Interwoven texture-based description of interest points in images

Ghahremani, Morteza; Zhao, Yitian; Tiddeman, Bernard; Liu, Yonghuai

Published in:
Pattern Recognition

DOI:
10.1016/j.patcog.2021.107821

Publication date:
2021

Citation for published version (APA):
Ghahremani, M., Zhao, Y., Tiddeman, B., & Liu, Y. (2021). Interwoven texture-based description of interest points in images. *Pattern Recognition*, *113*, Article 107821. https://doi.org/10.1016/j.patcog.2021.107821

Interwoven Texture-Based Description of Interest Points in Images

Morteza Ghahremani[a], Yitian Zhao[b], Bernard Tiddeman[a] and Yonghuai Liu[c,*]

[a]Department of Computer Science, Aberystwyth University, Ceredigion, United Kingdom
[b]Cixi Instuitue of Biomedical Engineering, Ningbo Institute of Industrial Technology, Chinese Academy of Sciences, Ningbo, China
[c]Department of Computer Science, Edge Hill University, Lancashire, United Kingdom

ARTICLE INFO

Keywords:
Interest point
Descriptive signature
Interwoven texture
Locality
Globality
Robustness

ABSTRACT

Local feature description is to assign a unique signature to a key-point such that it becomes distinctive from the others regardless of changes in viewpoint, illumination, rotation, scale as well as distortions and noise. This paper proposes a novel approach to construct such a descriptor. For preserving both homogeneous and heterogeneous features of a given support region, we interweave the texture information so that the key-point is more likely to be assigned a distinctive signature and neighboring key-points will be less likely to share the same texture information. The main idea behind our descriptor is to increase the areas of our observations in the given scene while the length of the local support region is fixed. Gradient magnitude and divergence, as measurement parameters of texture information, are applied to a group of pixels instead of employing a pixel-wise strategy that make the descriptor more resistant to noise, distortions and illumination variation. The required storage of the proposed descriptor is just 72 floats and its computational complexity is much lower than those of existing ones. A comparative study between the proposed method and the selected state-of-the-art ones over multiple publicly accessible datasets with different characteristics shows its superiority, robustness and computational efficiency under various geometric changes, illumination variation, distortions and noise. The code and supplementary materials can be found at https://github.com/mogvision/InterTex-Feature-Descriptor.

1. Introduction

Thanks to technological development, we are currently facing a large volume of images that need to be processed and evaluated automatically with little human intervention. The link between computer vision and other disciplines is narrowing, and computer vision is highly in demand in a large number of applications from 2D image processing [1] to 3D including simultaneous localization and mapping (SLAM) [2], structure from motion (SfM) [3], robotics [4], etc.

Feature detection and feature description are the preliminary steps of many computer vision applications. As their names reflect their functionalities, a local feature detector aims to detect distinctive key-points/interest points/feature

*Corresponding author.
mog9@aber.ac.uk (M. Ghahremani); yitian.zhao@nimte.ac.cn (Y. Zhao); bpt@aber.ac.uk (B. Tiddeman); yonghuai.liu@edgehill.ac.uk (Y. Liu)
ORCID(s):

points in images and they are then assigned signatures by a local feature descriptor [5; 6]. The signatures help to find corresponding regions/points in different images as required by the subsequent tasks such as SLAM [2] and SfM [3]. Developing fast descriptors with a high match-rate is highly demanding. This becomes more challenging when we deal with smooth/texture-less images or photometric and geometric changes: such complicated images usually drastically degrade the performance of feature descriptors.

Plenty of feature descriptors have been proposed over the last two decades. Based on two criteria whether the extracted features are informed and training data is required, they can be divided into hand-crafted and learning-based techniques. The handcrafted feature descriptors can be broadly categorized into three classes: (*i*) binary descriptors, which rely on patch-wise comparison of adjacent pixels in intensity and binary encoding their difference; (*ii*) distribution-based representations, which encode the local regions using the distributions of intensities, gradients, etc of their pixels; and (*iii*) hybrid techniques, which aim to benefit from well-structured descriptors and improve them via appropriate coding algorithms [7; 8].

Binary representations aim at minimizing the computational time and the required storage. Though it is impossible to enumerate all of them, we refer to the well-structured ones, including binary robust independent elementary features (BRIEF) [9], fast retina key-point (FREAK) [10], and local binary pattern (LBP) and its variants [11]. Compared to the binary descriptors, the distribution-based feature descriptors generally require high computational time and storage, while their performance and reliability are much higher. Scale invariant feature transform (SIFT) [5], speeded up robust features (SURF) [12] and gradient local orientation histogram (GLOH) [13] (and its improved version [14]) are probably the most representative methods of the second category. Detailed surveys on the handcrafted feature descriptors are reported in [15; 16].

Traditional learning methods [17; 7; 18; 19] learn from low level features like gradient, intensity, etc. Recently, convolutional neural network (CNN) learning-based descriptors have attracted intensive attention [20; 21; 22]. In contrast to the traditional learning techniques, CNN-based descriptors learn directly from the raw image patches. Recent deep learning-based descriptors [20; 22; 23; 24; 25] show improvements and robustness to changes in viewpoints, scales and distortions [26]. Even though they produce impressive results over similar training and testing images, they usually degrade their performances significantly over databases with different characteristics. It is commonly challenging to collect a large set of representative examples for training. The requirement of training data normally limits the applicability of learning based descriptors in the real world. A comprehensive comparative evaluation of the learning and hand-crafted descriptors is reported in [27; 28; 29]. Other interesting feature descriptors can be found in [30; 31; 32]. Determining which feature descriptor or which category of aforementioned feature descriptors works well for a particular application is usually a complicated task.

Despite the vast literature on feature descriptors, few of them are both highly computationally efficient and reliable

and can always achieve the best performance simultaneously across datasets with different characteristics. In this paper, we are seeking to assign each key-point a unique signature even though the key-points are so close to each other; it is obvious that adjacent key-points usually share a common area of pixels and this increases the ambiguity when matching their representations. To this end, firstly we develop three basic principles that a robust descriptor should follow and then propose a new strategy to design a robust descriptor. Specifically, we consider a local window/region around the interest point and then divide it into 36 bins of length 8 pixels in such a way that the total size of the window is just 28-by-28 pixels. Any two adjacent bins are interwoven by 50% in space, where all the pixels used for the representation of each bin in the interwoven area are different from those for the representation of its adjacent ones. Interweaving adjacent bins aims at mixing smooth bins with the textured ones. This increases the area of coverage and the probability of having unique texture information inside each bin and subsequently improves the expressiveness of the description. Bins are encoded by gradient magnitude and divergence operators, which provide complementary texture information for the bins. In order to make sure that these parameters are powerful and stable in encoding local image regions, we propose a global-wise analysis instead of using traditional pixel-wise strategy. To further deal with noise, illumination change and distortions and emphasise the nearby pixels of the key point, the measurements are then weighted by Gaussian functions of their distances and finally normalized using the Hellinger kernel. The storage required for representing each key-point is 72 floats only. Henceforth, for the sake of simplicity we abbreviate our interwoven texture-based descriptor as 'InterTex'. According to the experimental results over several publicly accessible datasets and a comparative study between the proposed and the selected state-of-the-art feature descriptors, InterTex is the fastest one that can successfully deal with noisy, distorted, illumination varied and texture-limited images.

The rest of this paper is structured as follows. In the next section, we critically review feature descriptors and their pros and cons, and meanwhile establish our principles for constructing a robust descriptor. The proposed InterTex is detailed in Section 3. Section 4 reports a series of quantitative and qualitative experimental results. Finally, Section 5 draws conclusions and points out possible future work.

2. Related Work

Although the proposed feature descriptor relies on texture information and several texture-based feature descriptors have been proposed [33; 34; 25; 35; 36; 26; 37; 38; 39], we critically review the concepts and methodologies of all the categories of feature descriptors and meanwhile establish our framework. The goal is to encode the region around a key-point in such a way that it is robust to imaging noise, illumination change and distortions and there is low similarity between the representations of adjacent key-points in an image. To this end, two concerns need to be considered: (*i*) The size and structure of partitioning windows and (*ii*) The employed measurement types.

A high density of key-points generally influences the homogeneity and heterogeneity of the region of interest. A small support region cannot be distinctly recognized as such regions increase the risk of homogeneity. More precisely, the probability of having similar textures is increased by narrowing the applied partitioning windows. This incident is more observable for smooth data, where the level of distinctive information is relatively low. A wider support region can reduce the homogeneity and subsequently enhances heterogeneity. Though considering a large support region is usually favorable, it introduces two issues: (*i*) It will be shared with more adjacent key-points and (*ii*) A larger proportion of this region will be shared among the adjacent key-points. In either case, the discriminability of the representation is reduced, yielding a smaller number of matches. This is depicted in Fig. 1. In this experiment, we plotted the average performance of benchmark SIFT for various sizes of the support region in more than 500 images. Reduction of the support region by 50% leads to more than 20% false match-rate. Reducing the support regions tends to increase homogeneity and in this case, even an increase in measurement parameters may not markedly help the discriminability of the descriptions as small regions usually do not contain sufficiently distinctive texture information.

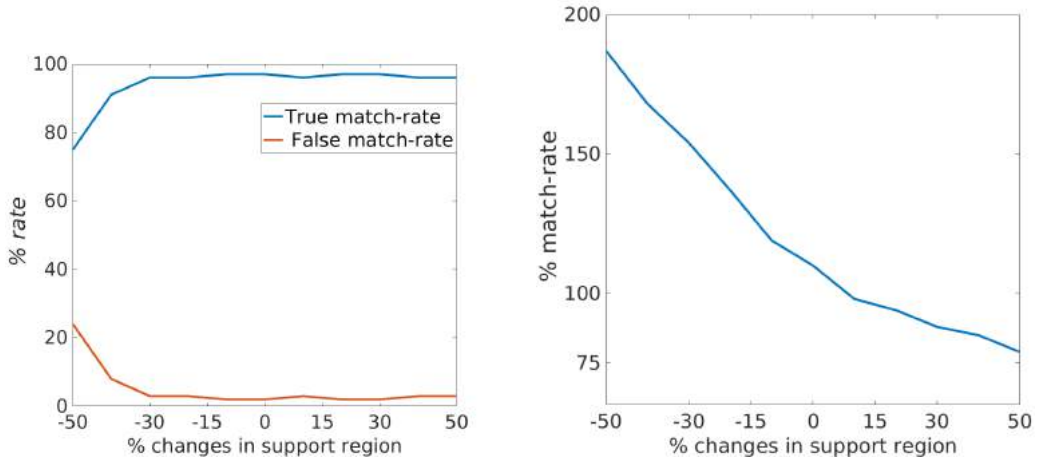
By widening the support region, the discriminability of the regions will be increased. As seen in the figure, the wrong matches are reduced from more than 20% to below 5% by widening the support region. However, continually enlarging the support region not only doesn't improve the true match-rate but also reduces the overall match-rate [Fig. 1(b)].

- Principle I: The support region around a key-point should not be too small in which case the representation is not discriminative, nor should it be too large in which case we lose the concept of the locality and the uniqueness of the region.

Now, we discuss the types of measurements and their characteristics. Generally, the measurements could be:

- Comparison of the intensity values of patches which are dominantly used by binary descriptors;
- Statistical analyses like histogram distribution of pixels used in [5; 14; 19] and higher-order statistical models developed by Koniusz *et al.* [7]; and
- Parameters/functions like gradient orientation, gradient magnitude and filter banks [8; 10; 12; 13].

In the construction of a feature descriptor, there is interest in the overall expressive information from a region rather than the exact behaviour of its pixels. Pixel-wise analysis of a region provides diverse features for the given region and the representation is often dominated by few pixels whose intensity values are quite different. Seidenari *et al.* [40] proposed pyramidizing descriptors by varying the size and density of pooling regions. A similar strategy can be seen in CNN-based learning techniques. Extracting features at multiple levels via pyramidizing preserves stable features of



(a) True and false match-rates (b) Overall match-rate

Figure 1: Performance of SIFT under different changes in the support region (positive values stand for widening and negative values for narrowing). In (b), overall match-rate is the combination of all true and wrong matches. SIFT at its original support region (0% change) is considered as a base value with 100% match-rate for the results of other sizes of the support region.

a region; however, multi-scale decomposition requires high computational time and increases implementation complexity. Group-wise analysis is another alternative to cope with the fragility of pixel-wise analysis. It can effectively extract the stable and dominant features of a region. This trend can be seen in binary methods. They often take small patches around the pixels of interest and then compare the patches' intensities rather than their pixels' themselves.

- Principle II: A feature descriptor should analyze features of a group of pixels rather than that of each individual pixel.

Feature descriptors must also be robust. Robustness for a feature descriptor can be defined as "being resistant to any unwanted/undesirable factors including noise, illumination change, any types of distortions and projection error." Images are often captured outdoors and contain considerable noise, illumination change and distortions. It is necessary to minimize their side effects. The types of measurements can determine their robustness. Intensity values are highly sensitive to changes in illumination and rotations [41]. Orientation-based measurements are usually susceptible to noise and distortions. This is more evident in smooth regions, wherein a small level of noise or distortions can severely affect the measurements. Robustness will be further discussed in Section 3.2 below.

- Principle III: A feature descriptor should be robust as much as possible to undesirable external factors introduced in the imaging and storing process.

More studies on feature descriptors can be found in [15; 26; 27; 28; 42].

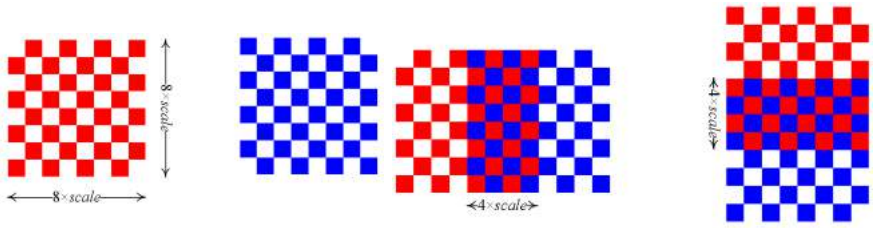
3. Proposed Interwoven Texture-Based Feature Descriptor (InterTex)

The proposed descriptor will be presented in three parts: (*i*) Design of an interwoven partitioning window; (*ii*) Measurement parameters and their group-wise handling; and (*iii*) Formation and complexity analysis of InterTex. While Part 1 is mainly guided by Principle I and Part 2 is mainly guided by Principle II, Part 3 is mainly guided in particular by Principle III. It is assumed that the locations, scales and orientations of all the key-points in a given image are provided by a typical feature detector such as SIFT [5], and the region around each key-point is scaled and rotated along its dominant orientation, unless stated otherwise.

3.1. Interwoven window

Of critical concern for partitioning is the expense of texture information loss. The partitioning strategy of the traditional methods generates bins that often do not contain sufficient texture information. Dependent on the size of the bins, they generally contain either pure texture or pure texture-less patterns. This is typically because of their small size; for example, SIFT typically uses a bin size of '4×4' pixels. In order to increase the probability of having texture information inside the bins, their size has to be increased. Increasing the size of the bins, however, should not increase the overall size of the support region because of the risk of losing the discriminability of the given region, stated by Principle I above. One possible solution is to overlap the adjacent expanded bins. By overlapping the adjacent bins, the size of the support region can be decreased. Smooth transitions between the bins via overlap can also improve the rotational robustness. This idea is employed in [8]. Although this strategy improves up to a point the robustness of a descriptor, further increasing it causes the descriptor to start losing its discriminability.

In this study, we propose an interwoven partitioning strategy. Slow change of pixels in intensity indicates that adjacent pixels inside each bin share the same texture pattern. As adjacent pixels usually do not introduce distinctive information, the measurement operators can be applied to every other pixel rather than to all the pixels continuously. Thus, the step size is increased from 1 (which is taken by the traditional descriptors) to 2 in our descriptor and the size of the bins is increased twice along each axis. To this end, we design two complementary windows in such a way that any two adjacent bins are interwoven. The windows are depicted in Fig. 2(a). The white and colored pixels are treated differently and they are used to represent different bins. The pixels used to represent a bin are completely different from those for the representation of the others. The measurements for each bin are calculated over the colored pixels only. The interwoven area between any two adjacent bins is 50% in space and they cover different sub-regions [Fig. 2(b)]. The interwoven bins do not share identical pixels at all but do share similar texture patterns. In this way, each bin becomes more independent on its adjacent ones. Interweaving bins render texture information of adjacent bins more distinctive and less repetitive in the overlapped regions, yielding unique encoding and less ambiguity in matching. This will be discussed later in more depth. The number of bins, on the other hand, should be sufficiently large to provide a



(a) (b)

Figure 2: Proposed interwoven windows. (a) The window (left-hand side) and its complementary one (right-hand side) applied to two adjacent bins; (b) Any two horizontally or vertically adjacent bins are interwoven by 50% in space.

fine grained characterization of the changes in intensity of the pixels in each bin. Interweaving adjacent bins enables us not only to stretch our observation over each bin but also to increase the number of observations/bins.

Stretching our observation over each bin and increasing the number of bins while fixing the size of the support region is the novelty of the proposed partitioning technique. Its main feature is to more effectively encode structural information in both texture rich and limited regions. This is shown by some examples in Fig. 3. Let us assume that there are a few sample regions of size '20×20' pixels, and the task is to partition them into 16 bins. The resultant bins with the conventional descriptors are shown in Fig. 3(b), wherein the samples are partitioned into 16 bins of size '5×5' pixels. It is clear that along each axis, there are 4 bins. It can be seen that most of the bins are texture-less (those are indicated by red squares) and such bins do not contribute valuable information to the descriptor. This issue is more obvious in the second example on the right-hand side, where all the bins are smooth. For the same samples and for the same number of bins, the proposed interwoven technique considers 16 bins of size '8×8' pixels instead[1]. From the left to the right and from the top to the bottom, the samples are partitioned into '8×8' bins, in which the overlap width between any two adjacent bins is 4 pixels. The texture information included in different bins due to the proposed partitioning strategy are depicted in Fig. 3(c). The technique can provide a wider observation of the same region, making sure the presence of texture information in all the bins. All the bins contain different texture patterns, so there are 16 distinctive non-zero values that can significantly help to provide a distinctive signature for the region. It is worth noting that the proposed partitioning method does not require excessive computational time for arranging the bins and pixels (see Section 3.3).

In practice, the proposed descriptor contains 36 interwoven bins. The arrangement of all these bins is shown in Fig. 4. All the bins are symmetric. Along each axis, 6 bins of size '8×8' pixels are considered and as they are interwoven by 50% in space, the size of the support region is thus '28×28' pixels. Among these bins, there are four whose designs are different from the others. These bins, i.e. $\{15, 16, 21, 22\}$, are located at the interest point, which

[1] We assume that the support region is square and its length M is calculated via $M = (\sqrt{k}+1) \times \frac{m}{2}$, in which k is the total number of bins and m is the length of each bin. The length of bins can be obtained via $m = \frac{2M}{\sqrt{k}+1}$. Here, we have '$M = 20$' and '$k = 16$' so the size of bins is 8.

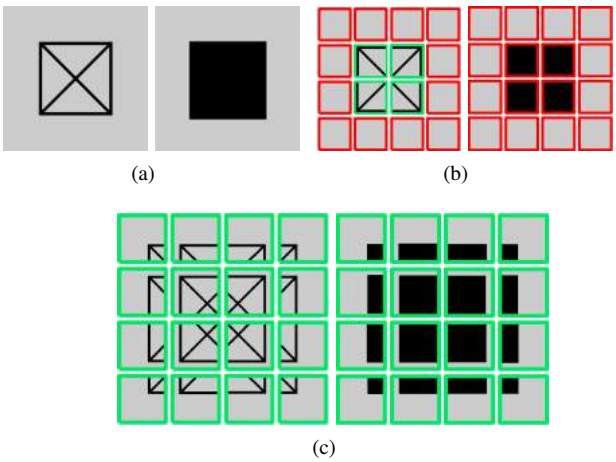


Figure 3: Examples of how the proposed window can increase the probability of having texture information in different bins. (a) Samples of size '20×20' pixels. (b) Traditional partitioning strategy. (c) Proposed partitioning strategy. The red squares show texture-less bins, and the green ones stand for the bins containing texture information.

ensure that they contain texture information. Fig. 4 also reveals the relation between these bins. Each bin is interwoven with its adjacent ones located in the cardinal directions and overlaps with its adjacent ones located in the intermediate directions. For example, the 8^{th} bin is interwoven with a bin set $\{2, 7, 9, 14\}$ and overlaps with a bin set $\{1, 3, 13, 15\}$. Partial overlap of the bins, which are of size '4×4' and include 8 pixels, improves the robustness of the representation. At the same time, the independency between interwoven bins, which is of size '8×4' along the x axis or '4×8' along the y axis and includes 32 pixels [Fig. 2(b)], increases the distinctiveness of the representation as the interwoven bins encode the region of interest from different views more effectively. Section 4.6 discusses the influence of these parameters.

In the following, we provide a real example of how the proposed partitioning window can improve the representation. Fig. 5(a) shows two images of the Semper dataset [42]. We chose a baluster region, indicated with red rectangles in Fig. 5(a), that has a symmetric and repetitive texture pattern as well as a considerably high density of the key-points[2]. Adjacent key-points reflect that they share a common neighbourhood and their representations are therefore more challenging. The matching results obtained by different methods are depicted in Figs. 5(b)-(d), wherein Fig. 5(b) shows the matching result of the proposed interwoven method whose score is 53%; and Fig. 5(c) shows the result of the proposed descriptor without using the interwoven partitioning technique that yields a matching score of 31% only. These results show that a considerable number of possible matches may be missed in the absence of an appropriate partitioning strategy. We also show the matching results of two texture-based descriptors, SURF and a recently developed deep learning-based technique [25], in Figs. 5(d) and 5(e), respectively. We observe that these descriptors failed to find sufficient matches (less than 5%).

[2]The key-points were extracted by the SIFT detector. The matching strategy and details are discussed in Section 4.

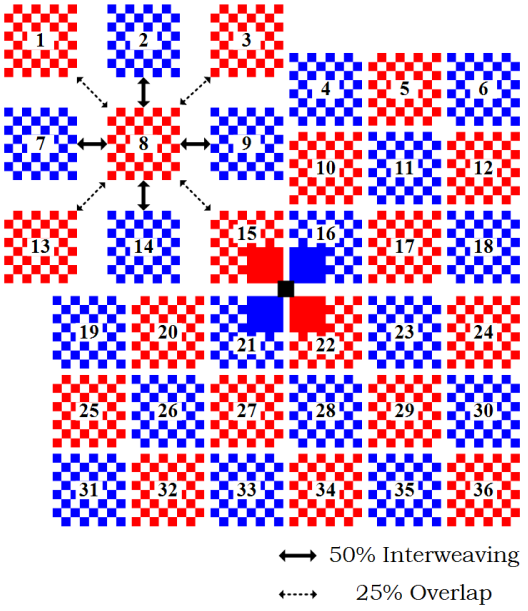


Figure 4: The arrangement of the bins in InterTex. The interaction of a sample bin (the 8th bin) with its adjacent neighbours is shown in detail.

3.2. Measurements

Generally, key-points are located in blobs or at corners and suitable measurements are those that can capture all texture information. To this end, the gradient magnitude and divergence can meet our requirement. If the gradient $\bigtriangledown I$ of an image I located at pixel (x, y) is a vector of its derivatives (I_x, I_y) along the x and y axes, then its gradient magnitude $|\bigtriangledown I|$ is calculated via:

$$|\bigtriangledown I| = \sqrt{I_x^2 + I_y^2}. \tag{1}$$

Gradient magnitude measures the absolute and total changes in intensity or equivalently textures and it does not determine the dominant texture pattern of a region. Different spatial patterns exhibit different levels of expansion or compression. The dominant characteristics of texture patterns could be measured by divergence [35]. According to the definition, divergence, denoted by div, is "flux density" that determines the amount of flux entering or leaving a pixel:

$$div(I) = I_x + I_y. \tag{2}$$

Divergence is a suitable parameter to measure whether a region is expandable ($div > 0$), compressible ($div < 0$) or even neutral ($div = 0$). The amplitude of this parameter determines the level of compression or expansion. Fig. 5(f) shows the effectiveness of divergence in encoding regions.

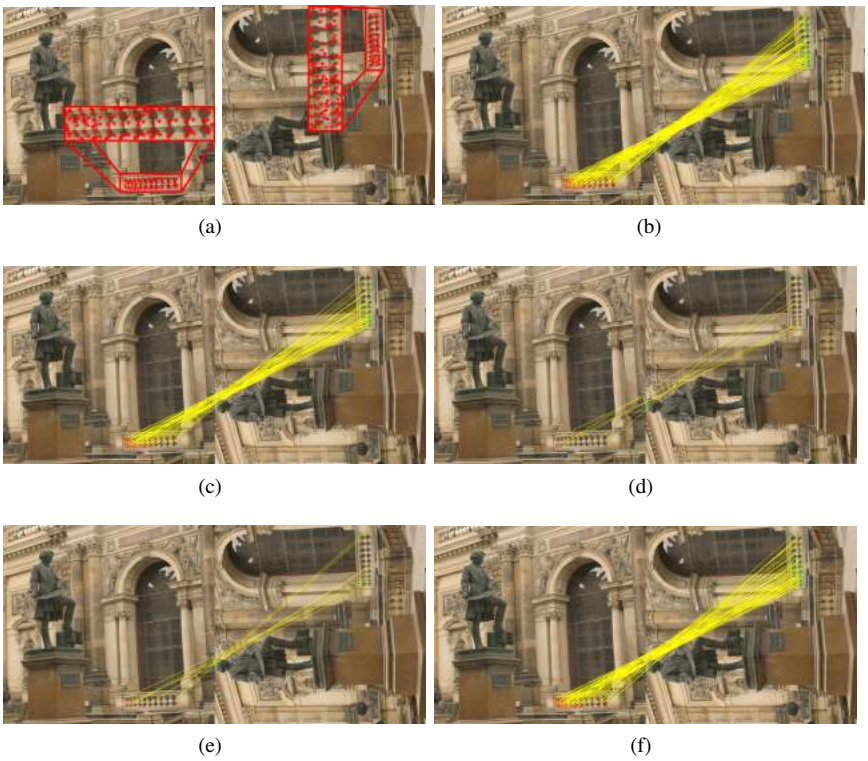
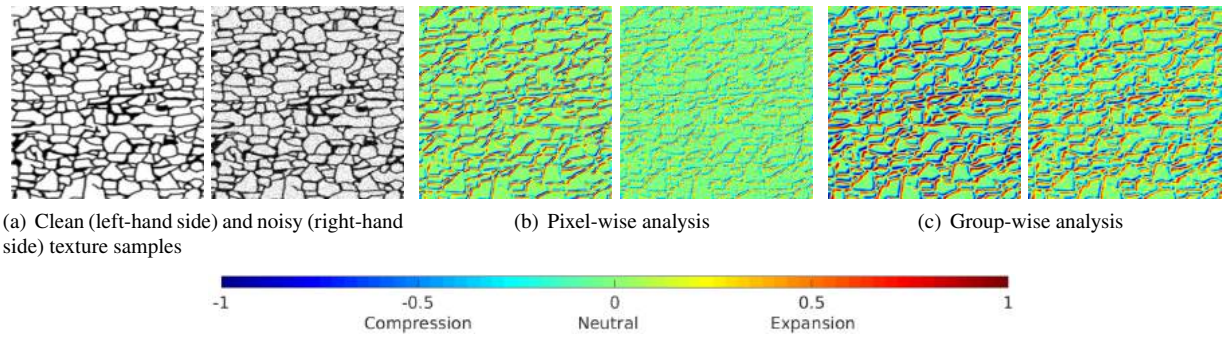


Figure 5: An example of feature matching using different techniques for a region with a repetitive pattern indicated by the red rectangles. (a) Reference and query images from the Semper dataset [42]. (b) Results of InterTex. (c) InterTex without using the interwoven window (using just a simple window without interweaving and overlapping concepts). (d) Matching results obtained by SURF. (e) Matching results obtained by the deep-learning method developed in [25]. (f) Matching results of InterTex using only divergence as a measurement parameter.

Our experiments have shown that a small area around a pixel is not able to show the exact behavior of divergence for the given pixel. Pixel-wise use of Eq. (2) involves just 4 pixels (or 8 pixels if the diagonal neighbours are considered) in computation. Often, a small number of pixels is not sufficient to reflect the exact spatial pattern. One possible solution is to apply the divergence operator to a group of pixels rather than to each individual pixel, as stated by Principle II above. This strategy ensures that engaging a suitable number of pixels in the computation leads to a more stable representation of the texture patterns. This is shown in Figs. 6(a)-(c), wherein the heat maps of the divergence responses for group- and pixel-wise analyses are depicted. The majority of pixel-wise responses for noise-free texture [Fig. 6(b)] are neutral (zero divergence) while the group-wise results [Fig. 6(c)] show more compressible/expansible responses that better reflect the texture patterns of the given image.

Images often contain noise and distortions. Compared to pixel-wise analysis, group-wise analysis is capable of representing the stable features of a region in the spirit of Principle II above. If a region contains K pixels with Independent and Identically Distributed (IID) noise, the probability distribution of the average of K IID variables with finite variance σ^2 approaches a normal distribution with the sigma of σ^2/K according to the central limit theorem.



(a) Clean (left-hand side) and noisy (right-hand side) texture samples (b) Pixel-wise analysis (c) Group-wise analysis

Figure 6: The results of pixel-wise and group-wise analyses in divergence for a noise-free and noisy image respectively.

We examine the robustness of pixel- and group-wise analyses against noise by adding 5% zero-mean white Gaussian noise to the sample image. The results verify that the group-wise approach is less susceptible to noise compared to the pixel-wise approach.

In summary, the structural properties of a group of pixels can provide stable features of a region and they can effectively reduce the side effects of noise and distortion. We use Haar wavelets for group-wise computation of the derivatives and they are implemented via an integral image and box filters. Initially we form the integral image and then compute the Haar kernels via the box filters of size '$4 \times scale$' along each axis. According to our experiments (see Section 4.6), this kernel size yields the best and most stable results.

3.3. Formation of InterTex and computational complexity

There are 36 bins and each bin contains 32 pixels, except the 15^{th}, 16^{th}, 21^{st} and 22^{nd} bins that contain 40 pixels as illustrated in Fig. 4. We calculate the gradient magnitude and divergence for all the pixels in a bin using the group-wise analysis. To emphasise the importance of nearby pixels and bins and combat the imaging noise, illumination change and distortions as guided by Principle III above, the measurements are firstly weighted by the Gaussian function with a sigma of '$2.2 \times scale$' of the distance from a pixel to the particular bin and then summed up, while the bins themselves are weighted by the Gaussian function with a sigma of 3.3 of the distance from a particular bin to the interest point. Since there are 36 bins and each bin yields two measurements, we have 72 values altogether. To further combat the change of the bins in size and intensity/illumination from one image to another, all the measurements are finally normalized by a square root (Hellinger) kernel proposed by Arandjelovic and Zisserman [43]. Given an image I and its key-points $p_i = (x_i, y_i, \sigma_i, \theta_i)$, $i \in \{1, ..., L\}$, where (x, y), σ, θ and L are the 2D location, scale, orientation and the total number of key-points, respectively. If the values b_i^j $(j = 1, 2, \cdots, 72)$ of bin $\mathbf{b}_i$ of key-point p_i are arranged in a vector and shown as

$$\mathbf{b}_i = sgn(\mathbf{b}_i) \times \left|\mathbf{b}_i\right| = \left[sgn(b_i^1), ..., sgn(b_i^{72})\right] \left[\, \left|b_i^1\right|, ..., \left|b_i^{72}\right| \,\right]^T, \tag{3}$$

then, its L_2-normalized version, $\mathbf{b}_{i,L2}$, is

$$\mathbf{b}_{i,L2} = sgn(\mathbf{b}_i) \times \left|\mathbf{b}_{i,L2}\right| = sgn(\mathbf{b}_i) \times \frac{\left|\mathbf{b}_i\right|}{\sqrt{\sum_{n=1}^{72} \left|b_i^n\right|^2}}. \tag{4}$$

In the above equations, $sgn(.)$ and $|.|$ are the sign and absolute value functions. The Hellinger kernel of the above equation is computed as

$$f_i = sgn(\mathbf{b}_i) \times \sqrt{\frac{\left|\mathbf{b}_{i,L_2}\right|}{\sum_{n=1}^{72} \left|b_{i,L2}^n\right|}}, \tag{5}$$

where f_i is the representation vector of key-point p_i.

In the following, we summarize InterTex. After forming the integral image of I, for each key-point p_i, $i \in \{1, ..., L\}$, do:

1. Apply the rotation θ_i to the region of size '28×28' pixels around key-point p_i by the similarity matrix $\begin{bmatrix} \sigma_i \cos(\theta_i) & -\sigma_i \sin(\theta_i) \\ \sigma_i \sin(\theta_i) & \sigma_i \cos(\theta_i) \end{bmatrix}$;
2. Compute the derivatives of all the pixels inside the region of interest using the integral image and the box filters;
3. Compute the gradient magnitude and divergence via Eqs. (1) and (2), respectively;
4. Arrange the pixels by the proposed partitioning window illustrated in Fig. 4.
5. Multiply the measurements of the pixels inside each bin by the Gaussian function with a sigma of '$2.2 \times \sigma_i$' of their distances and then sum up;
6. Multiply the measurements of the bins by the Gaussian function with a sigma of 3.3 of the distances to the interest point p_i; and
7. Arrange the values of the bins into a vector and then normalize them via the Hellinger kernel described in Eqs. 3-5. The output is the representation vector f_i of the key-point p_i.

From the computational complexity perspective, the support region contains $28 \times 28 = 784$ pixels. For each pixel, we need 2 and 3 addition operations for computing the integral image and Haar wavelets, respectively. Each derivative in the x and y directions needs 2 box filters. The computational cost of the box filters is independent of their size. The Gaussian coefficients of the bins are fixed and stored while the Gaussian coefficients of the pixels in each bin are calculated for each key-point. The majority of the computational time is assigned to these steps (about 85%). The proposed partitioning window does not need arithmetic operations as it is dominated by the cost of the memory accesses. Since all the operations are linear, the descriptor has linear computational complexity in the number of key-points.

4. Experimental Results

We validate InterTex over multiple datasets with different characteristics and applications. The proposed descriptor was compared with several well-known feature descriptors from different categories including:

- Root-SIFT (R-SIFT) [43], SURF [12], learned arrangements of three patch codes (LATCH) [44] and FREAK [10] from the hand-crafted category;
- BinBoost [17] and mixed intensity order pattern (MIOP) [19] from the learning-based category; and
- GeoDesc [20], HardNet [21], D2-Net [22], SuperPoint (SP) [24] and CD [25] from the deep learning-based category.

All the selected methods are among the best ones of their categories. We used the implementations of the descriptors from OpenCV[3] for all except the CNN-based descriptors that are available in [20; 21; 22; 24; 25]. For CNN-based descriptors, we used the pre-trained models released by the authors. R-SIFT was computed via applying the distance proposed in [43] to the SIFT descriptor implemented in OpenCV. The former yields better results than the latter. The length of the LATCH descriptor was set to 64 bytes for obtaining its best performance. Likewise, the length of BinBoost was set to 32 bytes.

SURF, LATCH, MIOP, FREAK and BinBoost were extracted at image locations detected using their own original key-point detector. D2-Net and SP used trainable networks for joint key-point description and detection. The SIFT detector was employed to extract key-points for the rest of the feature descriptors. The features were matched via mutual nearest neighbour distance ratio (NNDR), where the descriptor of a point in one image should be the nearest neighbour of that in the feature space of its correspondent in the other and vice versa. This strategy effectively reduces the false matching-rate [27; 42]. The ratio between the responses of the first nearest neighbour and the second nearest neighbour was set to 0.9. The Euclidean and the Hamming distances were used to match the float-based and the binary descriptors, respectively. The experiments were carried out over several benchmark datasets and the results are organized into the following subsections: homography datasets, 3D reconstruction, visual localization, descriptors' performance under different types of feature points, illumination changes, parameters evaluation, and run time and storage analysis.

4.1. Homography datasets

Here we evaluate the descriptors over the publicly available homography datasets with ground-truth geometric transformations. The HSequences benchmark [45] includes 59 sequences with viewpoint changes and 57 sequences with illumination changes. In each sequence, there are six images where the first image serves as a reference and the

[3] https://opencv.org

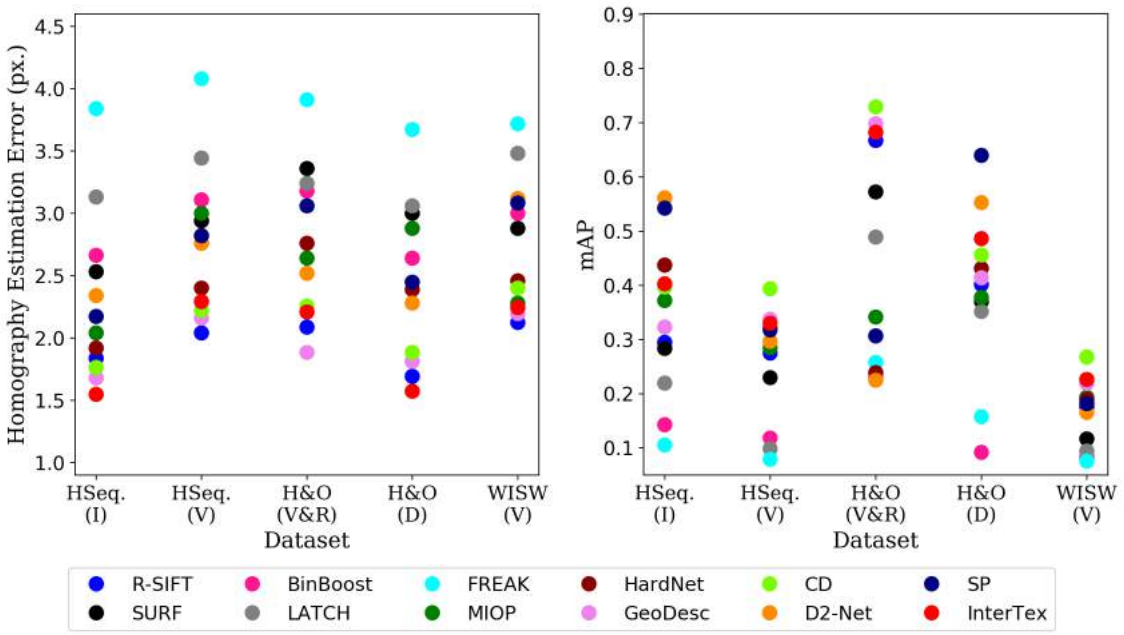


Figure 7: Evaluation results of different descriptors for the homography datasets including: Illumination (I) and Viewpoint (V) of the HSequences benchmark (HSeq.); Distortion (D) and Viewpoint & Rotation (V&R) of Heinly and Oxford benchmark (H&O); and WISW benchmark with viewpoint changes.

remaining five are test ones. Heinly [42] and Oxford [13] datasets are other benchmarks which evaluate the descriptor performance under changes in viewpoint, scale, illumination, blur, rotation and compression. WISW benchmark [28] includes 5 sequences with changes in viewpoint. The features were assessed by mean average precision (mAP). Feature points p_i in one image and p'_j in another are labelled as a true match if $||p_i - Hp'_j|| < \epsilon$, where H is the ground-truth homography and ϵ is the projection error in pixels. In this experiment, ϵ takes values in the range of [2.5, 7.5] in steps of 0.5 pixels and the average of the results is finally reported.

The evaluation results over the four homography datasets are reported in Fig. 7. For each benchmark, the average of the results over each dataset was first computed and their average over all the datasets was then reported. We can see that CD, InterTex and GeoDesc perform well over the viewpoint datasets and in the cases of illumination variation and distortion, D2-Net and SP gain the best mAP scores. Our experiments show that $\epsilon > 5$ could not considerably improve the precision results of InterTex. We also estimate the homography correctness. The homography is estimated using the OpenCV function *findHomography* and the maximum average reprojection error of the image corners was set to 4 pixels, which is commonly used in most software like COLMAP. According to the results shown in Fig. 7, InterTex gains the minimum error under distortions, i.e. HSeq. (I) and H&O (D) and performs on par with other descriptors, especially the deep learning ones, under viewpoint changes, despite the fact that InterTex did not manipulate a large number of similar images, which are not always available, and thus has an advantage of easy implementation.

We further examine the robustness of the feature descriptors against noise and distortion. The experiments were

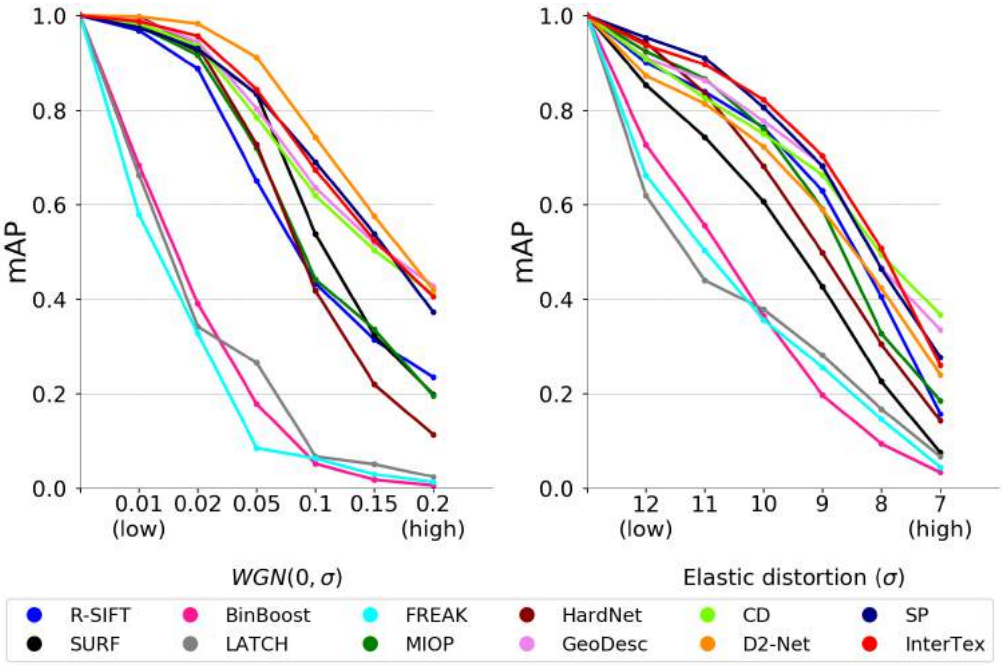


Figure 8: The mAP of different descriptors over the HSequences-Viewpoint benchmark subject to random white Gaussian noise and elastic distortions.

run over the HSequences-Viewpoint benchmark and the average of mAP results is reported here. An additive white Gaussian noise (WGN) with a standard deviation from 0.01 to 0.2 was added to the images, even though noise can be space-variant in practical imaging. The results on the synthesized data are reported in Fig. 8. Since the robustness is evaluated here, we normalized the mAP results of noisy images to the noise-free ones to simply compare the rate of drop at different levels of noise. All the descriptors decrease their performance in mAP over the increased noise, as expected. The binary descriptors are more susceptible to noise than the descriptors in the other categories. D2-Net, InterTex and SP show more resistance against noise than the other feature descriptors.

The images were also deformed through elastic distortions [46] with standard deviation ranged from 7 to 12. Similar to the results over the data with the synthesized noise, we normalized the mAP results over the distorted images to the distortion-free ones to facilitate the comparison of the performance of different descriptors. Unlike the results over the data with the synthesized noise where D2-Net shows the best performance, it is not so robust against shape deformation. For small deformation, SP and InterTex are less affected by the deformation distortion and for the harsh condition, i.e. $\sigma = 7$, CD and GeoDesc work well. In short, the synthesized noise and distortion analyses show that while all the descriptors decrease their performance in mAP over the increased noise and distortion, as expected, the proposed InterTex descriptor is relatively stable and handles noise and distortion well. This aspect of InterTex will be further assessed by the 3D reconstruction and the visual localization tasks in the following sections.

4.2. 3D reconstruction

For evaluation of the feature descriptors for the task of image-based 3D reconstruction, several pipelines have been developed [27; 29]. For image-based 3D reconstruction, it is necessary to calibrate in advance the cameras that is done via SfM. MVS is then applied to the output of SfM to obtain a dense reconstruction of the given scene. The quality of 3D models, which are the outputs of MVS, directly depends on the accurate and complete estimation of camera parameters in the first step, i.e. SfM. We follow the same metrics and protocols developed in [27] for analyzing the 3D models. According to this pipeline, the SfM and MVS analyses are made via COLMAP[4] and the metrics used are *the number of registered images*, *mean reprojection error*, *mean track length*, *reconstructed sparse points* and *reconstructed dense points*. The datasets employed here are Fountain, Herzjesu, South Building, Madrid Metropolis, Gendarmenmarkt and Tower of London. Exhaustive image matching was employed for all the datasets and they do not need image retrieval. More information about these datasets is available in [27].

In order to save space, we reported the results of R-SIFT, GeoDesc, D2-Net, CD and SP as they have gained the best scores in all the metrics among the aforementioned existing methods in the last section. From Table 1, it can be seen that in terms of the number of sparse and dense points, our method is comparable to existing ones. However, a higher number of points obtained via InterTex does not mean that it yields larger errors. If we compare the reprojection errors of all the descriptors, we can see that InterTex estimates the most accurate locations of the cameras over 5 datasets out of six. The results show that there is a close link between the track length and the descriptor length and a long length of a descriptor often yields a high track length. Even though InterTex has a short length for description, it is still able to register a comparable number of images.

3D reconstruction of images with smooth and repetitive textures is a challenging task. The number of matched points in such images is often insufficient and repetitive textures also increase the probability of false matches. In order to evaluate the performance of the descriptors for such tasks, we applied them to some ETH3D datasets[5] using COLMAP. To save space, we just reported two sample results of R-SIFT that have the best performance among the existing descriptors. Fig. 9 depicts the reconstruction results of these challenging images by different methods. An insufficient number of correct matches may not be the main reason for inaccurate estimations of the locations and orientations of the cameras or equivalently 6DoF[6]. In fact, nowadays' cameras provide high-resolution images that usually contain sufficient correspondences in most cases. Low precision of the matched points can also lead to inaccurate estimation of the locations and orientations of the cameras as shown in the sparse results of R-SIFT in Fig. 9 and in the homography results. This topic will be further investigated in the following subsection.

[4] https://colmap.github.io/
[5] https://www.eth3d.net/slam_datasets#test-data
[6] 6-Degree-of-Freedom that indicates the translations and orientations of the cameras in 3D space.

Table 1
Evaluation results of different descriptors over the 3D reconstruction benchmark. The first and second best results are highlighted in boldface and with underline respectively.

Dataset (# Images)	Descriptor	Regis. (#)	Track Length (#)	Sparse Points (#)	Dense Points (#)	Reproj. Error (px.)
Herzjesu	*R-SIFT*	8	4.01	4,916	242K	0.32
(8)	*GeoDesc*	8	4.03	8,720	243K	0.42
	CD	8	4.01	9,429	247K	0.41
	D2-Net	8	**4.07**	6,129	240K	0.65
	SP	8	3.96	**13,181**	251K	0.87
	InterTex	8	3.93	6,893	**266K**	**0.27**
Fountain	*R-SIFT*	11	4.63	10,004	304K	0.30
(11)	*GeoDesc*	11	4.71	16,687	306K	0.39
	CD	11	4.68	16,965	305K	0.38
	D2-Net	11	**4.75**	11,365	306K	0.76
	SP	11	4.70	**19,127**	303K	0.85
	InterTex	11	4.43	12,865	**308K**	**0.25**
South Building	*R-SIFT*	128	5.75	89K	2,022K	0.48
(128)	*GeoDesc*	128	5.67	148K	2,087K	0.53
	CD	128	5.65	**151K**	2,096K	0.54
	D2-Net	128	5.97	107K	2,069K	0.93
	SP	128	**6.32**	132K	2,057K	0.91
	InterTex	128	5.56	113K	**2,115K**	**0.39**
Madrid	*R-SIFT*	463	**6.12**	117K	**1,621K**	**0.58**
Metropolis	*GeoDesc*	486	5.93	145K	1,560K	0.66
(1,344)	*CD*	493	5.91	**151K**	1,573K	0.69
	D2-Net	498	5.97	135K	1,505K	1.05
	SP	**505**	5.91	141K	1,546K	1.19
	InterTex	478	5.81	138K	1,577K	0.60
Gendarmenmarkt	*R-SIFT*	975	**5.45**	321K	3,827K	0.69
(1,463)	*GeoDesc*	1002	5.12	**458K**	3,925K	0.73
	CD	1011	5.09	447K	3,893K	0.71
	D2-Net	**1038**	4.98	243K	3,532K	0.99
	SP	1029	5.04	362K	3,629K	1.06
	InterTex	953	4.87	388K	**4,010K**	**0.64**
Tower of	*R-SIFT*	711	**7.02**	227K	3,109K	0.62
London	*GeoDesc*	767	6.69	325K	2,985K	0.66
(1,576)	*CD*	771	6.66	**341K**	3,074K	0.68
	D2-Net	755	5.62	171K	2,806K	1.04
	SP	**785**	6.73	316K	2,934K	1.10
	InterTex	724	6.55	287K	**3,409K**	**0.57**

4.3. Visual localization

Visual localization is another important computer vision task that requires an accurate estimation of the camera position and orientation. Real-world conditions like day-night transitions severely affect the contents of the images for the same scene and local feature matching of such images is usually challenging. Aachen Day-Night dataset [47]

(a) Stereo images

(b) R-SIFT (c) InterTex

Figure 9: The sparse (left-hand side) and dense (right-hand side) reconstruction results (bottom row) of R-SIFT and InterTex for two sets of texture-limited images (top row) from the ETH3D dataset.

contains 4,328 day-time images and 98 queries taken at night. The performance of local feature descriptors is evaluated by a pre-defined visual localization pipeline[7]. In this experiment, the results of successfully localized images are reported with three tolerances in estimation errors of position and orientation: (0.5m, 2 deg.), (1m, 5 deg.) and (5m, 10 deg.). Table 2 reports the results of visual localization. Our approach yields comparable results to the baselines. For strict accuracy thresholds for the estimated localization, InterTex actually works better than all the others. These results are consistent with those reported in the previous sections.

As the pipeline defined in [22] is available for the Aachen dataset, we applied the descriptors to other datasets including RobotCar Seasons and CMU Seasons with various conditions such as day-night and seasons[8] and a part of the results is depicted in Fig. 10. This figure further shows the reliability of the proposed descriptor under severe changes in illumination.

4.4. Descriptors' performance under different types of feature points

Since different feature detectors provide different types of key-points in terms of quantity and quality like blob, corner, region etc, this section investigates the influence of feature points on the performance of feature descriptors.

[7] https://github.com/tsattler/visuallocalizationbenchmark/tree/master/local_feature_evaluation
[8] https://www.visuallocalization.net/benchmark

(a) Day-Night(Aachen dataset)

(b) Season (CMU Seasons dataset)

(c) Dawn-Day (CMU Seasons dataset)

(d) Night-Sun (RobotCar Seasons dataset)

Figure 10: Feature matching results (after applying RANSAC) of different descriptors over data with changes in time and illumination. From the left to the right: R-SIFT, GeoDesc, D2-Net and proposed InterTex.

Table 2
Evaluation results of different descriptors for visual localization on Aachen dataset.

Descriptor	(0.5m, 2 deg.)	(1m, 5 deg.)	(5m, 10 deg.)
Upright R-SIFT	36.7	54.1	72.5
HardNet	37.9	54.0	75.5
GeoDesc	38.1	54.7	73.4
CD	39.2	55.0	73.8
D2-Net	39.8	55.1	74.5
SP	38.7	54.9	**76.3**
Upright InterTex	**40.2**	**55.8**	74.2

To this end, considering the information such as location, scale and dominant orientation provided and required, we selected various representative feature detectors including SIFT, SURF, KAZE [48], FAST [49], MSER [50] and BRISK [51] and feature descriptors including the InterTex, R-SIFT and HardNet for detecting and describing the feature points in the homography images in Section 4.1. In order to provide fair comparison, we selected the best 6000 key-points per image of each feature detector. Similar to Section 4.1, the projection error ϵ takes values in the range of [2.5, 7.5] in steps of 0.5 pixels and the average of the results is finally reported.

The results are reported in Fig. 11. According to the results, blob detectors (i.e. SIFT, KAZE and SURF) provide better feature points for descriptors compared to the corner detectors including FAST and BRISK, and also to

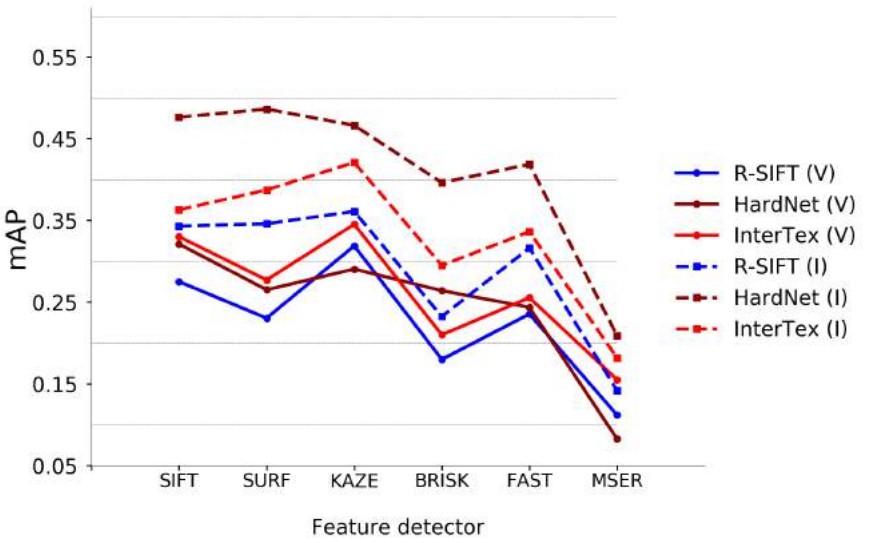


Figure 11: The mAP of InterTex, HardNet and R-SIFT for matching the key-points detected using different techniques over the homography HSequences-Viewpoint (V) and -Illumination (I) benchmarks, detailed in Section 4.1

the region-based MSER feature detector. All the feature descriptors exhibit similar behaviours over different feature detectors in both the HSequences-Viewpoint and HSequences-Illumination datasets, except the HardNet over the BRISK descriptor in the HSequences-Illumination dataset. These results show that the feature detectors do not impose a significant impact on the feature descriptors for their matching. While the HardNet performs best over the HSequences-Viewpoint dataset, the proposed InterTex performs on the whole best over the HSquences-Illumination dataset. Note that throughout this paper we reported the InterTex and R-SIFT results on the key-points extracted by the SIFT detector to keep consistency with the other feature descriptors.

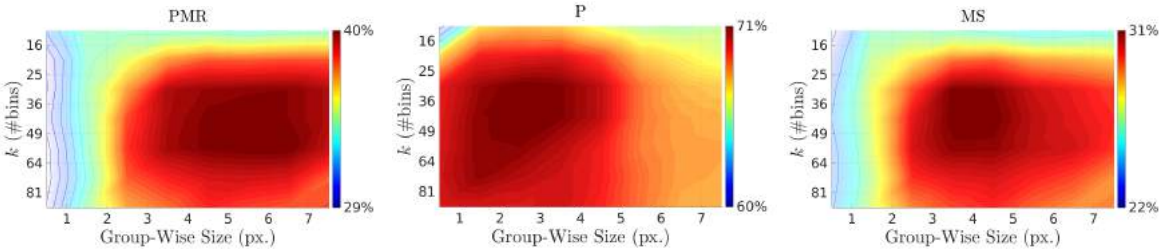
4.5. Illumination changes

It is always challenging to handle illumination changes in image analysis due to different imaging weather, environment, time, directions, and positions. Here, we reported the results about loop closure detection by R-SIFT, SP and ours for SLAM. We used HTMap [52] as an appearance-based approach for topological mapping and it is based on a hierarchical decomposition of the environment.

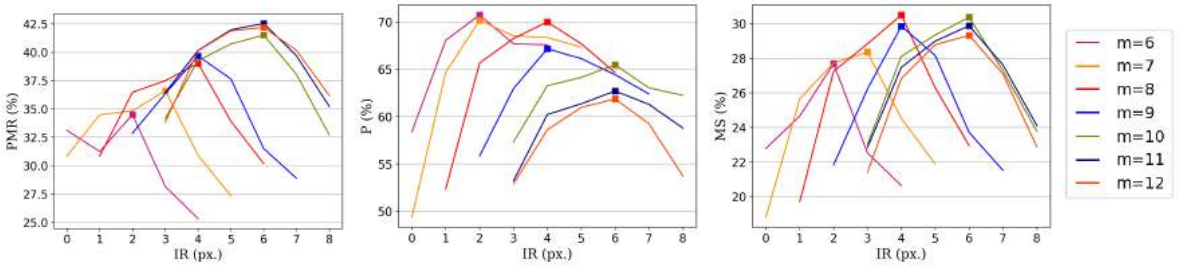
HTMap uses FAST corner detector[53] and ORB [54] feature descriptors to select the most similar images inside the retrieved nodes. We applied this technique to the City Center and the New College datasets, which consist of 1237 and 1073 pairs of images with a size of 640×480 captured by two cameras. Since HTMap has been developed to be used with monocular cameras, the left and right frames were merged into images of size 1280×480. We also used the KITTI [55] sequences 00 and 05 as a representative set of the benchmark. While HTMap uses FAST corner detector and ORB feature descriptor by default, we replace them with R-SIFT, SP and InterTex and appropriate feature detectors. The results are tabulated in Table 3 and show that InterTex could gain high performance over outdoor images.

Table 3
The recall results at 100% of precision for HTMap using different feature descriptors

Descriptor	Detector	City Center	New College	KITTI 00	KITTI 05
ORB	*FAST*	78.3	71.7	90.1	75.2
R-SIFT	*SIFT*	80.2	74.2	<u>90.5</u>	<u>77.3</u>
SP	*SP*	<u>81.3</u>	**76.6**	89.4	76.2
InterTex	*SIFT*	**81.7**	<u>76.3</u>	**91.3**	**78.4**



(a) Influence of the number k of bins and size of Group-wise box filter analysis for '$m = 8$' and '$IR = 50\%$'.



(b) Influence of parameters 'IR' and 'm' for fixed box filters with size=4 and '$k = 36$'.

Figure 12: Influence of parameters on the performance of InterTex.

4.6. InterTex parameters evaluation

As discussed in Section 3.1, the support region has a square shape of size $M = (\sqrt{k} + 1) \times \frac{m}{2}$, where k and m denote the number of bins and their size, respectively. The interwoven rate, IR, was 50% and for group-wise analysis, we set the size of group-wise box filters to 4. Below, we study the influence of these parameters on the performance of InterTex. All the following experiments were conducted on the HSequences-Illumination benchmark.

Fig. 12(a) illustrates the matching results for different values of group-wise box filter size and the number k of bins. For this experiment, m and IR were set to 8 pixels and 50%, respectively. A group-wise box filter size of 1 pixel indicates the pixel-wise analysis. For a fixed number of bins, involving 4 and more pixels in the box filters markedly increases the number of candidate matches and meanwhile, the precision $P = \#inlier-matches/\#putative-matches$ starts to drop. It indicates that an averaging operator is more suitable for detection of more similar corresponding points (higher $PMR = \#putative-matches/\#features$) while it is not favorable for precision. The interaction of

Table 4
The required storage, extraction time and matching run-time per key-point of different descriptors.

Descrip.	Storage	platform	Extraction (μs per kp.)	Matching (ms)
R-SIFT	128 floats	CPU	69	256
SURF	64 floats	CPU	28	194
BinBoost	256 bits	CPU	190	**52**
LATCH	512 bits	CPU	41	85
FREAK	512 bits	CPU	18	86
MIOP	128 floats	CPU	104	254
HardNet	128 floats	**GPU**	15	264
GeoDesc	128 floats	**GPU**	101	270
CD	128 floats	**GPU**	234	273
D2-Net	512 floats	**GPU**	43	651
SP	256 floats	**GPU**	29	376
InterTex	72 floats	CPU	**12**	201

these parameters, which is expressed as a matching score $MS = \#inlier - matches/\#features$, reaches the best performance at around 4 pixels. Lower PMR and P are the main weakness of small and large support regions, which is compatible with the locality concept. InterTex gains its high matching scores for k between 36 and 49 bins.

The impacts of parameters m and IR of the performance of the proposed InterTex are shown in Fig. 12(b). A sharp increase in the matching results using an interwoven window ($IR > 0\%$) confirms its importance in unique representation of regions. Increasing IR up to 60% for different values of m improves both parameters P and PMR while its effectiveness drops by further increase. Moreover, different bin sizes m gain their highest performance when the size of the interwoven area is even and this is probably due to the symmetric properties of even numbers. According to the MS results, interwoven rate relies on the bin size m; as a rule of thumb, an interwoven area of 4 pixels is suitable for $m < 10$ and 6 for $m \geq 10$. The former is more favorable as it yields a higher precision than the latter. It is worth noting that $m \geq 12$ endangers the precision of the matches while $m < 8$ leads to a low number of matches. The choice $8 \leq m \leq 10$ is a good compromise between the accuracy/quality and the quantity of the matches.

4.7. Run time and storage analyses

The required storage and computational time for each descriptor is reported in Table 4. InterTex was implemented in C++ & OpenCV (without boost). All the experiments were run on a 64-bits computer with 32 cores Intel(R) Xeon(R) Gold 6130 CPU @ 2.10GHz processors, 48 GB RAM and two Tesla P100-PCIE-16GB GPU devices. The required storage of the proposed descriptor is 72 floats for each key-point that is about 43.8% lower than that of R-SIFT or GeoDesc. From the computational time perspective, binary methods often need less time. FREAK has the lowest time cost among the existing CPU-based descriptors while our descriptor needs just two-thirds of that run-time. Although the extraction time of D2-Net is much shorter than that of GeoDesc, its larger description length causes it to

require more time for matching. From the matching time-cost perspective, reliable descriptors like R-SIFT often need considerable time. InterTex could reduce the matching time of R-SIFT by about 25%. However, these schemes are far from the results of binary descriptors that need much lower running time, especially BinBoost whose length is just 256 bits and has the best performance in terms of matching time.

5. Conclusion and Future Work

In this paper, we have proposed a novel descriptor called interwoven texture-based feature descriptor (InterTex). It has been built on the framework that comprises three principles. We recall these principles in short:

- Principle I (Locality): The length of the local region around the given key-point should be suitable so that the concept of locality for the key-point is reserved and it is large enough for containing discriminative texture information.
- Principle II (Globality): The operations of a group of pixels can better capture the features of an interest area rather than considering the behaviours of individual pixels.
- Principle III (Robustness): All the sources of changes, distortions and noise should be recognized, and appropriate strategies should be taken into account.

Although we have applied these principles to our descriptor, they can be applied to other descriptors or can be used partly in the framework of other descriptors as well. For example, as shown in the experimental section, the gradient orientation in SIFT is drastically changed in the presence of noise; a possible suggestion to enhance its robustness is to apply Principle II above. Likewise, a large window decreases the uniqueness of the local region around the key-point that can be seen in the binary-based methods; this shortcoming can be handled via using Principle I (the locality term) as well as widening our observation from the region as described in Section 2. An interesting direction for future work is to investigate the locality and globality concepts on different pattern shapes and measurement metrics, specifically region-based ones.

Many images are captured outdoors where noise, illumination change and distortions are inevitable. Thus, being robust to the unwanted factors plays an important role in the reliability of a descriptor. During the design process, we have applied these principles effectively while keeping our descriptor simple and easy to implement as much as possible. We have evaluated InterTex over several standard databases with different characteristics for varied applications. Results have shown that our descriptor is on the whole more accurate and robust than the state-of-the-art ones to different types of changes, distortions and noise. High accuracy and low computational cost of InterTex make it more appealing to real-time applications like object detection, mobile robotics, autonomous driving, 3D visual SLAM and 3D phenotyping of plants.

Acknowledgments

The authors acknowledge Supercomputing Wales (SCW) for providing HPC resources that have contributed to the 3D reconstruction results. M. Ghahremani and Y. Zhao are sponsored by DCDS and Zhejiang Provincial Natural Science Foundation (LZ19F010001), respectively; Y. Liu is partially supported by BBSRC grant BB/R02118X/1 and UKIERI-DST grant CHARM (DST UKIERI-2018-19-10). We thank the Associate Editor and the anonymous reviewers for their constructive comments that have improved the quality of the paper.

References

[1] T. D'Orazio, M. Leo, A review of vision-based systems for soccer video analysis, Pattern Recognition 43 (8) (2010) 2911–2926.

[2] R. Muñoz-Salinas, R. Medina-Carnicer, Ucoslam: Simultaneous localization and mapping by fusion of keypoints and squared planar markers, Pattern Recognition 101 (2020) 107193.

[3] N. Snavely, S. M. Seitz, R. Szeliski, Modeling the world from internet photo collections, International Journal of Computer Vision 80 (2) (2008) 189–210.

[4] P. Loncomilla, J. Ruiz-del Solar, L. Martínez, Object recognition using local invariant features for robotic applications: A survey, Pattern Recognition 60 (2016) 499–514.

[5] D. G. Lowe, Distinctive image features from scale-invariant keypoints, International Journal of Computer Vision 60 (2) (2004) 91–110.

[6] M. Ghahremani, Y. Liu, B. Tiddeman, Ffd: Fast feature detector, IEEE Transactions on Image Processing (2020) 1–1doi:10.1109/TIP.2020.3042057.

[7] P. Koniusz, F. Yan, P.-H. Gosselin, K. Mikolajczyk, Higher-order occurrence pooling for bags-of-words: Visual concept detection, IEEE Transactions on Pattern Analysis and Machine Intelligence 39 (2) (2016) 313–326.

[8] E. Tola, V. Lepetit, P. Fua, Daisy: An efficient dense descriptor applied to wide-baseline stereo, IEEE Transactions on Pattern Analysis and Machine Intelligence 32 (5) (2009) 815–830.

[9] M. Calonder, V. Lepetit, C. Strecha, P. Fua, Brief: Binary robust independent elementary features, in: European Conference on Computer Vision, Springer, 2010, pp. 778–792.

[10] A. Alahi, R. Ortiz, P. Vandergheynst, Freak: Fast retina keypoint, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2012, pp. 510–517.

[11] X. Qi, R. Xiao, C.-G. Li, Y. Qiao, J. Guo, X. Tang, Pairwise rotation invariant co-occurrence local binary pattern, IEEE Transactions on Pattern Analysis and Machine Intelligence 36 (11) (2014) 2199–2213.

[12] H. Bay, A. Ess, T. Tuytelaars, L. Van Gool, Speeded-up robust features (surf), Computer Vision and Image Understanding 110 (3) (2008) 346–359.

[13] K. Mikolajczyk, C. Schmid, A performance evaluation of local descriptors, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (10) (2005) 1615–1630. doi:10.1109/TPAMI.2005.188.

[14] F. Bellavia, C. Colombo, Rethinking the sgloh descriptor, IEEE Transactions on Pattern Analysis and Machine Intelligence 40 (4) (2017) 931–944.

[15] B. Fan, Z. Wang, F. Wu, Local image descriptor: modern approaches, Vol. 108, Springer, 2015.

[16] C. Celik, H. S. Bilge, Content based image retrieval with sparse representations and local feature descriptors: a comparative study, Pattern Recognition 68 (2017) 1–13.

[17] T. Trzcinski, M. Christoudias, V. Lepetit, Learning image descriptors with boosting, IEEE Transactions on Pattern Analysis and Machine Intelligence 37 (3) (2014) 597–610.

[18] B. Fan, F. Wu, Z. Hu, Aggregating gradient distributions into intensity orders: A novel local image descriptor, in: CVPR 2011, IEEE, 2011, pp. 2377–2384.

[19] Z. Wang, B. Fan, G. Wang, F. Wu, Exploring local and overall ordinal information for robust feature description, IEEE Transactions on Pattern Analysis and Machine Intelligence 38 (11) (2015) 2198–2211.

[20] Z. Luo, T. Shen, L. Zhou, S. Zhu, R. Zhang, Y. Yao, T. Fang, L. Quan, Geodesc: Learning local descriptors by integrating geometry constraints, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 168–183.

[21] A. Mishchuk, D. Mishkin, F. Radenovic, J. Matas, Working hard to know your neighbor's margins: Local descriptor learning loss, in: Advances in Neural Information Processing Systems, 2017, pp. 4826–4837.

[22] M. Dusmanu, I. Rocco, T. Pajdla, M. Pollefeys, J. Sivic, A. Torii, T. Sattler, D2-net: A trainable cnn for joint detection and description of local features, arXiv preprint arXiv:1905.03561 (2019).

[23] F. Huang, C. Jin, Y. Zhang, K. Weng, T. Zhang, W. Fan, Sketch-based image retrieval with deep visual semantic descriptor, Pattern Recognition 76 (2018) 537–548.

[24] D. DeTone, T. Malisiewicz, A. Rabinovich, Superpoint: Self-supervised interest point detection and description, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2018, pp. 224–236.

[25] Z. Luo, T. Shen, L. Zhou, J. Zhang, Y. Yao, S. Li, T. Fang, L. Quan, Contextdesc: Local descriptor augmentation with cross-modality context, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 2527–2536.

[26] L. Liu, J. Chen, P. Fieguth, G. Zhao, R. Chellappa, M. Pietikäinen, From bow to cnn: Two decades of texture representation for texture classification, International Journal of Computer Vision 127 (1) (2019) 74–109.

[27] J. L. Schonberger, H. Hardmeier, T. Sattler, M. Pollefeys, Comparative evaluation of hand-crafted and learned local features, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1482–1491.

[28] F. Bellavia, C. Colombo, Which is which? evaluation of local descriptors for image matching in real-world scenarios, in: International Conference on Computer Analysis of Images and Patterns, Springer, 2019, pp. 299–310.

[29] B. Fan, Q. Kong, X. Wang, Z. Wang, S. Xiang, C. Pan, P. Fua, A performance evaluation of local features for image-based 3d reconstruction, IEEE Transactions on Image Processing 28 (10) (2019) 4774–4789.

[30] T. V. Hoang, S. Tabbone, Invariant pattern recognition using the rfm descriptor, Pattern Recognition 45 (1) (2012) 271–284.

[31] Y. Xu, G. Lu, Y. Lu, D. Zhang, High resolution fingerprint recognition using pore and edge descriptors, Pattern Recognition Letters 125 (2019) 773–779.

[32] L. Cerkezi, C. Topal, Towards more discriminative features for texture recognition, Pattern Recognition (2020) 107473.

[33] A. R. Backes, D. Casanova, O. M. Bruno, Color texture analysis based on fractal descriptors, Pattern Recognition 45 (5) (2012) 1984–1992.

[34] F. Sandid, A. Douik, Robust color texture descriptor for material recognition, Pattern Recognition Letters 80 (2016) 15–23.

[35] J. Lira, A. Rodriguez, A divergence operator to quantify texture from multi-spectral satellite images, International Journal of Remote Sensing 27 (13) (2006) 2683–2702.

[36] S. K. Roy, B. Chanda, B. B. Chaudhuri, S. Banerjee, D. K. Ghosh, S. R. Dubey, Local directional zigzag pattern: A rotation invariant descriptor for texture classification, Pattern Recognition Letters 108 (2018) 23–30.

[37] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, A. Vedaldi, Describing textures in the wild, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 3606–3613.

[38] A. R. Rivera, J. R. Castillo, O. Chae, Local directional texture pattern image descriptor, Pattern Recognition Letters 51 (2015) 94–100.

[39] P. Banerjee, A. K. Bhunia, A. Bhattacharyya, P. P. Roy, S. Murala, Local neighborhood intensity pattern–a new texture feature descriptor for image retrieval, Expert Systems with Applications 113 (2018) 100–115.

[40] L. Seidenari, G. Serra, A. D. Bagdanov, A. Del Bimbo, Local pyramidal descriptors for image recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 36 (5) (2013) 1033–1040.

[41] V. Balntas, L. Tang, K. Mikolajczyk, Binary online learned descriptors, IEEE Transactions on Pattern Analysis and Machine Intelligence 40 (3) (2017) 555–567.

[42] J. Heinly, D. Enrique, F. Jan-Michael, Comparative evaluation of binary features, in: Proceedings of the European Conference on Computer Vision (ECCV), 2012, pp. 759–773.

[43] R. Arandjelović, A. Zisserman, Three things everyone should know to improve object retrieval, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2012, pp. 2911–2918.

[44] G. Levi, T. Hassner, Latch: learned arrangements of three patch codes, in: 2016 IEEE Winter Conference on Applications of Computer Vision (WACV), IEEE, 2016, pp. 1–9.

[45] V. Balntas, K. Lenc, A. Vedaldi, K. Mikolajczyk, Hpatches: A benchmark and evaluation of handcrafted and learned local descriptors, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 5173–5182.

[46] P. Y. Simard, D. Steinkraus, J. C. Platt, et al., Best practices for convolutional neural networks applied to visual document analysis., in: Proceedings of the Seventh International Conference on Document Analysis and Recognition, Vol. 2, 2003.

[47] T. Sattler, W. Maddern, C. Toft, A. Torii, L. Hammarstrand, E. Stenborg, D. Safari, M. Okutomi, M. Pollefeys, J. Sivic, et al., Benchmarking 6dof outdoor visual localization in changing conditions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 8601–8610.

[48] P. F. Alcantarilla, A. Bartoli, A. J. Davison, Kaze features, in: European Conference on Computer Vision, Springer, 2012, pp. 214–227.

[49] E. Rosten, R. Porter, T. Drummond, Faster and better: A machine learning approach to corner detection, IEEE transactions on pattern analysis and machine intelligence 32 (1) (2008) 105–119.

[50] M. Faraji, J. Shanbehzadeh, K. Nasrollahi, T. B. Moeslund, Extremal regions detection guided by maxima of gradient magnitude, IEEE Transactions on Image Processing 24 (12) (2015) 5401–5415.

[51] S. Leutenegger, M. Chli, R. Y. Siegwart, Brisk: Binary robust invariant scalable keypoints, in: 2011 International conference on computer vision, Ieee, 2011, pp. 2548–2555.

[52] E. Garcia-Fidalgo, A. Ortiz, Hierarchical place recognition for topological mapping, IEEE Transactions on Robotics 33 (5) (2017) 1061–1074.

[53] E. Rosten, T. Drummond, Machine learning for high-speed corner detection, in: European conference on computer vision, Springer, 2006, pp. 430–443.

[54] E. Rublee, V. Rabaud, K. Konolige, G. Bradski, Orb: An efficient alternative to sift or surf, in: 2011 International conference on computer vision, Ieee, 2011, pp. 2564–2571.

[55] A. Geiger, P. Lenz, R. Urtasun, Are we ready for autonomous driving? the kitti vision benchmark suite, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2012, pp. 3354–3361.