVID dataset are described in Sec. 3. Conclusion and future work are described in Sec. 4.

2 Proposed Method

The procedure of the proposed image and video semantic indexing based on q-Gaussian mixture models (q-GMMs) is shown in Fig. 2. First, low-level features (e.g. SIFT features) are extracted from image/video data. Second, a q-GMM for a background model is estimated from low-level features in training data. The background model is used instead of a codebook for BoW. Finally, we propose two separate methods based on q-GMMs: histogram-based representation (Sec. 2.3) and q-GMM kernel (Sec. 2.4). These methods are used as input for a discriminative classifier such as a support vector machine (SVM).

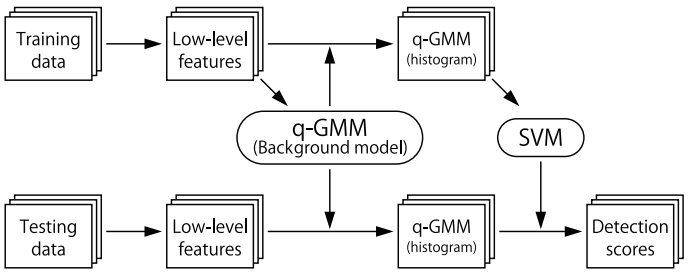

Fig. 2. The framework of image and video semantic indexing using q-Gaussian mixture models.

2.1 q-Gaussian Mixture Models

The q-Gaussian distribution derived in the non-extensive statistics is a generalized Gaussian distribution with a parameter q to control its tail-heaviness. The probability density function of the q-Gaussian distribution is given by

$$\mathcal{N}_q(x|\mu,\Sigma) = \begin{cases} \frac{1}{Z_q}\left(1-\frac{1-q}{3-q}(x-\mu)^T\Sigma^{-1}(x-\mu)\right)^{\frac{1}{1-q}}, \\ \qquad \text{if } (x-\mu)^T\Sigma^{-1}(x-\mu) < \frac{3-q}{1-q} \\ 0, \qquad \text{otherwise} \end{cases} \tag{1}$$

where μ is a mean vector, Σ is a covariance matrix, and Z_q is a normalising constant to make the integral over Eq. (1) to 1. Fig. 1 shows the shape of q-Gaussian distributions for some q-values. The q-Gaussian distribution has a longer tail than the Gaussian distribution when $q > 1$, and approximates to the Gaussian distribution when $q \to 1$. The long-tailed distribution obtained for $q > 1$ is expected to be effective for representing complexly correlated data.

To further improve the expressiveness of the q-Gaussian distribution, we introduce a mixture model of q-Gaussian distributions, namely a q-Gaussian mixture model (q-GMM), by

$$p_q(x|\theta) = \sum_{k=1}^{K} w_k \mathcal{N}_q(x|\mu_k,\Sigma_k), \tag{2}$$

where K is the number of mixtures, w_k is a mixture coefficient, and $\theta = \{w_k,\mu_k,\Sigma_k\}_{k=1}^K$ is a set of q-GMM parameters.

2.2 Training q-GMM for a Background Model

From a set of low-level features in training data, we estimate q-GMM parameters for a background model which is used instead of a codebook for BoW. The most popular way to estimate parameters of probabilistic models with hidden variables

such as GMMs is to use the expectation maximization (EM) algorithm. However, it is difficult to directly apply the EM algorithm to q-GMMs since we face a non-linear equation in the maximization step of the algorithm. To avoid this problem, we first estimate q-GMM parameters in the same way as the EM algorithm for a normal GMM (See Appendix for details) and then update them for a q-GMM.

Let $X = \{x_i\}_{i=1}^N$ be a set of low-level features extracted from training data. We randomly initialize q-GMM parameters $\hat{\theta} = \{\hat{w}_k, \hat{\mu}_k, \hat{\Sigma}_k\}_{k=1}^K$ and update them iteratively as follows:

$$\hat{\mu}_k = \frac{1}{C_k} \sum_{i=1}^{N} c_{ik} x_i, \tag{3}$$

$$\hat{\Sigma}_k = \frac{1}{C_k} \sum_{i=1}^{N} c_{ik} (x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^T, \tag{4}$$

$$\hat{w}_k = \frac{C_k}{\sum_{k=1}^K C_k}, \tag{5}$$

where c_{ik} is the posterior probability of x_i being at the k-th q-Gaussian component given by

$$c_{ik} = \frac{\hat{w}_k \mathcal{N}_q(x_i|\hat{\mu}_k, \hat{\Sigma}_k)}{\sum_{k=1}^K \hat{w}_k \mathcal{N}_q(x_i|\hat{\mu}_k, \hat{\Sigma}_k)}, \tag{6}$$

and $C_k = \sum_i c_{ik}$.

While the estimated $\hat{\Sigma}_k$ is a covariance matrix, covariance of x that follows $\mathcal{N}_q(\cdot|\mu_k, \Sigma_k)$ is given by,

$$\mathbb{V}[x] = \frac{3-q}{5-3q} \Sigma_k. \tag{7}$$

Therefore, for q-GMMs, we multiply $\hat{\Sigma}_k$ by $\frac{5-3q}{3-q}$ after the iterations converge, i.e., the final update equation is given by

$$\hat{\Sigma}_k = \frac{5-3q}{3-q} \hat{\Sigma}_k. \tag{8}$$

Here, we assume $q < \frac{5}{3}$ since a q-Gaussian distribution has an infinite variance if $q \geq \frac{5}{3}$.

2.3 q-GMM for histogram-based image representation

To represent an image by a feature vector, we create a histogram of low-level features $H(X')$ from a set of low-level features $X' = \{x_i\}_{i=1}^{N'}$ extracted from an image as follows:

$$H(X') = \begin{pmatrix} C_1 \\ C_2 \\ \vdots \\ C_K \end{pmatrix}, \; C_k = \sum_i c_{ik}, \tag{9}$$

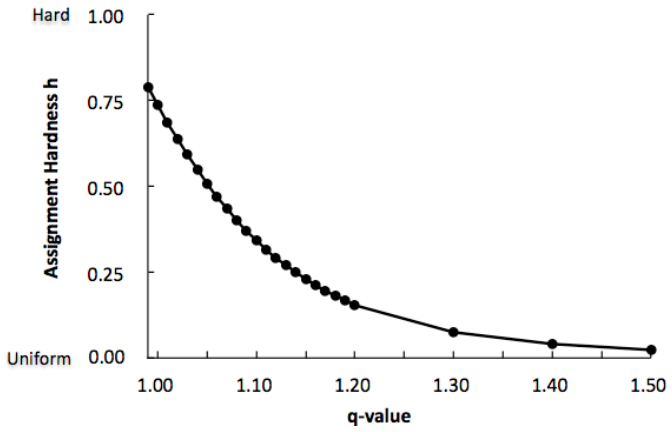

Fig. 3. Assignment hardness h with different q-values.

where c_{ik} is the posterior probability of x_i being at the k-th q-Gaussian component given by Eq. (6). The posterior propabilities c_{ik} can be viewed as weights in soft-assignment of visual words since they satisfy

$$\sum_{k=1}^{K} c_{ik} = 1, \ 0 \leq c_{ik} \leq 1. \tag{10}$$

Thus, the q-GMM can be regarded as an extension of BoW to a probabilistic framework.

We found that the assignment using the q-GMM comes close to the hard-assignment (i.e., only one of c_{ik} is equal to 1.0 and others are 0.0) as q decreases, and comes close to the uniform-assignment (i.e., all c_{ik} have equivalent values) as q increases. To measure how much the assignment is close to the hard-assignment, we introduce the assignment *hardness* h defined by

$$h = \frac{1}{N'} \sum_{i=1}^{N'} \frac{\max_k c_{ik} - K^{-1}}{1 - K^{-1}}. \tag{11}$$

The assignment hardness h is designed to reach 1.0 for the hard-assignment and 0.0 for the uniform-assignment.

Fig. 3 shows the assignment hardness with different q-values. To improve the final performance of image and video indexing, the assignment should not be too hard nor too uniform. Here, we employ a q-value of 1.05 that has the middle value (0.5) of assignment hardness.

2.4 q-GMM Kernel

Although histograms are utilized to represent the distribution of low-level features in the BoW method, probabilistic models are expected to perform bet-

Table 1. The targeted semantic concepts for PASCAL VOC 2010 and TRECVID 2010.

PASCAL VOC 2010
Aeroplane, Bicycle, Bird, Boat, Bottle, Bus, Car, Cat, Chair, Cow, Diningtable, Dog, Horse, Motorbike, Person, Pottedplant, Sheep, Sofa, Train, Tvmonitor.
TRECVID 2010
Airplane Flying, Animal, Asian People, Bicycling, Boat Ship, Bus, Car Racing, Cheering, Cityscape, Classroom, Dancing,Dark-skinned People, Demonstration Or Protest, Doorway, Explosion Fire, Female Human Face Closeup, Flowers, Ground Vehicles, Hand, Mountain, Nighttime, Old People, Running, Singing, Sitting down, Swimming, Telephones, Throwing, Vehicle, Walking.

ter than the histograms. Here, we introduce q-GMMs instead of the BoW histograms.

Generally, the number of low-level features extracted from an image is limited and may not be enough to estimate q-GMM parameters robustly. Thus, we use the maximum a posteriori criteria for GMMs which provides robust parameter estimation. For each image that has low-level features $X' = \{x_i\}_{i=1}^{N'}$, we only update q-GMM mean vectors from the background model as follows:

$$\tilde{\mu}'_k = \frac{\tau\hat{\mu}_k + \sum_{i=1}^{N'} c_{ik}x_i}{\tau + \sum_{i=1}^{N'} c_{ik}}, \tag{12}$$

where N' is the number of the low-level features, $\hat{\mu}_k$ is a q-GMM parameter for the background model, τ is a prefixed parameter, and c_{ik} is the posterior probability given by Eq. (6).

For a kernel to train support vector machines (SVMs), we introduce the following RBF-based kernel, namely q-GMM kernel,

$$k(X', X'') = \exp\left(-\gamma \sum_{k=1}^{K} \hat{w}_k (\tilde{\mu}'_k - \tilde{\mu}''_k)^T \hat{\Sigma}_k^{-1} (\tilde{\mu}'_k - \tilde{\mu}''_k)\right), \tag{13}$$

where X' is a set of low-level features extracted from an image, $\tilde{\mu}'_k$ is an updated q-GMM mean vector, $\hat{\Sigma}_k, w_k$ are q-GMM parameters for the background model, and γ is a scaling parameter. The weighted sum of Mahalanobis distance between the k-th q-Gaussian components is utilized in the q-GMM kernel.

3 Experiments

3.1 Experimental Conditions

Data set and evaluation measure We evaluate the proposed method on the PASCAL VOC 2010 dataset (VOC 2010 classification (validation) challenge

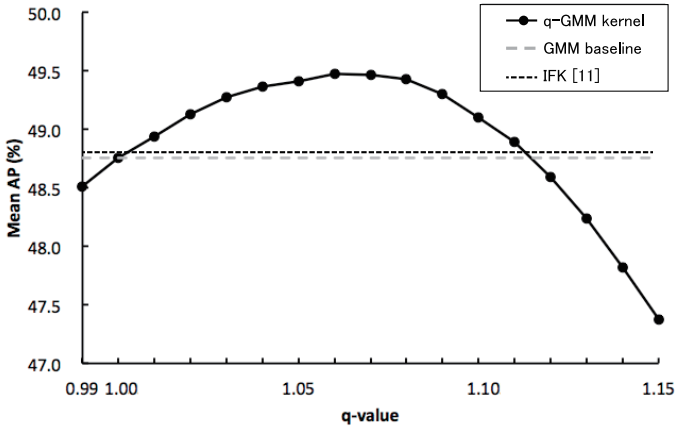

Fig. 4. The performance comparison of q-GMM kernels with different q-values on the PASCAL VOC 2010 dataset. The q-GMM kernel outperforms the GMM baseline ($q = 1.00$) and the improved Fisher kernel [11] of GMM means.

benchmark) [6] and the TRECVID 2010 dataset (TRECVID 2010 semantic indexing task benchmark) [5]. The PASCAL VOC 2010 dataset has 4,998 training images and 5,105 testing images with annotated labels for 20 object classes (Table 1). The TRECVID 2010 dataset, which consists of 400 hours of Internet archive videos with their shot boundaries, has 119,685 training video shots and 146,788 testing video shots with annotated labels for 30 semantic concepts (Table 1). Key-frame images are also provided for each video shot. Mean average precision (Mean AP) is used as an evaluation measure which is the mean of APs over all targeted semantic concepts.

Experimental Settings For low-level features, SIFT features (128-dimension) [7] with hue histogram (36-dimension) [10] are extracted from a 100x100 grid with 3 different scales. Principal component analysis is applied to reduce their dimention to 32. The number of mixture components K and the parameter τ in Eq. (12) are set to 512 and 20.0, respectively.

3.2 Experimental Results

Mean AP on the PASCAL VOC 2010 dataset was 30.9% for the (hard-assignment) BoW method, and it was improved to 32.1% by using a q-GMM (histogram). For the q-GMM, $q = 1.05$ is used since it has hardness of 0.5 as shown in Subsec. 2.3. The q-GMM kernel performed better than these methods and achieved 49.4% in Mean AP. Fig. 4 compares the performance of q-GMM kernels with different q-values. It is shown that the q-GMM performed better than the normal GMM ($q = 1.00$).

On the TRECVID 2010 dataset, the q-GMM kernel ($q = 1.05$) achieved 7.11% in Mean AP and also performed better than the normal GMM as shown

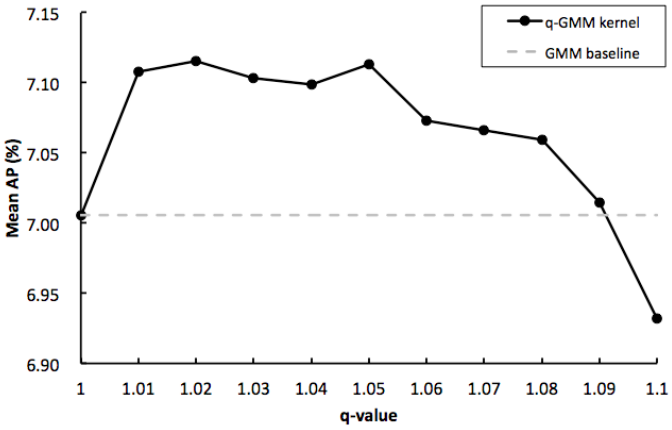

Fig. 5. The performance comparison of q-GMM kernels with different q-values on the TRECVID 2010 dataset.

in Fig. 5. Fig. 6 shows some examples of detected video shots. We also found that the best choice of the q-value was different for each of targeted semantic concepts. The q-GMM kernel achieved 7.50% in Mean AP if we use the best q-values for each semantic concept. Thus, a q-value selection method is needed to improve the performance in the future work.

3.3 Comparison with other methods

We compare our method with the improved Fisher kernel (IFK) [11, 12] in Fig. 4. In IFK, we extracted Fisher vectors for GMM means and applied L2 and power normalization as in [11]. The parameter of the power normalization was set to 0.4 which performed the best in our experiments. The q-GMM kernel performed better than the IFK which achieved state-of-the-art results in the recent PASCAL VOC challenges from 2009 to 2011. Spatial modeling such as spatial pyramid matching [13] may further improve its performance.

Fig. 7 shows performance comparison with other methods used in the TRECVID 2010 Semantic Indexing Task [5]. Mean AP of 7.11%, which was obtained by using our q-GMM kernel ($q = 1.05$), ranked 10-th among 87 runs. We conclude the q-GMM kernel performed well since other methods typically used more than 5 types of low-level features while we used only one type of low-level featres (SIFT with hue histogram). Furthermore, we achieved Mean AP of 10.9%, which is better than the best performance on the TRECVID 2010, by combining q-GMMs for 4 additional types of low-level features: SIFT (Harris-affine detector), SIFT (Hessian-affine detector), dense HOG, and MFCC audio features.

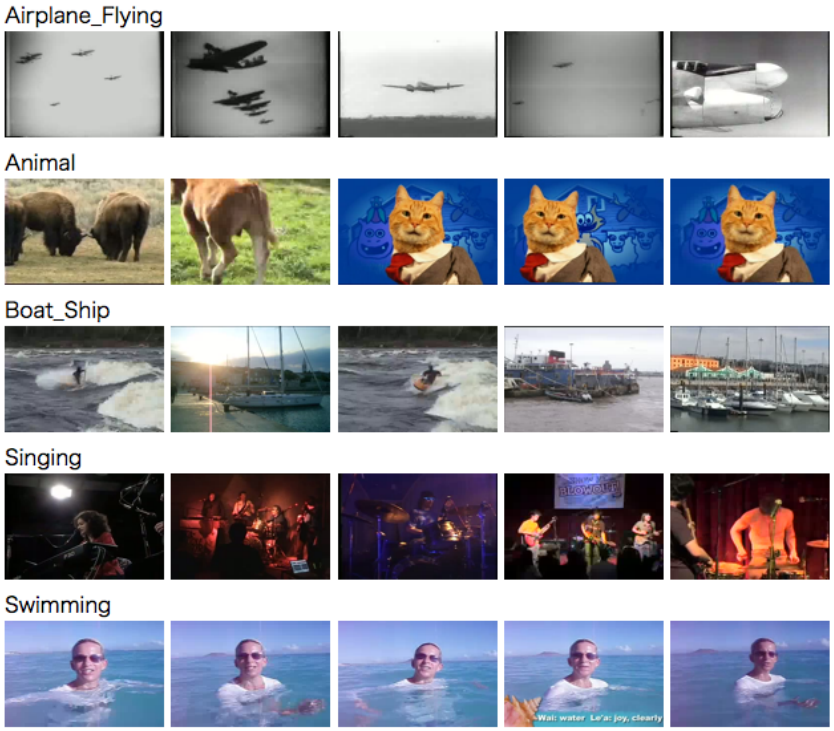

Fig. 6. Examples of detected video shots in TRECVID 2010 dataset. Top 5 video shots are shown for five semantic concepts.

4 Conclusion

We proposed q-Gaussian mixture models (q-GMMs) and their application to image and video semantic indexing systems. It has been shown in our experiments that the q-GMM kernels outperform both of the BoW method and the normal GMM. Our future work will focus on optimization of q-values and other applications of q-Gaussian distribution and non-extensive statistics.

References

1. G. Csurka, et al. Visual categorization with bags of keypoints. In Proc. of *ECCV SLCV workshop*, pp. 1–22, 2004.
2. K. E. A. van de Sande, et al. Evaluating color descriptors for object and scene recognition. In *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 32(9), pp. 1582–1596, 2010.
3. X. Zhou, et al. Image classification using super-vector coding of local image descriptors. In Proc. of *ECCV*, pp. 141–154, 2010.
4. N. Inoue, and K. Shinoda. A Fast MAP Adaptation Technique for GMM-supervector-based Video Semantic Indexing Systems. In Proc. of *ACM Multimedia*, pp. 1357–1360, 2011.

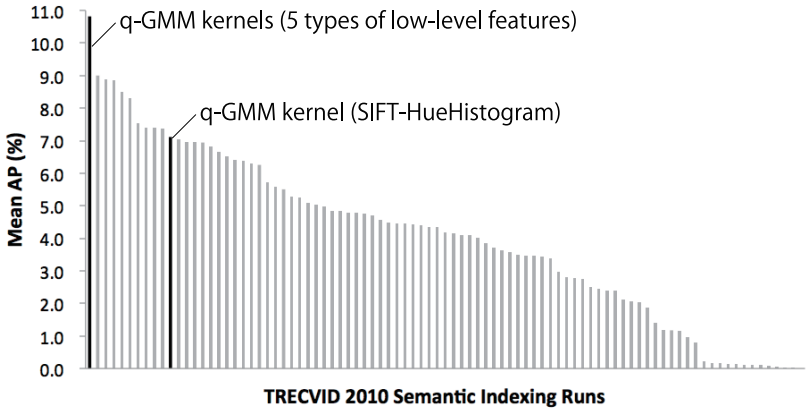

Fig. 7. The performance comparison with other methods in TRECVID 2010. We achieved 7.11% in Mean AP by using a q-GMM kernel with SIFT-HueHistogram features and achieved 10.9% with additional 4 types of low-level features.

5. A. F. Smeaton, et al. Evaluation campaigns and trecvid. In Proc. of *ACM Multimedia MIR workshop*, pp. 321–330, 2006.
6. M. Everingham, et al. The PASCAL Visual Object Classes Challenge 2010 (VOC2010) Results. http://www.pascal-network.org/challenges/VOC/
7. D. G. Lowe. Distinctive image features from scale-invariant keypoints. *In IJCV*, vol. 60 (2), pp. 91–110, 2004.
8. F. Perronnin, et al. Adapted Vocabularies for Generic Visual Categorization. In Proc. of *ECCV*, pp. 464–475, 2006.
9. C. Tsallis, Possible generalization of boltzmann-gibbs statistics, In *Journal of Statistical Physics*, vol. 52, pp. 479–487, 1988.
10. J. van de Weijer and C. Schmid. Coloring local feature extraction. In Proc. of *ECCV*, pp. 334–348, 2006.
11. F. Perronnin, J. Sanchez, and T. Mensink, Improving the Fisher Kernel for Large-Scale Image Classification. In Proc. of *ECCV*, pp. 143–156, 2010.
12. F. Perronnin, and C. Dance, Fisher kernels on visual vocabularies for image categorization. In Proc. of *CVPR*, pp. 1–8, 2007.
13. S. Lazebnik, C. Schmid, and J. Ponce. Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. In Proc. of *CVPR*, pp. 2169–2178, 2006.

Appendix

EM algorithm for q-GMMs

The most popular way to estimate parameters of a mixture model is the expectation maximization (EM) algorithm. Followings are the details of the EM algorithm for q-GMMs.

E-step Evaluate posterior probabilities c_{ik} as follows:

$$c_{ik} = \frac{\hat{w}_k \mathcal{N}_q(x_i|\hat{\mu}_k, \hat{\Sigma}_k)}{\sum_{k=1}^{K} \hat{w}_k \mathcal{N}_q(x_i|\hat{\mu}_k, \hat{\Sigma}_k)}, \tag{13}$$

M-step To derive the parameter-update rules for the M-step, we introduce a Q-function given by

$$Q(\theta) = \log \prod_i p_q(x_i|\theta) + \lambda \left(1 - \sum_{k'} w_{k'}\right) \tag{14}$$

where p_q is a pdf of a q-GMM defined by Eq. (2). A lagrangian multiplier λ is introduced to obtain w_k such that

$$\sum_{k'} w_{k'} = 1. \tag{15}$$

The derivations of the Q-function for each parameter are given by

$$\frac{\partial}{\partial \mu_k} Q(\theta) = \sum_i \frac{a_{ik} w_k \mathcal{N}_q(x_i|\mu_k, \Sigma_k)}{\sum_{k'} w_{k'} \mathcal{N}_q(x_i|\mu_{k'}, \Sigma_{k'})} \Sigma_k^{-1}(x_i - \mu_k) \tag{16}$$

$$= \Sigma_k^{-1} \sum_i a_{ik} c_{ik} (x_i - \mu_k), \tag{17}$$

$$\frac{\partial}{\partial \Sigma_k} Q(\theta) = \frac{1}{2} \sum_i \frac{w_k \mathcal{N}_q(x_i|\mu_k, \Sigma_k)}{\sum_{k'} w_{k'} \mathcal{N}_q(x_i|\mu_{k'}, \Sigma_{k'})} \Sigma_k^{-2} \left(a_{ik}(x_i - \mu_k)(x_i - \mu_k)^T - \Sigma_k\right) \tag{18}$$

$$= \frac{1}{2} \Sigma_k^{-2} \sum_i c_{ik} \left(a_{ik}(x_i - \mu_k)(x_i - \mu_k)^T - \Sigma_k\right), \tag{19}$$

$$\frac{\partial}{\partial w_k} Q(\theta) = \sum_i \frac{\mathcal{N}_q(x_i|\mu_k, \Sigma_k)}{\sum_{k'} w_{k'} \mathcal{N}_q(x_i|\mu_{k'}, \Sigma_{k'})} - \lambda \tag{20}$$

$$= \frac{1}{w_k} \sum_i c_{ik} - \lambda, \tag{21}$$

where

$$a_{ik} = \frac{2}{3 - q - (1 - q)(x_i - \mu_k)^T \Sigma_k^{-1}(x_i - \mu_k)}. \tag{22}$$

Parameter-update rules for the M-step are obtained by setting the derivatives of Q to zero. For mixture coefficients w_k, we obtain

$$\hat{w}_k = \frac{C_k}{\sum_{k=1}^{K} C_k}, \tag{23}$$

from Eq. (21) where $C_k = \sum_i c_{ik}$. However, it is difficult to obtain $\hat{\mu}_k$ and $\hat{\Sigma}_k$ from Eqs. (17) and (19) analitically since μ_k and Σ_k appear in a_{ik}.

For a case $q \simeq 1$, we obtain

$$\hat{\mu}_k \simeq \frac{1}{C_k} \sum_{i=1}^{N} c_{ik} x_i, \tag{24}$$

$$\hat{\Sigma}_k \simeq \frac{1}{C_k} \sum_{i=1}^{N} c_{ik} (x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^T, \tag{25}$$

by setting Eq. (17) and Eq. (19) to zero since we have $a_{ik} \simeq 1$ if $q \simeq 1$. These update rules are computationally efficient and give reasonable parameter values for the background model in Sec.2.2.