NeSS-ST: Detecting Good and Stable Keypoints with a Neural Stability Score and the Shi-Tomasi detector

Konstantin Pakulev
Skolkovo Institute of Science and Technology
Konstantin.Pakulev@skoltech.ru

Alexander Vakhitov
SLAMcore
alexander.vakhitov@gmail.com

Gonzalo Ferrer
Skolkovo Institute of Science and Technology
G.Ferrer@skoltech.ru

Abstract

Learning a feature point detector presents a challenge both due to the ambiguity of the definition of a keypoint and, correspondingly, the need for specially prepared ground truth labels for such points. In our work, we address both of these issues by utilizing a combination of a hand-crafted Shi-Tomasi detector, a specially designed metric that assesses the quality of keypoints, the stability score (SS), and a neural network. We build on the principled and localized keypoints provided by the Shi-Tomasi detector and learn the neural network to select good feature points via the stability score. The neural network incorporates the knowledge from the training targets in the form of the neural stability score (NeSS). Therefore, our method is named NeSS-ST since it combines the Shi-Tomasi detector and the properties of the neural stability score. It only requires sets of images for training without dataset pre-labeling or the need for reconstructed correspondence labels. We evaluate NeSS-ST on HPatches, ScanNet, MegaDepth and IMC-PT demonstrating state-of-the-art performance and good generalization on downstream tasks. The project repository is available at: `https://github.com/KonstantinPakulev/NeSS-ST`.

1. Introduction

Feature point detection (keypoint detection) is usually the first step in camera localization and scene reconstruction pipelines based on sparse features commonly used in robotics [29], computer vision [33], augmented, mixed and virtual reality [7, 22] and other systems.

Whilst feature description is successfully approached in the literature as metric learning [27, 36] problem, applying deep learning to the feature detection task still poses a challenge with classical solutions showing competitive results on the state-of-the-art benchmarks [15]. The difficulty is caused by the innate vagueness of point of interest definition [40] that greatly complicates the formulation of feature detection as a learning problem.

The data required to learn keypoints with sufficient robustness to illumination and viewpoint changes presents another challenge. Structure-from-Motion (SfM) [33, 10] and Multi-View Stereo (MVS) [34, 10] reconstructions that are used by some methods [41, 30, 31, 13, 37] to obtain pixel-accurate correspondences are hard to build properly and lack full image coverage [19, 9].

We approach the problem of "defining a keypoint" by leveraging principled assumptions that make up reasonable keypoints [35]: we employ the Shi-Tomasi [35] detector to provide locations for keypoints as well as to perform sub-pixel localization. At the same time, we propose a quantitative metric to measure the stability of detected keypoints to viewpoint transformations, the *stability score (SS)*, that randomly perturbs detected points and their surrounding patches for later aggregation of their statistics. This metric can be calculated in an online fashion for a set of keypoints from a single image, so it is ideal for automatically generating a supervised signal for training a neural network that predicts the *neural stability score (NeSS)*. We empower a classical method, the **S**hi-**T**omasi detector, with a neural approach, NeSS, getting a method, NeSS-ST, that provides accurate locations of keypoints that are more likely to remain stable under viewpoint changes.

The design presented in our work does not rely on ground-truth poses or reconstructed correspondences alongside [11, 3, 18]. These methods are sometimes referred to as self-supervised [11, 18]. Compared to the prior art, NeSS-ST needs only a set of real images without any la-

bels for training. Out of self-supervised methods, only Key.Net [3] and REKD [18] work in a similar setting while SuperPoint [11] relies on pre-labeled datasets and requires pre-training a base detector on a synthetic dataset (see Table 1).

Our contributions are as follows. We present a novel metric, the stability score, to estimate the quality of keypoints. We propose a method, NeSS-ST, that employs the stability score to learn a neural network and requires only images for training. In terms of pose accuracy, NeSS-ST surpasses state-of-the-art on MegaDepth [19], is on par on IMC-PT [15], ScanNet [9] and HPatches [2] being the best among the self-supervised methods (see Table 6).

Methods	Pre-training base detector	Dataset pre-labeling	Ground-truth generation
SuperPoint [11]	Yes	Yes	Offline
Key.Net [3]	No	No	Offline
REKD [18]	No	No	Offline
NeSS-ST	No	No	Online

Table 1: Comparison of self-supervised methods for learning keypoints.

2. Related work

Handcrafted detectors. Since Moravec [28], feature detection methods focused on finding local extrema in signals derived from images that correspond to meaningful structures, *e.g*. corners [40]. Therefore, the earliest designs of detectors employ the grayvalue analysis via differential expressions [40, 14, 5, 35, 20].

Depending on image resolution or scale, both location and strength of keypoints response change [40], thus, spatial extrema do not necessarily constitute good features. The most productive way to tackle this problem is modeling scale via the scale-space [20] - the backbone of some famous methods like SIFT [21], SURF [4], Harris-Laplace [23], Harris-Affine [24], KAZE [1]. Features are detected either as extrema in the scale-space [21, 4, 1], or extrema are found at scales first, and then features are selected among extrema by using transformation invariant scores [23, 24].

Learned detectors. Existing designs of learned detectors are quite diverse. The majority of the methods employ correspondence labels for training which are obtained via SfM and MVS [41, 30, 13, 37] or optical flow [31]. Preparing dense pixel-accurate ground-truth correspondence labels poses the problem by itself [34, 10] - special preprocessing of data and validation of obtained results is required [31, 9, 19]. Although avoiding surface reconstruction can ease the problem formulation [33, 34], accurate poses can still be hard to obtain [43]. In this regard, self-supervised methods that do not require reconstructed correspondences, such as [11, 3, 18] or the method proposed here, present a viable alternative.

Self-supervised methods rely on sampling of homographies [11], affinities [3] and in-plane rotations [18] to generate ground truth correspondences. To detect keypoint candidates SuperPoint [11] employs a base detector that is trained on a synthetic dataset with corner-like structures. Key.Net [3] and REKD [18] follow an anchorless approach for detection: keypoints are discovered as a result of a loss function optimization. To address the problem of scale SuperPoint [11] prepares ground truth feature points using accumulated repeatability-like scores obtained with the help of the base detector and generated correspondences. Key.Net [3] and REKD [18], on the other hand, average values of the loss function over different window sizes.

In our method, similarly to [23, 24], we find extrema at a scale using the detector based on the structure tensor [14, 35], the Shi-Tomasi [35] detector (see Sec. 3.1 and Sec. 4.3.1), and perform feature selection using the scores predicted by the neural network that assess the stability of points to viewpoint changes. Like SuperPoint [11], we utilize a combination of a base detector and homography generation to calculate ground truth scores. We leverage a specially designed keypoint quality measure that, unlike SuperPoint, assesses feature points locally, using only a small neighbourhood (see Sec. 3.2 and Sec. 4.3.2). That, coupled with the handcrafted detector, allows us to prepare the required ground truth during the training using the same GPU (see Table 1, Sec. 3.3 and Sec. 3.4).

3. Method

NeSS-ST is the combination of the handcrafted Shi-Tomasi [14, 35] detector and the neural network (see Fig. 1). We briefly present the formulation of the Shi-Tomasi detector as well as describe the sub-pixel localization procedure in Sec. 3.1. In Sec. 3.2 we describe the measure that allows us to quantitatively assess keypoints, the *stability score (SS)*. Sec. 3.3 discusses how the stability score can be used to prepare the ground truth for training the neural network to predict the stability score for good feature points, the *neural stability score (NeSS)*. Finally, Sec. 3.4 gives an account of implementation details.

3.1. Shi-Tomasi detector

We use the Shi-Tomasi [14, 35] detector to get locations of feature points (see Sec. 4.3.1 for the motivation behind the choice). We calculate the second-moment matrix for each pixel using the Gaussian weighting function [14, 35] and assign the pixel with a score that is equal to the smallest eigenvalue of the second-moment matrix [35]. By applying

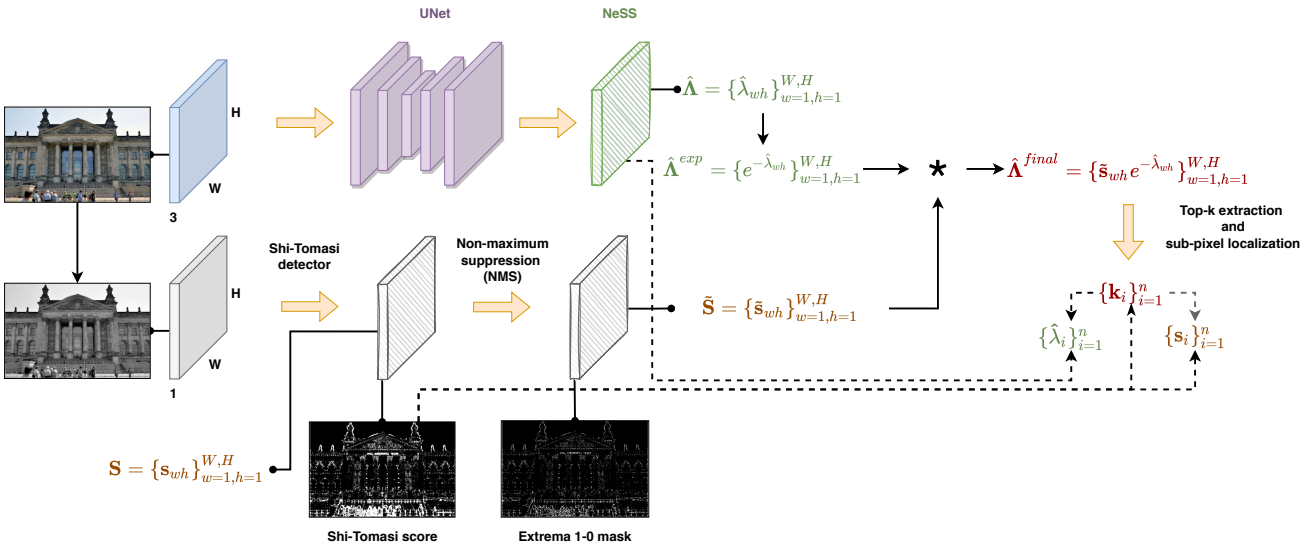

Figure 1: The method applies the Shi-Tomasi detector to an input image to get the Shi-Tomasi score $\mathbf{S}$. Next, a binary mask of extrema $\tilde{\mathbf{S}}$ is obtained via non-maximum suppression of $\mathbf{S}$. Simultaneously, the method uses the neural network to regress the neural stability score $\hat{\mathbf{\Lambda}}$. A set of best feature points $\{\mathbf{k}_i\}_{i=1}^n$ is selected from the combined score map $\hat{\mathbf{\Lambda}}^{final}$ that is a multiplicative combination of $\tilde{\mathbf{S}}$ with the negative exponential of $\hat{\mathbf{\Lambda}}$. Obtained keypoints are localized using $\mathbf{S}$ and are provided with corresponding $\{\mathbf{s}_i\}_{i=1}^n$ and $\{\hat{\lambda}_i\}_{i=1}^n$.

the non-maximum suppression over the obtained score map, we get the locations of keypoints.

We further refine the locations of feature points following [21]. By assuming that the Shi-Tomasi score $\mathbf{S}$ can be approximated by a function $\mathcal{S}(\mathbf{x})$, we perform the second order Taylor expansion around a point $\mathbf{x}$ and find the correction $\mathbf{dx}$ that maximizes $\mathcal{S}(\mathbf{x})$ by solving:

$$\mathbf{dx} = -\frac{\partial^2 \mathcal{S}}{\partial \mathbf{x}^2}^{-1} \frac{\partial \mathcal{S}}{\partial \mathbf{x}}. \tag{1}$$

3.2. Stability score

In this work, we design a quantitative description to assess the stability of feature points to viewpoint transformations. Given a keypoint $\mathbf{k}$ we apply to it a set of generated homographies $\{\mathcal{H}_j\}_{j=1}^m$ to produce a set of m warped keypoints $\{\mathbf{k}'_j\}_{j=1}^m$:

$$\mathbf{k}'_j = \mathcal{H}_j \mathbf{k}. \tag{2}$$

We generate homographies by sampling random perspective distortions while restricting the deformation along the x and y axes, see the details in the supplementary material. We do not add rotation or translation transformations since the Shi-Tomasi detector with Gaussian weighting that we use is invariant to them [14, 35, 12].

Next, we create a grid $\mathbf{G}_j$ of the predetermined size p around each of the warped points $\mathbf{k}'_j$, warp the grid back to the image coordinates and sample values in these locations to get a deformed patch $\mathbf{P}_j$ around $\mathbf{k}'_j$:

$$\mathbf{P}_j = sample(\mathcal{H}_j^{-1} \mathbf{G}_j). \tag{3}$$

The Shi-Tomasi detector is not invariant to scale [12] or perspective transformations, hence, in general, we cannot accumulate its scores from different warped patches [20]. However, we can accumulate locations of maximum scores and analyze their distribution.

We run the Shi-Tomasi detector f_{Shi} on each patch $\mathbf{P}_j$ getting a score patch $\mathbf{S}_j^{patch} = \{\mathbf{s}_l^{patch}\}_{l=1}^{p^2}$ and extract from it the location $\hat{\mathbf{l}}_j$ that corresponds to the maximum score:

$$\hat{\mathbf{l}}_j = \underset{\mathbf{s}_l^{patch}}{argmax}\, f_{Shi}(\mathbf{P}_j). \tag{4}$$

Next, we warp each $\hat{\mathbf{l}}_j$ back to the original reference frame and calculate the sample covariance $\mathbf{\Sigma}$ with respect to the original position $\mathbf{k}$:

$$\mathbf{\Sigma} = \sum_{j=1}^{m} \frac{(\mathcal{H}_j^{-1}\hat{\mathbf{l}}_j - \mathbf{k})(\mathcal{H}_j^{-1}\hat{\mathbf{l}}_j - \mathbf{k})^\top}{m}. \tag{5}$$

Finally, we characterize a feature point by the largest of eigenvalues λ_1 and λ_2 of $\mathbf{\Sigma}$:

$$\lambda = \max(\lambda_1, \lambda_2) = \|\Sigma\|_2. \tag{6}$$

The stability score λ captures the maximum deviation of a point from its location under perspective transformations of the neighbourhood of the point. The measure prioritizes feature points that deviate the least from the initial location $\mathbf{k}$ in any direction and, hence, are more likely to be accurately detected and localized. For further discussion of the design of the stability score see Sec. 4.3.2.

Since λ assesses keypoints based on their local perturbations, the patch size p is set to be small. It allows to quickly calculate λ with a sufficient number of samples m for a reasonable number of keypoints n. This particular trait makes a foundation for the training process of the NeSS regression network which is discussed next.

3.3. Neural stability score

Not every point with a good stability score λ makes a good training target: given that λ is calculated over a small window, image artifacts in low-textured regions can be perfectly stable. These unreliable responses can be filtered by the Shi-Tomasi detector assuming a threshold t_{Shi}. Thus, the neural network is trained on feature points that both have good stability scores and are more likely to correspond to meaningful patterns in an image. This gives rise to a new, better, kind of score, the neural stability score, see Sec. 4.3.2 for more details. In this regard, the stability score can be viewed as the necessary (but not sufficient) condition for a good keypoint. See the supplementary material for results on the influence of t_{Shi} on the quality of the model.

In line with several works [30, 3, 37], we get keypoints for training the neural network by running keypoint extraction using the trained-so-far weights of the model. More specifically, for each image we extract n feature points $\{\mathbf{k}_i\}_{i=1}^n$, their Shi-Tomasi scores $\{\mathbf{s}_i\}_{i=1}^n$ and neural stability scores $\{\hat{\lambda}_i\}_{i=1}^n$ regressed by the neural network (see Fig. 1). Such an approach not only narrows the gap between train and test regimes of the neural network, but it also allows to significantly reduce the time for preparing the ground truth. The combination of during-the-training feature extraction and fast calculation of λ enables online ground truth generation.

For each point $\mathbf{k}_i$ we calculate the ground-truth stability score λ_i (Eq. 6). Next, we define $\mathbb{1}(\mathbf{s}_i, t_{Shi})$ as an indicator function that gives 1 if $\mathbf{s}_i > t_{Shi}$ and 0 otherwise. Finally, we learn $\{\hat{\lambda}_i\}_{i=1}^n$ by formulating the training objective as a regression problem:

$$L = 0.5 \frac{\sum_{i=1}^{n} (\hat{\lambda}_i - \lambda_i)^2 \mathbb{1}(\mathbf{s}_i, t_{Shi})}{\sum_{i=1}^{n} \mathbb{1}(\mathbf{s}_i, t_{Shi})}. \tag{7}$$

3.4. Implementation details

We train our detector from scratch on the same subset of the MegaDepth [19] dataset as used in DISK [37]. We do not use poses, depth maps or any other information from the dataset other than images. We use full-resolution images cropped to a square of length 560 pixels for training. We perform validation and model selection on the validation subset of the IMC-PT [15] dataset.

We set $p = 5$ to correspond to the size of the non-maximum suppression kernel. The value of $n = 1024$ was found to be enough to cover the most of salient points above t_{Shi} for our choice of the threshold and resolution of training images. We pick $m = 100$ as it provides consistent estimates of λ. Further increasing m gives diminishing returns.

We use a U-Net [32] architecture with 4 down-sampling layers and 3×3 convolutions. To train the model we employ Adam [16] optimizer with learning rate 10^{-4}. The model that is used in experiments in Sec. 4 was trained on a single NVIDIA 2080 Ti GPU for 22 hours.

4. Experiments

The means of detector evaluation deserve special attention. Originally, detectors were mostly evaluated using classical metrics like repeatability and matching score [26]. Since feature points require supplying them with a description in order to obtain the correspondences, a descriptor has to be accounted for to ensure a proper assessment of the quality of detected keypoints. A common solution is to fix a descriptor for all detectors in the evaluation [26, 38, 44, 3]. More recent publications reported evidence that gains in classical metrics do not necessarily translate to the gains in the downstream tasks performance [15]. For this reason, in our work, we mostly perform the evaluation on a range of downstream tasks. To get a more comprehensive assessment, we test our method on a variety of datasets and tasks.

In our evaluation, we consider the following methods: Shi-Tomasi [14, 35], SIFT [21], SuperPoint [11], R2D2 [31], Key.Net [3], DISK [37] and REKD [18]. Some of these methods use detectors that have assigned names, *e.g*. SIFT utilizes the difference of Gaussians (DoG) detector, while some don't, *e.g*. the detector of R2D2 doesn't have a name. Since we report the evaluation results for "detector+descriptor" pairs, for consistency, we refer to both detectors and descriptors by the name of a complete solution that they belong to. For example, "SIFT+DISK" implies that the detector of SIFT, DoG, is used with the descriptor of DISK.

Image resolution plays an important role in the accuracy of correspondences, thus we provide images in the original resolution to all methods in our evaluation. It also allows to assess the ability of a detector to generalize beyond the training resolution. The number of keypoints that is extracted from each image plays no lesser role: we found the regime of 2048 keypoints per image from [15, 37] to be a good trade-off between performance and consumption of computational resources. We use mutual nearest neigh-

bour matching and employ the Lowe ratio test [21, 15]. As choosing proper hyper-parameters for a method is of utmost importance for downstream tasks [15], we employ a hyper-parameter tuning procedure similar to one in [15]. We extract feature points for all methods using only the original scale with the exception of Key.Net [3] and REKD [18] that have multi-scaling as an essential built-in part of the method. To ensure a fair comparison, we do not employ the orientation estimation for REKD [18] and SIFT [21] since other methods in the evaluation don't have it. As long as the sub-pixel localization is a part of our solution, the version of the Shi-Tomasi [14, 35] detector employed in our evaluation includes it since we focus on assessing the influence of NeSS.

4.1. Evaluation on HPatches

The HPatches [2] dataset features sequences with planar surfaces related by homographies under a variety of illumination and viewpoint changes. We use the same test subset as in [13] totaling 540 image pairs from 108 scenes among which 260 image pairs are with illumination changes and 280 - with viewpoint.

We report Mean Matching Accuracy (MMA) [25, 13] under different pixel thresholds following [11, 13, 39, 37, 18]. We evaluate on a downstream task of homography estimation using a protocol similar to [11, 39] and report the homography estimation accuracy for different pixel thresholds. Additionally, we integrate the accuracy-threshold curve up to a 5-pixel threshold to get a single-valued quality measure, mean Average Accuracy [42, 15].

The DISK [37] descriptor shows the state-of-the-art performance on HPatches, thus we pick it for this dataset. We employ OpenCV [6] routines for homography estimation. For tuning hyper-parameters we use sequences of HPatches [2] left out from the test set [13] as well as hyper-parameters obtained from validation sequences of IMC-PT [15] on the task of relative pose estimation. This combination provides the best results for all methods, see the details in the supplementary material.

Results. Our method shows inferior performance to the Shi-Tomasi [14, 35] detector when evaluated on the MMA metric as illustrated in Fig. 3. However, the evaluation on the task of homography estimation completely changes the ranking, and our method shows strong performance on scenes with viewpoint changes (see Table 2 and Fig. 2). These results correlate with recent findings that classical methods can show state-of-the-art performance if tuned properly, and that classical metrics might not fully capture the complicated dependency between features and downstream tasks [15]. Since we do not address the problem of illumination invariance in our work, NeSS-ST shows average results on illumination sequences. Refer to the supplementary material for more results.

4.2. Evaluation on downstream tasks

We evaluate on a downstream task of relative pose estimation following the protocol of [15]. We use our own evaluation pipeline to provide consistency in evaluations across different datasets. We calculate pose estimation accuracy for different angular thresholds and report mean Average Accuracy (mAA) [42, 15] for both rotation and translation by integrating the area under the accuracy-threshold curve up to a 10-degree threshold. Errors for rotation and translation are calculated in degrees [42, 39, 15].

4.2.1 Evaluation on IMC-PT

The IMC-PT [15] dataset is a collection of photo-tourism images supplied with depth maps and poses, which are reconstructed via SfM and MVS, that features landmarks (mostly buildings). We use a full test set release that consists of 800 unique images from 8 different locations. By considering image pairs with co-visibility larger than 0.1 [15], we get 37k test image pairs.

Like on HPatches, we choose the DISK [37] descriptor as it shows the state-of-the-art performance on this dataset. We utilize a robust fundamental matrix estimator with DEGENSAC [8]. We calculate the essential matrix from the estimated fundamental matrix using ground-truth intrinsics and then recover the poses using OpenCV [6]. The tuning of hyper-parameters is performed on the validation subset of IMC-PT [15], see the details in the supplementary material.

Results. Comparison in Table 3 shows that we considerably outperform other self-supervised approaches like SuperPoint [11], Key.Net [3] and REKD [18]. We explain the gap between DISK [37] and our method by the difference in the strategies that the methods employ to detect points. DISK [37] tends to densely detect points on semantically meaningful objects (mostly, buildings), whereas our method doesn't employ any knowledge of scene semantics and, instead, relies on the local properties of points, which results in sparsified detections. Given that IMC-PT [15] contains a lot of extreme viewpoint and scale changes, the former strategy looks more advantageous in such an environment. Refer to the supplementary material for additional results.

4.2.2 Evaluation on MegaDepth

The MegaDepth [19] dataset is another photo-tourism collection of images that also provides depth maps and poses. As the IMC-PT dataset has a limited diversity of scenes and their semantics and has only 800 unique images, we create

Methods	Overall mAA (5px)	Illumination mAA (5px)	Viewpoint mAA (5px)
Shi-Tomasi [14, 35] + DISK [37]	**0.716**	**0.892**	**0.552**
SIFT [21] + DISK [37]	0.688	0.877	0.512
SuperPoint [11] + DISK [37]	**0.706**	0.883	0.541
R2D2 [31] + DISK [37]	0.690	**0.888**	0.506
Key.Net [3] + DISK [37]	0.678	0.844	0.524
DISK [37]	0.699	0.867	**0.542**
REKD [18] + DISK [37]	0.689	**0.895**	0.498
NeSS-ST + DISK [37]	**0.714**	0.883	**0.556**

Table 2: Evaluation on HPatches [2] with 2048 keypoints and full resolution images. We report homography estimation mAA [11, 39, 42, 15] up to a 5-pixel threshold. Best results are marked in **red**, 2nd best - in **green**, 3rd best - in **blue**.

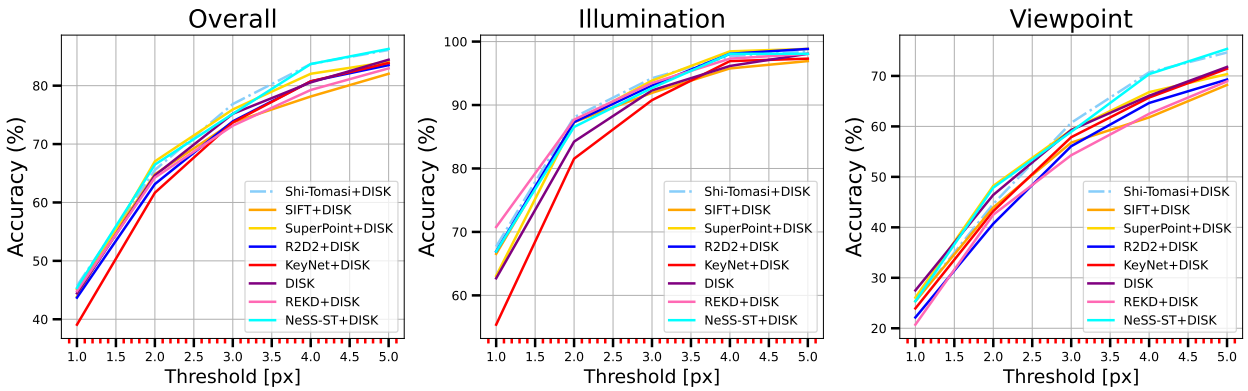

Figure 2: Evaluation on HPatches [2] with 2048 keypoints and full resolution images. We report homography estimation accuracy [11, 39] in %.

Methods	Rotation mAA (10°)	Translation mAA (10°)
Shi-Tomasi [14, 35] + DISK [37]	0.744	**0.422**
SIFT [21] + DISK [37]	0.7	0.387
SuperPoint [11] + DISK [37]	0.708	0.37
R2D2 [31] + DISK [37]	**0.751**	0.403
Key.Net [3] + DISK [37]	0.666	0.327
DISK [37]	**0.813**	**0.489**
REKD [18] + DISK [37]	0.601	0.271
NeSS-ST + DISK [37]	**0.767**	**0.438**

Table 3: Evaluation on IMC-PT [15] with 2048 keypoints and full resolution images. We report mAA [42, 15] up to a 10 degrees threshold for rotation and translation. Best results are marked in **red**, 2nd best - in **green**, 3rd best - in **blue**.

a test set from the MegaDepth dataset to perform the assessment with a larger diversity of data. In particular, our test set consists of 7.5k image pairs with 6k unique images sampled from 25 scenes belonging to 5 semantically different categories. This test set doesn't have any intersections neither with validation nor with test sequences of IMC-PT.

Since both IMC-PT and MegaDepth belong to the category of outdoors/photo-tourism datasets, we use the same descriptor, pose estimation routines and hyper-parameters for the evaluation.

Results. Contents of Table 4 show that the evaluation on a more diverse set of data narrows the gap between methods with our method being marginally better than DISK [37]. SuperPoint [11] is the second-best method in the category of methods that don't employ reconstructed correspondence labels. Although our gains in rotation mAA compared to it are marginal, we obtain noticeably better translation estimates. Refer to the supplementary material for more results.

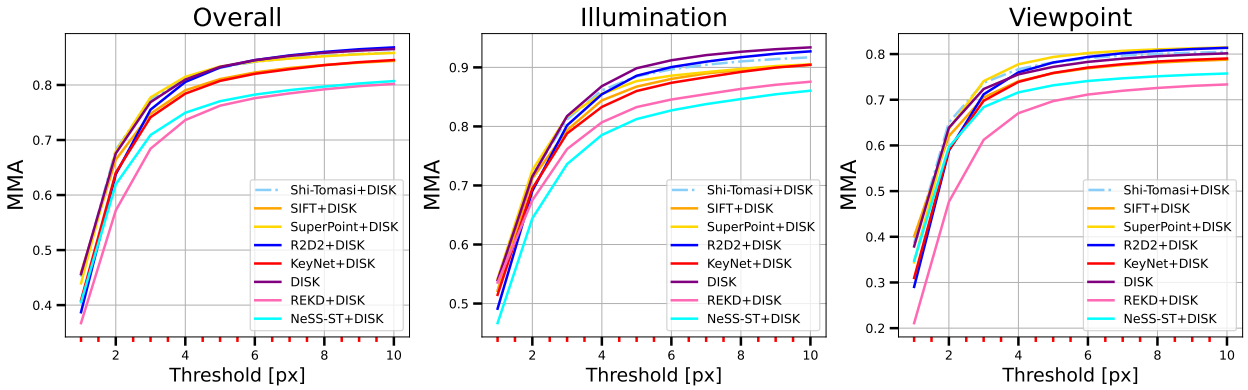

Figure 3: Evaluation on HPatches [2] with 2048 keypoints and full resolution images. We report MMA [25, 13].

Methods	Rotation mAA (10°)	Translation mAA (10°)
Shi-Tomasi [14, 35] + DISK [37]	0.858	0.31
SIFT [21] + DISK [37]	0.83	0.299
SuperPoint [11] + DISK [37]	**0.873**	**0.318**
R2D2 [31] + DISK [37]	0.871	0.312
Key.Net [3] + DISK [37]	0.84	0.278
DISK [37]	**0.877**	**0.326**
REKD [18] + DISK [37]	0.848	0.274
NeSS-ST + DISK [37]	**0.878**	**0.335**

Table 4: Evaluation on MegaDepth [19] with 2048 keypoints and full resolution images. We report mAA [42, 15] up to a 10 degrees threshold for rotation and translation. Best results are marked in **red**, 2nd best - in **green**, 3rd best - in **blue**.

Methods	Rotation mAA (10°)	Translation mAA (10°)
Shi-Tomasi [14, 35] + HardNet [27]	0.568	0.196
SIFT [21] + HardNet [27]	0.578	0.197
SuperPoint [11] + HardNet [27]	**0.611**	**0.217**
R2D2 [31] + HardNet [27]	**0.642**	**0.235**
Key.Net [3] + HardNet [27]	0.606	0.213
DISK [37] + HardNet [27]	0.528	0.162
REKD [18] + HardNet [27]	0.464	0.142
NeSS-ST + HardNet [27]	**0.621**	**0.222**

Table 5: Evaluation on ScanNet [9] with 2048 keypoints and full resolution images. We report mAA [42, 15] up to a 10 degrees threshold for rotation and translation. Best results are marked in **red**, 2nd best - in **green**, 3rd best - in **blue**.

4.2.3 Evaluation on ScanNet

To assess the generalization ability of our method we perform the evaluation on the ScanNet [9] dataset that contains indoor video sequences provided with camera poses and depth maps. Following [30, 39], we create validation and test sets by sampling pairs from video sequences with different gaps between frames. Our test set consists of 21k image pairs with 39k unique images sampled from 100 test sequences of the dataset.

We perform the evaluation using the HardNet [27] descriptor since on this dataset it shows performance that is superior to DISK [37] for every method. Because ScanNet is captured on a single RGB camera, we employ a robust essential matrix estimator from OpenGV [17] with the rest of the pipeline remaining the same as in previous evaluations. We use the validation subset of ScanNet for tuning hyperparameters, see the details in the supplementary material.

Results. Contrary to photo-tourism datasets that depict objects with a lot of texture, ScanNet [9] indoor environments contain a lot of surfaces with little to no texture. We believe that this is the main reason why DISK [37] shows significantly inferior performance on this set of data (see Table 5). Our method, on the other hand, can consistently cope with the challenge and shows the second-best result achieving significant improvements compared to SuperPoint [11], Key.Net [3] and REKD [18]. R2D2 [31] achieves the best results on this dataset; we found that the method is able to provide consistent matches on images with little texture and poor illumination conditions by performing detection along the contours of objects. Refer to the supplementary material for additional results.

To sum up, the experiments show that NeSS-ST is the only method in the top three across all the datasets (see Table 6) as well as has the best performance among the self-supervised methods listed in Table 1. It has better general-

	HPatches	IMC-PT	MegaDepth	ScanNet
First	Shi-Tomasi	DISK	**NeSS-ST**	R2D2
Second	**NeSS-ST**	**NeSS-ST**	DISK	**NeSS-ST**
Third	SuperPoint	R2D2/Shi-Tomasi	SuperPoint	SuperPoint

Table 6: The top three ranking of detectors on downstream tasks.

ization ability compared to most of state-of-the-art, compact model size and a fast running time (see Table 7).

4.3. Ablation study

4.3.1 Base detector ablation

Our method operates on top of the handcrafted Shi-Tomasi [14, 35] detector, however, NeSS can be used in combination with other detectors. The choice of a handcrafted detector over a learned detector is motivated by the size of patch p that is required to maintain online ground truth generation (see Sec. 3.2). Specifically, handcrafted detectors require a small neighbourhood to calculate the score compared to learned detectors that have a large receptive field. Another reason for giving the preference to handcrafted detectors is their good performance on downstream tasks, DoG, in particular, reported in [15], which is also confirmed in our evaluation.

We pick the Shi-Tomasi [14, 35] detector over the Harris [14] detector since the keypoint selection criterion of the former better fits NeSS. More precisely, the Shi-Tomasi detector doesn't impose any restrictions on the shape of the second-moment matrix; an edge-like pattern and a weak blob-like pattern can look the same to the Shi-Tomasi detector. The Harris detector, on the other hand, draws a clear distinction between points and edges: eigenvalues of the second-moment matrix need to be relatively well-proportioned for a high Harris score.

We explore the performance of various handcrafted detectors by using them as base detectors for NeSS. Based on the summary given in Sec. 2, handcrafted detectors can be categorized into those that detect only spatial extrema, and those that also model the invariance to viewpoint changes. Since NeSS, by design, assesses the invariance of spatial extrema to viewpoint transformations (see Sec. 3.2), we consider only the former kind of methods to serve as base detectors. In particular, we examine the determinant of the Hessian (DoH) [20] and the Laplacian of the Gaussian (LoG) [5, 20] detectors. The latter detector presents a particular interest since DoG is the approximation of scale-normalized LoG [20, 21]. All base detectors in the evaluation use sub-pixel localization.

Results. Table 8 shows that NeSS noticeably improves the performance of all base detectors; however, at the same time, the results indicate that the gains provided by NeSS depend on the performance of an employed base detector. Shi-Tomasi [14, 35] has the best performance among all base detectors and demonstrates the best results when combined with NeSS. Interestingly, Shi-Tomasi shows noticeable improvements over Harris on translation estimation that correlates well with the theoretical justification behind the design of the Shi-Tomasi detector [35]. More results can be found in the supplementary material.

4.3.2 Stability score design ablation

We highlight the importance of accounting for the uncertainty of a keypoint location in the stability score (see Eq. 5 and Eq. 6) by conducting experiments with another score. In particular, using the notation from Eq. 5, we formulate the *repeatability score (RS)* that acts like ϵ-pixel repeatability measure:

$$\mathbf{r} = \sum_{j=1}^{m} \frac{\mathbb{1}^{rep}(\mathcal{H}_j^{-1}\hat{\mathbf{l}}_j - \mathbf{k}, \epsilon)}{m}. \tag{8}$$

We define $\mathbb{1}^{rep}(\mathcal{H}_j^{-1}\hat{\mathbf{l}}_j - \mathbf{k}, \epsilon)$ as an indicator function that gives 1 if $|\mathcal{H}_j^{-1}\hat{\mathbf{l}}_j - \mathbf{k}|_\infty < \epsilon$ and 0 otherwise. Setting $\epsilon = 1$, we train the neural network to predict the *neural repeatability score (NeRS)* in the similar to Eq. 7 manner.

To emphasize the role of the neural network in our design we build detectors that operate without it by directly calculating SS and RS for all potential feature points on an image. Apart from the computational concerns, such design choice requires picking a threshold to filter the noise (see Sec. 3.3). This poses a problem since the optimal threshold depends on the intensity of depicted patterns that varies from image to image. Utilizing filtering during the training is exempt from this limitation: on the contrary, it allows to select only reliable points for learning a new rule from the data and, hence, to a certain extent, overcome the limitations of homography sampling. Refer to the supplementary material for the details.

Results. Results presented in Table 9 indicate that the assessment of the local uncertainty of a keypoint location (SS, NeSS) yields a better criterion compared to repeatability (RS, NeRS) when the downstream task of pose estimation is considered. Notably, although the RS criterion provides keypoints for training that have better repeatability and MMA compared to SS (see Table 10), the model trained using the latter criterion shows much better results on the downstream task (see Table 9). Table 9 and Table 7 show the benefits of utilizing the neural network (NeSS, NeRS)

	SuperPoint [11]	R2D2 [31]	Key.Net [3]	DISK [37]	REKD [18]	NeSS-ST	SS-ST
Size (MB)	4.96	1.85	0.02	4.17	99.12	3.54	-
Inference time (ms)	4.0	24.4	5.4	19.5	59.5	8.2	403.9
Post-processing time (ms)	1.7	2.2	2.2	0.7	2.0	7.4	

Table 7: Comparison of models sizes and running-time for 640×480 resolution images on NVIDIA 2080 Ti GPU. Please note that post-processing procedures (keypoint selection, sub-pixel localization) are not optimized.

Method	Rotation mAA (10°)	Translation mAA (10°)
Shi-Tomasi [14, 35]	**0.744**	**0.422**
Harris [14]	0.743	0.405
DoH [20]	0.694	0.363
LoG [5, 20]	0.708	0.376
NeSS-ST	**0.766**	**0.438**
NeSS-DoH	0.742	0.39
NeSS-LoG	0.763	0.421

Table 8: Base detector ablation on IMC-PT [15] with DISK [37] descriptor, 2048 keypoints and full resolution images. We report mAA [42, 15] up to a 10 degrees threshold for rotation and translation. Best results are marked in **red**.

Method	Rotation mAA (10°)	Translation mAA (10°)
SS-ST	0.757	0.417
RS-ST	0.622	0.273
NeSS-ST	**0.766**	**0.438**
NeRS-ST	0.75	0.41

Table 9: Stability score design ablation on IMC-PT [15] with DISK [37] descriptor, 2048 keypoints and full resolution images. We report mAA [42, 15] up to a 10 degrees threshold for rotation and translation. Best results are marked in **red**.

over the algorithmic approach (SS, RS) from both performance and computational standpoints. More results can be found in the supplementary material.

5. Conclusion

In this work, we proposed the NeSS-ST detector that combines the handcrafted Shi-Tomasi detector and the neural stability score. The method doesn't require any reconstructed correspondence labels and can be trained from arbitrary sets of images without the need for dataset prelabeling. It achieves state-of-the-art performance on a variety of datasets and downstream tasks, has good generalization and consistently outperforms other self-supervised methods. In the future, we plan to address the main limitation of our method which is the lack of illumination invariance as well as use inferences from the evaluation of methods like DISK and R2D2 to improve our keypoint detection strategy. Additionally, NeSS may be used as a weight in non-linear pose refinement or metric learning of feature descriptors.

Method	Repeatability (3 px)	MMA (3 px)
SS-ST	0.339	0.655
RS-ST	**0.423**	**0.668**

Table 10: Stability score design ablation on HPatches [2] with DISK [37] descriptor, 2048 keypoints and full resolution images. We report MMA [25, 13] and repeatability [26] under a 3-pixel threshold. Best results are marked in **red**.

References

[1] Pablo Fernández Alcantarilla, Adrien Bartoli, and Andrew J Davison. Kaze features. In *Eur. Conf. on Computer Vision (ECCV)*, pages 214–227. Springer, 2012.

[2] Vassileios Balntas, Karel Lenc, Andrea Vedaldi, and Krystian Mikolajczyk. Hpatches: A benchmark and evaluation of handcrafted and learned local descriptors. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 5173–5182, 2017.

[3] Axel Barroso-Laguna, Edgar Riba, Daniel Ponsa, and Krystian Mikolajczyk. Key. net: Keypoint detection by handcrafted and learned cnn filters. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 5836–5844, 2019.

[4] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *Eur. Conf. on Computer Vision (ECCV)*, pages 404–417. Springer, 2006.

[5] Dorothea Blostein and Narendra Ahuja. A multiscale region detector. *Computer Vision, Graphics, and Image Processing*, 45(1):22–41, 1989.

[6] Gary Bradski. The opencv library. *Dr. Dobb's Journal: Software Tools for the Professional Programmer*, 25(11):120–123, 2000.

[7] Robert Castle, Georg Klein, and David W Murray. Video-rate localization in multiple maps for wearable augmented reality. In *12th IEEE International Symposium on Wearable Computers*, pages 15–22, 2008.

[8] Ondrej Chum, Tomas Werner, and Jiri Matas. Two-view geometry estimation unaffected by a dominant plane. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 772–779. IEEE, 2005.

[9] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 5828–5839, 2017.

[10] Angela Dai, Matthias Nießner, Michael Zollhöfer, Shahram Izadi, and Christian Theobalt. Bundlefusion: Real-time globally consistent 3d reconstruction using on-the-fly surface reintegration. *ACM Trans. Graph.*, 36(4):1, 2017.

[11] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. SuperPoint: Self-supervised interest point detection and description. In *IEEE Conf. on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 224–236, 2018.

[12] Y. Dufournaud, C. Schmid, and R. Horaud. Matching images with different resolutions. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 612–618 vol.1, 2000.

[13] Mihai Dusmanu, Ignacio Rocco, Tomas Pajdla, Marc Pollefeys, Josef Sivic, Akihiko Torii, and Torsten Sattler. D2-net: A trainable cnn for joint detection and description of local features. *arXiv preprint arXiv:1905.03561*, 2019.

[14] C. Harris and M. Stephens. A combined corner and edge detector. In *Proceedings of the 4th Alvey Vision Conference*, pages 147–151, 1988.

[15] Yuhe Jin, Dmytro Mishkin, Anastasiia Mishchuk, Jiri Matas, Pascal Fua, Kwang Moo Yi, and Eduard Trulls. Image matching across wide baselines: From paper to practice. *Intl. J. of Computer Vision*, 129(2):517–547, 2021.

[16] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR (Poster)*, 2015.

[17] Laurent Kneip and Paul Furgale. Opengv: A unified and generalized approach to real-time calibrated geometric vision. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pages 1–8. IEEE, 2014.

[18] Jongmin Lee, Byungjin Kim, and Minsu Cho. Self-supervised equivariant learning for oriented keypoint detection. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 4847–4857, 2022.

[19] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 2041–2050, 2018.

[20] Tony Lindeberg. Feature detection with automatic scale selection. *Intl. J. of Computer Vision*, 30(2):79–116, 1998.

[21] David G Lowe. Distinctive image features from scale-invariant keypoints. *Intl. J. of Computer Vision*, 60(2):91–110, 2004.

[22] Sven Middelberg, Torsten Sattler, Ole Untzelmann, and Leif Kobbelt. Scalable 6-dof localization on mobile devices. In *Eur. Conf. on Computer Vision (ECCV)*, pages 268–283, 2014.

[23] Krystian Mikolajczyk and Cordelia Schmid. Indexing based on scale invariant interest points. In *Intl. Conf. on Computer Vision (ICCV)*, volume 1, pages 525–531. IEEE, 2001.

[24] Krystian Mikolajczyk and Cordelia Schmid. An affine invariant interest point detector. In *Eur. Conf. on Computer Vision (ECCV)*, pages 128–142. Springer, 2002.

[25] Krystian Mikolajczyk and Cordelia Schmid. A performance evaluation of local descriptors. *IEEE Trans. Pattern Anal. Machine Intell.*, 27(10):1615–1630, 2005.

[26] Krystian Mikolajczyk, Tinne Tuytelaars, Cordelia Schmid, Andrew Zisserman, Jiri Matas, Frederik Schaffalitzky, Timor Kadir, and L Van Gool. A comparison of affine region detectors. *Intl. J. of Computer Vision*, 65:43–72, 2005.

[27] Anastasiia Mishchuk, Dmytro Mishkin, Filip Radenovic, and Jiri Matas. Working hard to know your neighbor's margins: Local descriptor learning loss. *Advances in Neural Information Processing Systems (NIPS)*, 30, 2017.

[28] Hans P. Moravec. Rover visual obstacle avoidance. In *Proceedings of the 7th International Joint Conference on Artificial Intelligence - Volume 2*, IJCAI'81, page 785–790, San Francisco, CA, USA, 1981. Morgan Kaufmann Publishers Inc.

[29] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. ORB-SLAM: a versatile and accurate monocular SLAM system. *IEEE Trans. Robotics*, 31(5):1147–1163, 2015.

[30] Yuki Ono, Eduard Trulls, Pascal Fua, and Kwang Moo Yi. LF-Net: Learning local features from images. *Advances in Neural Information Processing Systems (NIPS)*, 31, 2018.

[31] Jerome Revaud, Cesar De Souza, Martin Humenberger, and Philippe Weinzaepfel. R2d2: Reliable and repeatable detector and descriptor. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems (NIPS)*, volume 32. Curran Associates, Inc., 2019.

[32] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer Assisted Intervention (MICCAI)*, pages 234–241. Springer, 2015.

[33] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 4104–4113, 2016.

[34] Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *Eur. Conf. on Computer Vision (ECCV)*, pages 501–518. Springer International Publishing, 2016.

[35] Jianbo Shi and Tomasi. Good features to track. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 593–600, 1994.

[36] Yurun Tian, Xin Yu, Bin Fan, Fuchao Wu, Huub Heijnen, and Vassileios Balntas. Sosnet: Second order similarity regularization for local descriptor learning. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 11016–11025, 2019.

[37] Michał Tyszkiewicz, Pascal Fua, and Eduard Trulls. Disk: Learning local features with policy gradient. *Advances in Neural Information Processing Systems (NIPS)*, 33:14254–14265, 2020.

[38] Yannick Verdie, Kwang Yi, Pascal Fua, and Vincent Lepetit. Tilde: A temporally invariant learned detector. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 5279–5288, 2015.

[39] Qianqian Wang, Xiaowei Zhou, Bharath Hariharan, and Noah Snavely. Learning feature descriptors using camera pose supervision. In *Eur. Conf. on Computer Vision (ECCV)*, pages 757–774. Springer International Publishing, 2020.

[40] Andrew P. Witkin. Scale-space filtering. In *IJCAI*, 1983.

[41] Kwang Moo Yi, Eduard Trulls, Vincent Lepetit, and Pascal Fua. Lift: Learned invariant feature transform. In *Eur. Conf. on Computer Vision (ECCV)*, pages 467–483. Springer, 2016.

[42] Kwang Moo Yi, Eduard Trulls, Yuki Ono, Vincent Lepetit, Mathieu Salzmann, and Pascal Fua. Learning to find good correspondences. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 2666–2674, 2018.

[43] Shishun Zhang, Longyu Zheng, and Wenbing Tao. Survey and evaluation of rgb-d slam. *IEEE Access*, 9:21367–21387, 2021.

[44] Xu Zhang, Felix X Yu, Svebor Karaman, and Shih-Fu Chang. Learning discriminative and transformation covariant local feature detectors. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 6818–6826, 2017.