Local Feature Augmentation with Cross-Modality Context for Geometric Matching

Anonymous CVPR submission

Paper ID 325

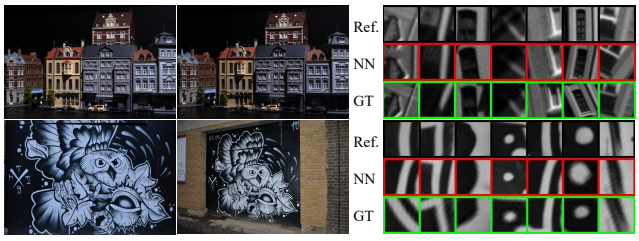
Abstract

Most existing studies on learning local features focus on appropriate descriptions of individual image patches, whereas neglecting spatial location relationship of descriptors in the image. In this paper, we bridge isolated patches by their keypoint coordinates, and go beyond the local detail by introducing context awareness to augment raw local feature. Specifically, we propose a unified learning framework that leverages cross-modality contextual information, consisting of (i) visual context from high-level image understandings, and (ii) geometric context from 2D keypoint distribution. Moreover, we propose an effective technique to alleviate the scale affects of N-pair loss. The proposed augmentation scheme costs only 6% extra time compared with raw local feature description, but improves remarkably on several large-scale benchmarks with diversified scenes, which demonstrates both strong generalization and practicality in geometric matching applications.

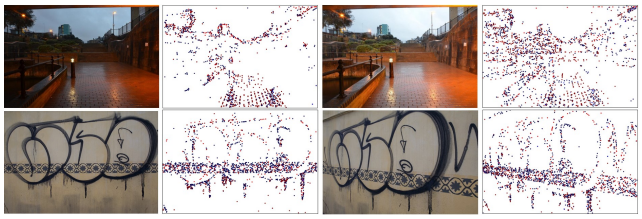
1. Introduction

Designing powerful local feature descriptor is a fundamental problem in applications such as wide-baseline matching [23], image retrieval [26], and structure-from-motion (SfM) [32]. Despite of notable achievement by recent advance, the performance of state-of-the-art learned descriptors is observed to be somewhat saturated on standard benchmarks. As shown in Fig. 1a, due to visual repetitiveness, the matching process often finds nearest neighbors that are hardly distinguishable from true matches unless validated by geometry, e.g., homography. Essentially, such visual ambiguity may not be easily resolved with only local information. In this spirit, we propose to enhance the feature description with prior knowledge, which we refer to as introducing *context awareness* to augment local feature.

As a common practice, a multi-scale-like architecture helps to capture *visual context* of different levels, termed as aggregating domains of multiple sizes by DSP-SIFT [8]



(a)



(b)

Figure 1: (a) Saturated results on standard benchmark [2] by the recent advance [22]. The search of nearest neighbors (NN) returns false matches though visually similar to groundtruth (GT), indicating the limitation of relying on only local visual information. (b) 2D keypoints distribute structurally, on which we human beings are capable of establishing coarse matches even without color information.

and adopted by recent learned advances [44, 19, 37]. Beside of the challenge on properly selecting domain sizes and the concern of deviating the teachings from scale-space theory [21], a naïve multi-scale implementation may cost excessive computation such as doubled inference time and doubled dimensionality as in [44, 19, 37]. Seeking for a more acceptable accuracy-efficiency trade-off, we instead resort to well-studied high-level image representation that has essentially incorporated rich contextual information, and aim to effectively enhance the local feature description with off-the-shelf visual understandings.

In addition to visual information, it would be interesting to exploit context in other modality. In particular, as shown in Fig. 1b, since keypoint is principally designed to be repeatable in the same underlying scene, its distribution thus has revealed comprehensive structure that allows we human

beings to establish coarse matches even without color information. In essence, keypoints can be bridged by their coordinates, from which we can explore *geometric context* to help to alleviate the ambiguity of local visual feature.

Thus far, we have discussed two context candidates, referred to as *visual context* and *geometric context* that incorporate high-level visual understandings and geometric cues of 2D keypoint distribution, respectively. Instead of learning a completely new descriptor, in the present work, we target to flexibly leverage the above context awareness to augment off-the-shelf raw local features, in which process we consider the key challenges threefold:

- Proper integration of local geometric feature and high-level visual understandings. As keypoint description requires sub-pixel accuracy, the integration is not supposed to mix up raw representation of local details.
- Instability of 2D keypoints. Due to image appearance changes, keypoint distribution often suffers from substantial variations of sparsity, non-uniformity or perspective, which raises a challenge on acquiring strong invariance property of the context encoder.
- Effective learning scheme. Input signals and features in different modalities are supposed to be efficiently processed and aggregated in a unified framework.

Finally, regarding practicability, the augmentation is not supposed to be too complex to introduce excessive computational cost, as the local feature description is often regarded as part of preprocessing in practical pipelines.

Although contextual information has been widely applied in computer vision tasks, the challenges faced by local feature learning are substantially different, posing many non-trivial technical and systematic issues to overcome. In this paper, we propose a unified augmentation scheme that leverages and aggregates cross-modality context, of which the contributions are summarized threefold: 1) a novel *visual context encoder* that integrates high-level visual understandings from *regional image representation*, a technique often used by image retrieval [29], 2) a novel *geometric context encoder* that exploits geometric cues from raw 2D keypoint distribution while being robust to complex variations. 3) A novel learning scheme with a new loss that requires no manual parameter tuning and improves the convergence by mitigating scale affects in N-pair loss. To our best knowledge, it is the first work that emphasizes the importance of context awareness in 2D local feature learning.

The proposed augmentation scheme is extensively evaluated and achieves state-of-the-art results on several large-scale benchmarks, including patch-level homography dataset, image-level wild outdoor/indoor scenes and 3D reconstruction image sets, with only 6% extra time cost compared with raw local description, demonstrating both strong generalization ability and practicability.

2. Related Work

Learned local features. Initially, local descriptors are jointly learned with a new metric [9, 44], which is later simplified as direct comparison in Euclidean space [34, 42, 3, 19, 1]. More recently, attention is drawn on efficient training data sampling [37, 24, 11], effective regularizations [37, 46], and geometric shape estimation of input patches [25, 7]. However, most of above methods take *individual* image patches as input, whereas in the present work, we aim to make use of contextual cues beyond the local detail and take advantage of features in multiple modalities.

Context awareness. Although widely introduced in many tasks, context awareness has received little attention in learning 2D local descriptors. In terms of visual context, the central-surround (CS) structure [44, 19, 37] leverages multi-scale information by additionally extracting features from the central part of patch to boost the performance, whereas sacrificing computational efficiency due to doubled extraction time and doubled feature dimensionality. Regarding semantics, previous practice [18] designs a new comparison metric and describes features directly from histogram of semantic labels. On the other hand, a family of studies has focused on finding semantic correspondences [39, 30] across *different* objects of the same category, of which the purpose is substantially different from our geometric matching. Beside of visual information, a recent advance [43] explores to encode motion context for identifying outlier from image correspondences, i.e., 4-d coordinate pairs, whereas we aim to exploit geometric context from *single* image without any reference. Overall, encoding proper context is non-trivial and still unclear in 2D local feature learning.

Point feature learning. In the present work, one of our goals is to explore geometric features from keypoint distribution, we thus resort to PointNet [27] and its variants [28, 5, 43] to consume unordered points. Although great success has been shown in learning tasks on 3D points, there are only few studies exploiting the potential outcome of 2D keypoints. In essence, keypoint structure is not intuitively meaningful and robust, as being highly dependent on the performance of interest point detectors and strongly affected by image variations. However, in descriptor learning, we consider the keypoint location as an important cue that bridge each individual local feature, constructing a unified instance that reveals high-level contextual information.

Loss formulation. Recent local descriptors are often evolved with advanced variants of N-pair losses. Initially, L2-Net [37] adopts a log-likelihood formulation, which is later extended by HardNet [24] with hard negative triplet margin loss. Furthermore, GeoDesc [22] applies an adaptive margin value to improve the convergence in terms of different data sampling strategies, where AffNet [7] approaches the same issue by fixing the distance of hardest

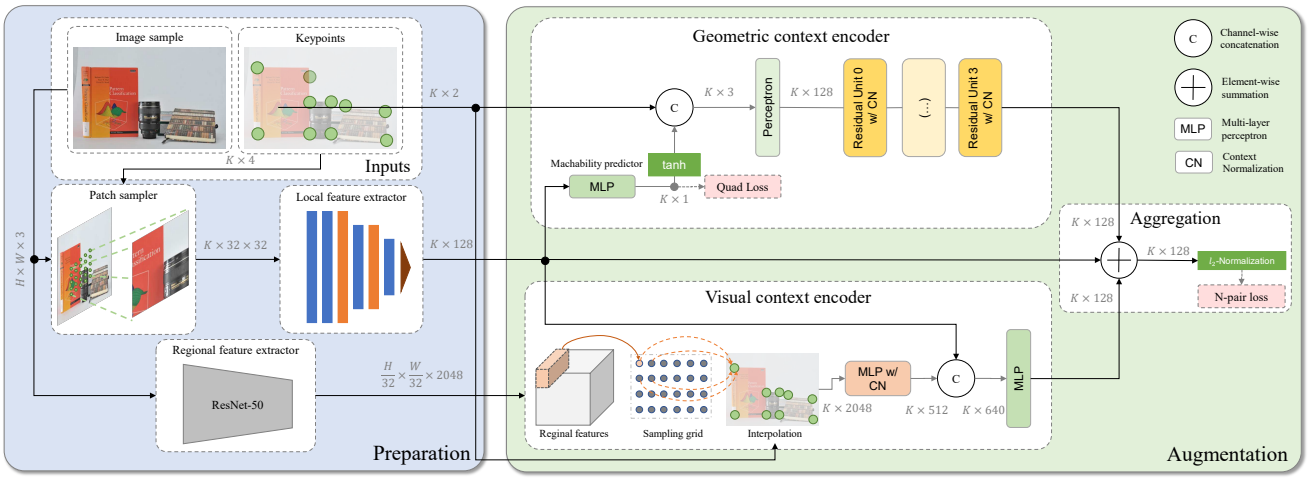


Figure 2: The network architecture of local feature augmentation.

negative sample in the training. Meanwhile, on the other hand, DOAP [11] extends the N-pair loss to a list-wise ranking loss, while [17] points out and studies the scale affects in N-pair losses while introducing additional manual tuning of hyper-parameters. Principally, the loss is supposed to encourage similar patches to be close while dissimilar ones to be distant in the descriptor space. In this spirit, we aim to further resolve scale affects in an self-adaptive manner, without the need of complex heuristics or manual tuning.

3. Local Feature Augmentation

Overview. As illustrated in Fig. 2, the proposed framework consists of two main modules: *preparation* (left) and *augmentation* (right). The *preparation* module provides input signals in different modalities (raw local feature, high-level visual feature and keypoint location), which is then fed to the *augmentation* module and aggregated into compact feature descriptions. At test time, the augmentation needs to be performed once per image, resulting in K feature vectors for K respective keypoints.

3.1. Preparation

Patch sampler. This module takes images and their keypoints as input, producing 32×32 gray-scale patches. Akin to [42, 22], the patch is sampled as applying similarity transformation parameterized by keypoints (coordinates, orientation and scale) from the SIFT detector, implemented by a spatial transformer [16]. The patch has the same size with the supporting region of SIFT descriptor.

Local feature extractor. This module takes image patches as input, producing 128-d feature descriptions as output. We borrow the lightweight 7-layer convolutional networks, as used in several recent works [37, 24, 22].

Regional feature extractor. In contrast to aggregating features of different domain sizes [44, 19, 37], in the present work, we fix the sampling scale of patches, and exploit contextual cues by inspiration of well-studied regional representation in image retrieval tasks [38, 29]. Without the loss of generality, we reuse features from an off-the-shelf deep image retrieval model of ResNet-50 [12]. As in [38], feature maps are extracted from the last bottleneck block, across which each response is regarded as a regional feature vector effectively corresponding to a particular region in the image. As a result, we derive regional features of $\frac{H}{32} \times \frac{W}{32} \times 2048$, where H and W denote original image height and width. The aggregation of regional and local features will be later discussed in Sec. 3.3.

3.2. Geometric context encoder

This module takes K unordered points as input, and outputs K corresponding feature vectors. Each input point is represented as 2D coordinate of the keypoint, and can be associated with other attributes.

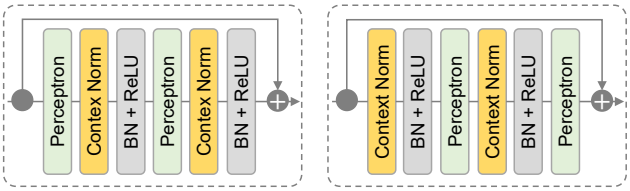
2D point processing. At first glance, 2D keypoint is inappropriate to serve as robust contextual cues, as its presence is heavily dependent on image appearance and thus affected by various image variations. As a result, keypoint distribution depicting the same scene may suffer from significant density or structure variations, as examples shown in Fig. 1b. Hence, acquiring strong invariance property is the key challenge when designing the context encoder.

Initially, we attempt to approach the goal by PointNet [27] and its variants [28, 5]. Although having shown great success on processing 3D point clouds, the family of PointNet methods fails to achieve consistent improvement in terms of processing 2D points (Sec. 5.4.1). Instead, we resort to [43], which originally focuses on outlier rejection in image matching and consumes *putative matches* (4-d coordinate pairs) of image pairs as their network input. In particular, we aim to further explore the potential of context

normalization (CN) proposed in [43], and extend its usability to processing 2D points in *single* image.

Formally, context normalization is a non-parametric operation that simply normalizes feature maps according to their distribution, expressed as $\hat{\boldsymbol{o}}_i^l = \frac{(\boldsymbol{o}_i^l - \boldsymbol{\mu}^l)}{\boldsymbol{\sigma}^l}$, where $\boldsymbol{o}_i^l$ is the output of i-th point in layer l, and $\boldsymbol{\mu}^l, \boldsymbol{\sigma}^l$ are mean and standard deviation of the output in layer l. To equip the operation, we borrow the residual architecture in [43], where each residual unit is built with two perceptrons followed by context and batch normalization, as illustrated in Fig. 3a.

However, the above construction leads to a *non-negative* output from the residual branch that impacts the representational ability as investigated in [13] and witnessed in our experiments (Sec. 5.4.1). Following the teachings of [13], we propose to re-arrange the operations in residual unit by adopting the *pre-activation* construction, which is compatible with the context normalization as presented in Fig. 3b. We then construct four such units as the final geometric context encoder, as shown in Fig. 2. We will show that this simple revision plays an important role to ease the optimization.



(a) Original design (b) Proposed pre-activation

Figure 3: Different designs of residual unit with context normalization, where the proposed construction improves by a considerable margin than its original counterpart.

Intuitively, the observation that context normalization (CN) performs better than PointNet in our task can be interpreted as the 'inexactness' nature of CN, which makes a global association of keypoints with simple statistics about the distribution and is thus less sensitive to the instability of 2D keypoints. In principle, the 'inexactness' suffices to extract the context prior as we essentially expect to establish only *coarse* matches on keypoint distribution.

Matchability predictor. In 3D point cloud processing, low-level color and normal [27] information or more complex geometric attributes [5] are often adopted to enhance the representation. Similarly, associating 2D coordinate input with other meaningful attributes would be promising to boost the performance. However, due to the substantial variations, e.g. perspective change, it is non-trivial to define appropriate intermediate attributes on 2D points.

Although this issue has been merely discussed, we draw inspiration from [10], which poses a problem named *matchability prediction* and targets to decide *whether a keypoint descriptor is matchable before the matching stage*. In practice, the learned matchability can be used to guide the keypoint sampling and accelerate the matching without sacrificing accuracy. In contrast to criteria such as cornerness or edgeness of a keypoint, matchability implies high-level prior knowledge derived from data. Dependent on individual keypoint, matchability can be naturally associated as a representative attribute in addition to 2D coordinate.

Previously in [10], the matchability prediction is casted as a binary classification problem, taking individual descriptor as input and inferred by a random forest. In the present work, we approach this problem with deep learning techniques, and apply a more strict constraint requiring consistent prediction between images. Similar to [31, 45], we resort to an unsupervised learning scheme that aims to rank points. Specifically, given K corresponding keypoint pairs (p_1^n, p_2^n), $n \in [1, K]$ from an image pair, we first extract local features $(\boldsymbol{f}_1^n, \boldsymbol{f}_2^n)$ of each keypoint, then construct *feature quadruples* as $(\boldsymbol{f}_1^i, \boldsymbol{f}_1^j, \boldsymbol{f}_2^i, \boldsymbol{f}_2^j)$, satisfying $i, j \in [1, K], i \neq j$ and holding that:

$$\begin{cases} H(\boldsymbol{f}_1^i) > H(\boldsymbol{f}_1^j) & \& \quad H(\boldsymbol{f}_2^i) > H(\boldsymbol{f}_2^j) \\ & \text{or} \\ H(\boldsymbol{f}_1^i) < H(\boldsymbol{f}_1^j) & \& \quad H(\boldsymbol{f}_2^i) < H(\boldsymbol{f}_2^j) \end{cases}, \tag{1}$$

where $H(:)$ absorbs the raw local feature into a single real-valued matchability, implemented as standard multi-layer perceptrons (MLPs). Here, Cond. 1 aims to preserve a ranking of each keypoint, hence improves the repeatability of prediction. The condition can be re-written as:

$$\begin{aligned} &R(\boldsymbol{f}_1^i, \boldsymbol{f}_1^j, \boldsymbol{f}_2^i, \boldsymbol{f}_2^j) = \\ &(H(\boldsymbol{f}_1^i) - H(\boldsymbol{f}_1^j))(H(\boldsymbol{f}_2^i) - H(\boldsymbol{f}_2^j)) > 0, \end{aligned} \tag{2}$$

the final objective can be obtained with hinge loss:

$$\mathcal{L}_{quad} = \frac{1}{K(K-1)} \sum_{i,j,i \neq j} \max(0, 1 - R(\boldsymbol{f}_1^i, \boldsymbol{f}_1^j, \boldsymbol{f}_2^i, \boldsymbol{f}_2^j)). \tag{3}$$

In the proposed framework, the matchability is learned as an auxiliary task, which is then activated by a $\tanh$ and associated with keypoint coordinate before fed into the encoder, as in Fig. 2. Beside of Eq. 3, the gradient from final augmented feature will flow through the matchability predictor, allowing a joint optimization of the entire encoder. We illustrate the efficacy of matchability in Fig. 4.

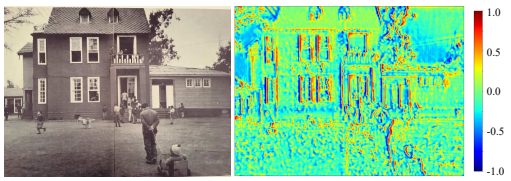


Figure 4: Visualization of matchability responding to the entire image (best viewed in color).

3.3. Visual context encoder

This module takes regional features of $\frac{H}{32} \times \frac{W}{32} \times 2048$ in Sec. 3.1, K raw local features and their location as input, producing K corresponding feature vectors. As widely-adopted in other tasks, the main purpose of this module is to integrate visual information in different levels, e.g., semantics. In our context, the key issue is to handle the regional features and keypoints of different numbers. One option as in [5] is to concatenate global representation of entire image on raw local features, where the global feature in our framework can be derived by applying Maximum Activations of Convolutions (MAC) aggregation [29] which simply max-pools over all dimensions per feature map. However, such compact representation can mess up the raw local description, due to the lack of distinctiveness (Sec. 5.4.1).

To better preserve the regional distinction, we associate regional features to a regular sampling grid on the image, then interpolate $\frac{H}{32} \times \frac{W}{32}$ grid points at coordinates of the K keypoints. For interpolation, we use the inverse distance weighted average based on k nearest neighbors (in default we use $k = 4$), formulated as:

$$\boldsymbol{f}(\hat{p}_i) = \frac{\sum_{j=1}^{k} w(p_j)\boldsymbol{f}(p_j)}{\sum_{j=1}^{k} w(p_j)}, \text{and } w(p_j) = \frac{1}{d(\hat{p}_i, p_j)}, \quad (4)$$

where $\boldsymbol{f}(:)$ is the regional feature located at a certain grid point. $\hat{p}_i$, $i \in [1, N]$ indicates interpolated point, while p_j, $j \in [1, \frac{H}{32} \times \frac{W}{32}]$ indicates original grid point. Next, the dimensionality is reduced by applying point-wise MLPs, where we also insert CN after each perceptron in order to capture global context. Finally, raw local features are concatenated and further mapped by MLPs, forming the final 128-d features. The above process is illustrated in Fig. 2.

3.4. Feature aggregation with raw local feature

To aggregate the above two types of contextual features, similar to the CS structure, one option is to concatenate them together and forms features of, in our case, 384-d (128×3). However, the increased dimensionality will introduce excessive computational cost in the matching stage of $\mathcal{O}(n^2)$ complexity. Instead, as shown in Tab. 2, we propose to combine different feature streams into a single vector by summing and L2-normalizing them in the end, i.e., without the change of feature dimensionality. Beside of the simplicity, such strategy allows flexible use of the augmentation. For example, in situations where regional features are not available, one may aggregate with only geometric context without the need of retraining the model.

4. Learning Scheme

4.1. N-pair loss with softmax temperature

N-pair losses have been primarily used by recent works. Empirically, the subtractive hinge loss [24, 22, 7] has reported better performance, of which the main idea is to push similar samples away from dissimilar ones to a certain *margin* in the descriptor space. However, setting the appropriate margin is tricky, which does not always assure convergence as observed in [22, 7]. More generally, the aspects of making a good loss are studied in [17], from which guidelines are provided about tuning loss coefficients on particular dataset. In this spirit, we aim to ease the pain of parameter searching in [17], and obtain an adaptive loss that allows fast convergence regardless of the learning difficulty.

We use the log-likelihood version of N-pair loss [37] as a base, which originally does not involve any tunable parameter. Formally, given L2-normalized feature descriptors $\mathbf{F}_1 = [\boldsymbol{f}_1^1 \boldsymbol{f}_1^2 ... \boldsymbol{f}_1^N]^T, \mathbf{F}_2 = [\boldsymbol{f}_2^1 \boldsymbol{f}_2^2 ... \boldsymbol{f}_2^N]^T \in \mathbb{R}^{N \times 128}$, the distance matrix $\mathbf{D} = [d_{ij}]_{N \times N}$ can be obtained by $\mathbf{D} = \sqrt{2(1 - \mathbf{F}_1 \mathbf{F}_2^T)}$. By applying, e.g., row-wise softmax, we derive the final loss as:

$$\begin{aligned} \mathcal{L}_{N\text{-}pair} &= -\sum_i \log s_{ii}, \\ \text{where } [s_{ij}]_{N \times N} &= \text{softmax}(2 - \mathbf{D}). \end{aligned} \quad (5)$$

Noted that since input features are L2-normalized, the resulting d_{ij} is bounded between 0 and 2, which causes convergence issues due to the scale sensitivity of softmax function [15]. Similarly, we introduce a single scalar parameter α, referred to as *softmax temperature*, to amend the inability of re-scaling the input. The loss now becomes:

$$[s_{ij}]_{N \times N} = \text{softmax}(\alpha(2 - \mathbf{D})), \quad (6)$$

where α is initialized to 1 and regularized with the same weight decay in the network, which does not require any manual tuning or complex heuristics. In the experiments in Sec. 5.4.2, we show this simple technique improves drastically than the original formulation in [37], whose performance we believe is hindered due to the scale sensitivity of softmax. In the proposed framework, we compute the N-pair loss on augmented features, and obtain the total loss:

$$\mathcal{L}_{total} = \mathcal{L}_{N\text{-}pair} + \lambda \mathcal{L}_{quad}, \quad (7)$$

where we choose $\lambda = 1$ in the experiment.

4.2. Learning with noisy keypoints

The training of proposed framework, apparently, needs to be conducted between image pairs instead of isolated patches, which is referred to as simulating image matching in [22]. However, the simulation in [22] is not complete, as it considers only matchable keypoints that are paired with correspondences, whereas in real situation, only a subset of keypoints is repeatable in other images. In practice, as illustrated in Fig. 5, we divide keypoints obtained from SfM as in [42, 22] into three categories: i) *Matchable*: repeatable and verified by SfM; ii) *Undiscovered*: repeatable but

did not survive the SfM. iii) *Unrepeatable*: unable to be re-detected in other images. In the training, we randomly sample a number of matchable keypoints as well as some undiscovered and unrepeatable (*noisy keypoints*) to have a complete simulation, which is necessary to acquire the generalization ability, otherwise the training considers only ideal setting which is inconsistent with real applications. In this setting, the loss is still computed on matchable keypoints, of which the formal expression is provided in the appendix.

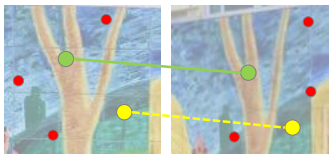


Figure 5: We divide keypoints after SfM into three categories: matchable (green), undiscovered (yellow) and unrepeatable (red), and aim to perform a complete simulation in training that incorporates all three types of keypoints.

5. Experiments

5.1. Implementation

Training details. Although the framework is end-to-end trainable, we *fix* the local and regional feature extractors in Sec. 3.1 during the training, in order to make a clear demonstration of the proposed augmentation scheme. We train the networks using SGD with a base learning rate of 0.05, weight decay of 0.0001 and momentum at 0.9. The learning rate exponentially decays by 0.1 for every 100k steps. The batch size is set to 2, and each time 1024 keypoints are randomly sampled including random numbers of matchable and noisy keypoints. Input patches are standardized to have zero mean and unit norm, while input keypoint coordinates are normalized to $[-1, 1]$ regarding the image size.

Training dataset. Although UBC Phototour [4] is used as a common practice, this dataset consists of only three scenes with limited diversity of keypoint distribution. In order to achieve better generalization ability, we resort to large-scale photo-tourism dataset [40, 29] as in [42, 22], and generate groundtruth matches from SfM. We manually exclude the data that is used in the evaluation.

Data augmentation. For image patches, we apply random affine transformations including rotation, anisotropic scaling and translation w.r.t. the detection scale. For keypoint augmentation, we perturb the coordinate with random homography transformation as in [6].

5.2. Evaluation datasets

Homography dataset. HPatches [2] is a large-scale patch dataset for evaluating local features regarding illumination and viewpoint changes. As groundtruth homographies and raw images are provided, HPatches can also be used to evaluate image matching performance, which we accordingly refer to as HPSequences as in [20], consisting of 116 sequences and 580 image pairs.

Wild dataset. Similar to settings in [43], we use outdoor YFCC100M [36] (1000 pairs) and indoor SUN3D [41] (539 pairs). The two datasets additionally introduce variations such as self-occlusions, and in particular, repetitive or feature-poor patterns in indoor scenes, which is generally considered challenging for sparse keypoint methods.

SfM dataset. Following [33], we evaluate on SfM dataset such as well-known *Fountain* and *Herzjesu* [35], or landmark collections [40]. We integrate the proposed framework into SfM pipeline, i.e., COLMAP [32], and use the keypoints provided in [33] to compute the local features.

5.3. Evaluation protocols

Patch level. Following the same protocols of HPathces [2], we use mean average precision (mAP) for its three subtasks , including patch verification, matching, and retrieval.

Image level. For HPSequences, we use *Recall = # Correct Matches / # Correspondences* defined in [14], to quantify the image matching performance, where *# Correct matches* are matches found by nearest neighbor searching and verified by groundtruth geometry, e.g., homography, while *# Correspondences* are matches that should have been identified by the given keypoint locations. Following [14], a match point is determined to be correct if it is within 2.5 pixels from the wrapped keypoint in the reference image. We use a standard SIFT detector to localize the keypoints, of which the number is randomly sampled to 2048. For YFCC100M [36] and SUN3D [41], we follow the same setting in [43] and report the median number of inlier matches after RANSAC for each dataset.

Reconstruction level. For clarity, we report metrics in [33] that quantify the completeness of SfM, including the number of registered images (*# Registered*), sparse points (*# Sparse Points*) and image observations (*# Observations*).

5.4. Ablation study

5.4.1 Design of context encoder

In this section, we evaluate two splits of HPSequences [2]: *illumination (i)* and *viewpoint (v)*, regarding different image transformations. We report *Recall* as defined in Sec. 5.3. If not specified, we use GeoDesc [22] as a baseline model (*baseline (GeoDesc)*) to extract raw local features, whose parameters are *fixed* during the training of augmentation.

Visual context. We compare four designs, including i) *CS (256-d)*: the central-surround (CS) structure [44, 19, 37] as described in Sec. 2, which leverages visual information of

Vsiual context encoder			Geometric context encoder			Comparison with other methods		
Strategy	*Recall i/v*		*Network architecture*	*Recall i/v*		*Method*	*Recall i/v*	
baseline (GeoDesc)	59.27	71.44	baseline (GeoDesc)	59.27	71.44	SIFT [21]	47.41	53.19
CS (256-d) [44, 19, 37]	59.64	71.47	PointNet [27]	59.61	71.16	L2-Net [37]	47.55	54.10
w/ global feature [5]	58.92	71.22	w/ CN (pre) + xy	61.05	72.47	HardNet [24]	57.61	63.45
w/ regional feature	63.45	73.57	w/ CN (pre) + xy + raw local feature	60.61	72.69	GeoDesc [22]	59.27	71.44
w/ regional feature + CN	**63.79**	**73.63**	w/ CN (orig.) + xy + matchability	59.94	71.25	**multi-context**	**65.35**	**74.70**
			w/ CN (pre) + xy + matchability	**61.52**	**72.83**	**multi-context+**	**65.76**	**75.64**

Table 1: Comparisons on HPSequences [2] of different designs of visual and geometric context encoder, and the performance of entire augmentation scheme. 'i/v' denotes two evaluations on *illumination* and *viewpoint* sequences, respectively.

different domain sizes. ii) *w/ global feature*: the integration with global features [5], which is originally designed for improving 3D local descriptors. iii) *w/ regional feature*: the proposed integration with interpolated regional features, and its variant iv) *w/ regional feature + CN*: with context normalization to exploit global visual information.

As shown in Tab. 1 (left columns), the CS structure [44, 19, 37] delivers only marginal improvements despite of the doubled dimensionality. By contrast, though being effective in 3D descriptor learning, the integration with global features [5] instead harms the performance, which we attribute to the weak relevance of local geometric and global semantic features. Finally, the proposed integration with interpolated regional features clearly shows advantages as it better preserves distinctions from a smaller visual scale. Moreover, to strengthen global context awareness, we show that the performance can be further boosted by equipping context normalization to associate regional features.

Geometric context. We study five options: i) PointNet-like architecture, i.e., segmentation networks in [27] without the final classifier. ii) Pre-activated context normalization (CN) networks in Sec. 3.2 with 2D xy input, and its variants iii) with additional raw local feature input or iv) with matchability. We also compare the use of pre-activation of the residual unit in context normalization networks.

As presented in Tab. 1 (middle columns), though widely used in processing 3D points, PointNet [27] does not work well in our context, where the similar phenomenon is also observed in [43] when processing 2D correspondences. Besides, it is noticed that input with additional raw local feature does not help to boost the performance, which we attribute to the weak relevance between local features as extracted from different levels of scale space pyramid. Instead, the cooperation with matchability is beneficial, as matchability is more interpretable as a high-level abstraction of local feature. Finally, the pre-activation is clearly a preferable alternative than the original design in this task.

Integration with multiple context. Finally, we evaluate the full augmentation with both visual and geometric context (*multi-context*). As shown in Tab. 1 (right columns), the simple summation aggregation in Sec. 3.4 effectively takes advantage of both context, delivering remarkable improvements over the state of the art.

5.4.2 Efficacy of softmax temperature

To make a clear demonstration on HPathces [2], we train *only* the local feature extractor with the proposed loss and adopt image matching simulation as in Sec. 4.2. We compare different losses including: i) the proposed loss and ii) its original form [37] without scale temperature, also iii) the loss in [17] with its original parameters.

	SIFT [21]	L2-Ne [37]	HardNet [24]	GeoDesc [22]	w/ loss [37]	w/ loss [17]	proposed
Verification, mAP [%]							
Easy	80.0	91.4	93.6	**94.0**	83.3	87.5	93.6
Hard	59.2	83.7	87.6	91.8	78.9	82.1	**91.8**
Tough	44.9	72.1	77.3	87.6	72.6	73.9	**88.4**
Mean	61.4	82.4	86.2	91.1	78.3	81.2	**91.3**
Matching, mAP [%]							
Easy	46.7	63.2	69.7	69.5	35.2	52.8	**70.1**
Hard	20.4	43.7	53.3	60.0	22.8	40.8	**62.3**
Tough	0.09	26.4	35.4	48.0	13.9	28.1	**51.5**
Mean	25.5	44.5	52.8	59.1	23.9	40.5	**61.3**
Retrieval, mAP [%]							
Easy	64.7	78.6	81.9	80.7	56.2	72.2	**81.8**
Hard	37.9	65.44	71.7	75.9	47.0	65.2	**78.4**
Tough	22.7	48.6	55.8	67.9	37.3	54.8	**72.1**
Mean	41.7	64.2	69.8	74.9	46.8	64.0	**77.4**

Table 2: Evaluation results on HPatches [2] of three complementary tasks: patch verification, matching and retrieval.

As shown in Tab. 2, the proposed loss delivers notable improvements over the previous best-performing GeoDesc [22] under similar training settings except for the loss design. Besides, the proposed loss clearly shows better convergence compared with [37] and [17]. Although we suspect that the loss in [17] may perform better with careful parameter searching, the proposed loss is advantageous due to its self-adaptivity without the need of complex heuristics or manual tuning. In addition, once equipped with the resulting model as a base, the augmentation results can be further improved by a healthy margin, denoted as *multi-context+* in Tab. 1 (right columns). We will use this model to complete the following experiments.

5.5. Generalization

Wild dataset. The evaluation results on two challenging datasets (*outdoor* YFCC100M [36] and *indoor* SUN3D [41]) are presented in Tab. 3. The proposed multi-context augmentation delivers $\sim$35% and $\sim$125% improvements over the previous state of the art, which effectively

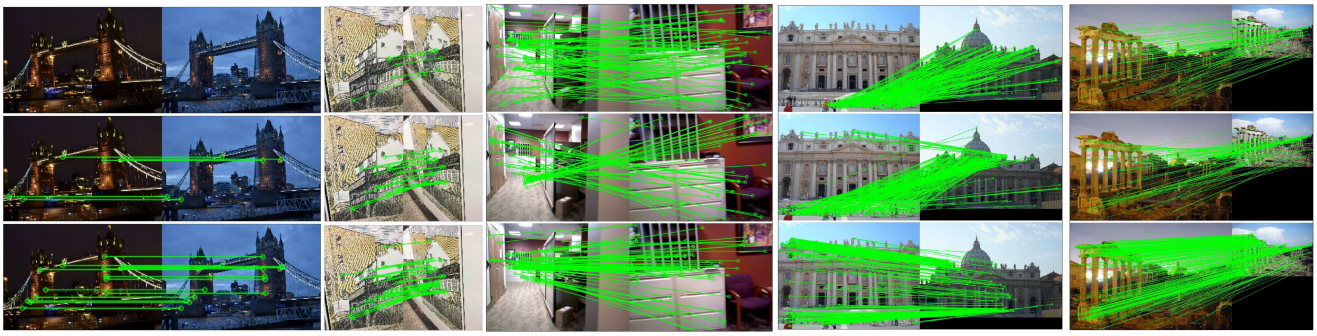


Figure 6: Matching results after RANSAC in different challenging scenarios. From top to bottom: SIFT, GeoDesc and ours. The augmented feature not only helps to find more inlier matches, but allows a more accurate recovery of camera geometry.

demonstrates the strong generalization ability of the learned context features in practical scenes.

	SIFT	L2-Net	HardNet	GeoDesc	Ours
	median number of inlier matches				
indoor	138	153	239	271	**365**
outdoor	168	173	219	214	**482**

Table 3: Evaluation results on wild datasets: *indoor* SUN3D [41] and *outdoor* YFCC100M [36].

SfM dataset. We further demonstrate the improvement in complex SfM pipeline. As shown in Tab. 4, the integration of augmented feature generalizes well among different scenes even in large-scale SfM tasks, meanwhile consistently boosts the completeness of sparse reconstruction. Some matching results are presented in Fig. 6, and more visualizations can be found in the appendix.

		# Images	# Registered	# Sparse Points	# Observations
Fountain	*SIFT*	11	11	10,004	44K
	GeoDesc		11	16,687	83K
	Ours		11	**16,965**	**84K**
Herzjesu	*SIFT*	8	8	4,916	19K
	GeoDesc		8	8,720	38K
	Ours		8	**9,429**	**40K**
South Building	*SIFT*	128	128	62,780	353K
	GeoDesc		128	170,306	887K
	Ours		128	**174,359**	**893K**
Roman Forum	*SIFT*	2,364	1,407	242,192	1,805K
	GeoDesc		1,566	770,363	5,051K
	Ours		**1,571**	**848,319**	**5,484K**
Alamo	*SIFT*	2,915	743	120,713	1,384K
	GeoDesc		893	353,329	3,159K
	Ours		**921**	**424,348**	**3,488K**

Table 4: Evaluation results on SfM dataset [33].

5.6. Discussions

Invariance property. We again use *Recall* and evaluate on Heinly benchmark [14] to quantify the invariance property. As shown in Tab 5, the proposed method improves remarkably over the previous best-performing descriptor, except for some minor underperformance regarding *Rotation* change where images are rotated up to 180°, which may be caused by the essential inability of being fully rotation-invariant especially for the regional feature extractor.

Computational cost. Towards practicability, we only use basic and shallow MLPs or non-parametric context normalization in our framework design, which thus introduces little computational overhead, i.e., $\sim$6% time cost on a NVIDIA GTX 1080 compared with raw local feature, as reported in Tab. 6. Besides, we consider that regional features are often off-the-shelf in practical pipelines, e.g., from a retrieval model deployed in SfM pipeline for accelerating image matching, which can be thus reused without introducing extra cost, achieving system-level efficiency and integrity.

	SIFT	GeoDesc	Ours
JPEG	60.7	66.1	**77.8**
Blur	41.0	47.7	**56.9**
Exposure	78.2	86.4	**87.8**
Day-Night	29.2	*39.6*	**44.6**
Scale	81.2	85.8	**87.9**
Rotation	82.4	**87.6**	87.3
Scale-Rotation	29.6	33.7	**38.0**
Planar	48.2	59.1	**61.3**

Table 5: Evaluation results regrading different transformations on Heinly benchmark [14].

	Preparation		*Augmentation*		
	local feat.	**regional feat.**	**geo. context**	**vis. context**	**multi-context**
Time (ms)	351	49	8	14	21
FLOPs (B)	802.9	123.4	3.0	13.9	16.9
Params (M)	2.4	24.5	0.2	3.1	3.3

Table 6: The computational cost of proposed framework. The evaluation tests 10k keypoints and 896×896 images.

6. Conclusion

In contrast to current trends, we have addressed the importance of introducing *context awareness* in learning local descriptors. The augmentation framework takes keypoint location, raw local and high-level regional feature as input, from which two types of context are encoded, including *geometric* and *visual* context. The training process adopts a novel learning scheme and a new loss that is self-adaptive to the task difficulty. We have extensively evaluated the proposed framework on several large-scale datasets that cover diverse practical scenes, outperforming the state of the art by a significant margin and showing strong generalization and the practicability of the proposed method.

References

[1] V. Balntas, E. Johns, L. Tang, and K. Mikolajczyk. Pn-net: Conjoined triple deep network for learning local image descriptors. In *arXiv*, 2016. 2

[2] V. Balntas, K. Lenc, A. Vedaldi, and K. Mikolajczyk. Hpatches: A benchmark and evaluation of handcrafted and learned local descriptors. In *CVPR*, 2017. 1, 6, 7

[3] V. Balntas, E. Riba, D. Ponsa, and K. Mikolajczyk. Learning local feature descriptors with triplets and shallow convolutional neural networks. In *BMVC*, 2016. 2

[4] M. Brown and D. G. Lowe. Automatic panoramic image stitching using invariant features. In *IJCV*, 2007. 6

[5] H. Deng, T. Birdal, and S. Ilic. Ppfnet: Global context aware local features for robust 3d point matching. In *CVPR*, 2018. 2, 3, 4, 5, 7

[6] D. DeTone, T. Malisiewicz, and A. Rabinovich. Deep image homography estimation. In *arXiv*, 2016. 6, 11

[7] J. M. Dmytro Mishkin, Filip Radenovic. Repeatability is not enough: learning discriminative affine regions via discriminability. In *ECCV*, 2018. 2, 5

[8] J. Dong and S. Soatto. Domain-size pooling in local descriptors: Dsp-sift. In *CVPR*, 2015. 1

[9] X. Han, T. Leung, Y. Jia, R. Sukthankar, and A. C. Berg. Matchnet: Unifying feature and metric learning for patch-based matching. In *CVPR*, 2015. 2

[10] W. Hartmann, M. Havlena, and K. Schindler. Predicting matchability. In *CVPR*, 2014. 4

[11] K. He, Y. Lu, and S. Sclaroff. Local descriptors optimized for average precision. In *CVPR*, 2018. 2, 3

[12] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 3

[13] K. He, X. Zhang, S. Ren, and J. Sun. Identity mappings in deep residual networks. In *ECCV*, 2016. 4

[14] J. Heinly, E. Dunn, and J.-M. Frahm. Comparative evaluation of binary features. In *ECCV*. 2012. 6, 8

[15] E. Hoffer, I. Hubara, and D. Soudry. Fix your classifier: the marginal value of training the last weight layer. In *ICLR*, 2018. 5

[16] M. Jaderberg, K. Simonyan, A. Zisserman, et al. Spatial transformer networks. In *NIPS*, 2015. 3

[17] M. Keller, Z. Chen, F. Maffra, P. Schmuck, and M. Chli. Learning deep descriptors with scale-aware triplet networks. In *CVPR*, 2018. 3, 5, 7

[18] N. Kobyshev, H. Riemenschneider, and L. Van Gool. Matching features correctly through semantic understanding. In *3DV*, 2014. 2

[19] B. Kumar, G. Carneiro, I. Reid, et al. Learning local image descriptors with deep siamese and triplet convolutional networks by minimising global loss functions. In *CVPR*, 2016. 1, 2, 3, 6, 7

[20] K. Lenc and A. Vedaldi. Large scale evaluation of local image feature detectors on homography datasets. In *BMVC*, 2018. 6

[21] D. G. Lowe. Distinctive image features from scale-invariant keypoints. 2004. 1, 7, 11, 12

[22] Z. Luo, T. Shen, L. Zhou, S. Zhu, R. Zhang, Y. Yao, T. Fang, and L. Quan. Geodesc: Learning local descriptors by integrating geometry constraints. In *ECCV*, 2018. 1, 2, 3, 5, 6, 7, 11, 12

[23] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. In *Image and vision computing*, 2004. 1

[24] A. Mishchuk, D. Mishkin, F. Radenovic, and J. Matas. Working hard to know your neighbor's margins: Local descriptor learning loss. In *NIPS*, 2017. 2, 3, 5, 7

[25] K. Moo Yi, Y. Verdie, P. Fua, and V. Lepetit. Learning to assign orientations to feature points. In *CVPR*, 2016. 2

[26] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *CVPR*, 2006. 1

[27] C. R. Qi, H. Su, K. Mo, and L. J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *CVPR*, 2017. 2, 3, 4, 7

[28] C. R. Qi, L. Yi, H. Su, and L. J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *NIPS*, 2017. 2, 3

[29] F. Radenović, G. Tolias, and O. Chum. Cnn image retrieval learns from bow: Unsupervised fine-tuning with hard examples. In *ECCV*, 2016. 2, 3, 5, 6, 11

[30] I. Rocco, R. Arandjelovic, and J. Sivic. End-to-end weakly-supervised semantic alignment. In *CVPR*, 2018. 2

[31] N. Savinov, A. Seki, L. Ladicky, T. Sattler, and M. Pollefeys. Quad-networks: unsupervised learning to rank for interest point detection. In *CVPR*, 2017. 4

[32] J. L. Schönberger and J.-M. Frahm. Structure-from-motion revisited. In *CVPR*, 2016. 1, 6

[33] J. L. Schönberger, H. Hardmeier, T. Sattler, and M. Pollefeys. Comparative evaluation of hand-crafted and learned local features. In *CVPR*, 2017. 6, 8

[34] E. Simo-Serra, E. Trulls, L. Ferraz, I. Kokkinos, P. Fua, and F. Moreno-Noguer. Discriminative learning of deep convolutional feature point descriptors. In *CVPR*, 2015. 2

[35] C. Strecha, W. Von Hansen, L. Van Gool, P. Fua, and U. Thoennessen. On benchmarking camera calibration and multi-view stereo for high resolution imagery. In *CVPR*, 2008. 6

[36] B. Thomee, D. A. Shamma, G. Friedland, B. Elizalde, K. Ni, D. Poland, D. Borth, and L.-J. Li. Yfcc100m: The new data in multimedia research. In *CACM*, 2016. 6, 7, 8, 11, 12

[37] Y. Tian, B. Fan, F. Wu, et al. L2-net: Deep learning of discriminative patch descriptor in euclidean space. In *CVPR*, 2017. 1, 2, 3, 5, 6, 7

[38] G. Tolias, R. Sicre, and H. Jégou. Particular object retrieval with integral max-pooling of cnn activations. In *ICLR*, 2016. 3

[39] N. Ufer and B. Ommer. Deep semantic feature matching. In *CVPR*, 2017. 2

[40] K. Wilson and N. Snavely. Robust global translations with 1dsfm. In *ECCV*, 2014. 6

[41] J. Xiao, A. Owens, and A. Torralba. Sun3d: A database of big spaces reconstructed using sfm and object labels. In *ICCV*, 2013. 6, 7, 8, 11, 12

[42] K. M. Yi, E. Trulls, V. Lepetit, and P. Fua. Lift: Learned invariant feature transform. In *ECCV*, 2016. 2, 3, 5, 6

[43] K. M. Yi, E. Trulls, Y. Ono, V. Lepetit, M. Salzmann, and P. Fua. Learning to find good correspondences. In *CVPR*, 2018. 2, 3, 4, 6, 7, 11

[44] S. Zagoruyko and N. Komodakis. Learning to compare image patches via convolutional neural networks. In *CVPR*, 2015. 1, 2, 3, 6, 7

[45] L. Zhang and S. Rusinkiewicz. Learning to detect features in texture images. In *CVPR*, 2018. 4

[46] X. Zhang, X. Y. Felix, S. Kumar, and S.-F. Chang. Learning spread-out local feature descriptors. In *ICCV*, 2017. 2

A. Supplementary appendix

A.1 Implementation details

Network architecture details. In terms of the matchability predictor, we construct simple 4-layer MLPs whose output node numbers are 128, 32, 32, 1, respectively. The visual context encoder is composed of of two MLPs, located before/after the concatenation with raw local features. We insert context normalization only into the former MLPs in the way shown in Fig. 7, while insertion in the latter one is observed to harm the performance.

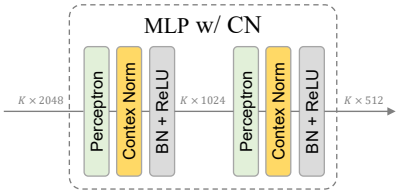


Figure 7: The construction of MLP module with context normalization in visual context encoder.

Performance of the retrieval model. The retrieval model is trained on *Large-Scale Landmark Recognition Challenge*[1], which consists of more than 1M landmark images. Instead of adopting the training scheme in [29], we find that the model pretrained on landmark classification task (containing 15K classes) suffices to produce satisfactory results in practice. We have validated the retrieval model with MAC aggregation on standard Oxford dataset, resulting in mAP of 0.83 on par with [29] whose mAP is 0.80. The performance of proposed augmentation is expected to be further boosted with the evolution of its high-level feature extractor, which we leave as a future work to further achieve the system-level efficacy.

Keypoint coordinate augmentation. Similar to [6], we choose to use the 4-point parameterization, which represents a homography as follows:

$$H_{4point} = \begin{Bmatrix} u_1 + \triangle u_1 & v_1 + \triangle v_1 \\ u_2 + \triangle u_2 & v_2 + \triangle v_2 \\ u_3 + \triangle u_3 & v_3 + \triangle v_3 \\ u_4 + \triangle u_4 & v_4 + \triangle v_4 \end{Bmatrix},$$

where $(u_1, v_1), (u_2, v_2), (u_3, v_3), (u_4, v_4)$ are four corner points at $(-1, 1), (1, 1), (-1, -1), (1, -1)$, and $\triangle u_i, \triangle v_i$ are random variables between $(-s, s)$. One can easily convert H_{4point} to a standard 3×3 homography by, e.g., normalized Direct Linear Transform (DLT) algorithm. We choose $s = 0.5$, which means that the keypoint set can be perturbed by a maximum of one quarter of the total image size. We then apply the random homography on the keypoint coordinate before fed into geometric context encoder.

[1] https://landmarkscvprw18.github.io/

Loss computation with noisy keypoints. Formally, given index sets $C_m = \{i_1, ..., i_{K_m}\}$ and $C_n = \{i_1, ..., i_{K_n}\}$, where K_m and K_n are numbers of matchable and noisy keypoints for an image pair, the losses of Eq. 3 and Eq. 5 are now rewritten as:

$$\mathcal{L}'_{quad} = \frac{1}{K_m(K_m - 1)} \sum_{i,j \in C_m, i \neq j} \max(0, 1 - R(\boldsymbol{f}_1^i, \boldsymbol{f}_1^j, \boldsymbol{f}_2^i, \boldsymbol{f}_2^j)), \tag{8}$$

and

$$\mathcal{L}'_{N\text{-}pair} = -\sum_{i \in C_m} \log s_{ii}. \tag{9}$$

Subsequently, adding noisy keypoints will influence $\mathcal{L}_{N\text{-}pair}$ as all keypoints have been cross-paired, while it also affects the encoding of geometric context.

A.2 Training with softmax temperature

We plot the growth of softmax temperature and its respective loss decrease in Fig. 8. As can be seen, the softmax temperature fast grows at the beginning and gradually converges to a constant value. As mentioned in Sec. 4.1, the softmax temperature is regularized with the same network weight decay, whereas we have observed that the eschewing of regularization does not harm the performance but results in a larger temperature value, e.g., $\sim$42.

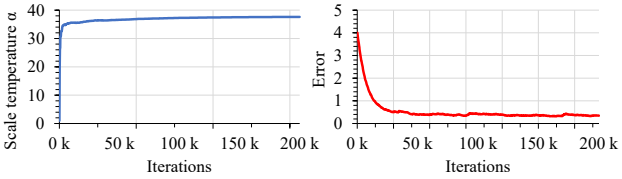


Figure 8: Left: the growth of scale temperature. Right: the respective decrease of loss.

A.3 Ratio test

In previous experiments of image matching, we did not apply any outlier rejection (e.g., cross check, ratio test [21]) for all methods for fair comparison, whereas the early rejection is generally crucial and necessary to later geometry computation, e.g., recovering camera pose. In particular, ratio test [21] has demonstrated remarkable success, we thus follow the practice in [22] to determine the ratio criteria of proposed augmented feature. Specifically, given *# Correct Matches* defined in Sec. 5.3, we test on HPSequence and aim to find a proper ratio that achieves *Precision = # Putative Matches / # Correct Matches* similar to SIFT. As a result, we choose 0.89 for the proposed descriptor.

To demonstrate the efficacy of obtained ratio, we evaluate on the wild indoor/outdoor data [41, 36] with an error metric of relative camera pose. Following the protocols defined in [43], we use mean average precision (mAP) of a certain threshold (e.g., 20°) to quantify the error of rotation

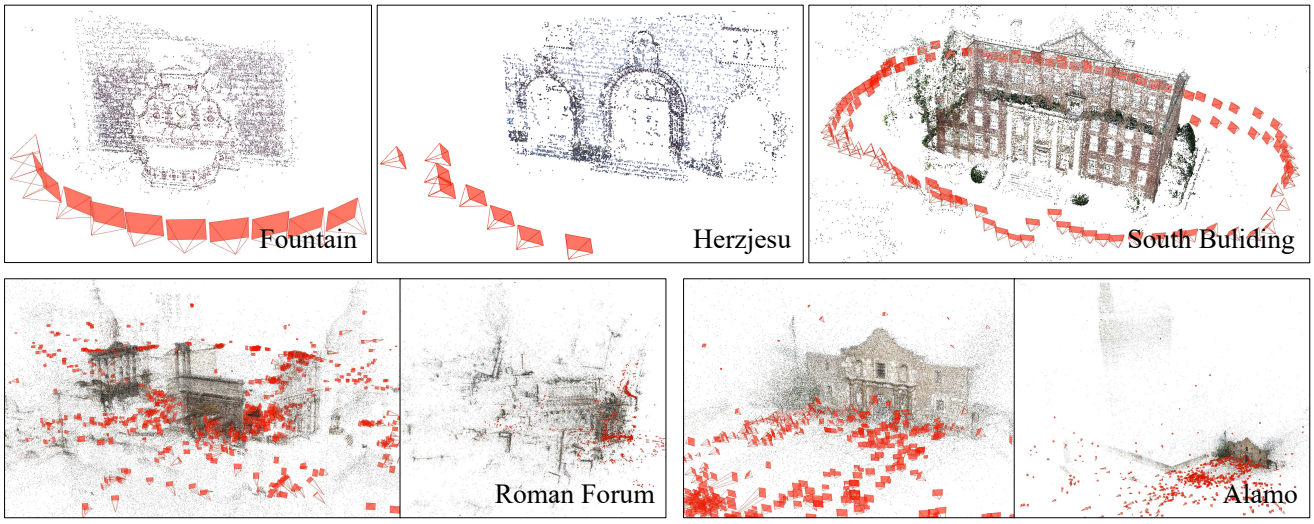


Figure 9: Visualizations of SfM results of Sec. 5.5.

	SIFT	GeoDesc	Ours
mAP of pose (error threshold 20°)			
indoor	37.4	41.8	**42.9**
outdoor	17.9	20.5	**22.5**

Table 7: Pose evaluation on wild datasets with ratio test applied: *indoor* SUN3D [41] and *outdoor* YFCC100M [36].

and translation. For comparison, we use ratio criteria of 0.80 for SIFT [21] and 0.89 for GeoDesc [22], and present evaluation results in Tab. 7, showing consistent improvements on pose estimation with proper outlier rejection.

A.4 Different domain sizes

Somewhat counter-intuitively, the CS structure improves marginally on image matching as reported in Tab. 1. To further study this phenomenon, we compare the patch sampling from different domain sizes, including the original SIFT's ($1\times$) as used in previous experiments, half ($0.5\times$) or double ($2\times$) sizes. We also compare the aggregation of multiple sizes, i.e., the original and halved $(1+0.5)\times$ or the original and doubled $(1+2)\times$. Instead of concatenating features as used by CS structure, we apply the simple summing-and-normalizing aggregation in Sec. 3.4 to avoid increasing the dimensionality. We experiments with our *context+* model, and as shown in Tab. 8, when only single size is adopted, the original '$1\times$' performs best as consistent with the training setting. In addition, when combining a larger size, we can further boost the proposed method by a considerable margin, yet leading to excessive computational cost and doubling the inference time. In practice, it is compatible with the proposed framework and can be applicable where high accuracy is in demand.

domain size	Recall i/v	
$0.5\times$	59.27	68.19
$2\times$	62.17	71.67
$1\times$	**65.76**	**75.64**
$(1+0.5)\times$	65.32	75.05
$(1+2)\times$	**66.72**	**77.30**

Table 8: The efficacy of extracting local features from different domain sizes.

A.5 Invariance of density change

We further demonstrate the robustness regarding density change on HPSequences, of which images are feature-rich and have keypoints up to 15k. Beside of sampling keypoints of different numbers, we consider a more challenging case where *no sampling* and *all detected keypoints* are used. As presented in Fig. 10, the proposed method delivers consistent improvements in terms of all cases, which demonstrates the reliable invariance property acquired by context encoders.

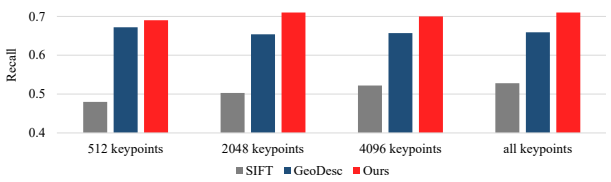
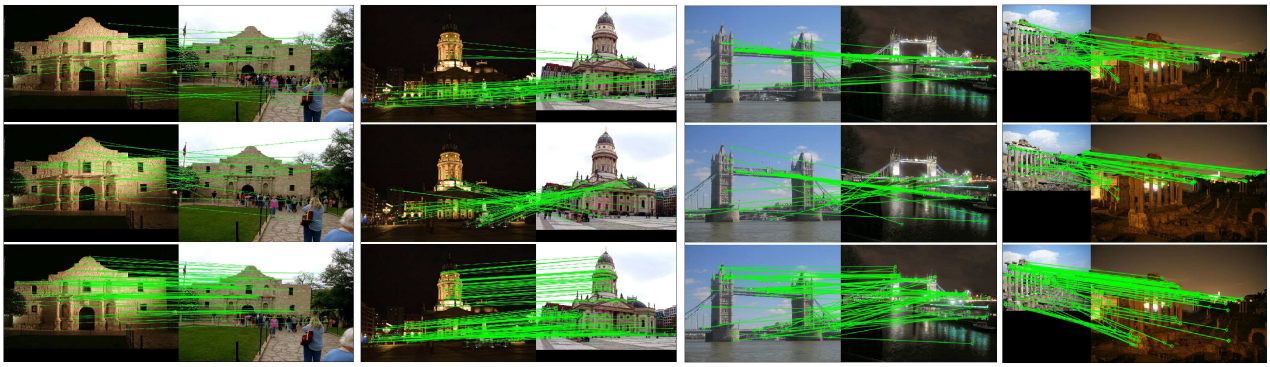


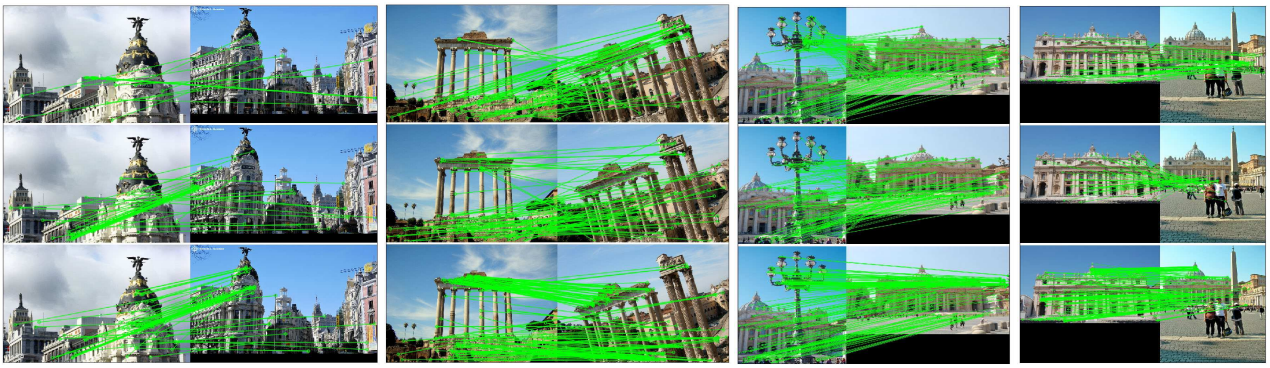
Figure 10: The performance of proposed augmentation scheme regarding density change of keypoints.

A.6 More visualizations

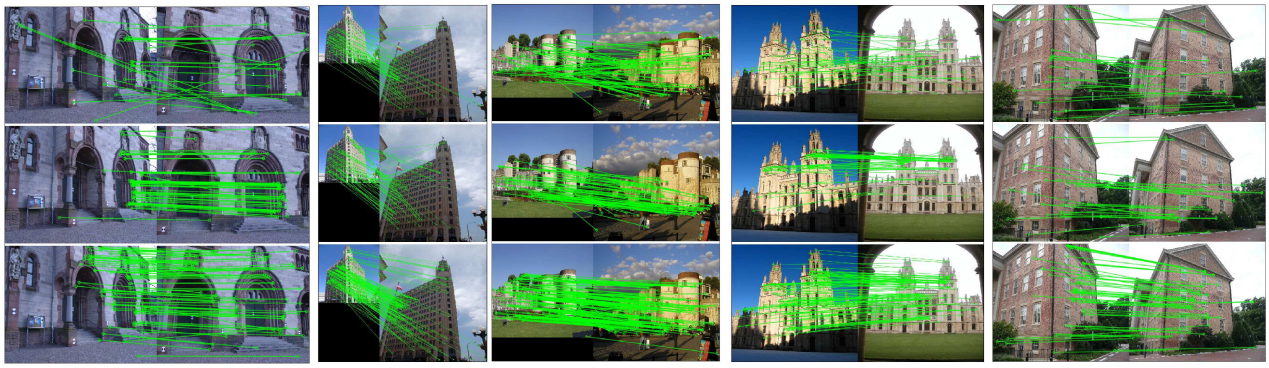
We have provided more visualizations of previous experiments in Fig. 9 (SfM results in Sec. 5.5) and Fig. 11 (image matching results w.r.t different image transformations).



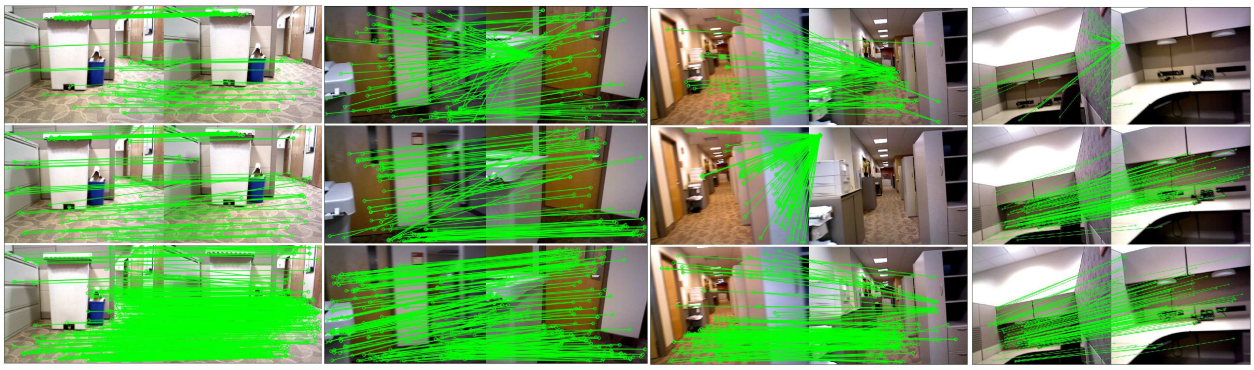
Illumination change



Scale or rotation change



Perspective change



Indoor scene (repetitive or texture-less pattern)

Figure 11: Image matching results after RANSAC. From top to bottom: SIFT, GeoDesc and proposed augmented feature.