ACCURATE FEATURE MATCHING AND SCORING FOR RE-RANKING IMAGE RETRIEVAL RESULTS

Yusuke Uchida, Shigeyuki Sakazawa

KDDI R&D Laboratories, Inc., Japan

ABSTRACT

In this paper, a new reranking approach is proposed to refine the results obtained with a bag-of-visual words (BoVW) image retrieval method. First, a simple but effective criterion to reject unreliable feature matches is proposed, where the information of nearest neighbors from a large dataset is used to accurately estimate feature density. Second, by adopting a product quantization-based nearest neighbor method in both the voting and reranking steps, it becomes possible to reuse the information obtained in the BoVW method in the reranking step. Finally, a density ratio-based scoring method is naturally integrated to calculate a new score from inliers.

Index Terms— Specific object recognition, bag-of-visual words, product quantization, geometric verification, feature matching

1. INTRODUCTION

With the advancement of both stable interest region detectors [1, 2] and robust and distinctive descriptors [1, 3], local feature-based image or object retrieval has attracted a great deal of attention. It has become particularly applicable to large-scale databases with a bag-of-visual words (BoVW) framework [4]. In the BoVW framework, local feature points or regions are detected from an image, and feature vectors are extracted from them. These feature vectors are quantized into visual words (VWs) using a visual codebook, resulting in a histogram representation of VWs. Image similarity is measured by L_1 or L_2 distance between the normalized histograms. As VW histograms are generally very sparse, an inverted index data structure and a voting function enables an efficient similarity search. A term frequency-inverse document frequency (TF-IDF) weighting scheme is naturally integrated with the voting function [4].

Although the BoVW framework realizes efficient retrieval, there is some room for improvement in terms of accuracy. One significant drawback of VW-based matching is that two features are matched if and only if they are assigned to the same VW [5]. There are two major expansions of the voting function to alleviate this problem: post-filtering [5, 6] and multiple assignment [7, 5]. In post-filtering, after VW-based matching, unreliable matches are filtered out according to (estimated) distances between query and reference features. In multiple assignment, query features vote not only for reference features in the nearest VW but also for reference features in the k-nearest VWs. The use of weak geometric information in conjunction with the BoVW framework also improves the performance [5, 8], where the IDF scores are voted for orientation and scale space associated with feature matches [5], or position space of target objects [8]. Finally, query expansion [9] and geometric verification [10] are performed to refine scores obtained with the voting function.

Geometric verification is a very important step to refine the scores obtained with the voting function by filtering out outliers of feature matches and reranking the first result. However, reranking based on geometric verification sometimes degrades accuracy in real applications because target objects are generally assumed to be planar and rigid. This is the case when recognizing non-planar product packages or deformable objects such as magazines or books. Although there are a few approaches to detect or track deformable objects [11], it is time consuming in one-to-many matching which is required in reranking.

In this paper, in order to improve the results obtained with the voting function without geometric information, a new reranking approach based on effective feature matching and scoring is proposed. In Section 2, conventional feature matching methods and their problems are explained. In Section 3, a simple and effective criterion to reject unreliable feature matches based on the probabilistic density function (PDF) of local features is proposed, where the information of nearest neighbors from a large dataset is used to accurately estimate the PDF. In Section 4, the proposed criterion is integrated with a state-of-the-art product quantization-based framework [6, 12], where the information of nearest neighbors obtained in the BoVW framework is reused in the reranking step for efficiency. An effective density ratio-based scoring method [13] is adopted in the framework to obtain final scores.

2. RERANKING BASED ON GEOMETRIC VERIFICATION

The results obtained in voting can be improved with reranking based on geometric verification [10], where the spatial consistency among a query image and reference images is used as a constraint to filter out outliers. Because geometric verification is more time consuming than BoVW scoring, it is performed only on a shortlist of top scoring images. Reranking based on geometric verification mainly consists of the following steps: (1) for each feature q in a query image Q, perform nearest neighbor matching against features in a target image and obtain a tentative match, (2) estimate an affine or homography matrix using the tentative matches, and (3) replace the score between the query image and the target image by the number of inliers or the sum of IDF scores of inliers which satisfy the affine or homography matrix constraint. Because tentative matches include many outliers, it is necessary to filter out unreliable matches to obtain stable results.

There are two directions to improve the accuracy of tentative matches. One is to use sophisticated (dis)similarity measures: Chi-squared distance [14], Hellinger kernel [9], or earth mover's distance (EMD) [15] is used as (dis)similarity measures. The other is to filter out unreliable matches based on a matching criterion. The most frequently used criterion is ratio criterion [1, 3, 14, 16] to filter out unreliable matches: if the ratio of the best distance d_1 and the second best distance d_2 is greater than a threshold α (typically 0.8), the tentative match is rejected. This ratio criterion works better than the distance criterion [3] because SIFT features are not distributed uniformly in the feature space [12] and a constant threshold cannot cope with this aspect. The ratio criterion is adaptive to some extent according to feature distribution, and it is also generic and not restricted to the Euclidean distance (e.g., it is applied to the Hamming distance [16], the Chi-squared distance [14], and EMD [15]). However, the ratio criterion does not work well in some situations. First, it does not work if multiple similar features are detected from near regions; this is often the case for the Hessian-Affine detector [2] or the multi-scale FAST detector [17], for instance. Second, its performance degrades if approximate distances are used because feature distribution is estimated using only a single target image, and is thus sensitive to noise. This often happens when features are encoded into compact codes [6, 16, 5] for memory and time efficiency. In the next section, to address these issues, we propose a new criterion based on the density of local features, which is estimated using a large dataset independent of the target image.

3. PROPOSED CRITERION

Using the proposed criterion, the tentative match between q and its nearest neighbor $\hat{r}$ is rejected if the likelihood ratio $p(q|\hat{r})/p(q)$ is smaller than the threshold th, where $\hat{r}$ denotes the nearest neighbor feature in the target image R, $p(q|\hat{r})$ denotes the probability that q is generated from $\hat{r}$, and $p(q)$ denotes the probability that q is generated independently of R. Denoting by $p(q|R)$ the probability that q is generated from one of the features in the target image R, it is clear that an inequality $p(q|\hat{r}) \leq p(q|R)$ holds. Due to the long-tail characteristic of feature distributions, almost all of the features are rather isolated in the feature space [18]. In other words, $p(q|r) \approx 0$ for $r \neq \hat{r}$. Hence, $p(q|R)$ is a good approximation of $p(q|\hat{r})$. The likelihood ratio $p(q|\hat{r})/p(q)$ is now approximated by $p(q|R)/p(q)$.

Here, we assume a distance list $(d_1, \cdots, d_T)$ with size T is available, where d_t is the distance between q and the t-th nearest neighbor feature in a large, independent database as shown in Figure 1 (a). Using the distance list, $p(q|R)$ and $p(q)$ are estimated via k-nearest neighbor density estimation:

$$p(q|R) = \frac{1}{|\mathcal{R}| \cdot V_{\hat{d}}},\ p(q) = \frac{\hat{t}}{|\mathcal{R}_{\mathrm{DB}}| \cdot V_{\hat{d}}}, \tag{1}$$

where $|\mathcal{R}|$ is the number of features in R, $|\mathcal{R}_{\mathrm{DB}}|$ is the number of features in the database, $V_{\hat{d}}$ is the volume of a hypersphere with radius $\hat{d}$, $\hat{d}$ is the distance between q and $\hat{r}$, and $\hat{t}$ is the smallest t that satisfies $\hat{d} < d_t$. As shown in Figure 1 (b), for $p(q|R)$ and $p(q)$, $k = 1$ and $k = \hat{t}$ are respectively used in k-nearest neighbor density estimation; in the case of $p(q|R)$, one feature $\hat{r}$ out of $\mathcal{R}$ exists in the hypersphere, while, in the case of $p(q)$, $\hat{t}$ features including $\hat{r}$ out of $\mathcal{R}_{\mathrm{DB}}$ exist in the hypersphere. Now the criterion to reject the tentative match associated with q is:

$$\frac{p(q|\hat{r})}{p(q)} \approx \frac{|\mathcal{R}_{\mathrm{DB}}|}{\hat{t} \cdot |\mathcal{R}|} < th. \tag{2}$$

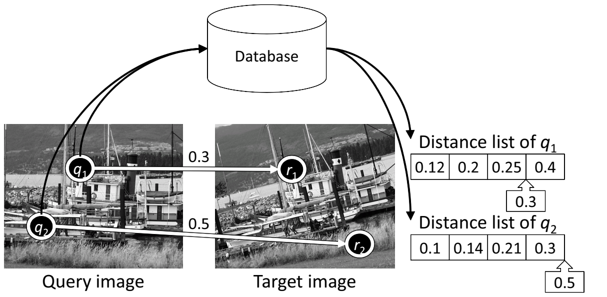
As Eq. (2) can be written as $\hat{t} > |\mathcal{R}_{\mathrm{DB}}|/(|\mathcal{R}| \cdot th)$, we can use $T' = |\mathcal{R}_{\mathrm{DB}}|/(|\mathcal{R}| \cdot th)$ $(T' \leq T)$ as a new threshold for $\hat{t}$.

To summarize, the proposed method consists of the following simple procedures: (1) calculate the distance $\hat{d}$ from q to its nearest neighbor $\hat{r}$ in a target image R, (2) calculate the distances $d_1, \cdots, d_T$ from q to its T nearest neighbors in a large dataset, and (3) reject the tentative match associated with q if $\hat{d} > d'_T$.

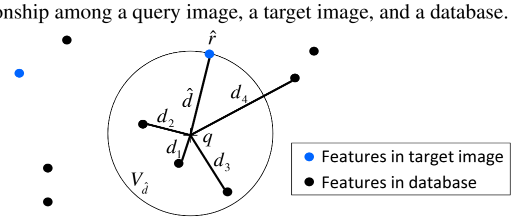
In this paper, we refer to the criterion as a rank criterion because $\hat{t}$ corresponds to the rank of $\hat{r}$ in a distance list. In the case of Figure 1 (a), where $T' = T = 4$, the tentative match associated with q_1 is considered to be an inlier, while that with q_2 is rejected. It is desirable to obtain distance lists from a large database, because, if an infinite number of samples is available, the density estimated using the k nearest neighbor method converges to the true density. However, obtaining distance lists from a large database imposes a heavy computational burden. In the next section, we propose the framework where a overhead becomes negligible because we reuse distance lists of query features which have been already obtained in the process of index search.

4. PROPOSED FRAMEWORK

In this section, we describe the proposed framework, which naturally integrates the state-of-the-art product quantization-based image retrieval method [6, 12, 13] with the proposed



(a) The relationship among a query image, a target image, and a database.



(b) An example of k-nearest neighbor density estimation.

Fig. 1: Overview of the proposed method.

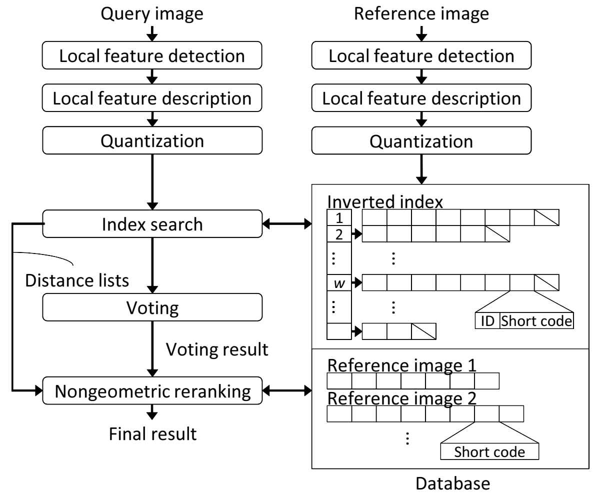


Fig. 2: Framework of the proposed image retrieval system.

reranking method. In the reranking step, we also adopt the product quantization-based distance calculation method with different settings. This enables efficient matching and makes distance lists obtained in the index search compatible with distances calculated in the reranking step. New scores are calculated and voting results are refined according to feature matches which satisfy the proposed criterion described in Section 3. Figure 2 provides an overview of the proposed framework and the data structure used in the framework.

4.1. Feature detection and description

From query and reference images, a set of feature vectors is extracted. We adopt Hessian-Affine [19] and SIFT [1] as the feature detector and descriptor, respectively. We denote the i-th feature vector of the query image by $q_i \in \mathcal{Q}$ and the h-th feature vector of the j-th reference image by $r_{jh} \in \mathcal{R}_j$.

4.2. Feature indexing with product quantization

We adopt a product quantization-based method [6] to improve the BoVW framework, namely IVFADC. In the indexing (offline) step in IVFADC, a reference vector r_{jh} with d dimension is quantized with a coarse quantizer in the same way as the BoVW framework. We refer to the codebook used in coarse quantization as the CQ codebook. This is the same as what is referred to as visual words or a visual codebook in the context of BoVW-based image retrieval or recognition.

In the indexing step, a reference vector r_{jh} is first quantized into $c_{\hat{a}}$ using the CQ codebook C with k' centroids $c_1, \cdots, c_{k'} \in \mathbb{R}^d$, where

$$\hat{a} = \arg \min_{1 \leq a \leq k'} ||r_{jh} - c_a||^2. \tag{3}$$

Subsequently, the residual vector $\bar{r}_{jh}$ from the corresponding centroid $c_{\hat{a}}$ is calculated as $\bar{r}_{jh} = r_{jh} - c_{\hat{a}}$. Then, the residual vector $\bar{r}_{jh}$ is decomposed into u subvectors $\bar{r}^1_{jh}, \cdots, \bar{r}^u_{jh} \in \mathbb{R}^{d^*}$, where $d^* = d/u$. Subsequently, these subvectors are quantized separately using u codebooks $P_1, \cdots, P_u$. This is referred to as product quantization. In this paper, a codebook used in product quantization is referred to as a PQ codebook. We assume that each PQ codebook P_l has k^* centroids $p_{l1}, \cdots p_{lk^*} \in \mathbb{R}^{d^*}$. Using the l-th PQ codebook, the l-th subvector $\bar{r}^l_{jh}$ is quantized into p_{lb_l}, where

$$b_l = \arg \min_{1 \leq b \leq k^*} ||\bar{r}^l_{jh} - p_{lb}||^2. \tag{4}$$

Finally, the short code $(b_1, \cdots, b_u)$ is stored in the $\hat{a}$-th list of the inverted index with the identifier j of the reference image. The size of the short code is represented by $u \log_2 k^*$ bits.

In addition, local features are stored independently of the inverted index for reranking. This is needed because in reranking each local feature of a query image should be compared with the features of each of the images in a shortlist. In IVFADC, residual vectors are encoded by product quantization, while original SIFT vectors are encoded here, which is referred to as ADC in [6]. Note that we can choose an arbitrary number u' of vector decomposition independent of u.

4.3. Index search

In the search step in IVFADC, the T nearest features $\mathcal{N}_T(q_i)$ of q_i are obtained from the inverted index. A query vector q_i is first quantized using the CQ codebook, and the residual vector $\bar{q}_i$ from the corresponding centroid is calculated in the same manner as the indexing. Subsequently, the distance between the residual vector $\bar{q}_i$ and short codes $(b_1, \cdots, b_u)$ in the corresponding list in the inverted index are calculated.

These distances correspond to the approximate distances between the query vector q_i and the reference vectors r_{jh}:

$$d(q_i, r_{jh}) = d(\bar{q}_i, \bar{r}_{jh}) \approx \sqrt{\sum_{l=1}^{u} ||\bar{q}_i^l - p_{lb_l}||^2}. \quad (5)$$

This distance calculation is performed efficiently using a precomputed lookup table [6]. Finally, the T nearest features $\mathcal{N}_T(q_i)$ of q_i are obtained by sorting according to the distances. The distances obtained in this process are reused in the reranking step described in Section 4.5.

4.4. Voting scores

Once the T nearest neighbors $\mathcal{N}_T(q_i)$ of q_i are obtained with the index search, the scores are voted for the reference images associated with $\mathcal{N}_T(q_i)$. We adopt density ratio-based scoring [13], which has been shown to be superior to IDF scoring. The voting score s_{ij} from query feature q_i to reference image R_j is calculated using the k nearest neighbors of q_i as

$$s_{ij} = \log\left(\frac{\lambda}{1-\lambda}\frac{|\mathcal{N}_T(q_i)_j|/|\mathcal{R}_j|}{|\mathcal{N}_T(q_i)|/|\mathcal{R}_{\mathrm{DB}}|} + 1\right), \quad (6)$$

where $|\mathcal{N}_T(q_i)| = T$, and $|\mathcal{N}_T(q_i)_j|$ is the number of features in $\mathcal{N}_T(q_i)$ associated with R_j, and λ denotes an adjustable parameter that controls the strength of the prior distribution of $p(q_i|R_j)$. For each q_i, the voting score s_{ij} is assigned to each R_j. The resulting $\sum_i s_{ij}$ corresponds to the similarity measure between Q and R_j, and a voting result is obtained.

4.5. Nongeometric reranking with density ratio scoring

For each of the top M images in a voting result (shortlist), new scores are calculated to refine the first result. First, approximate distances are calculated in the manner described in Section 4.3, and the distance $\hat{d}_i$ from q_i to its nearest feature $\hat{r}_i$ in a target image in the shortlist is obtained.

The distance lists from the index search are used to reject unreliable matches: given a distance list $d_{i1}, \cdots, d_{iT}$ for query feature q_i, the tentative match associated with q_i is rejected if $\hat{d}_i > d_{iT'}$. In this paper, as the proposed criterion is very reliable, geometric verification is skipped and reranking is performed using the non-rejected tentative matches. Instead of simply counting the number of inliers, sum of modified version of scores in Eq. (6) is used as the new score:

$$s_i = \log\left(\frac{\lambda}{1-\lambda}\frac{1/|\mathcal{R}|}{\hat{t}/|\mathcal{R}_{\mathrm{DB}}|} + 1\right), \quad (7)$$

where the same density estimation is performed as Eq. (1).

5. EXPERIMENTAL EVALUATION

5.1. Evaluation of matching accuracy

The proposed rank criterion described in Section 3 is compared with the conventional ratio criterion in terms of the matching accuracy of local features. The Graffiti dataset [2] is used for the evaluation. The dataset consists of 8 scenes and each of the scenes contains 6 images with a gradual geometric or photometric transformation such as viewpoint changes, image blur, JPEG compression, or illumination. We use the first and second images from each of the scenes for the evaluation. In order to obtain distance lists, 10,000 images are used out of the MIRFLICKR-1M dataset[1]. Approximate distance lists are obtained with the approximate nearest neighbor search method described in Section 4.3. The ground truth homographies between the first image and the others in each scene are also provided, tentative matches with projection error greater than 5.0 pixels are considered as outliers.

Figure 3 shows 1−precision vs. recall curves for the proposed rank criterion and the conventional ratio criterion. Both exact distances and approximate distances are used in matching: the prefixes A8 and A16 indicate that approximate distances are calculated using 8-byte ($u' = 8$) and 16-bytes ($u' = 16$) short codes, respectively. For the proposed method, the threshold T' ranges from 1 to 10, and for the conventional method, the threshold α ranges from 0.8 to 1.0. It is shown that the proposed criterion outperforms the conventional criterion even if approximate distances are used for the proposed criterion and exact distances are used for the conventional criterion. It can also be said that, with the proposed criterion, the degradation of the performance caused by approximate distances is also alleviated owing to the use of an independent, large dataset. The memory requirement is the same for both the proposed method and the conventional method; A8, A16, and Exact requires 8, 16, and 128 bytes per descriptor, respectively. Computational costs are explored in the following section.

5.2. Evaluation of reranking methods

In this section, the proposed framework is evaluated in terms of image retrieval accuracy using the University of Kentucky recognition benchmark dataset[2] provided by the authors of [20]. The dataset includes 2,550 different objects or scenes. Each of these objects is represented by four images taken from four different angles, giving a total of 10,200 images. These images are used as both reference and query images. Mean average precision (MAP), the area under the precision-recall curve, is used as an indicator of performance [20, 5]. Standard parameter settings [5, 6, 12] are used in our experiments. The size k' of the visual codebook is set to 20K. In indexing, reference features are divided into 8 16-dimensional

[1] http://press.liacs.nl/mirflickr/
[2] http://www.vis.uky.edu/~stewe/ukbench/

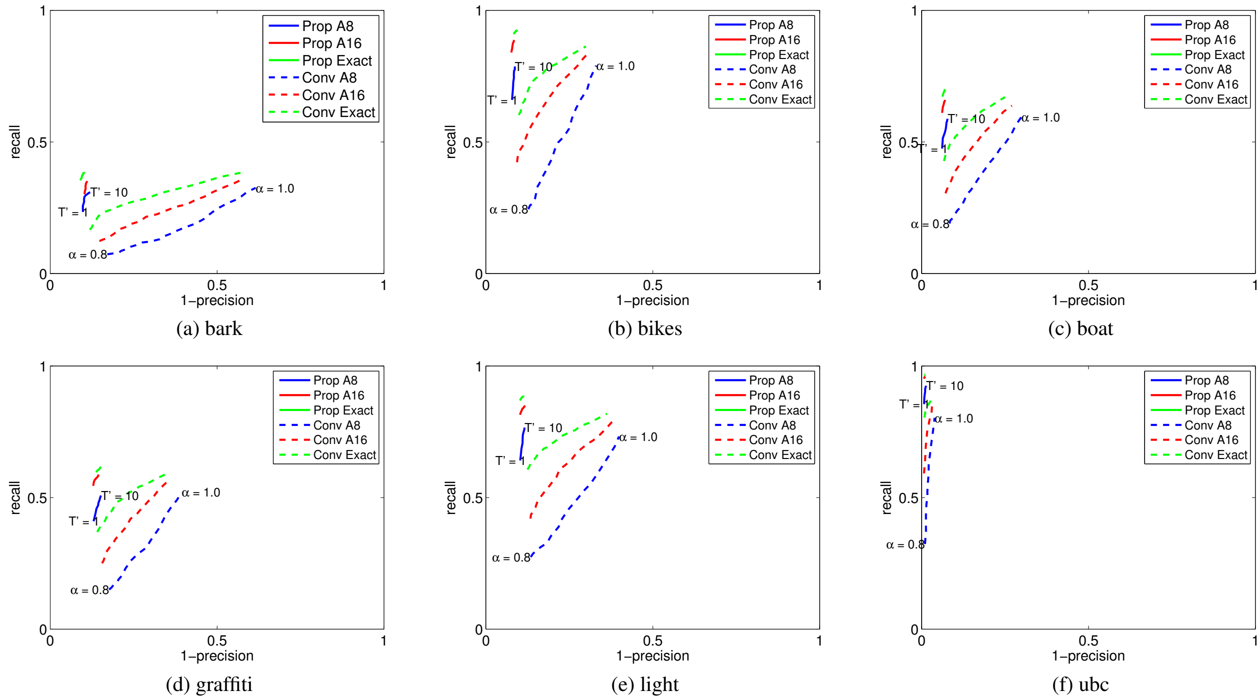


Fig. 3: 1−precision vs. recall curves for the ratio criterion and the proposed criterion. Both exact distance and approximate distance are used in matching. From 3,000 to 5,000 features are extracted with default parameters.

subvectors ($u = 8$), and encoded by a product quantizer with 256 centroids ($k^* = 256$), resulting in 8×8 bit code. For the proposed method, we set $T = 24$ and $T' = 10$. These parameters were not particulary sensitive to the performance (greater is better) in preliminary experiments.

Figure 4 shows accuracy as a function of λ before and after reranking. The MAP score of the baseline system (before reranking) is 0.882. The scores of the top 20 images in a shortlist are refined with the proposed method for Prop using exact and approximate distances. We can see that the accuracy in image retrieval is also improved by the proposed method without geometric information. It is also shown that the accuracy of A16 is comparable with Exact, which is consistent with the results in Section 5.1.

Table 1 shows the processing time required for the proposed method. The processing time required for A8 and A16 can be decomposed into two parts; processing time for table construction (for each query) and distance calculation with table lookups (for each of the images in the shortlist). Top20, Top50, and Top100 represent the processing times required to rerank the top 20, 50, and 100 images in the shortlist, respectively. All distance calculations are optimized with SIMD operations. In the experiment more than 3,000 features are used. As computational cost is proportional to the square of the number of features, if the number of features is reduced to around 1,000, the computational cost is reduced by an order of magnitude. Table 2 also summarizes the MAP scores of

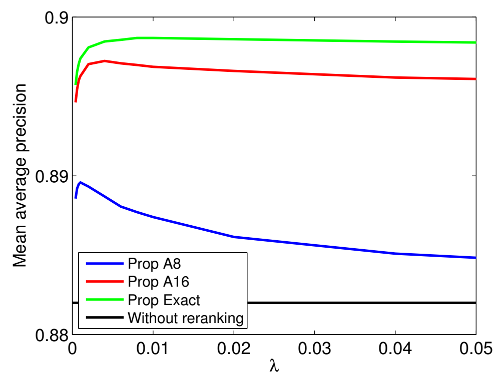


Fig. 4: Accuracy of the proposed framework with different distance calculation scenarios as a function of λ.

the proposed method with the same parameter settings. We observe: (1) there is no significant differences in accuracy between A16 and Exact while Exact requires a large computational cost (almost three times larger than A16), (2) using top 50 and 100 images does not improve accuracy greatly, while causing a linear increase in computational cost.

We also conducted reranking based on geometric verification, where a score in a shortlist is replaced by a new score if the number of inliers is greater than a threshold. The MAP was degraded from 0.882 to 0.871 due to failures in homography estimation for nonplanar or textureless objects and false

Table 1: Processing time required in the proposed system [sec].

	Table	Distance	Top20	Top50	Top100
A8	0.027	0.058	1.193	2.942	5.857
A16	0.040	0.102	2.074	5.125	10.210
Exact	-	0.291	5.810	14.525	29.050

Table 2: Comparison of different parameter settings for the proposed method.

	Top20	Top50	Top100
A8	0.890	0.891	0.890
A16	0.897	0.902	0.904
Exact	0.899	0.904	0.907

positives. Figure 5 shows examples of nonplanar or textureless objects in the database for which homography estimation failed. The proposed method can improve accuracy even against these queries.

6. CONCLUSIONS

In this paper, we proposed a new feature matching criterion and nongeometric reranking method for large-scale image retrieval. The reuse of the information obtained in the BoVW framework enables accurate feature matching and sophisticated scoring in the reranking. In the experiments, we confirmed that the proposed method can improve the results even if geometric information is not used. As a future research topic, we are interested in the utilization of the proposed rank criterion as a priority measure for tentative matches in PROSAC [21].

7. REFERENCES

[1] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *IJCV*, vol. 60, no. 2, pp. 91–110, 2004.

[2] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. V. Gool, "A comparison of affine region detectors," *IJCV*, vol. 60, no. 1-2, pp. 43–72, Nov. 2005.

[3] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *TPAMI*, vol. 27, no. 10, pp. 1615–1630, Oct. 2005.

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

[5] H. Jégou, M. Douze, and C. Schmid, "Improving bag-of-features for large scale image search," *IJCV*, vol. 87, no. 3, pp. 316–336, 2010.

[6] H. Jégou, M. Douze, and C. Schmid, "Product quantization for nearest neighbor search," *TPAMI*, vol. 33, no. 1, pp. 117–128, 2011.

[7] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Lost in quantization: Improving particular object retrieval in large scale image databases," in *Proc. of CVPR*, 2008, pp. 1–8.

[8] Xiaohui Shen, Zhe Lin, Jonathan Brandt, Shai Avidan, and Ying Wu, "Object retrieval and localization with spatially-constrained similarity measure and k-nn re-ranking," in *Proc. of CVPR*, 2012, pp. 3013–3020.

[9] R. Arandjelović and A. Zisserman, "Three things everyone should know to improve object retrieval," in *Proc. of CVPR*, 2012, pp. 2911–2918.

[10] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *Proc. of CVPR*, 2007, pp. 1–8.

[11] M. Cho, J. Lee, and K. M. Lee, "Feature correspondence and deformable object matching via agglomerative correspondence clustering," in *Proc. of WACV*, 2009, pp. 1280–1287.

[12] Y. Uchida, K. Takagi, and S. Sakazawa, "Ratio voting: A new voting strategy for large-scale image retrieval," in *Proc. of ICME*, 2012.

[13] Y. Uchida, K. Takagi, and S. Sakazawa, "An alternative to idf: Effective scoring for accurate image retrieval with non-parametric density ratio estimation," in *Proc. of ICPR*, 2012.

[14] P. E. Forssen and Lowe D. G, "Shape descriptors for maximally stable extremal regions," in *Proc. of ICCV*, 2007.

[15] O. Pele and M. Werman, "A linear time histogram metric for improved sift matching," in *Proc. of ECCV*, 2008, pp. 495–508.

[16] M. Ambai and Y. Yoshida, "Card: Compact and real-time descriptors," in *Proc. of ICCV*, 2011.

[17] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *Proc. of ICCV*, 2011, pp. 2564–2571.

[18] O. Boiman, E. Shechtman, and M. Irani, "In defense of nearest-neighbor based image classification," in *Proc. of CVPR*, 2008, pp. 1–8.

[19] K. Mikolajczyk and C. Schmid, "Scale & affine invariant interest point detectors," *IJCV*, vol. 60, no. 1, pp. 63–86, 2004.

[20] D. Nistér and H. Stewénius, "Scalable recognition with a vocabulary tree," in *Proc. of CVPR*, 2006, pp. 2161–2168.

[21] O. Chum and J. Matas, "Matching with prosac -progressive sample consensus," in *Proc. of CVPR*, 2005.

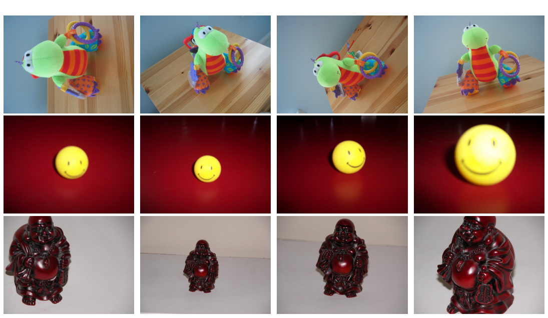


Fig. 5: Examples of nonplanar or textureless objects in the database. The top, middle, and bottom row correspond to the 94, 384, and 2274-th objects, respectively.