INTERACTIVE ON-DEVICE MOBILE LANDMARK RECOGNITION WITH COMPACT BINARY CODES

Tao Guan[1], Liujuan Cao[2], Ling Cai[2], and Rongrong Ji[2]
qd_gt@126.com, {caoliujuan, lcai, rrji}@xmu.edu.cn
[1]College of Computer Science, Huazhong University of Science and Technology
[2]School of Information Science and Engineering, Xiamen University

ABSTRACT

Interactive mobile vision applications, such as Mobile Landmark Recognition (MLR), have recently attracted ever increasing research attention due to the exponential growth of mobile devices. However, the recognition accuracy retains as a bottleneck hesitating the proliferation of such applications. To address this challenge, in this paper we design a novel framework based on interactive image segmentation and multiple visual features fusion to improve the accuracy of on-device MLR systems. Firstly, we propose a simple but effective vector binarization method to reduce the memory usage of image description significantly without decreasing the search accuracy. Secondly, we design a location aware fusion algorithm which can integrate multiple visual features into a compact yet discriminative image descriptor on-device. Thirdly, a user-friendly interaction scheme is developed to enable interactive foreground/background segmentation to improve the recognition accuracy. Experimental results demonstrate the effectiveness of the proposed algorithm for on-device MLR applications.

Index Terms—Mobile Landmark Recognition, On-Device, Binarization, Feature Fusion, User Interaction

1. INTRODUCTION

With the development of mobile computing techniques, interactive mobile vision applications, such as Mobile Landmark Recognition (MLR) have attracted extensive research attention in these days. As a sort of location base service [6], MLR systems enable the mobile user to capture the image by using his/her camera phone, from which the location can be recognized. The recognized location and its corresponding information can be used to enhance the user experience, e.g., assistant city guide or navigation tools [18]. For example, when a tourist wants to know more about a landmark, he can capture an image by his mobile phone and upload it to the search server. The server performs vision-based landmark recognition and then feedbacks the information about the recognized landmark, i.e., the name, history, or the related dining, entertainment, shopping and traffic suggestions.

The user experience of MLR applications highly depends on the recognition accuracy as well as the response time. On one hand, the recognition should be as accurate as possible to avoid user trial-and-failure. On the other hand, the returned result should be as timely as possible to avoid the loss of user patience.

To obtain accurate MLR, most current systems [1][5][9][11][14][18] rely on client-server mode in which a city (million) scale database is stored at a server and the landmark recognition is accomplished via visual features based matching. Although promising, limitations still exist to largely affect the user experience. For example, the response time of these systems critically depends on how much information is transferred in the possibly unstable wireless link, while in some cases the recognition has to be performed in regions with no cell phone service. Moreover, the user's personal information such as the route of travel can easily be disclosed by transferring the captured image and the GPS information to a remote server.

In view of the above problems, it is foreseeable that if the landmark recognition can be performed directly on a mobile device, the user experience can be improved substantially thanks to the ever increasing computation capability of the mobile devices. For example, the system can still work even in regions with no cell phone service. Moreover, the user's personal information will not be disclosed because no data will be uploaded to a remote server. While promising, it is really challenging to scale up the recognition to the entire city due to the storage limitation on mobile devices. Firstly, the search engine should be compact enough to be deployed directly on the RAM of a mobile device. Secondly, to ensure good user experience, the recognition accuracy should be still comparable to the client-server based MLR systems. Despite the ongoing research in recent years [4; 8; 13], such issues retain open in the state-of-the-art systems. For example, the method of [4] used hundreds of bytes to represent a single image, which is still not compact enough to manage millions of images directly on the mobile end. The method of [8] can compress each image descriptor into several bytes. However, the user experience needs to be improved as the significant compression of image descriptors will reduce the recognition accuracy. In sum, the design of compact and accurate landmark search method in the city scale has become a vital problem in improving the user experience of on-device MLR systems.

In this research, we tackle this problem from both the system and the user level. In the system level, we design a multi-feature fusion scheme that innovates at the following aspects: Firstly, a simple but effective vector binarization method which can reduce the memory usage of image descriptors significantly without decreasing the recognition accuracy. Secondly, we design a location aware fusion algorithm to fuse multiple visual features into a compact and discriminative descriptor. Thirdly, we jointly optimize the feature fusion and indexing procedures to further improve the recognition accuracy. In the user level, we propose an interactive foreground and background segmentation approach to improve both the recognition accuracy and the user experience.

2. THE PROPOSED FRAMEWORK

In the offline part, we firstly extract multiple visual features from each reference image, which are then converted into binary vectors as detailed in Section 3. We then propose a location-aware fusion scheme to fuse these binary features (Section 3). In the online part, the user interaction is presented in Section 4 to segment the target landmark from the query image. Then, multiple binary visual features of the segmented landmark are fused to generate the visual descriptor for search.

3. MULTIPLE VISUAL FEATURE BINARIZATION

Visual Features: We briefly introduce the image descriptors used in our system: (1) *VLAD (Vector of Locally Aggregated Descriptors)*: We detect 64 dimensional gravity-aligned SURF [2][10] features from the reference images and then use K-means algorithm to generate 64 cluster centroids. To generate the VLAD descriptor of an image, we firstly assign each detected gravity-aligned SURF feature to its nearest cluster centroid. Then, a 4,096 dimensional VLAD descriptor is obtained by directly concatenating the aggregated residual vector in all the centroids. (2) *BOF*: We build a vocabulary tree (depth=4 and branch=10) with 1K visual words using the same set of gravity-aligned SURF features used in VLAD. We generate the BOF descriptor of each image by using the standard TF-IDF [15] method. (3) *PHOG*: We firstly extract canny edges and then quantize the gradient orientation on the edges (0° to 180°) into 20 bins. Three spatial pyramid levels are used (1×1 , 2×2 and 4×4). We also align each input image according to the direction of gravity to make the generated PHOG descriptors rotation-invariant.

Binarization: We quantize each visual descriptor into binary vector to reduce the memory usage. Most of the existing methods, for instance codebook based or Fisher vector based approaches, rely on analyzing the distribution of the training set with a time consuming learning process to generate the quantizer to achieve binarization. In this paper, we design a simple binarization method targeting at finding a threshold for each descriptor separatively.

Without loss of generality, it is reasonable to target at the goal that the descriptors in the same landmark should have similar values on each corresponding dimension. We can reasonably assume that the similar thresholds for visual descriptors from the same landmark can result in the similar binarized descriptors. To this end, numerous approaches are there to determine the value of the needed threshold. For example, we can set the value of the threshold as the mean of the normalized descriptor. In this paper, a more stable threshold computation and binarization method is written as

$$b_t = \begin{cases} 1 & if \left|\sqrt{v_t}\right| \geq \sum_{t=1}^{T} \left|\sqrt{v_t}\right| \Big/ T \\ 0 & Otherwise \end{cases} \quad (1)$$

where $V = (v_1 \quad v_2 \quad \cdots \quad v_T)$ is the normalized input image descriptor to be binarized. $B = (b_1 \quad b_2 \quad \cdots \quad b_T)$ is the obtained binary vector corresponding to V . T is the dimensionality of the image descriptor. The square root operation in Equation 1 is used to reduce the influence of peaky dimensions.

3. MULTIPLE VISUAL FEATURE FUSION

We design an early fusion strategy to combine multiple binary visual descriptors into a compact and discriminative binary descriptor, with two observations: (1) for each landmark, there will be some representative features or dimensions which can be used to distinguish this landmark from the others efficiently. (2) in MLR, the entire geographical map is commonly partitioned into different landmarks to facilitate the recognition process.

We use Boosting to select the most discriminative dimensions from multiple features in each landmark region individually. The concatenation of multiple binary visual descriptors B_{VLAD} , B_{BOF} and B_{PHOG} is denoted as $B = (B_{VLAD}, B_{BOF}, B_{PHOG}) = (\underbrace{b_1,\cdots,b_{4096}}_{B_{VLAD}}, \underbrace{b_{4097},\cdots,b_{5096}}_{B_{BOF}}, \underbrace{b_{5097},\cdots,b_{5516}}_{B_{PHOG}})$. Our idea is to minimize the ranking loss of all sample queries. Since our algorithm is based on the binary vectors by using the fused binary vectors, Hamming distance is adopted to rank the search results.

Each dimension is regarded as a weak ranker. Then, in the $k-th$ iteration of our algorithm, we select the dimension which minimizes the ranking loss of correct matches, i.e.,

$$\sum_{n=1}^{N} \alpha_n^k \sum_{r=1}^{R} Rank_H(I_n^r) \cdot D_H(Q_n, I_n^r) \quad (2)$$

where $Rank_H(I_n^r)$ is the current ranking position of I_n^r by the metric of the Hamming distance $D_H(\cdot)$ where. is computed by using the current k selected dimensions from B . α_n^k is the error weighting of sample query Q_n to make the

dimension selecting process benefiting the entire set of training images.

With Hamming distance, there may be the case when multiple dimensions give the same minimal ranking loss. Let $\widetilde{S}_1$ be the set of such dimensions. From $\widetilde{S}_1$, the dimension maximizing the ranking loss of mismatches is selected, i.e.,

$$\sum_{n=1}^{N}\beta_n^k\sum_{s=1}^{S}Rank_H(M_n^s)\cdot D_H(Q_n,M_n^s) \tag{3}$$

where $Rank_H(M_n^s)$ is the current ranking position of M_n^s by considering $D_H(\cdot)$. β_n^k is the error weighting which is used to make the dimension selecting process punishing all mismatches instead of a small part of them. For the computational process of $(k+1)-th$ iteration, , we then compute α_n^{k+1} and β_n^{k+1} based on the selection result of $k-th$ dimension as

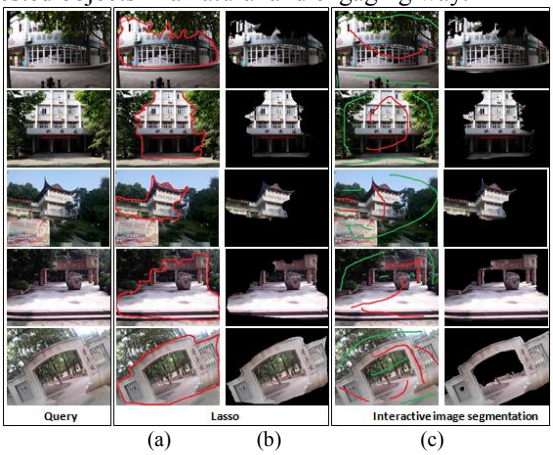
$$\alpha_n^{k+1}=\sum_{r=1}^{R}Rank_H(I_n^r)\cdot D_H(Q_n,I_n^r) \tag{4}$$

$$\beta_n^{k+1}=\frac{1}{\sum_{s=1}^{S}Rank_H(M_n^s)\cdot D_H(Q_n,M_n^s)} \tag{5}$$

where $Rank_H(\cdot)$ and $D_H(\cdot)$ is computed by using the k dimensions selected in the previous k iterations.

4. INTERACTION DESIGN

We further design an efficient interface by using interactive image segmentation (Fig.1) so that users can segment interested objects in a natural and engaging way.



(a) (b) (c)

Fig.1. Illustration of Lasso and interactive image segmentation

To segment a foreground object, a few lines are marked manually through the touch screen of mobile device. The red lines and green lines indicate the foreground and background markers respectively. The segmentation process is started once the user clicks the "Segment" button after drawing each marking line. In case of the segmentation result is not satisfied, the user can mark more lines and re-click the "Segment" button to refine the segmentation result. Different from the Lasso-based interactive segmentation technique proposed recently in [14], the segmentation algorithm used in our system is the interactive graph cut algorithm proposed in [16], which requires a minimum user interaction to identify the target of interest, as shown in Fig. 1

5. RESULTS

Benchmark: In our experiments, the San Francisco dataset [5] is employed in quantitative evaluation, which contains two parts. PCI (perspective central images) set contains approximately 1.06 million images which are generated from the center of the panoramas. PFI (perspective frontal images) set contains approximately 0.638 million images each of which is generated by shooting a ray through the center of projection of a PCI and computing the ray intersection point with the scene geometry. The database provides 803 query images of landmarks in San Francisco taken with several different camera phones by various people at several months after the database images are collected. We divide the whole workspace into 64 regions by considering the GPS information. For each query image, we firstly use GPS to find a candidate region and then obtain the search results by directly computing the hamming distances between binary descriptors. In this experiment, we directly use the original query images instead of the segmented ones to test the performance of different method. The heading information is also ignored.

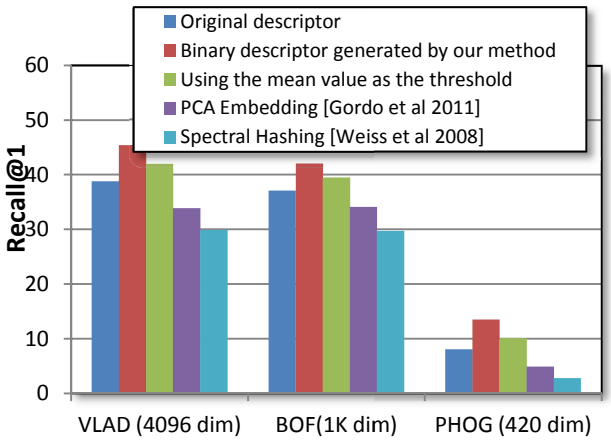


Fig.2. Search accuracy of the proposed binarization method on San Francisco PFI datasets respectively.

Binarization: We implement the following three methods for comparison: (1) The PCA embedding method proposed in [7]. (2) The Spectral Hashing method proposed in [17]. (3) Using the mean value as the threshold to fulfill the binarization process. To make a fair comparison, we generate the binary codes with the same length (identical to

the dimensionality of input vectors) when using different methods to binarize input vectors. We have tested the performance of different methods on normalized VLAD, BOF and PHOG respectively. The results are shown in Fig. 2, which indicates that the binary descriptor obtained by using the method can obviously improve the recognition accuracy than that of the others.

Feature Fusion: For each query image, we use GPS to find a candidate region and then obtain the searching results by computing the Hamming distances between the fused binary visual descriptors. We also test the performance of the following two strategies for comparison use. Moreover, the training and searching processes are performed by using the original query images instead of the segmented ones.

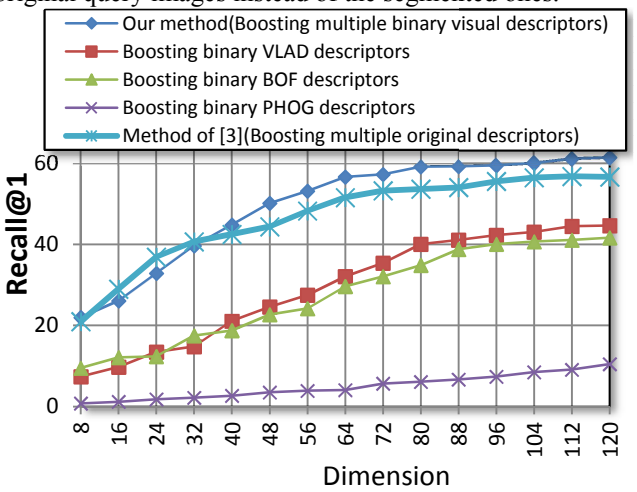


Fig.3. Search accuracy of the proposed feature fusion method on San Francisco PFI datasets respectively.

The results are shown in Fig.3 from which we can draw the following conclusions: (1) Boosting of multiple features can achieve better accuracy than boosting a single kind of feature. (2) Compared with the method of [9], our method not only reduces the memory usage by a factor of 32, but also improves the search accuracy significantly. The above results demonstrate the effectiveness of the proposed feature fusion method.

Comparison with State-of-the-Art: The proposed multiple-feature based landmark recognition is compared with previous methods, including: (1) The late fusion strategy proposed in [4]: Both SURF and CHoG features are extracted to generate two different VLAD descriptors for each image. (2) The method of [5]: We build a vocabulary tree (depth=6 and branch factor=10, 64-dimensional SURF features) with 1 million leaf nodes to generate the BOF descriptor of each image. To perform landmark recognition, only the database images that are within 300 meters of the query image will be scored. Finally, a geometric verification (RANSAC with a 2D affine model) process is carried out to refine the recognition results. (3) The method of [8]: We use the TC-RVQ method [8] to encode each PCA compressed VLAD descriptor (256 dimensional) into 10 bytes and then use the index structure of Fig.2 to index the generated codes. (4) The method of [9]: We use the boosting method proposed in [9] to compress each 1-million dimensional BOF descriptor to 80 dimensionalities. As shown in Fig.4, our multiple-feature based landmark recognition method can provide better recognition accuracy than that of other compared methods.

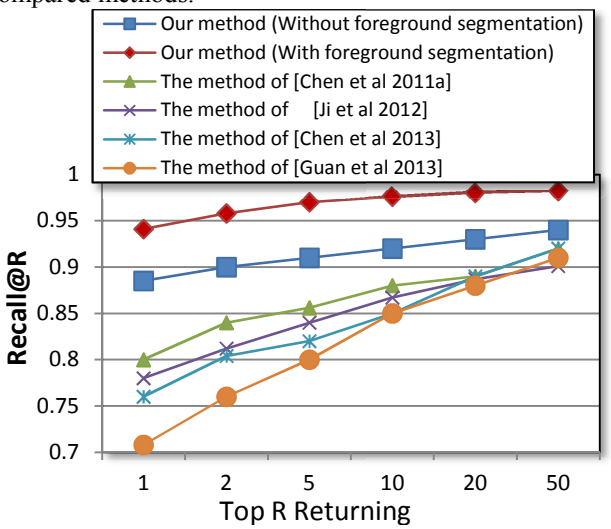


Fig.4. Comparison with previous methods.

Time and Memory: Our method can fulfill the multi-feature based landmark recognition within 1.71s. The computation time of our method is slightly longer than that (1.3s) of the method proposed in [8]. We also record the memory usage of our method. The index nodes take about 108K bytes (608 index nodes, each region ID takes 2 bytes, the serial numbers take 80× 2=160 bytes, each GPS centroid takes 8 bytes and the 12 heading nodes take 12 bytes). Each image descriptor takes 15 bytes (10 bytes for fused visual descriptor, 3 bytes for image ID, and 2 bytes for GPS code). The vocabulary tree used for BOF generating takes 271K bytes. Therefore, our image searching engine can load a database of 1.295 million images into the RAM of a mobile device by costing about 18.87M bytes.

5. CONCLUSIONS

This paper targets at fusing multiple visual features in a compact manner with an intelligent interactive interface design to enhance the user experience of city scale on-device mobile landmark recognition. We innovates at the following aspects: binarizing visual descriptors to reduce the memory usage, fusing multiple visual features to get more compact and discriminative image descriptors, and finally, an interactive foreground/background segmentation to improve the recognition accuracy and user experience.

ACKNOWLEDGEMENT

This work is supported by the Special Fund for Earthquake Research in the Public Interest No.201508025, the Nature Science Foundation of China (No. 61422210 and No. 61373076), the Fundamental Research Funds for the Central Universities (No.2013121026).

REFERENCES

[1]. Baatz, G., Koeser, K., Chen, D., Grzeszczuk, R., and Pollefeys, M. Handling Urban Location Recognition as a 2D Homothetic Problem. In Proceedings of European Conference on Computer Vision, 2010.

[2]. Bay, H., Ess, A., Tuytelaars, T. and Gool, L, V. SURF: Speeded Up Robust Features. Computer Vision and Image Understanding. 110, 3, 346-359, 2008.

[3]. Boykov, Y., AND Jolly, M, P., 2001. Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images. In Proceedings of IEEE International Conference on Computer Vision, 2001.

[4]. Chen, D., Tsai, S., Chandrasekhar, V., Takacs, G., Vedantham, R., Grzeszczuk, R., and Girod, B. Residual enhanced visual vector as a compact signature for mobile visual search. Signal Processing, 93. 8. 2316-2327, 2013.

[5]. Chen, D., Baatz, G., Koeser, K., Tsai, S., Vedantham, R., Pylvanainen, T., Roimela, K., Chen, X., Bach, J., Pollefeys, M., Girod, B., and Grzeszczuk, R. City-scale landmark identification on mobile devices. In Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition, 737-744, 2011.

[6]. Dey, A., Hightower, J., de Lara, E., Davies, N. Location-Based Services. IEEE Pervasive Computing, 9, 1, 11-12 ,2010.

[7]. Gordo, A. and Perronnin, F. Asymmetric Distances for Binary Embeddings. In Proceedings of International Conference on Computer Vision and Pattern Recognition, 729-736, 2011.

[8]. Guan, T., He, Y, F., Gao, J., Yang, J, Z., Yu, J, Q. On-Device Mobile Visual Location Recognition by Integrating Vision and Inertial Sensors. IEEE Trans. Multimedia, 2013.

[9]. Ji, R, R., Duan, L, Y., Chen, J., Yao, H, X., Yuan, J, S., Rui, Y., Gao, W. Location Discriminative Vocabulary Coding for Mobile Landmark Search. International Journal of Computer Vision. 96, 3, 290-314, 2012.

[10]. Kurz, D. and Benhimane, S. Inertial sensor-aligned visual feature descriptors. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2011.

[11]. Liu, H., Mei, T., Luo, J, B., Li, H, Q. and Li, S, P. Finding Perfect Rendezvous On the Go: Accurate Mobile Visual Localization and Its Applications to Routing. In Proceedings of ACM Multimedia, Nara, Japan, 2012.

[12]. Nister, D. and Stewenius, H. 2006. Scalable Recognition with a Vocabulary Tree. In Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2161–2168, 2006.

[13]. Schroth, G., Huitl, R., Chen, D., Abu-Alqumsan, M., Al-Nuaimi, A., and Steinbach, E. Mobile Visual Location Recognition. IEEE Signal Processing Magazine. 28, 4, 77-89, 2011..

[14]. Sang, J, T., Mei, T., Xu, Y, Q., Zhao, C., Xu, C, S. and Li, S, P. Interaction Design for Mobile Visual Search. IEEE Trans. on Multimedia, 2013.

[15]. Sivic, J. and Zisserman, A. Video Google: A Text Retrieval Approach to Object Matching in Videos. In Proceedings of the International Conference on Computer Vision, 1470–1477, 2003.

[16]. Tian, Y., Guan, T., Wang, C., Li, L,J., Liu, W. Interactive foreground segmentation method using mean shift and graph cuts. Sensor Review, 29 , 157-162, 2009.

[17]. Weiss, Y., Torralba, A., and Fergus, R. Spectral hashing. In Proceedings of Advances in Neural Information Processing Systems, 2008.

[18]. Yap, K, H., Li, Z., Zhang, D, J., Ng, Z, K. Efficient Mobile Landmark Recognition Based on Saliency-Aware Scalable Vocabulary Tree. In Proceedings of ACM International Conference on Multimedia, 1001-1004, 2012.

[19]. Yap, K, H., Chen, T., Li, Z., Wu, K. A Comparative Study of Mobile-Based Landmark Recognition Techniques. IEEE Intelligent Systems, 25, 1, 48-57, 2010.