# Evaluating Gaussian Like Image Representations over Local Features

Yu-Chuan Su⋆ Guan-Long Wu† Tzu-Hsuan Chiu† Winston H. Hsu⋆† Kuo-Wei Chang‡
⋆ *Dept. of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan*
†*Graduate Institute of Networking and Multimedia, National Taiwan University, Taipei, Taiwan*
‡*Telecommunication Laboratories, Chunghwa Telecom Co., Ltd*
*spooky@cmlab.csie.ntu.edu.tw, {garywgl,r96098,winston}@csie.ntu.edu.tw, muslim@cht.com.tw*

***Abstract*—Recently, several gaussian like image representations are proposed as an alternative of the bag-of-word representation over local features. These representations are proposed to overcome the quantization error problem faced in bag-of-word representation. They are shown to be effective in different applications; the Extended Hierarchical Gaussianization reached excellent performance using single feature in VOC2009, Vector of Locally Aggregated Descriptors and Fisher Kernel reached excellent performance using only signature like representation on Holiday dataset. Despite their success and similarity, no comparative study about these representations has been made. In this paper, we perform a systematic comparison about three emerging different gaussian like representations: Extended Hierarchical Gaussianization, Fisher Kernel and Vector of Locally Aggregated Descriptors. We evaluate the performance and the influence of feature and parameters of these representations on Holiday and CC_Web_Video datasets, and several important properties about these representations have been observed during our investigation. This study provides better understanding about these gaussian like image representations that are believed to be promising in various applications.**

***Keywords*-Image Representation; Local Feature; Gaussian Mixture Model; Extended Hierarchical Gaussianization; Vector of Locally Aggregated Descriptors; Fisher Kernel; Performance Comparison**

## I. Introduction

In recent years, local feature, or as it is widely described as image keypoint, has emerged as the most powerful visual feature in both image classification and retrieval. Keypoints are local image patches that contain important visual information. These keypoints are represented by keypoint descriptors, where 128 dimensional SIFT descriptor [1] is generally used. These local features have proved to be discriminative, but since the number of keypoints in each image is usually different, they do not form a compact representation for images and videos and hence is not efficient enough to apply to large databases.

To form a compact representation for images, the “Bag of Words” (BoW) [2] method is widely used. In BoW method, keypoints are represented by a set of basis keypoints, or as they are generally called “visual vocabulary.” Each keypoint is assigned to a particular visual vocabulary according to their visual similarity. By doing so, keypoints become analogous to words in text domain, and images become the analogy of documents. Each image can now be represented by a vector containing the weighted visual word count, as documents in text domain information retrieval (IR). The BoW method has proved its efficacy and is adopted in many state of the art classification or retrieval systems.

A well known problem of BoW is that quantization error leads to information loss and limit the performance [3]. Some studies try to overcome the problem by adopting image representations that are conceptually different from BoW. While BoW method is an analogy of vector space model in IR, language model [4] is also analogized to image representation and introduces gaussian like image representations. As in language model, assume the keypoints of an image are generated by an unknown probability distribution, the problem of image representation becomes how to capture and represent the probability distribution. Several different image representations following this probability point of view has been introduced, including the Extended Hierarchical Gaussianizatoin (EHG) [5], Fisher Kernel (Fk) [6] and Vector of Locally Aggregated Descriptors (VLAD) [7]. These gaussian like representations exhibited excellent performance on various benchmarks. The classification accuracy using EHG with simple nearest centroid classifier is comparable to the state of art result using BoW and coplicate nonlinear classifier on Caltech256 dataset, and achieved top performance in the Visual Object Classes Challenge 2009 (VOC2009). Fk and VLAD show excellent performance on Holiday and UKB dataset, using only simple vector difference in retrieval.

Despite their success, no comparative study about these similar representations has been conducted. While previous works mainly focus on introducing new representations and the comparison with BoW, the similarity and difference of these representations have not been investigated. In this paper, we perform a systematic comparison on the three gaussian like image representations and investigate the influence of feature and structural parameters on the performance. The goal is to link these representations together, and provides further understanding about them. We try to figure out the best combination of feature, parameters and representation. We also provide a comparison on the time efficiency of representation formation.

The evaluation is carried out on two public benchmarks, including the CC_Web_Video [8] dataset for video retrieval and the Holiday [9] dataset for image retrieval. The experiments lead to two conclusions: (1) no singe gaussian like representation outperforms others in all situations; (2) VLAD is the most time efficient representation. We also observed several important properties about these representations.

In Section 2, we briefly review the existing work on image representation based on local feature. We describe the three different representation in Section 3. The experiment result is in Section 4, with more detail discussion in Section 5.

## II. Related Work

Bag-of-words representation is an early attempt to generate compact image representation [2] over local feature. Beside BoW, gaussian like representations serve as another promising direction, which do not suffer from quantization error. These representations describe the image on the basis of keypoints distribution. Fk is the pioneer of these representations [6], which describes images by the gradient of the likelihood of image keypoints. EHG [5] represents images directly by their keypoints distribution using gaussian mixture model (GMM). The VLAD [7] is a simplification of Fk.

## III. Gaussian Like Image Representations

Three gaussian like image representations are considered in our evaluation, the Extended Hierarchical Gaussianization, Fisher Kernel and Vector of Locally Aggregated Descriptors. In this section, we will briefly introduce the three representations and the relationship between them.

### A. Background Model

Similar to language model, a background model that describes the overall keypoint distribution of all images is necessary, as illustrated in Figure 1. In both Fk and EHG, a gaussian mixture model (GMM) is used to capture the background model, while it is simplified in VLAD. The background model is similar to the "visual vocabulary," where each gaussian component of the background GMM describes the distribution of a group of similar keypoints. There are three main reasons for its necessity.

- **Provide efficient image comparison method:** Although Kullback-Leibler divergence (KLD) may be used to measure the similarity between two arbitrary distribution, the evaluation of KLD requires a numerical integral over the entire feature space and is very inefficient. However, if image specific GMMs are adapted from the background GMM, under the assumption that image specific distributions do not deviate far away from the background distribution, efficient comparisons between two GMMs may be achieved by considering one pair of gaussian components at a time where the pair of components come from the same background component.

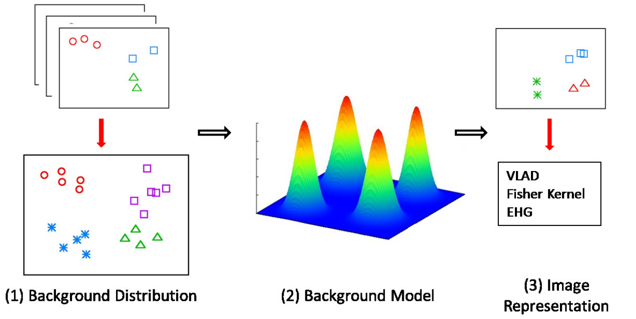

Figure 1. An overview of the general process for extracting gaussian like image representations.

- **Smoothing for unseen data:** As mentioned before, each gaussian component describes the distribution of a particular kind of keypoints. Since some kind of keypoints might be missing in a particular image, smoothing using background distribution is necessary.
- **Avoid over fitting:** The number of keypoints in one image may not be sufficient to learn a GMM, which requires the determination of hundreds or thousands of variables. Background model can constraint the image specific GMM and avoid overfitting.

The background model can be learned using EM algorithm under maximum likelihood criterion. And for simplicity, the covariance matrices of GMM are assumed to be diagonal in both Fk and EHG.

### B. Extended Hierarchical Gaussianization (EHG)

Given the background model, EHG [5] learns a image specific GMM under Maximum A Posteriori (MAP) criterion. The prior of image specific GMM is assumed to be

$$(w_{a,1}, \cdots, w_{a,K}) \sim Dir(Tw_{0,1}, \cdots, Tw_{0,K}) \quad (1)$$

$$\mu_{a,i} \sim N(\mu_{0,i}, \Sigma_{0,i}/r), k = 1 : K \quad (2)$$

given there are K gaussian components in GMM, where $N$ stands for normal distribution and $Dir$ for dirichlet distribution. $\mu_{a,k}$ is the mean of the $k$-th gaussian component from image $a$ (0 stands for background), $\Sigma_{a,k}$ is the covariance matrices, $w_{a,k}$ is the weight of the $k$-th gaussian component and $(r, T)$ are two empirical parameters. The MAP estimation can be obtained via EM algorithm.

After the image specific GMM is learned, a compact representation that allows efficient evaluation of the difference of GMMs is then extracted. In EHG, the difference between two GMMs is defined as the upper bound of the symmetrized KLD obtained using log-sum inequality. Under the assumption $\Sigma_{a,i} \approx \Sigma_{0,i}$, the upper bound can be approximated as

$$U_s(g_a, g_b) \approx \sum_{i=1}^{K} w_{0,i} \frac{1}{2} tr[(\Sigma_{a,i} - \Sigma_{b,i})\Sigma_{0,i}^{-2}(\Sigma_{a,i} - \Sigma_{b,i})] + \sum_{i=1}^{K} w_{0,i}(\mu_{a,i} - \mu_{b,i})^T)\Sigma_{0,i}^{-1}(\mu_{a,i} - \mu_{b,i})) \tag{3}$$

According to equation 3, the bound is divided into two parts, with one being the difference of means and the other being the difference of variance. A super-vector representation for the image specific GMM can be obtained as $\phi_a = [m_a; v_a]$ where

$$m_a = [w_{0,1}^{1/2}\Sigma_{0,1}^{-1/2}\mu_{a,1}; \cdots ; w_{0,M}^{1/2}\Sigma_{0,M}^{-1/2}\mu_{a,M}] \tag{4}$$

is the mean super-vector, denoted by EHG-m,

$$v_a = [\sqrt{\frac{1}{2}} w_{0,1}^{1/2}\Sigma_{0,1}^{-1}\Sigma_{a,1}; \cdots ; \sqrt{\frac{1}{2}} w_{0,M}^{1/2}\Sigma_{0,M}^{-1}\Sigma_{a,M}] \tag{5}$$

is the variance super-vector denoted by EHG-c. The $L_2$ distance of the super-vector representation will reduce to the approximated KLD upper bound, so the difference of two image can be evaluated efficiently, comparing to the KLD integral.

### C. Fisher Kernel (Fk)

In Fk [6], instead of extracting information from image specific GMM, the information about how to update the background GMM is used to represent the image. In practice, the information is obtained from the derivative with respect to means and variances of the log-likelihood function of the image. The detailed formulation is

$$\frac{\partial L(X|\lambda)}{\partial \mu_i^d} = \sum_{t=1}^{N} \gamma_t(i)[\frac{x_t^d - \mu_{0,i}^d}{(\sigma_{0,i}^d)^2}] \tag{6}$$

$$\frac{\partial L(X|\lambda)}{\partial \sigma_i^d} = \sum_{t=1}^{N} \gamma_t(i)[\frac{(x_t^d - \mu_{0,i}^d)^2}{(\sigma_{0,i}^d)^3} - \frac{1}{\sigma_{0,i}^d}] \tag{7}$$

given there are N keypoints in the image, where $\gamma_t(i)$ is the occupancy probability

$$\gamma_t(i) = p(i|x_t, \lambda) = \frac{w_{0,i} p_i(x_t|\lambda)}{\sum_{j=1}^{K} w_{0,j} p_j(x_t|\lambda)} \tag{8}$$

where $d$ denotes the $d$-th dimension of the SIFT descriptor, $x_t$ represents the keypoint descriptor of keypoint $k$, and $L(X|\lambda)$ is the log-likelihood function with $\lambda$ denoting the background GMM variables. Derivative with respect to weight $w_i$ is not included in our evaluation, because it is less informative than the means and variances in our early test.

| Method | mAP | Retrieval Time |
|---|---|---|
| CHSIG [8] | 0.891 | 1 sec |
| HIRACH [8] | 0.952 | 7 days |
| QIP [11] | 0.951 | 1.7 days |
| TNP [11] | 0.935 | 3.1 hours |
| VLAD | 0.958 | 49.5 sec |

Table I
COMPARING CC_WEB_VIDEO RESULT WITH EXISTING METHODS. THE VLAD WITH 64 CENTERS IS USED IN COMPARISON.

### D. Vector of Locally Aggregated Descriptors (VLAD)

VLAD is a simplification of Fk [7]. The evaluation of Eq.6 in Fk can be viewed as a weighted sum of the normalized difference of every keypoints with respect to each centers, where weight is given by the occupancy probability $\gamma_t(i)$ and normalization is done by variance $\sigma_{0,i}$. VLAD simplifies Fk mean vector as followed:

- **Ignore normalization**. $\sigma_{0,i}$ is set to $1$ reqardless of $i$.
- **Binary weight**. $\gamma_t(i)$ becomes a binary function of $t$ and $i$, where $\gamma_t(i)$ equals to $1$ iff center $i$ is the nearest center to keypoint $t$ in feature space and equals to $0$ otherwise.

VLAD therefore does not require the estimation of background GMM, but only need the cluster centers of keypoints, as if in BoW method. Also, the Fk variance vector is ignored in VLAD.

## IV. EXPERIMENT RESULTS

In this section, we present the result of the three different representations on different datasets. For Fk and EHG, the performance of using only mean vector (-m) and variance vector (-c) are also included. The two local features we used include the Hessian affine-invariant (HA) region detector and dense sampling, both described by SIFT descriptor. The original 128-dimension SIFT descriptor is used, without performing any dimension reduction. For dense sampling, the VLFeat package [10] is used for feature extraction.

In all datasets, the performance of using 8, 16 and 64 gaussian components (or cluster centers in VLAD) is evaluated. The dimension of VLAD is 128 times the number of centers. Because EHG and Fk contain both means and variances in the representations, the dimension is twice that of VLAD. The dimension of representations involved in the experements thus range from 1024 to 16384.

### A. Holiday Dataset

The INRIA Holiday dataset [9] is a collection of 1491 holiday images, with 500 of them are used as queries. The performance is measured by the mean Average Precision (mAP). The image resolution in the dataset is higher than most existing datasets and daily used images, so except evaluating the performance using original images, we also evaluate the performance on scale-down images. Overall, there are four different settings for local feature extraction:

- Original image + Hessian Affine-invariant (HA)
- Original image + Dense Sampling (Dense)
- Scale down image (400×300) +
  Hessian Affine-invariant (HA_Scale)
- Scale down image (320×240)
  + Dense Sampling (Dense_Scale)

The size of scale down images is different for HA_Scale and Dense_Scale so the average keypoints number is the same in the two settings. For dense sampling, 20×20 patches with overlapping windows shifted by 5 pixels is used. Notice for dense sampling using original images, many image patches contain only small variation and are not informative. And for Hessian affine-invariant with scale down images, the number of keypoints in each image has a large variation, where some images have only less than ten keypoints.

Table II shows the result of our experiment. For comparison, the state of the art result using BoW representation [12] is listed in Table III, in which Hessian affine-invariant feature is used. We can see that VLAD and Fk outperform BoW when the dimensions are at the same order: both of them reach mAP higher than 0.5 using only 2048 dimension (HA, k=16), where the mAP of BoW is only 0.414 at 1k dimension and 0.446 at 20000 dimension. While VLAD reaches the mAP of 0.541 using 8192 dimension, BoW needs 200000 dimension, which is 24 times higher than VLAD, to reach the same performance.

The best representation for each setting are not the same. For HA and Dense_Scale, VLAD has the best performance using 64 centers. For HA_Scale and Dense, Fk has the best performance, with 16 and 64 gaussian components respectively. EHG does not perform as well as VLAD and Fk in most settings, but performs pretty well in Dense_Scale, which is nearly as good as the best result of the setting.

When using only mean vector (Fk-m) or variance vector (Fk-c) of Fk, the performance of Fk-m is usually better and more stable. As the number of gaussian component increases, Fk-c becomes more discriminative, while Fk-m has no consistent trend in different settings. For EHG, the performance of mean vector and variance vector is more similar, comparing with that of Fk.

### B. CC_Web_Video Dataset

CC_Web_Video dataset [8] is a collection of 12790 web videos from YouTube, Google and Yahoo, which is collected using 24 distinct search queries. For each query, one seed video is used as an example for near-duplicate video retrieval. The mAP of the 24 queries is used as the performance measurement, with the retrieval time being another important criterion.

In our evaluation, we form a single gaussian like representation for each video as if it is an image. We first extract local features from each keyframe. The keypoints from all the keyframes of the video are then aggregated together, as if they were from the same image. The video representation is then computed like that of images. By doing so, we represent the video using its keyframes, where these keyframes can be imagined as being combined together into a large image. For dense sampling, 24×24 patches with overlapping window shifted by 6 pixels is used to keep the average number of keypoints of each video comparable with that of Hessian affine-invariant.

| method | k | mAP |
|---|---|---|
| BoW | 1,000 | 0.414 |
| BoW | 20,000 | 0.446 |
| BoW | 200,000 | 0.549 |
| binary BoW | 20,000 | 0.458 |
| binary BoW | 200,000 | 0.554 |

Table III
Result for the state of the art signature based method on Holiday dataset [12]. k is the size of visual vocabularies.

| | HA | | | Dense | | |
|---|---|---|---|---|---|---|
| | k=8 | k=16 | k=64 | k=8 | k=16 | k=64 |
| VLAD | 0.932 | 0.937 | **0.948** | 0.927 | 0.951 | **0.958** |
| Fk | 0.898 | 0.898 | 0.897 | 0.886 | 0.896 | 0.903 |
| Fk-m | 0.894 | 0.895 | 0.892 | 0.897 | 0.903 | 0.910 |
| Fk-c | 0.825 | 0.843 | 0.869 | 0.830 | 0.851 | 0.868 |
| EHG | 0.906 | 0.891 | 0.860 | 0.947 | 0.944 | 0.903 |
| EHG-m | 0.918 | 0.908 | 0.869 | 0.942 | **0.952** | 0.933 |
| EHG-c | 0.898 | 0.883 | 0.856 | 0.946 | 0.938 | 0.889 |

Table IV
mAP of CC_WEB_Video dataset. k is the number of gaussian components (or centers) in feature representation.

Table IV shows the result of our experiment, and the result of other state of the art methods are listed in Table I. Both VLAD and EHG can achieve comparable performance with HIRACH under suitable setting, while Fk usually has inferior performance. For both Hessian affine-invariant and dense sampling feature, VLAD has the best performance, while the mAP of EHG-m is comparable with VLAD when dense sampling is used.

Since a video wise compact vector representation is used, we only have to evaluate the $L_2$ distance between different videos during retrieval. For methods listed in Table I, except the simple color histogram (CHSIG), all other methods require complicated computation during retrieval and have longer retrieval time. The VLAD outperforms other methods in both mAP and retrieval time.

### C. Computation Time Analysis

In this section, we examine the computation time for representation formation given local features and background models. The time for background model learning is not considered because it can be evaluated off-line. We focus on two factors:

| | HA | | | HA_Scale | | | Dense | | | Dense_Scale | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | k=9 | k=16 | k=64 | k=8 | k=16 | k=64 | k=8 | k=16 | k=64 | k=8 | k=16 | k=64 |
| VLAD | 0.516 | 0.502 | **0.541** | 0.411 | 0.398 | 0.285 | 0.395 | 0.460 | 0.490 | 0.416 | 0.484 | **0.510** |
| Fk | 0.482 | 0.520 | 0.535 | 0.446 | **0.480** | 0.474 | 0.451 | 0.482 | **0.513** | 0.379 | 0.423 | 0.488 |
| Fk-m | 0.507 | 0.524 | 0.471 | 0.359 | 0.320 | 0.096 | 0.436 | 0.463 | 0.491 | 0.402 | 0.440 | 0.497 |
| Fk-c | 0.260 | 0.332 | 0.424 | 0.164 | 0.263 | 0.350 | 0.296 | 0.371 | 0.428 | 0.154 | 0.247 | 0.369 |
| EHG | 0.457 | 0.443 | 0.413 | 0.309 | 0.220 | 0.094 | 0.330 | 0.387 | 0.441 | 0.424 | 0.462 | 0.467 |
| EHG-m | 0.468 | 0.456 | 0.458 | 0.341 | 0.321 | 0.179 | 0.297 | 0.364 | 0.428 | 0.382 | 0.414 | **0.505** |
| EHG-c | 0.444 | 0.426 | 0.369 | 0.264 | 0.166 | 0.067 | 0.349 | 0.400 | 0.430 | 0.441 | 0.466 | 0.427 |

Table II

MAP OF HOLIDAY DATASET. K IS THE NUMBER OF GAUSSIAN COMPONENTS (OR CENTERS) IN FEATURE REPRESENTATION.

- The number of gaussian components
- The number of keypoints in each image

We perform the experiments on the Holiday dataset using 50 randomly chosen images. Dense sampling is used, with 20×20 patches and 5 pixels window shift. The experiments are run on a single machine with Intel Xeon Processor 5140 and 8GB of memory.

To demonstrate the effect of gaussian component number, all images are scale down to 320×240. The average computation time for the 50 images is reported, with the number of gaussian components being 8, 16, 64, 128 and 256 respectively. The result is in Figure 2. The computation time of Fk is the longest, followed by EHG. VLAD is the most time efficient representation. Also notice the computation time for Fk and EHG grows linearly with respect to the number of gaussian components, while that of VLAD remains about the same regardless of the number of centers. This nice property of VLAD is due to the tree structure used for finding the nearest center of each keypoint.

We also examine the computational time increase with respect to keypoint numbers. Because the number of keypoints in dense sampling is proportional to the image size, we compute the representations of the same 50 images with different size ranging from 320×240 to 2560×1920. The result is shown in Figure 3. We can see the computation time of all three representations increase linearly with respect to keypoints number, but with different slope. VLAD has the smallest slope, meaning it is more efficient for large images or long videos.

## V. DISCUSSIONS

Several interesting properties of the three representations have been observed during our experiments, and will be discussed in this section.

### A. Parameters of EHG

There are three empirical parameters when learning the image specific GMM in EHG. In previous work [5], the effect of the parameters was not discussed. In our experiments, we notice an unneclectable dependency between the performance and parameters. More precisely, although $T$, which determines the prior of $w_{a,i}$, has no significant effect on the retrieval result, the parameter that determines the variance of mean distribution, $r$, is important for the performance.

According to our experiments, the performance of EHG is poor when we fixed $r$ to a small value such as 1, but improves as $r$ becomes larger, which implies the means of image specific GMM is close to the background means. In practice, we set $r$ to a constant times of the number of keypoints the image has, where the constant is adjusted in a small range from 1 to 10. Because normal images usually have more than a hundred of keypoints, the prior distributions of means are actually very narrow.

### B. MAP Process of EHG

The EM algorithm used to obtained the MAP estimation may takes several iterations before convergence. However, it is not necessary to wait until the EM algorithm converges before we extract information from the image specific GMM. Actually, the performance of EHG is no better or even worse when more than 1 iteration of EM is allowed. So in practice, we perform only 1 iteration of EM algorithm when learning the image specific GMM. This further limit the deviation of image specific GMM from the background GMM.

In conclusion, the performance of EHG is dependent on the distance between image specific GMM and background GMM. Because EHG uses the approximated upper bound of KLD to evaluate the difference between two images for the sake of efficiency, the performance is dependent on how good the approximation is. When the image specific GMM deviates far from the background GMM, the approximation is poor and degrades the performance.

### C. EHG with Dense Sampling

While it is widely accepted that dense sampling usually yields better performance, we notice that dense sampling is especially suitable for EHG. For other representations like VLAD, the difference between Hessian affine-invariant and dense sampling is smaller than that of EHG, and Hessian affine-invariant sometimes even outperforms dense sampling. For EHG, the performance of dense sampling is always better, signifying that EHG works better with dense sampling.

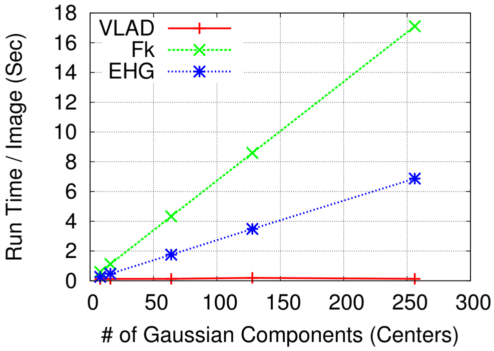

Figure 2. Average representation calulation time with respect to the number of gaussian components.

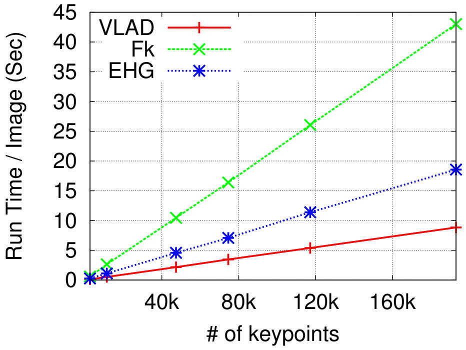

Figure 3. Average representation calulation time with respect to the number of keypoints.

### D. *Differences of Fk and VLAD*

Although VLAD is a kind of simplification of Fk, there is a significant difference between the two representations. The difference comes from the influence of each keypoint on the final representation. In Fk, given a gaussian component, the effect of each keypoint is weighted by a gaussian function. Therefore, keypoints that are close to gaussian center in feature space will be emphasized and are more important for the representation. For VLAD, the weighting function will ignore keypoints closer to other centers, while it gives the same weight for all keypoints that belong to the same center. As a consequence, the representation will be dominated by keypoints that are near the border of clusters.

This difference makes Fk more robust to noisy data, where the weighting may unweighed the noise. This can be seen in the result of Holiday dataset, where Fk outperforms VLAD in the two more noisy settings.

## VI. Conclusion

Several gaussian like image representations have been proposed, and are proved to be effective in various benchmarks. In this paper, we perform a systematic comparison between them. The study provides an illustration about the similarities and differences of these representations; and several observations have been discussed. The performance of these image representations depends on the feature and image quality, as well as the number of centers used. In our experiment, no single representation seems to significantly outperforms others in all situations, when only the performance is considered. But when the time for representation formation is also considered, VLAD turns out to be a more efficient representations, and is therefore more suitable for large images or long videos.

## References

[1] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *IJCV*, 2004.

[2] J. Sivic and A. Zisserman, “Video google: a text retrieval approach to object matching in videos,” in *ICCV*, 2003.

[3] J. Yang, K. Yu, Y. Gong, and T. Huang, “Linear spatial pyramid matching using sparse coding for image classification,” in *CVPR*, 2009.

[4] J. M. Ponte and W. B. Croft, “A language modeling approach to information retrieval,” in *SIGIR*, 1998.

[5] M. Xu, X. Zhou, Z. Li, B. Dai, and T. S. Huang, “Extended hierarchical gaussianization for scene classification,” in *ICIP*, 2010.

[6] F. Perronnin and C. R. Dance, “Fisher kernels on visual vocabularies for image categorization,” in *CVPR*, 2007.

[7] H. Jégou, M. Douze, C. Schmid, and P. Pérez, “Aggregating local descriptors into a compact image representation,” in *CVPR*, 2010.

[8] X. Wu, A. G. Hauptmann, and C.-W. Ngo, “Practical elimination of near-duplicates from web video search,” in *ACM MM*, 2007.

[9] H. Jégou, M. Douze, and C. Schmid, “Hamming embedding and weak geometric consistency for large scale image search,” in *ECCV*, 2008.

[10] A. Vedaldi and B. Fulkerson, “VLFeat: An open and portable library of computer vision algorithms,” http://www.vlfeat.org/, 2008.

[11] H.-K. Tan, C.-W. Ngo, R. Hong, and T.-S. Chua, “Scalable detection of partial near-duplicate videos by visual-temporal consistency,” in *ACM MM*, 2009.

[12] H. Jégou, M. Douze, and C. Schmid, “Packing bag-of-features,” in *ICCV*, 2009.