TRANSMITTING INFORMATIVE COMPONENTS OF FISHER CODES FOR MOBILE VISUAL SEARCH

Guixuan Zhang, Zhi Zeng, Shuwu Zhang, Qinzhen Guo

Institute of Automation, Chinese Academy of Sciences, Beijing, China
{guixuan.zhang, zhi.zeng, shuwu.zhang, qinzhen.guo}@ia.ac.cn

ABSTRACT

Existing techniques usually adopt compact descriptors such as Fisher vector for mobile visual search, since compact descriptors are memory-efficient and suitable for fast transmission. In common Fisher vector methods, in order to make the size of image representations small enough for efficient transmission, only a small number of visual words are used. However, this choice usually sacrifices the search accuracy. In this paper, a Soft-Assignment Adjusting approach is proposed to just select informative components of descriptors for query. With this method, we can adopt more visual words to improve accuracy, while the memory usage is still low. Furthermore, efficient bitrate scalable codes are proposed in order to accommodate the network bandwidth variation. Experiments performed on benchmark datasets show that our proposed approach outperforms the state-of-the-art methods for mobile visual search.

Index Terms— soft-assignment adjusting, bitrate scalable codes, informative components, mobile visual search, Fisher vector

1. INTRODUCTION

The problem of retrieving images of a given object from large datasets has attracted increasing attentions. In the past few years, the searching process has been operated on mobile devices more frequently. The user snaps a photo of an object and searches the information about it over the network. We call this mobile visual search (MVS). People usually search objects such as book/media cover, products, landmarks, artwork, and video frames. Recent commercial MVS platforms include Google Goggles and Amazon Flow.

A MVS system usually transmits query data from a mobile client to a remote server. Since the network bandwidth varies in different environments, the transmission latency becomes an uncertain factor. What we can do is to try to reduce the size of query data, so that the latency time may decrease. Compact descriptors are widely used because of their low memory usage [1-3]. Raw Bag-of-Features (BOF) [4-5] results in a compact representation, but its retrieval precision is low. Many approaches such as hamming embedding [6] are applied to improve BOF. Though the performance is good, more memory is required for the extra information, which is not suitable for MVS systems [1, 3]. Recently, compact descriptors such as Fisher Vector (FV) [7] and VLAD [8] are proposed for image retrieval. These compact descriptors are discriminative and the memory usage is low.

Chen et al. propose Residual Enhanced Visual Vector (REVV) [2] for their MVS system, which is similar to VLAD. The size of REVV is fixed and not adaptive to the variable network bandwidth. Lin et al. introduce Rate-adaptive Compact Fisher Codes (RCFC) [3] based on FV to produce a rate-adaptive image signature. Both REVV and RCFC adopt a small number of visual words, which leads to memory-efficient codes. However, since the query photos for MVS are usually with clutter and occlusions, the retrieval accuracy of REVV and RCFC, which are generated by a small vocabulary, is not as good as we expect. In order to improve the performance, adopting more visual words is a good choice. However, this will lead to large memory usage, which increases the transmission latency time and impacts the user experience.

In this paper, we address the problem of accuracy and fast transmission for MVS. We present an approach complementary to Fisher vector. Our first contribution is to propose the Soft-Assignment Adjusting (SAA) method to discard less-informative components to save the memory usage, while the performance does not decline. So adopting more visual words to improve accuracy can be allowed by employing SAA. The second contribution is that a more effective bitrate scalable approach different from RCFC is proposed in order to accommodate the bandwidth variation.

The rest of the paper is organized as follows. Section 2 briefly reviews the original Fisher vector. Our Soft-Assignment Adjusting method is proposed in Section 3. The approach for bitrate scalable codes is proposed in Section 4. Section 5 presents the experimental results of our approach. The final conclusions are given in Section 6.

2. REVIEW OF FISHER VECTOR

Let $X = \{X_t, t = 1, \ldots, T\}$ be a set of d-dimensional samples whose generation process can be modeled by an independent probability density function u_λ with parameters λ . Here X corresponds to local features

extracted from an image. The score function is given by the gradient of the log-likelihood on the model:

$$G_\lambda^X = \frac{1}{T}\nabla_\lambda \log u_\lambda(X). \quad (1)$$

This gradient describes the contribution of the parameters to the generation process. A natural kernel on these gradients is $K(X,Y) = G_\lambda^{X'} F_\lambda^{-1} G_\lambda^Y$, where $F_\lambda = E_{x\sim u_\lambda}[G_\lambda^X G_\lambda^{X'}]$ is the Fisher information matrix of u_λ.

As F_λ is symmetric and positive definite, it has a Cholesky decomposition $F_\lambda = L_\lambda' L_\lambda$. Then Fisher Kernel $K(X,Y)$ can be rewritten as a dot-product between normalized vectors $g_\lambda^X = L_\lambda G_\lambda^X$. We will refer to g_λ^X as the Fisher vector of X.

Perronnin et al. [7] choose u_λ to be a Gaussian Mixture Model (GMM) with N centroids: $u_\lambda(x) = \sum_{i=1}^N \omega_i u_i(x)$ and $\lambda = \{\omega_i, \mu_i, \sigma_i, i = 1,\ldots,N\}$ where ω_i, μ_i and σ_i are respectively the weight, mean vector and variance matrix of Gaussian u_i. The GMM u_λ is trained on a large number of images by using Maximum Likelihood Estimation. The diagonal closed-form approximation of the Fisher information matrix is used, so that the normalization of the gradient by $L_\lambda = F_\lambda^{-1/2}$ is simply a whitening of the dimensions. Let $\gamma_t(i)$ be the soft assignment of the local descriptor x_t to Gaussian i:

$$\gamma_t(i) = \frac{\omega_i u_i(x_t)}{\sum_{j=1}^N \omega_j u_j(x_t)}. \quad (2)$$

Let g_i^X be the gradient with respect to the mean μ_i of Gaussian. We have g_i^X after standard mathematical derivations:

$$g_i^X = \frac{1}{T\sqrt{\omega_i}} \sum_{t=1}^T \gamma_t(i)\left(\frac{x_t - \mu_i}{\sigma_i}\right). \quad (3)$$

The final gradient vector g_λ^X is formed by concatenating the g_i^X vectors for $i = 1,\ldots,N$ and is Nd-dimensional.

N is typically ranging from 16 to 256 [8]. However, such a small vocabulary usually fails to give good results, since there is much clutter in the query images for MVS. For example, nearby buildings usually appear together in the photo when recognizing a landmark. In order to improve the accuracy, more visual words are necessary. We performed experiments on the Stanford Media Cover Dataset [1] and found that the precision at rank 1 increased from 0.6 to 0.93 when the number of visual words changed from 128 to 2048. However, the dimension of an image representation is up to 262144 when using raw SIFT descriptor [14]. If we use floating points, the memory usage per image is up to 1MB, which is really large for MVS [3].

3. SOFT-ASSIGNMENT ADJUSTING

The Fisher vector of an image is Nd-dimensional. This dense representation relies on that every Gaussian visual word has been visited by at least one local feature. If none of local features are assigned to word i, the corresponding Fisher sub-vector g_i^X would be null. These sub-vectors are treated as less-informative components. In order to achieve memory-efficient usage, we discard the less-informative sub-vector g_i^X. For purpose of marking which visual words are retained, we define $b_i = 0$ if g_i^X is less-informative and $b_i = 1$ otherwise. We call the 0/1 vector b as Flag Vector.

One of the advantages of FV is its soft assignment. In the Fisher framework, the probability of feature x_t assigned to word i is $\gamma_t(i)$. For word i, the lower $\gamma_t(i)$ is, the fewer contributions will be made to g_i^X by local feature x_t. Then we present an idea that discarding features with low $\gamma_t(i)$, i.e., we just select components with high $\gamma_t(i)$ and the value of low $\gamma_t(i)$ is set to zero. This will bring several benefits. One is that faster computing for g_i^X is allowed since some terms are abandoned. There are many centroids which are only visited by local features with low $\gamma_t(i)$, the corresponding sub-vectors are really less informative. The second benefit of this procedure is that the sub-vectors g^X of these centroids will become zero. Only informative centroids are retained after discarding. We concatenate the non-null sub-vectors g^X and the final dimension is Md if the number of informative centroids is M.

Since low $\gamma_t(i)$ has been set to zero, the sum $\sum_{i=1}^N \gamma_t(i)$ for local feature x_t would be less than 1. In order to keep the sum equal to 1, we add the difference between $\sum_{i=1}^N \gamma_t(i)$ and 1 to the value $\gamma_t^* = \max(\gamma_t(i))$.

We call the method mentioned above as Soft-Assignment Adjusting. The procedure of Soft-Assignment Adjusting is provided in Algorithm 1.

4. BITRATE SCALABLE CODES

Since the query data is transmitted over network, the network bandwidth and representation size will affect the response latency. In order to address the issue of scalability, MPEG CDVS (Compact Description for Visual Search) has set several operating points [10]. The query data size R_v ranges from 0.5kB to 16kB. The image signature should be scalable to these different bitrates. For instance, in a 2G network environment, the max size of one signature is 2kB. The size can be moderately increased to 8kB for more discriminative power if the scene becomes a WiFi or 4G environment. In order to achieve such a small size, we should compress the Fisher vector into binary codes first. Here we use an element-wise binarization function $f(x) = +1$ if $x \geq 0$, and 0 otherwise [15], which is also employed by REVV and RCFC.

Let s_i^X be the binary version of g_i^X. The similarity between two images is defined as follows:

$$S_{X,Y} = \frac{\sum_{i=1}^N b_i^X b_i^Y (d - 2H(s_i^X, s_i^Y))}{\sqrt{d\sum_{i=1}^N b_i^X}\sqrt{d\sum_{i=1}^N b_i^Y}}, \quad (4)$$

Algorithm 1 Compute Fisher Vector with SAA

Input: Local image descriptor $X = \{x_t \in \mathbb{R}^d, t = 1,\ldots,T\}$

Output: The number of informative sub-vectors M

The Flag Vector $b = (b_1,\ldots,b_N)$

Improved Fisher vector representation $g_\lambda^X \in \mathbb{R}^{Md}$

1. For $t = 1,\ldots,T$
2. Initialize the accumulator $da = 0$
3. For $i = 1,\ldots,N$
4. Compute $\gamma_t(i)$
5. If $\gamma_t(i) < threshold$
6. $da = da + \gamma_t(i)$, $\gamma_t(i) = 0$
7. $\gamma_t(\text{argmax}_i\, \gamma_t(i)) = \max \gamma_t(i) + da$
8. For $i = 1,\ldots,N$
9. $g_i^X = 1/(T\sqrt{\omega_i})\sum_{t=1}^{T}\gamma_t(i)(x_t - \mu_i)/\sigma_i$
10. Number the informative sub-vectors from 1 to M
11. Generate the Flag Vector b
12. Concatenate non-zero components $g_\lambda^X = (g_1^X,\ldots,g_M^X)$
13. Return g_λ^X, M, and b

where $H(.,.)$ is hamming distance between s_i^X and s_i^Y.

Though compressing the image representations into codes has saved memory, some signatures may be still larger than the setting size. Our rate scalable approach is based on selecting the most informative Fisher sub-vectors to form the final signature. Since each sub-vector g_i^X is formed by many local features with different $\gamma_t(i)$, we define the max assignment probability Mp to Gaussian word i as follows:

$$Mp_i = \max(\gamma_t(i)). \quad (5)$$

A sorting algorithm to the set $\{Mp_i, i = 1,\ldots,M\}$ is applied. The sub-vector g^X with the largest Mp is first selected, then g^X with the second largest Mp is selected, and so on. Each binary version of g^X needs d bits to represent it, where d is the dimension of local feature. The first m selected sub-vectors will occupy md bits. When md increases to the setting rate R_v, the selection process stops. This method is inspired by VLAD, which uses hard-assignment instead of soft-assignment.

5. EXPERIMENTS

We perform the experiments over the MPEG CDVS datasets [10]. The datasets have five categories of images: (1) Text and graphics dataset, including CD/DVD/book cover, text document and business card; (2) Museum painting dataset; (3) Video frame dataset; (4) Landmark dataset, including the Zurich buildings [11], the PKU dataset [12], etc; (5) The UKB dataset [5]. To evaluate large scale image search, we introduce distractor images downloaded from Flickr. Because most groups of images have only one database image, we use precision at rank 1 (P@1) for evaluation. For UKB, we use the average number N_S of relevant images at top 4 ($4\times R@4$) for evaluation.

Table 1. Testing different thresholds on Stanford Mobile Visual Search Dataset. P@1 means precision at top 1. NISV means the number of informative sub-vectors.

Threshold	0.01	0.05	0.1	0.2	0.3	0.5
P@1(1024)	0.73	0.73	0.73	0.70	0.69	0.64
NISV(1024)	419	389	307	245	218	184
P@1(2048)	0.79	0.80	0.80	0.77	0.75	0.72
NISV(2048)	672	571	415	352	309	248
P@1(4096)	0.85	0.84	0.84	0.80	0.78	0.75
NISV(4096)	1312	1020	705	688	569	502

Feature descriptors are obtained by Hessian-blob detector [13] and SIFT descriptor. Feature dimension is reduced from 128 to 32 by PCA.

5.1. Parameter Analysis

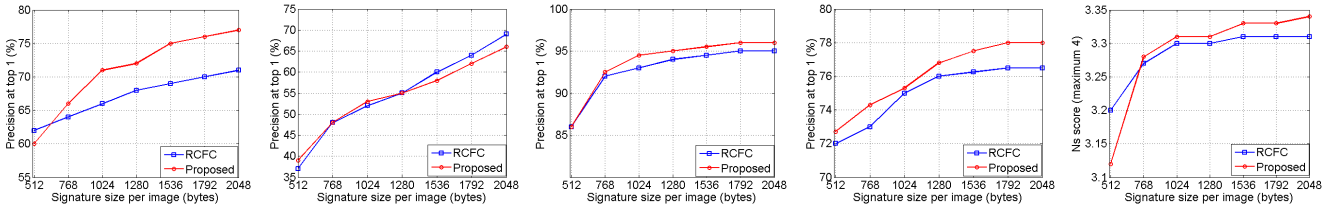
We test different discarding thresholds for SAA on the Stanford Mobile Visual Search Dataset [9]. We employ raw SIFT descriptor. 1024, 2048 and 4096 visual words are used respectively. From Tab.1 we can find that 0.1 is a good choice. 0.1 yields the competitive performance with 0.05 and 0.01, while the memory cost of 0.1 is less than the others'. When 2048 visual words and threshold 0.1 are used, there are only 415 informative sub-vectors left on average. The final dimension of the representation of an image is 53120, which is 5 times smaller than 262144 that we mentioned in Section 2.

5.2. Evaluation of Soft-Assignment Adjusting

In order to evaluate our SAA method, we make a comparison between the original Fisher vector and the Fisher vector incorporated with SAA. Tab. 2 shows that the vocabulary of 2048 visual words is significantly better than the 256 vocabulary. A larger vocabulary is necessary to boost the accuracy. Applying our SAA method to FV successfully preserves the good performance, while the memory usage is 5 times smaller than the original FV when 2048 visual words are used. Interestingly, SAA slightly improves the accuracy for the landmark dataset. It is because that SAA removes some components which are generated by the background clutter. So our SAA method makes FV more effective and memory-efficient for MVS.

5.3. The Evaluation of Fixed-Length Signature

We compare our proposed Fisher codes with REVV at a fixed signature size. A comparison with BOF is also included. REVV derives from VLAD, which builds a compact representation based on word residuals with hard-assignment. Both REVV and RCFC employ the same binarization function as ours. We set the fixed signature size 1kB and 2kB respectively. REVV should adopt 256 visual



(a)Graphics (b)Museum Paintings (c) Video Frames (d)Landmarks (e) UKB

Fig. 1. Evaluation of different signature sizes. Flickr 1M is used as the distractor dataset.

Table 2. Evaluation of SAA. The image categories include Text and graphics, Museum paintings, Video frames, Landmarks and UKB. Flickr 10k is used as the distractor database.

Category	Graph	Paint	Video	LanM	UKB
FV(256)	0.761	0.683	0.920	0.738	3.46
FV(2048)	0.858	0.805	0.981	0.839	3.50
FV(2048)+SAA	0.857	0.771	0.978	0.842	3.49

Table 3. Comparison of our proposed method, REVV and BOF. Flickr 1M distractor images are included.

Category	Fixed Size (1kB)			Fixed Size (2kB)		
	BOF	REVV	Ours	BOF	REVV	Ours
Graph	0.32	0.61	**0.69**	0.41	0.67	**0.78**
Paint	0.18	0.50	**0.60**	0.22	0.59	**0.66**
Video	0.46	0.86	**0.92**	0.51	0.89	**0.96**
LanM	0.30	0.68	**0.72**	0.37	0.72	**0.78**
UKB	2.02	3.09	**3.20**	2.25	3.18	**3.34**

words to generate the signature at the size of 1kB and 512 words at the size of 2kB. By employing SAA, we can adopt 1024 and 2048 visual words respectively. Our bitrate scalable method may be used if necessary. For example, when the size is set to 2kB and 2048 words are adopted, the Flag Vector b needs 128 bytes. Then 448 Fisher sub-vectors at most can be transmitted. If one image has more than 448 informative sub-vectors, we apply our bitrate method by selecting the most informative components with the largest Mp. Tab. 3 shows that our proposed method outperforms BOF and REVV over the five different datasets. The good performance of our method benefits from the soft assignment of Fisher framework and adopting more visual words to improve the accuracy.

5.4. The Evaluation of Rate Scalable Signature

To evaluate the performance of the rate scalable signatures, we set the size of image signatures varying from 512 bytes to 2048 bytes. We compare our proposed method with RCFC, which makes use of the original FV. For the RCFC, 512 visual words are used so that their image signature size ranges from 512 bytes to 2048 bytes adaptively. Our rate scalable approach adopts 2048 words. In order to generate a signature at the smallest size of 512 bytes, we have to select only 80 informative sub-vectors with the largest Mp .

Fig. 1 shows the results. In most cases, our proposed bitrate scalable codes outperform RCFC. The first reason is that we adopt more visual words to improve the accuracy, while the memory usage does not increase. The second reason is that our approach selects the most informative components for query, which makes the image signatures more discriminative. For the Paintings dataset, RCFC performs a little better than ours, this is because only a few useful features can be extracted from each image. Our approach discards some components of these useful local features, which may decrease the accuracy. Though there is a little accuracy loss, our bitrate scalable codes are the better choice in most cases.

5.5. System Latency

The feature extracting and aggregating time depends on the type of clients. The average client-processing time is 0.5s. The transmission delay for a 2kB signature over 2G and WLAN network is about 0.2s and 0.01s respectively. The searching time based on hamming matching for 1M database on a server with 3.40 GHz CPU is 1s on average.

6. CONCLUSION

This paper addresses the problem of accuracy and efficient transmission for MVS. Our SAA and bitrate scalable codes achieve memory-efficient and perform well in many scenarios. The proposed image signature accommodates the changes of network bandwidth in different environments. Though 4G is becoming common, it does not guarantee that high bandwidth is available everywhere. Memory-efficient signature is still necessary for providing a good user experience. The tradeoff between accuracy and efficient transmission is always an important issue for mobile visual search.

7. ACKNOWLEDGEMENTS

The work has been supported by the International S&T Cooperation Program of China under Grant No.2013DFG12980 and the National Key Technology R&D Program of China under Grant No.2012BAH75F03.

8. REFERENCES

[1] B. Girod, V. Chandrasekhar, and D. Chen, "Mobile visual search," *IEEE Signal Process. Mag.*, 2011.

[2] D. Chen, S. Tsai, V. Chandrasekhar, G. Takacs, R. Vedantham, R. Grezeszczuk, and B. Girod, "Residual enhanced visual vector as a compact signature for mobile visual search" *Signal Processing*, 2012.

[3] J. Lin, L.Y. Duan, Y. Huang, S. Luo, T, Huang, and W. Gao, "Rate-adaptive compact fisher codes for mobile visual search," *Signal Processing Letters, IEEE*, vol. 21, no. 2, pp. 195-198, Feb. 2014.

[4] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *ICCV*, pp. 1470–1477, October 2003.

[5] D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," in *CVPR*, pp. 2161–2168, June 2006.

[6] H. Jegou, M. Douze, and C. Schmid, "Improving bag-of-features for large scale image search," *International Journal of Computer Vision,* vol. 87, pp. 316–336, February 2010.

[7] F. Perronnin and C. Dance, "Fisher kernels on visual vocabularies for image categorization," in *CVPR*, June 2007.

[8] H. Jegou, F. Perronnin, M. Douze, J. Sanchez, P. Perez, and C. Schmid, "Aggregating local image descriptors into compact codes," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 34, no. 9, pp. 1704-1716, Sept. 2012.

[9] V. Chandrasekhar, D. Chen, S. Tsai, N.M. Cheung, H. Chen, G. Takacs, Y. Reznik, R. Vedantham, R. Grzeszczuk, J. Bach, and B. Girod. "The stanford mobile visual search data set," in *Proceedings of the second annual ACM conference on Multimedia systems*, pp. 117-122, 2011.

[10] ISO/IEC JTC1/SC29/WG11/N12202, Evaluation Framework for Compact Descriptors for Visual Search 2011.

[11] H. Shao, T. Svoboda, and L.V. Gool. "Zubud - zurich buildings database for image based recognition," *Technical Report 260*, Computer Vision Laboratory, Swiss Federal Institute of Technology, 2003.

[12] R. Ji, L.Y. Duan, J. Chen, S. Yang, T. Huang, H. Yao, and W. Gao, "PKUBENCH: A context rich mobile visual search benchmark," in *Image Processing (ICIP), IEEE International Conference on*, pp. 2545-2548, Sept. 2011.

[13] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Comput. Vis. Image Understand.*, vol. 110, no. 3, pp. 346–359, 2008.

[14] D. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis*., vol. 60, no. 2, pp. 91–110, 2004.

[15] F. Perronnin, Y. Liu, J. Sanchez, and H. Poirier, "Large-scale image retrieval with compressed fisher vectors," in *CVPR*, pp. 3384-3391, June 2010.