Pattern Recognition Letters 98 (2017) 123–129

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Lattice-Support repetitive local feature detection for visual search

Dipu Manandhar*, Kim-Hui Yap, Zhenwei Miao, Lap-Pui Chau

School of Electrical and Electronics Engineering, Nanyang Technological University, 50 Nanyang Avenue, Singapore 639798, Singapore

A R T I C L E I N F O

Article history:
Received 30 August 2016
Available online 12 September 2017

Keywords:
Repetitive pattern detection
Visual burstiness
Image search and retrieval

A B S T R A C T

Repetitive patterns such as building facades, floor tiles, vegetation, and wallpapers are commonly found in sceneries and images. The presence of such repetitive patterns in images often leads to visual burstiness and geometric ambiguity, which poses challenge for state-of-the-art visual search technologies. To alleviate these problems, we propose a new lattice-support repetitive local feature detection method to detect repetitive patterns, estimate the underlying lattice structure, and enhance descriptors used for subsequent visual image search. Existing methods for repetitive pattern detection are commonly based on determining the underlying lattice structures. However, these structures do not correspond directly to robust features that are scale- and rotation-invariant. This paper proposes a new lattice-support repetitive local feature (LS-RLF) detection method that aims to integrate lattice information into repeated local feature detection and extraction. The advantage of the proposed method is that the detected features can be directly used by current visual search technologies. The LS-RLF method estimates the undetected repeated features in the lattice structure using Hough transform-based feature estimation. Further, in order to handle the visual burstiness issue, a new LS-RLF based image retrieval framework is developed. Experiments performed on benchmark datasets show that the proposed method outperforms the state-of-the-art methods by mean Average Precisions (mAP) of 4.5%, 5.5% and 3.2% on Oxford, Paris, and INRIA holidays datasets respectively. This demonstrates the effectiveness of the proposed method in performing visual search for images which contain wide range of repeated patterns.

1. Introduction

The presence of repetitive patterns provides pleasing aesthetic to image, and are commonly found in many images. Repetitive patterns are frequently encountered in images from natural scene as well as man-made structures. For example, in the city landscape, images often contain repetitive patterns/structures such as building facades, windows, tiles, etc. Repeated structures can also present in background objects such as trees and bushes, which result in negative impact on the performance of visual recognition [15]. Sample images that contain repetitive patterns are shown in Fig. 1. In general, repetitive patterns are often considered as a challenge that hinders visual search and recognition in computer vision.

In recent year, local feature-based image representations such as Bag-of-Words (BoW) [8,21,22,25] have been widely used in image search systems. The BoW model finds wide applications in visual recognition tasks such as place and landmark recognition [2,3,26]. The model represents the local feature as discrete visual words (VWs), and then represents an image as a histogram of visual words. The visual words are computed by using unsupervised algorithms such as k-means clustering in high dimensional feature space. Variants of k-means like Approximate *K*-Means (AKM) [21] and Hierarchical *K*-Means (HKM) [19] have been proposed to achieve efficient computation of visual vocabulary. Eventually, local features from the database images are quantized using the visual vocabulary and indexed using inverted file structure. To retrieve similar images for a query image, database images are scored and ranked using weighting functions such as *term frequency-inverse document frequency* (tf-idf). To further enhance the performance, post-processing strategies such as geometric verification [21,25] and query expansion [5,6] are used to re-rank the images.

Nevertheless, the similarity scoring and geometric verification in the BoW framework are adversely affected by the presence of repetitive patterns in images. The repetitive patterns pose significant challenges to image matching due to the following observations:

(i) ***Violation of Feature Independence Assumption*** **:** The repeated patterns in images violate the i.i.d. (independent and identically distribution) assumption required by state-of-the-art algorithms in image matching, retrieval and reconstruction [7,13,24].

* Corresponding author.
E-mail addresses: dipu002@e.ntu.edu.sg, dips4717@gmail.com (D. Manandhar), ekhyap@ntu.edu.sg (K.-H. Yap), zwmiao@ntu.edu.sg (Z. Miao), elpchau@ntu.edu.sg (L.-P. Chau).

http://dx.doi.org/10.1016/j.patrec.2017.09.021
0167-8655/

Fig. 1. Sample images with different types of repetitive patterns in man-made structures and natural sceneries.

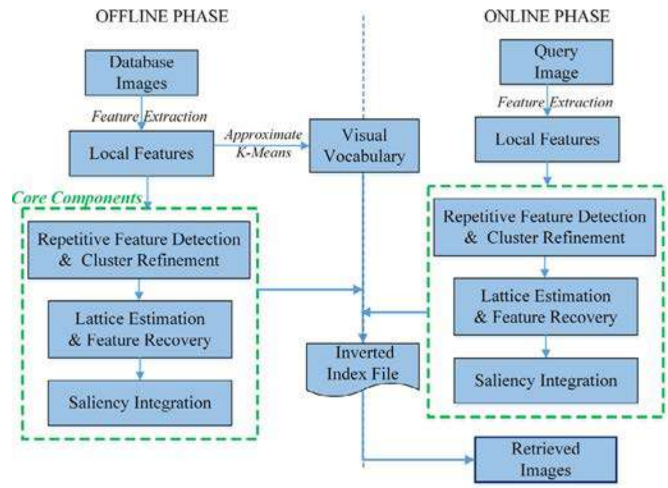


Fig. 2. Overview of the proposed LS-RLF framework in visual search.

Hence, the presence of repetitive patterns results in inferior retrieval performance of the algorithms.

(ii) ***Burstiness of Visual Words*** **:** Features from the same repetitive patterns are likely to be quantized to the same visual words, thus causing visual words from these repeated patterns to overwhelm the non-repeated ones. This phenomenon is known as visual burstiness [13]. The similarity score is hence dominated by visual words with large bin counts that correspond to repetitive features. The burstiness of visual words will often lead to incorrect retrievals and degrades performance of visual search systems.

(iii) ***Ambiguity for Geometric Matching*** **:** The spatial verification of images are performed on the putative correspondences established during visual word matching. Hence, multiple matches of features from the same repetitive patterns will cause ambiguity in determining the true correspondences during geometric verification.

To handle the above-mentioned challenges, detection of repeated patterns and incorporation of this information into the retrieval framework is needed. In recent years, a number of repetitive pattern detection methods have been studied. Leung et al.[16] used local patches to find similar elements in the neighborhood. Park et al. [20] and Hays et al. [11] modeled the repetitive patterns with deformed 2D lattice. They used normalized cross correlation (NCC) on lattice template and vectors to find the dominant lattice structure in the image. However, these lattice-based detection methods [11,16,20] are not able to handle different types of repetitive patterns. Further, not all the repeated patterns in images can be modeled using quadrilateral lattice elements. Most importantly, the detected lattice elements do not correspond to discriminative features that can be used to characterize the image for visual search/recognition directly.

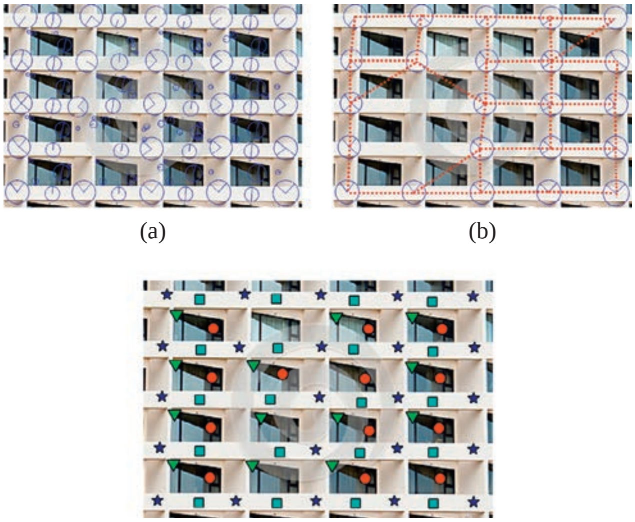
In image search and retrieval, repetitive building facades have been studied in [23] by measuring the similarity of motifs using NCC. Similarly, Doubek et al. [9] employed shift invariant descriptor for lattice representation. However, both Doubek et al. [9] and Schindler et al. [23] can only handle small dataset. This is because their methods do not produce discriminative features that are not scalable with respect to the dataset dimension.

For scalable retrieval, the study on the effect of repetitive patterns has been done in [13]. The authors different weighting functions such as square-root operator $\sqrt{(\cdot)}$, and $\log(\cdot)$ to down-weigh the contribution of repetitive features. However, the method does not exploit repetitive pattern detection. This leads to sub-optimal solution to the burstiness issue. Recent work on repttile method [26] used local features to detect the presence of repetitive patterns. The method relies on the top K nearest visual words to measure the feature similarity, and hence suffers from feature quantization errors. Moreover, Torii's repttile method only makes coarse estimation of repetitive patterns in images without utilizing the underlying lattice structure.

In view of this, this paper presents a new Lattice-Support Repetitive Local Feature (LS-RLF) detection method to detect repetitive patterns, estimate the underlying lattice structure, and extract descriptors that can used directly for visual image search. By using both geometric and descriptor information of local features, the proposed LS-RLF method is able to accurately detect different types of repetitive patterns at various scales. An advantage of this approach is that since the detected local features are invariant to scale change and rotation, the detected pattern can be used directly for image search and matching. Moreover, as the local descriptors can handle a certain amount of view, geometric and photometric variations, the proposed method is more robust than lattice patch approaches [9,11,20,23]. As compared to [26], the proposed LS-RLF method integrates both feature information and underlying lattice structure to estimate more accurate repetitive patterns. Undetected features in the patterns can also be recovered using Hough transform-based interpolation. The detected patterns can be readily used to address the visual burstiness issue. Since the bursty features from the image background such as trees, bushes, flora, etc. have negative impact on retrieval, feature saliency information is also extracted and incorporated into the framework to de-emphasize these background features. Experiments show that the proposed method can outperform the state-of-the-art method by an mAP of 4–5%.

2. Overview of the proposed method

The flowchart of the proposed LS-RLF framework is shown in Fig. 2. It consists of offline and online phases. In the offline phase, local features extracted from the database images are used for vocabulary construction and repetitive feature detection/extraction. The repetitiveness of features are analyzed based on their attributes, and the features are clustered using connected component analysis. As the initial clustering may suffer from misclassification due to variations in features of repeated patterns, cluster refinement is performed to improve the clustering outcome. Next, for each cluster, the repetitive properties and its underlying lattice structures are estimated. Missing feature in the lattice structures are identified, and recovered using Hough transformed-based interpolation. Finally, the saliency information are incorporated to emphasize the foreground regions of interest, and weaken the burstiness of repeated patterns in the backgrounds.



(a) (b) (c)

Fig. 3. Illustration of repetitive pattern detection and clustering. (a) Image with overlaid detected features. (b) Image with dominant features (nodes) and connections (edges) between them. (c) Connected features after analysis. Different labels/symbols represent different repetitive feature types.

During the online phase, query images are passed through the same process of feature detection, clustering refinement, and missing feature recovery. The detected feature of the query image is weighed by the saliency map and represented as weighted BoW histogram. Inverted index file is then employed to match the query with the database images. Images with top scores are returned by the system.

3. Lattice support repetitive local feature (LS-RLF) detection

3.1. Repetitive feature detection and clustering

Studies from repetitive features indicate that they share many common attributes and have similar scale, orientation and descriptor vector. These geometric as well as descriptor information can be used to measure the pairwise similarity between the features. In order to detect repetitive features and cluster them, we propose a set of similarity constraints (ζ) on the feature attributes as below:

1. Cosine similarity of central patch descriptors ($\Phi \in [0, 1]$)
2. Difference angle in orientations (θ *Radians*)
3. Ratio of relative scales (ρ)
4. Spatial proximity (L_2-distance) between features (Δ)

Following extensive empirical evaluation, the following thresholds on the constraint ζ are selected: $\Phi \geq 0.9, |\theta| \leq 0.5, (1/1.3) \leq \rho \leq 1.3, \Delta \leq 10 \cdot scale$.

We formulate repetitive feature detection using undirected graph where each feature represents a node in the graph. The edges are created if and only if the similarity between 2 nodes satisfy constraints on ζ. Fig. 3 illustrates the high-level idea of repetitive feature clustering. Fig. 3(a) shows an image of facades and detected features overlaid on it. First, we create the linkages between similar features (as shown in Fig 3(b)) which satisfy the pairwise similarity required by ζ. For clarity, only the connections between dominant features are shown. By performing the connectivity analysis, features are clustered into different groups as shown by different labels in Fig. 3(c). It can be noted that there are a number of missing features which were not detected by well-known feature detectors. These missing features can, however, be recovered using the proposed feature recovery technique in Section 3.3.

In this paper, SIFT descriptors are used to describe the local patches. As the descriptors from repeated pattern often experience some variations, direct use of the SIFT descriptors to measure the similarity between two repeated patches sometimes leads to erroneous results. We propose to measure the similarity between two patches by using the use the central 2×2 SIFT patches which correspond to $2 \times 2 \times 8 = 32$ dimensional feature descriptor. A further advantage of this similarity measure is that it is not prone to quantization error. In addition, the proposed LS-RLF method also uses orientation information to differentiate among various different patterns.

Formally, the graph is constructed by linking similar repetitive features with edges. To capture the pairwise similarity, a binary affinity matrix is computed where elements (edges) are set to 1 if and only if all the 4 requirements by the constraint ζ are fulfilled. Similar to [4,26], we extend connected component analysis to group similar features and obtain initial feature clustering. Usually, only few clusters contain significant number of features which represent corresponding patterns in the images whereas the rest are singleton clusters which represent unique features. Thus, only few clusters are analyzed for further processing and hence the computational overhead is greatly reduced.

3.2. Cluster refinement

Repetitive patterns in real-world images may experience variations due to difference in settings such as local occlusion, reflective surface, and viewpoint changes. As a result, extracted feature descriptors from the same repetitive patterns may be grouped into different clusters. To address this issue, the proposed LS-RLF method develops a cluster refinement strategy to improve the initial clustering described in Section 3.

The clustering of repetitive features is affected by local variations in image. The use of a stringent constraint ζ will differentiate different types of repetitive patterns, but may split a class of repetitive feature into different clusters. Therefore, to reduce erroneous clustering, a stringent constraint (ζ) is first employed to achieve the initial clustering. Next, to merge the clusters containing same type of repetitive features, the similarities between the clusters are measured using inter-intra class distance ratio. The intra-class distance provides a measure of cluster tightness whereas inter-class distance measures the separation of clusters in the feature space. Hence, the similarity between two clusters c_i and c_j can be using inter-intra class distance ratio (R_{ij}) defined in (1),

$$R_{ij} = \frac{\|\bar{d}_i - \bar{d}_j\|_2^2}{\frac{1}{n_i n_j} \Sigma_{l=1}^{n_i} \|d_i^{(l)} - \bar{d}_i\|_2 \, \Sigma_{m=1}^{n_j} \|d_j^{(l)} - \bar{d}_j\|_2} \tag{1}$$

where $\bar{d}_i$ and $\bar{d}_j$ are the mean descriptors of clusters c_i and c_j, respectively and n_i and n_j are their respective cardinalities. Two clusters with ratio $R_{ij} < 1$ implies that they are similar in the descriptor space and therefore are merged into a single cluster. This refinement strategy addresses the mis-classification of initial clustering.

3.3. Lattice estimation and feature recovery

In this section, we aim to estimate underlying lattice structures of repeated local features. The spatial occurrence of these repeated features is formulated with parametric models. In particular, Hough-transform is used for determining the parameters of different patterns and hence their lattice structures. The presence of large bins in Hough space refers to specific patterns in the image. Moreover, the method also aims at recovering the missing (undetected) features in the pattern. The lattice estimation and feature recovery technique in LS-RLF framework is summarized in Algorithm 1.

Algorithm 1 Lattice Estimation & Feature Recovery.

```
Input: Feature location (x_i, y_i) of repeated patterns
Output: Lattice pattern Λ^k, recovered features ϒ
for all Clusters c do
    Get repeated feature locations (x_i, y_i)
    Hough Space Mapping: (x_i, y_i) → (a_i, b_i)
    Find the large bins {B_k} in Hough accumulator cells.
    Compute hough-bin-pixels {P_k} for {B_k}
    for all P_k = {p_1^k, p_2^k, ..., p_n^k} do
        Estimate repetition interval Λ^k          ▷ Lattice Pattern
        if Interval (p_i^k, p_{i+1}^k) ≈ m · Λ^k then
            Interpolate locations κ               ▷ Missing Feature
        end if
    end for
    Compute descriptors ϒ on locations κ
    if ϒ is valid then                            ▷ Repetitive Consistency Check
        Add ϒ to the cluster                      ▷ Recovered Valid Features
    end if
end for
```

Local feature locations are used to estimate the underlying lattice structures in each cluster c. First, the locations (x, y) in the image space are mapped to the Hough parametric space using Hough transform. Large bin counts $\{B_k\}$ in Hough accumulator cells indicate presence of a specific structure in the image. To find the corresponding set of points $\{P_k\}$, bins with large counts $\{B_k\}$ are back-projected to the image space. Specifically, points $P_k = \left\{p_1^k, p_2^k, \cdots, p_n^k\right\}$ in the pattern repeat with an interval Λ. The undetected features in the image can then be predicted if the interval between two consecutive detected points is multiples of the interval Λ. Next, missing feature locations κ are interpolated where new descriptors Υ are computed to extract the undetected local repeated features. Finally, to ensure the repetitiveness consistency, recovered features Υ are validated by descriptor similarity using cosine measure.

3.4. Saliency integration

A common source of repetitive patterns is due to backgrounds such as vegetation, sky, and water surfaces etc. To de-emphasize the contribution of these background repeated patterns, we incorporate saliency information into the proposed LS-RLF framework. In recent years, several visual saliency detection approaches have been proposed to mimic human visual system. We extend and incorporate the graph-based visual saliency (GBVS) [10] into this work. GBVS estimates a saliency map S using graph in which the directed edge from a point (i, j) to point (p, q) has weight

$$e((i,j),(p,q)) = \left|\log\frac{M(i,j)}{M(p,q)}\right| \exp\left(-\frac{(i-p)^2+(j-q)^2}{2\sigma^2}\right), \quad (2)$$

where M is the feature map obtained using biologically inspired filters and σ is a free parameter. An activation map is generated by eigenvector computation based on graph theory, which is normalized and combined to obtain the final saliency map S. Fig. 4 shows the local features and the saliency map overlaid on an image. Highlighted orange and yellow regions indicate informative foreground regions with high visual saliency. Bursty features from the background are suppressed while those from foreground region of interest are boosted.

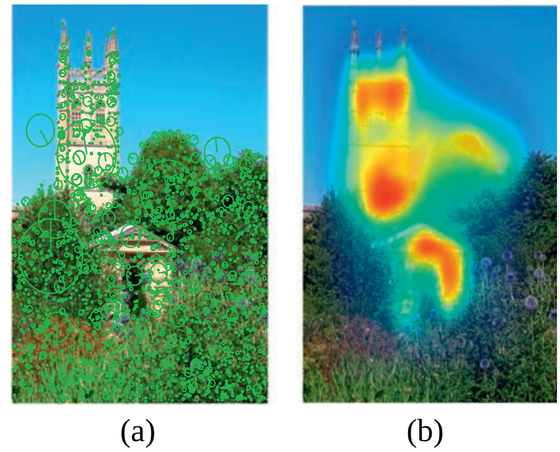


Fig. 4. (a) Detected features and (b) Saliency Map overlaid on the image.

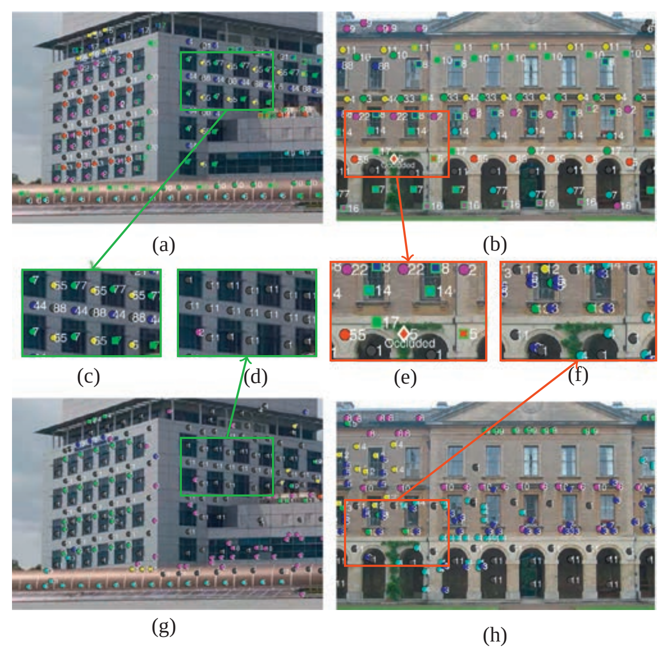


Fig. 5. Repetitive pattern detection on different images. Images on the top row (a) & (b) are obtained using the proposed LS-RLF method, and the bottom row (g) & (h) are based on Torii's method. Different groups of repetitive patterns are shown with different colors and annotated labels. Undetected but recovered new features are shown as square keypoints. See Section 4 for more explanations. Best viewed in color with 2× zoom. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

4. Qualitative analysis of LS-RLF

This section presents the qualitative results and analysis on repetitive pattern detection by the proposed LS-RLF method and compares it with the state-of-the-art repttile method in [26]. The comparison is shown in Fig. 5. The detection results obtained by the proposed method are given in the first row of Fig. 5(a) & (b). The proposed method is able to detect and differentiate various structures like facades, windows and motifs. Further, it utilizes feature recovery to estimate the missing features (shown as square keypoints). Compared to this, the detection results obtained using Torii's repttile method in Fig. 5(g) & (h) contain numerous clustering errors. As seen from the zoom-in area in Fig. 5(d), the method mis-classify different types of repetitive patterns into a single class (label 1). In contrast, each type of elements like facades, windows are correctly clustered in our proposed method in Fig. 5(c). Similarly, in Fig. 5(e), different structures from walls and gates are correctly differentiated and clustered into different pattern types (labels 1, 2 & 5). However, Torii's repttile method grouped them as same pattern type (label 1) in Fig. 5(f).

The proposed method can also estimate occluded repeated features using Hough transform-based feature interpolation. An example can be seen in Fig. 5(e), where an occluded feature is shown in white diamond (label 5). It can be seen from the results that the proposed LS-RLF method is more robust and discriminative in estimating the lattice pattern and cluster structures. This will result in improvement in the subsequent image recognition and retrieval.

5. LS-RLF based image matching and retrieval

Conventional BoW method is sensitive to the presence of repetitive patterns as the BoW histogram suffers from visual word burstiness. This often results in poor retrieval performance due to overwhelming false matches on visual words from repetitive patterns. In this paper, we propose to use the LS-RLF framework to address the burstiness issue in the BoW model.

Based on a vocabulary with z visual words, an image is represented as a BoW histogram as

$$I = (w_1, w_2, \cdots, w_i, \cdots w_z)^\intercal, \tag{3}$$

where w_i is the proposed weighting of i^{th} visual word in the image.

Using the tf-idf weighting strategy, the similarity score between a query image $Q = \{q_i\}_{i=1}^{z}$ and a database image $I = \{w_i\}_{i=1}^{z}$ is calculated using cosine angle similarity between the vectors,

$$Score(I, Q) = \frac{\sum_{i=1}^{z} w_i \cdot q_i \cdot idf(i)^2}{\|I\|_2 \, \|Q\|_2}, \tag{4}$$

where idf is the inverse document frequency of visual words.

For an image with local features $\{f_i\}_{i=1}^{Np}$, each feature is quantized and soft-assigned to $K = 3$ nearest visual words v^k with $k = \{1, 2, 3\}$. The i^{th} visual word weighting w_i of the BoW representation in (3) are determined using

$$w_i = \sum_{j=1}^{Np} \sum_{k=1}^{K} w_R \, w_S \, w_{SA}(k) \cdot 1[v_j^{(k)} = i], \tag{5}$$

where $1[v_j^{(k)} = i]$ is the indicator function that is equal to 1 if the k^{th} nearest neighbor of descriptor d_j is quantized to visual word index i. The proposed weighting w_i integrates 3 factors in its formulation: repetitiveness score (w_R), saliency weighting (w_S) and soft-assignment weighting (w_{SA}). The significances of these factors are explained as follows:

(i) Repetitiveness score (w_R)

The information from the detected repetitive patterns is incorporated into image representation and hence in image scoring. The cluster cardinality (n) of the detected features is an indicator of feature repetitiveness. The bigger the cluster, the more repetitive the feature is. Each feature f_i in the image is assigned a repetitiveness score $w_R \in (0, 1]$ based on the sigmoidal function in (6).

$$w_R = \frac{s}{1 + \exp(\alpha(n - \mu))} \tag{6}$$

The parameter s is the scaling factor, and parameters α and μ are positive real numbers that controls the decay and shift of the sigmoid function.

Fig. 6 shows the repetitiveness function which provides different weights to features with different repetitiveness. As n increases, the repetitiveness score decreases. The repetitiveness score ensures features with high repetitiveness are de-emphasized so that they do not overwhelm the BoW histogram. This will alleviate the burstiness issue due to the repeated patterns. The values

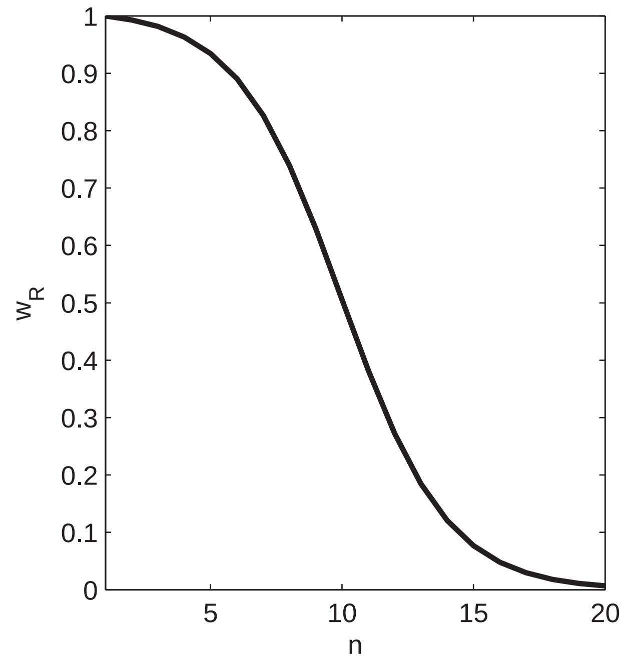


Fig. 6. Repetitiveness weighting function with $\alpha = 0.5$ and $\mu = 10$.

of the parameters α and μ in (6) are experimentally set to 0.5 and 10 respectively.

(ii) Saliency weighting (w_S)

Saliency weighting is integrated into the overall weighting to emphasize the salient foreground object of interest, and de-emphasize the background bursty features such as trees, vegetations etc. The factor $w_S \in [0, 1]$ is the saliency weight extracted from the saliency map $S \in [0, 1]$ using GBVS as explained in Section 3.4. To obtain the weights w_S from the map S, a gamma transformation is used for saliency value calibration i.e. $w_S = S(x, y)^\gamma$, where (x, y) is the feature location. For all the experiments, the value of γ is empirically set to 0.8 as it has been shown to achieve good performance.

(iii) Soft-assignment strategy (w_{SA})

We extend the soft quantization strategy in [14]. The factor w_{SA} is set to $1/(2^{k-1})$ which represents the k^{th} soft-assigned weight. This provides geometrically decreasing weights with respect to k and helps to reduce the impact of quantization loss.

6. Experiments

6.1. Dataset and evaluation

The proposed LS-RLF method is used to perform image retrieval and evaluated using three standard benchmark datasets: Oxford dataset [21], Paris dataset [22], and INRIA holidays dataset [12]. The Oxford dataset contains 5062 database images which are annotated with ground-truth for 55 query images from 11 distinct Oxford landmarks. This dataset is challenging as images are taken under significant variations of illumination, scale and viewpoint. The Paris dataset contains 6412 images of various Paris landmarks which are also annotated with labels for 55 query images. The INRIA Holidays dataset is a set of personal holiday images which include a large variety of scenes with 500 query images in total. We use mean Average Precision (mAP) to evaluate the retrieval performance.

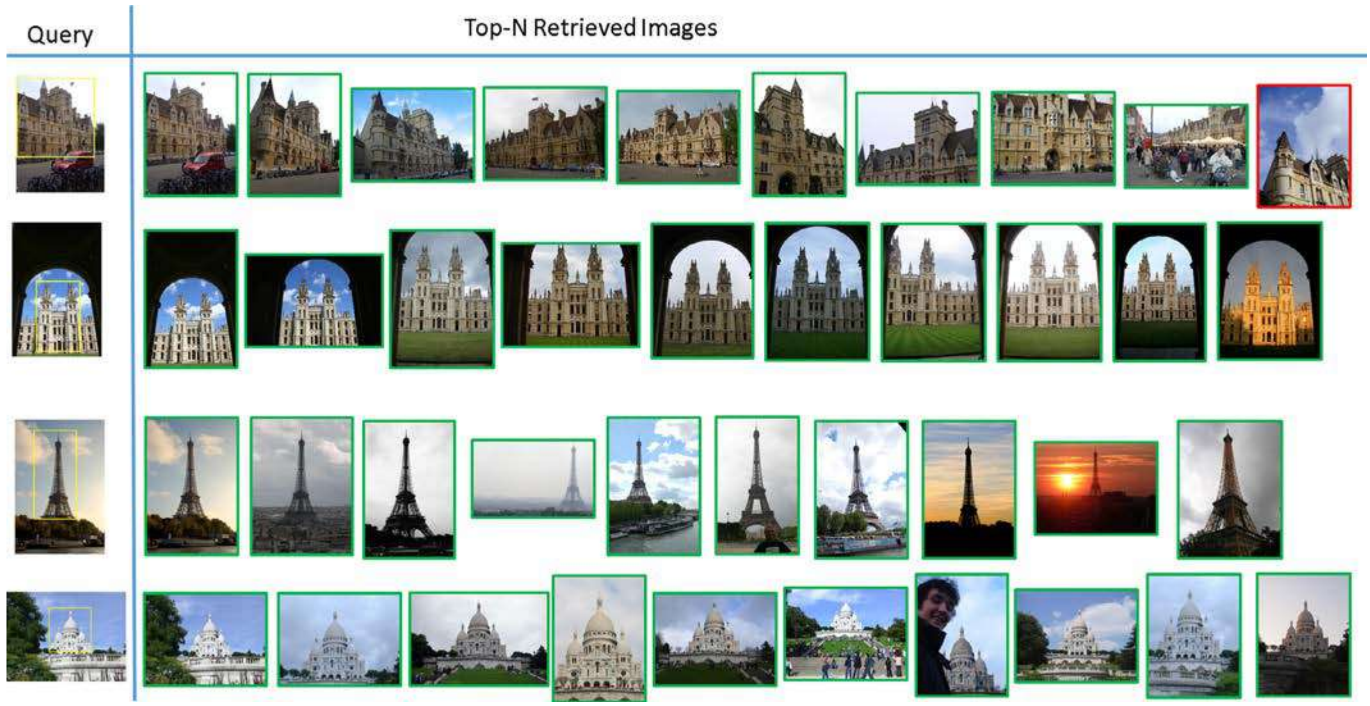


Fig. 7. Qualitative retrieval results using the proposed LS-RLF framework. Left-most column shows query images with the bounding box and the corresponding rows show the top 10 retrieved images. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 1
mean Average Precision (mAP) on the Oxford and Paris Datasets.

	Oxford Dataset		Paris Dataset		INRIA Holidays	
Method	200K VV	1M VV	200K VV	1M VV	200K VV	1M VV
TF-IDF [21]	57.74	63.65	70.44	72.56	66.9	78.3
BRST-IDF [13]	63.09	68.86	72.14	73.24	68.1	78.7
Repttile Method [26]	67.72	71.60	74.10	74.64	72.0	80.9
Proposed LS-RLF Framework	**70.25**	7**6.06**	**78.63**	**80.11**	**75.9**	**84.1**

6.2. Experimental settings

We use Difference of Gaussians (DoG) function to extract the interest points which are then wrapped by affine elliptical regions resulting in DoG-Affine detector. The affine regions are the described by 128 dimensional SIFT descriptor [17] followed by *Root*SIFT [1] normalization. We run experiments with visual vocabulary (VV) of different sizes. In particular, visual vocabularies with 1M and 200K visual words are constructed with approximate K-means using FLANN library [18]. An inverted index file is constructed using the visual vocabulary where database images and their respective features are indexed. We use the same settings for all experiments.

The algorithms are implemented using MATLAB and VLFeat library [27] and tested on a computer with i5-4570 CPU and 32GB memory. The larger the number of features, the longer is the computational time. It takes on average 1.8*sec* to query an image with approximately 1500 detected features. The computational time can be further reduced with implementation using compiling language such as C with code optimization.

6.3. Results and discussion

The performance of the proposed LS-RLF framework is compared with the standard retrieval method [21], burstiness weighting (BRST-IDF) [13] and Torii's repptile method [26]. The mAPs obtained using these methods with 200K and 1M visual vocabularies are compared with those obtained by the proposed LS-RLF are summarized in Table 1.

For the Oxford dataset, the proposed method with 1M VV achieves an mAP of 76% which outperforms the Philbin's method by 12%, Jegou's burstiness weighting by 7% and Torii's repptile method by 4.5%. This clearly shows the effectiveness of the proposed method. It is also observed that for all the methods, retrieval with 1M vocabulary consistently outperforms that with 200K. It is because larger vocabularies are more expressive compared to the smaller ones. The image representations with fine grained visual words are more discriminative and hence achieve better search performance.

Similar trend of retrieval performance can be observed for the Paris dataset. The proposed LS-RLF based framework with 1M VV achieves an mAP of 80.1% which outperforms the Philbin's method by 8%, Jegou's burstiness weighting by 7% and Torii's repttile method by 5.5%. For Holidays dataset, it is observed our proposed method outperforms other methods for both 200K and 1M VVs. The proposed method achieves an mAP of 75.9% and 84.1% for 200K VV and 1M VV, respectively. These results outperform the Torris Repttile method by 3.2–3.9%, Jegous BRST-IDF method by 5.4–7.8% and Philbins method by 5.8–9.0%.

The designed weighting strategy is able obtain balanced BoW histograms which allow unbiased similarity measure between images. The mAP obtained using the proposed method clearly show superior performance when compared with other methods. Fig 7 shows qualitative retrieval results using the proposed LS-

RLF framework. The query images with the search bounding box are given on the left column, while the top 10 retrieved images are shown in the corresponding rows in the right. Images with green outline are true positives (with respect to the ground truth), whereas those with red border is false positive. From the figures, it is clear that proposed method can offer good retrieval performances.

7. Conclusion

A new LS-RLF method is proposed in this paper. The proposed method integrates local features and underlying lattice structure to detect repetitive patterns. A new feature interpolation technique is also presented to estimate and recover undetected or missing features. The key advantage of the LS-RLF framework is that the detected repetitive features can be used directly in conjunction with existing visual search technologies. In addition, the proposed method is able to alleviate the burstiness issue in BoW-based image search. The experimental results clearly show the effectiveness of the proposed method.

Acknowledgment

This research was carried out at the Rapid-Rich Object Search (ROSE) Lab at the Nanyang Technological University, Singapore. The ROSE Lab is supported by the Infocomm Media Development Authority, Singapore.

References

[1] R. Arandjelovi, A. Zisserman, Three things everyone should know to improve object retrieval, in: IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 2911–2918, doi:10.1109/CVPR.2012.6248018.

[2] T. Chen, K.H. Yap, Context-aware discriminative vocabulary learning for mobile landmark recognition, IEEE Trans. Circuits Syst. Video Technol. 23 (9) (2013) 1611–1621, doi:10.1109/TCSVT.2013.2254978.

[3] T. Chen, K.H. Yap, L.P. Chau, Integrated content and context analysis for mobile landmark recognition, IEEE Trans. Circuits Syst. Video Technol. 21 (10) (2011) 1476–1486, doi:10.1109/TCSVT.2011.2161413.

[4] O. Chum, J. Matas, Unsupervised discovery of co-occurrence in sparse high dimensional data, in: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, 2010, pp. 3416–3423, doi:10.1109/CVPR.2010.5539997.

[5] O. Chum, A. Mikulk, M. Perdoch, J. Matas, Total recall ii: query expansion revisited, in: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, 2011, pp. 889–896, doi:10.1109/CVPR.2011.5995601.

[6] O. Chum, J. Philbin, J. Sivic, M. Isard, A. Zisserman, Total recall: Automatic query expansion with a generative feature model for object retrieval, in: 2007 IEEE 11th International Conference on Computer Vision, 2007, pp. 1–8, doi:10.1109/ICCV.2007.4408891.

[7] R.G. Cinbis, J. Verbeek, C. Schmid, Image categorization using fisher kernels of non-iid image models, in: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, 2012, pp. 2184–2191, doi:10.1109/CVPR.2012.6247926.

[8] G. Csurka, C. Dance, L. Fan, J. Willamowski, C. Bray, Visual categorization with bags of keypoints, in: Workshop on statistical learning in computer vision, ECCV, 1, Prague, 2004, pp. 1–2.

[9] P. Doubek, J. Matas, M. Perdoch, O. Chum, Image matching and retrieval by repetitive patterns, in: 20th International Conference on Pattern Recognition, 2010, pp. 3195–3198, doi:10.1109/ICPR.2010.782.

[10] J. Harel, C. Koch, P. Perona, Graph-based visual saliency, in: B. Schölkopf, J.C. Platt, T. Hoffman (Eds.), Advances in Neural Information Processing Systems 19, MIT Press, 2007, pp. 545–552.

[11] J. Hays, M. Leordeanu, A.A. Efros, Y. Liu, Discovering texture regularity as a higher-order correspondence problem, in: European Conference on Computer Vision (ECCV), Springer, 2006, pp. 522–535.

[12] H. Jegou, M. Douze, C. Schmid, Hamming embedding and weak geometric consistency for large scale image search, Comput. Vis. (2008) 304–317. ECCV 2008

[13] H. Jegou, M. Douze, C. Schmid, On the burstiness of visual elements, in: IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 1169–1176, doi:10.1109/CVPR.2009.5206609.

[14] Y.-G. Jiang, C.-W. Ngo, J. Yang, Towards optimal bag-of-features for object categorization and semantic video retrieval, in: Proceedings of the 6th ACM International Conference on Image and Video Retrieval, ACM, 2007, pp. 494–501.

[15] J. Knopp, J. Sivic, T. Pajdla, Avoiding confusing features in place recognition, in: European Conference on Computer Vision (ECCV), Springer, 2010, pp. 748–761.

[16] T. Leung, J. Malik, Detecting, localizing and grouping repeated scene elements from an image, in: European Conference on Computer Vision (ECCV), Springer, 1996, pp. 546–555.

[17] D.G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2) (2004) 91–110, doi:10.1023/B:VISI.0000029664.99615.94.

[18] M. Muja, D.G. Lowe, Fast approximate nearest neighbors with automatic algorithm configuration, VISAPP (1) 2 (331–340) (2009) 2.

[19] D. Nister, H. Stewenius, Scalable recognition with a vocabulary tree, in: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), 2, 2006, pp. 2161–2168, doi:10.1109/CVPR.2006.264.

[20] M. Park, K. Brocklehurst, R.T. Collins, Y. Liu, Deformed lattice detection in real-world images using mean-shift belief propagation, IEEE Trans. Pattern Anal. Mach. Intell. 31 (10) (2009) 1804–1816, doi:10.1109/TPAMI.2009.73.

[21] J. Philbin, O. Chum, M. Isard, J. Sivic, A. Zisserman, Object retrieval with large vocabularies and fast spatial matching, in: IEEE Conference on Computer Vision and Pattern Recognition, 2007, pp. 1–8, doi:10.1109/CVPR.2007.383172.

[22] J. Philbin, O. Chum, M. Isard, J. Sivic, A. Zisserman, Lost in quantization: improving particular object retrieval in large scale image databases, in: IEEE Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8, doi:10.1109/CVPR.2008.4587635.

[23] G. Schindler, P. Krishnamurthy, R. Lublinerman, Y. Liu, F. Dellaert, Detecting and matching repeated patterns for automatic geo-tagging in urban environments, in: IEEE Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–7, doi:10.1109/CVPR.2008.4587461.

[24] J.L. Schnberger, F. Radenovi, O. Chum, J.M. Frahm, From single image query to detailed 3d reconstruction, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 5126–5134, doi:10.1109/CVPR.2015.7299148.

[25] J. Sivic, A. Zisserman, Video google: a text retrieval approach to object matching in videos, in: Ninth IEEE International Conference on Computer Vision, 2003, pp. 1470–1477vol.2, doi:10.1109/ICCV.2003.1238663.

[26] A. Torii, J. Sivic, M. Okutomi, T. Pajdla, Visual place recognition with repetitive structures, IEEE Trans. Pattern Anal. Mach. Intell. 37 (11) (2015) 2346–2359, doi:10.1109/TPAMI.2015.2409868.

[27] A. Vedaldi, B. Fulkerson, Vlfeat: An open and portable library of computer vision algorithms, in: Proceedings of the 18th ACM International Conference on Multimedia, in: MM '10, ACM, New York, NY, USA, 2010, pp. 1469–1472, doi:10.1145/1873951.1874249.