𝕏Resolution Correspondence Networks

Georgi Tinchev[1]
gtinchev@robots.ox.ac.uk
Shuda Li[2]
shuda.li@xyzreality.com
Kai Han[3]
khan@robots.ox.ac.uk
David Mitchell[2]
david.mitchell@xyzreality.com
Rigas Kouskouridas[2]
rigas.kousk@xyzreality.com

[1] Oxford Robotics Institute
University of Oxford
Oxford, UK
[2] XYZ Reality
London, UK
[3] Visual Geometry Group
University of Oxford
Oxford, UK

Abstract

In this paper, we aim at establishing accurate dense correspondences between a pair of images with overlapping field of view under challenging illumination variation, viewpoint changes, and style differences. Through an extensive ablation study of the state-of-the-art correspondence networks, we surprisingly discovered that the widely adopted 4D correlation tensor and its related learning and processing modules could be de-parameterised and removed from training with merely a minor impact over the final matching accuracy. Disabling these computational expensive modules dramatically speeds up the training procedure and allows one to use 4 times bigger batch size, which in turn compensates for the accuracy drop. Together with a multi-GPU inference stage, our method facilitates the systematic investigation of the relationship between matching accuracy and up-sampling resolution of the native testing images from 1280 to 4K. This leads to discovery of the existence of an optimal resolution 𝕏 that produces accurate matching performance surpassing the state-of-the-art methods particularly over the lower error band on public benchmarks for the proposed network.

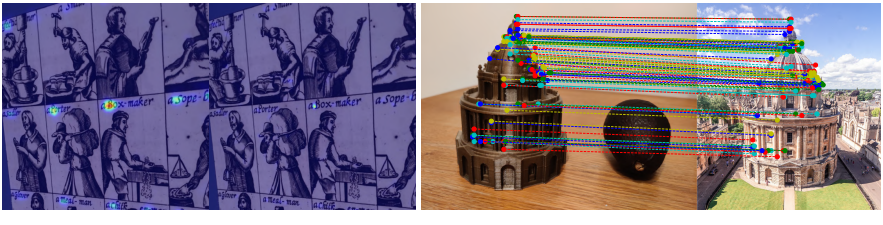


Figure 1: 𝕏RCNet highlights: **Left tuple**: The heatmaps represent the confidence of the location of the query key point ('a' on row 2 column 3) in the source image accurately pinpointed in the target image (blue - low confidence, red - high confidence) even with challenging repetitive patterns of the letter 'a' at low resolution (left) compared to high resolution (right). **Right**: 𝕏RCNet without using any geometric constraints can produce state-of-the-art matching accuracy and can reliably match under extreme illumination and style differences.

1 Introduction

Establishing dense image correspondences is a fundamental problem for many computer vision applications, from Structure-from-Motion (SfM) [1, 48, 49], visual Simultaneous Localisation and Mapping (SLAM) [37] to image retrieval [22], image style transfer [15, 30], and scene understanding [5, 10]. Traditionally, image correspondences are found through a sparse detection-description and matching pipeline. Particularly, a key point detector [16, 44] is first used to collect a set of sparse interest points from input images while a feature descriptor [3, 25, 33] extracts a unique description of a local image patch centred at the detected key point location. In the end, the point correspondences between the query image and the reference one are calculated by searching the candidate matching pairs for small descriptor distances or using the ratio test between the best and the second best matches [33].

In the last few years we have witnessed a dramatic improvement over all stages of the sparse correspondence pipeline mostly using machine learning [11, 12, 41, 41, 53]. In addition to the feature detectors and descriptors, the matching stage has also been extensively studied and new algorithms taking into account both inter-image and intra-image constraints make the matching stage more reliable than before [34, 45, 53, 60]. However, sparse correspondence methods are not straightforward to be adapted to produce per pixel matches which are often required for image warping, style transfer, or dense 3D reconstruction. A naive extension from sparse methods, for example, is to densely extract feature descriptors and use brute force matching. However, this is prohibitively expensive for high resolution images. Furthermore, to achieve the best performance, the detection-description and matching pipeline typically requires each stage to be trained separately, which introduces extra difficulties when being deployed to new sensory data. For example, the top performer on the visual localisation benchmark [61] combines the SuperPoint (SP) [11] and SuperGlue (SG) [45] and the SP detector-descriptor has to be trained separately with SG.

In contrast, the dense correspondence methods [32] and particularly Deep Correspondence Networks (DCN) [9, 38, 42, 43, 56, 59] that emerged in recent years, represent a highly competitive alternative for their capability of producing good quality per pixel correspondences. DCNs also unify the detection-description and matching pipeline into one single architecture using standard feature backbones such that it can be trained end-to-end. Moreover, DCNs are shown to be able to quickly adapt to images of high resolution or larger feature maps while being deployed into consumer products [17, 28, 42, 43, 56].

In this paper, we present a novel dense correspondence methodology that is capable of processing high resolution images and produce reliable and highly accurate matching results as shown in Fig. 1. More importantly, light-weight correspondence networks allows us to investigate intriguing questions for all DCNs: *Does up-sampling of the testing image always lead to higher accuracy*? If not, *does an optimal resolution* $\mathbb{X}$ *exist*? In this work, we introduce $\mathbb{X}$Resolution Correspondence Network ($\mathbb{X}$RCNet), a light-weight architecture designed to answer these questions while achieving state-of-the-art performance.

Our work is directly inspired by the recently introduced strategy of using extensive ablation studies to either achieve more accurate visual representations [6] or highly impactful training procedures or architecture refinements that improve model accuracy [20]. Approaching the dense correspondence problem with the same strategy, we start by carrying out extensive ablation studies with various training configurations over the state-of-the-art dense correspondence networks and made several key observations. First, the widely adopted 4D correlation tensor and its related filtering modules [17, 42, 43, 56] can be de-parameterised and even removed from the training stage at the cost of a small drop in accuracy. Sec-

ond, switching to a much shallower feature backbone also has limited impact to the overall matching results. Third, the combination of the first two discoveries results in the light-weight 𝕏RCNet. During the training stage, 𝕏RCNet enjoys a significantly smaller memory footprint and much higher speed than the state-of-the-art methods (see Tab. 1). This allows us to use 4 times larger batch size and increase the number of epochs within roughly the same amount of training time using the same hardware. The latter compensates for the slight deterioration of the accuracy levels. When combined with a multi-GPU inference method, it allows us to evaluate the matching accuracy of 𝕏RCNet using image size up to 4K, by which we discover the existence of an optimal up-sampling resolution for 𝕏RCNet to achieve the best accuracy. Interestingly, increasing the resolution is not always beneficial possibly because the relative size of the receptive field to the image might decrease, which then renders the network prediction less accurate. The contributions can be summarised as follow:

- We carried out an extensive study over state-of-the-art DCNs and made several key observations that lead to the introduction of a simple and light-weight multi-resolution neural network architecture named as the 𝕏Resolution Network (𝕏RCNet).
- 𝕏RCNet is capable of training with much larger batch size and faster per image learning speed. During inference, 𝕏RCNet can take in images with higher resolution than most of the previous work and allows us to search for the optimal resolution 𝕏 to up-sample the testing image for a correspondence task.
- 𝕏RCNet achieves state-of-the-art accuracy on two challenging datasets — HPatches [2] and InLoc [51], while performing competitively on Aachen Day-Night [46, 47].

2 Related work

The correspondence algorithm is a basic building block in computer vision which is widely explored. Existing methods range from sparse to dense correspondence estimation. Sparse correspondence algorithms typically adopt the three-stage pipeline of detection-description and matching. Each stage has received extensive research focus over the last two decades. For key point detection and description, handcrafted methods SIFT, SURF, BRIEF [3, 4, 33] and their variants [25, 50] were introduced for first detecting, then describing and finally matching a sparse set of key points. Taking into account the local region around each key point, a feature vector of floating points or binary numbers can be extracted to uniquely represent the key points for feature matching or scene description [13]. Most of the modern descriptors [12, 34, 41, 52, 53, 58] focus on data-driven learning approaches, while evaluating the matching performance of descriptors is performed either by measuring the distances between a pair of descriptors or through the ratio test [33]. Modern matching approaches take into account the constraints between feature descriptors to enhance the matching success rate [45, 60]. Particularly, SuperGlue [45] explores the inter/intra-image information.

Sparse correspondence algorithms achieve efficiency by attending to a small set of salient points in the images, however, for applications such as SfM [48, 49], style transfer [56] or view synthesis [40] where per-pixel correspondence maps are often required, simply scaling up the sparse approach becomes prohibitively expensive. In contrast, dense correspondence approaches focus on bridging this gap. One of the earliest dense methods [32] uses dense feature descriptors and regularising within the local region to achieve a consistent dense flow field. In recent years, deep semantic correspondence networks [18, 21, 24, 27, 35, 42] have demonstrated the potential of densely associating key points between a pair of input images. However, as these approaches focus on matching high level regions, they either require a large number of feature channels, typically larger than 1024 [24, 35, 36], or build on top of the 4D correlation tensor and expensive 4D filtering [21, 27, 42]. This fact

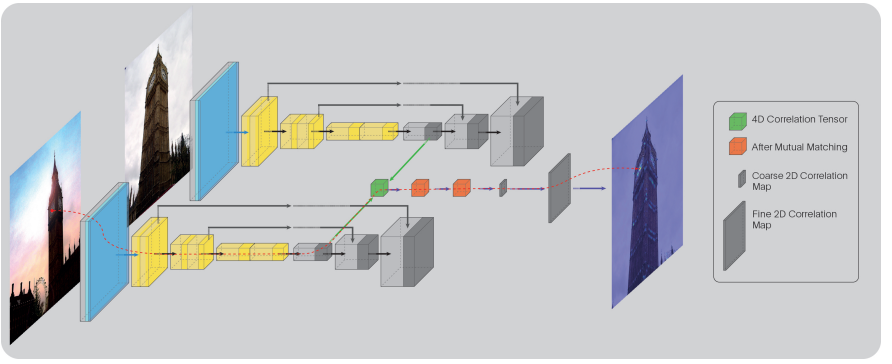


Figure 2: The architecture of 𝕏RCNet. The deep correspondence neural network follows a Siamese-like structure. Each branch is composed of a feature encoder (yellow) and FPN-like decoder (gray blocks). From the coarse layer of FPN, a 4D correlation tensor is calculated (green) and filtered by two mutual matching (MM) layers to get the filtered 4D tensor (orange). Given a query key point from the source image (bottom left), the corresponding features are selected from the FPN coarse layers and query into the 4D tensor.

makes it very difficult to scale up to higher image resolutions, which is critical for accurate data association [28, 42, 43]. SparseNC overcomes the scalability problem by projecting the memory consuming 4D correlation tensor into a sub-manifold and uses the Minkowski convolution [8] to approximate the 4D filtering, however, the approximation reduces the performance of the network. DualRC [28] keeps the original 4D correlation tensor in its original space, but relies on a coarse to fine re-weighting mechanism to guide the search in a fine resolution correlation map for the best match. In this work, we further reduce the network redundancy by limiting the operation of the 4D correlation tensor. Combined with a much shallower feature backbone, our proposed approach can process images with higher resolution than all previous dense networks on the same hardware setup.

Establishing dense correspondences is also relevant to stereo networks [55], deep visual odometry [57] and dense optical flow [53] since these algorithms also involve calculating a dense flow field associating two images. However, the stereo matching often assumes the input views are rectified so that the images are captured under the same lighting condition, while the viewpoints are relatively close to each other, which can be viewed as a simplified version of the correspondence problem. Similarly, both the tasks of optical flow estimation and visual odometry are considered to be much more constrained than the general correspondence problem, since both assume that the viewpoints of the input images are close both temporally and spatially in terms of the 6D manifold of the camera poses.

3 Methodology

In this work we present a new dense correspondence methodology working with input images of higher resolution than any other state-of-the-art dense method and attaining higher accuracy particularly for small error bands. In this section, we first describe the DCN framework illustrated in Fig. 2, then the redundant module is ablated to form the 𝕏RCNet.

Given a pair of images $\mathbf{I}$ and $\mathbf{I}'$, we want to estimate a per-pixel correspondence map that associates a 2D key point from the source image $(x,y) \in \mathbf{I}$ to a point in the target image $(x',y') \in \mathbf{I}'$. To reliably associate the point, we first adopt a standard multi-level feature backbone $\mathbf{F} = f(\mathbf{I};\theta_0)$, where θ_0 are learnable variables. Particularly, $\mathbf{F} = \{\mathbf{F}_f, \mathbf{F}_c\}$ where $\mathbf{F}_f \subset \mathbb{R}^{C \times \Omega_f}$ is one layer of the feature map within a 2D domain Ω_f and $\mathbf{F}_c \subset \mathbb{R}^{C \times \Omega_c}$ is

another layer of the feature map within Ω_c. Subscripts f and c represent the fine and coarse resolutions, while C stands for the number of feature channels. Previous works [28, 42, 43, 56] make use of a 4D correlation tensor $\mathbf{C} \subset \mathbb{R}^{\Omega \times \Omega'}$, where $\mathbf{C}_{(x,y,x',y')} = \mathbf{F}(x,y)^\top \mathbf{F}'(x',y')$. Note that all values in the feature maps are positive due to the ReLU activation layer in the feature backbone $f(\cdot)$ immediately before calculating the correlation. The features are typically normalised along the channels — $\|\mathbf{F}(x,y)\|_2 \triangleq 1$ and thus the dot product of two feature vectors is within the range $[0,1]$. The 4D correlation tensor represents all possible candidate matching pairs from the source to the target image.

3.1 Neighbourhood consensus

Initially introduced in NCNet [42], a set of 4D convolutions with learnable variables is trained to filter the noise from the raw correlation tensor. The local 4D volume contains all possible matching pairs within the neighbourhood of the source and target image from which a filtering process is employed in order to collect consensus from them. Neighbourhood Consensus (NC) filtering can be formulated as $\hat{\mathbf{C}} = N(\mathbf{C};\theta_1) + N(\mathbf{C}^\top;\theta_1)^\top$ where $N(\cdot)$ represents the NC filtering consisting of a sequence of 4D convolution layers. $\mathbf{C}^\top$ is the permutation operation such that $\mathbf{C}_{(x,y,x',y')} = \mathbf{C}^\top_{(x',y',x,y)}$ and θ_1 are the learnable parameters in the NC filtering. The first term corresponds to the matching direction from source to the target and the second term from target to the source. Since the matching direction is independent of the filter weights, θ_1 is shared by the two filtering stages. The result $\hat{\mathbf{C}}$ has the same dimensionality as $\mathbf{C}$ that contains the filtered correlation scores.

To improve accuracy soft Mutual Matching (MM) filtering layers can also be applied before and after the NC filtering to dynamically adjust the scale of the correlation tensor:

$$\mathbf{M}_{(x,y,x',y')} = \frac{\mathbf{C}_{(x,y,x',y')}}{\max_{\forall(x',y')\in\Omega'} \mathbf{C}_{(x,y,x',y')} + \varepsilon} \tag{1}$$

$$\mathbf{M}'_{(x,y,x',y')} = \frac{\mathbf{C}_{(x,y,x',y')}}{\max_{\forall(x,y)\in\Omega} \mathbf{C}_{(x,y,x',y')} + \varepsilon} \tag{2}$$

$$\hat{\mathbf{C}}_{(x,y,x',y')} = \mathbf{M}_{(x,y,x',y')} \mathbf{C}_{(x,y,x',y')} \mathbf{M}'_{(x,y,x',y')} \tag{3}$$

where ε is an infinitely small value to improve the numerical stability and prevent errors during the degenerating scenario when the maximum correlation in a domain is 0. In practice we use $\varepsilon = 1 \times 10^{-5}$. The MM layer contains no learnable parameters. As shown in equations (1) and (2) the MM layer first converts the correlation scores into probabilities by normalising using the maximum correlations with respect to the target domain Ω' and source domain Ω, respectively. The multiplication of $\mathbf{M}_{(x,y,x',y')}$ and $\mathbf{M}'_{(x,y,x',y')}$ can be viewed as the joint probability of matching from source to target and from target to source providing the matching along both directions are independent. Ablating MM layer reduces matching accuracy possibly because the MM layer adjusts the scores in the correlation tensor (Sec. 3.3).

In the end, given a query point $(x,y) \in \mathbf{I}$, the best matches can be found at $(\hat{x}',\hat{y}') = \arg\max_{\forall(x',y')\in\Omega'} \hat{\mathbf{C}}_{(x,y,x',y')}$. The dense correspondence map can be established by calculating $(\hat{x}',\hat{y}')$ for every pixel in the source image. In addition, the maximum correlation scores $\mathbf{S} \subset \mathbb{R}^{\Omega}$ represent a good indication of the matching reliability, where $\mathbf{S}(x,y) = \max_{\forall(x',y')\in\Omega'} \hat{\mathbf{C}}_{(x,y,x',y')}$. A sub-set of top k reliable matches $\mathbb{S} = \{\mathbf{S}\}_k$ can be collected accordingly, or alternatively set a threshold to remove unreliable matches [27, 42, 43].

3.2 Correlation re-weighting

The main bottleneck for the aforementioned NC filtering and MM layer lies in the fact that the 4D correlation is very expensive to calculate and difficult to scale up. To deal with the

Table 1: Ablation study on a Tesla V100-SXM2 GPU with batch size of 16, Adam optimiser [23], ResNet18, 15 epochs, strong supervision, and test image up-sample resolution of 1.6K. The Sum of Area represents the overall MMA over multiple error bands.

Component/Method	DualRC	SparseNC	NCNet	$\mathbb{X}$RC$_1$	$\mathbb{X}$RC$_2$	$\mathbb{X}$RC$_3$	$\mathbb{X}$RC$_4$
4D correlation tensor	✓	✓	✓	✓	✓	✓	✗
NC filtering	✓	✓	✓	✗	✗	✗	✗
Mutual matching	✓	✗	✓	✓	✓	✗	✗
DualRC re-weighting	✓	✗	✗	✓	✗	✓	✓
Memory (GB)	6.78	3.73	5.40	4.57	4.25	4.36	4.21
Training time (s)	2.73	0.49	0.73	0.48	0.26	0.35	0.27
Sum of Area	3.90	3.20	3.61	3.65	2.49	3.36	3.26

problem, SparseNC [43] projects the correlation tensor onto a sub-manifold that contains the top k highest correlation scores for each source or target pixels. The 4D filtering is then approximated using the Minkowski operation [8]. In this way, the memory footprint can be dramatically reduced. Higher resolution images can fit into the memory leading to improved performance. However, such an approximation affects the accuracy as shown in [28]. Li *et al.* [28] propose to use a hierarchical architecture where the coarse resolution feature map $\mathbf{F}_c$ is used to calculate the 4D tensor for NC filtering and MM filtering. Then, the 2D correlation map $\mathbf{C}^c(x,y) \subset \mathbb{R}^{\Omega'_c}$ at location $(x,y) \in \mathbb{R}^{\Omega_c}$ is used to guide the searching for the best matches in the fine feature map by re-weighting the correlation map at the fine resolution $\mathbf{C}^f(x,y)$. Specifically, $\hat{\mathbf{C}}^f(x,y) = U(\mathbf{C}^c(x,y)) \cdot \mathbf{C}^f(x,y)$, where $U()$ is a de-parameterised up-sampling function, $\cdot$ represents the element-wise multiplication, and $\hat{\mathbf{C}}^f(x,y)$ is the re-weighted correlation map with the fine resolution. More accurate matches can be localised at $(\hat{x}',\hat{y}')_f = \arg\max_{\forall(x',y')\in\Omega'_f} \hat{\mathbf{C}}^f_{(x,y,x',y')}$. The correlation re-weighting contains no learnables.

3.3 Ablation study and 𝕏RCNet

To better understand the pros and cons of the mainstream DCN architectures, we conducted an ablation study over several state-of-the-art methods, namely, NCNet [42], SparseNC [43], and DualRC [28]. The left column in Tab. 1 lists the key modules shared by the DCNs. We tested the performance of all possible combinations of key modules using the same training protocol on the MegaDepth dataset [29] following the work of [12] - all baselines are trained with strong supervision with ResNet18 with 256 channels and hard relocalization [43] for a fair comparison. Evaluation of feature backbones is presented in Fig. 3, bottom. For each configuration, we record the average memory consumption, training speed, and overall matching accuracy. The accuracy measurements are the sum of the area below the Mean Matching Accuracy (MMA) curve on the HPatches dataset, comprised of challenging scenes with illumination and viewpoint variation. Fig. 4 c) shows the accuracy of MMA.

From the experiments, we observe that although all the modules of DCN contribute to the accuracy, they come with a variety of costs. Particularly, 1) the 4D NC filter consumes nearly 50% more memory and is more than 5 times slower comparing DualRC and $\mathbb{X}$RC$_1$. SparseNC reduces the expense of NC filter using the sparse 4D correlation with Minkowski convolution [8] but at the cost of degrading accuracy. 2) The DualRC re-weighting often plays an important role to the accuracy comparing DualRC with NCNet and $\mathbb{X}$RC$_1$ with $\mathbb{X}$RC$_2$. 3) Mutual matching layer contributes relative less to accuracy but is also cheap to calculate comparing $\mathbb{X}$RC$_1$ with $\mathbb{X}$RC$_3$ and therefore we do not remove it. 4) Removing both NC filtering and DualRC re-weight dramatically increases the speed but also decreases the accuracy for $\mathbb{X}$RC$_2$ significantly. 5) Removing the 4D correlation tensor, similar to UCN [9], hurts the performance for $\mathbb{X}$RC$_4$ compared to $\mathbb{X}$RC$_3$. To summarise, we select $\mathbb{X}$RC$_1$ as the

default architecture of 𝕏RCNet, which is about 5 times faster than DualRC but nearly 50% smaller in terms of memory costs. Also, it allows us to adopt 4 times larger batch size during training and can run up to 40 epochs in about same amount of time of training DualRC for 15 epochs. Fig. 1 (left) shows qualitative examples of removing the NC module.

The prediction of 𝕏RCNet is 2D correlation maps (Fig. 2). The loss can be then calculated using the F-norm between the prediction and the ground truth distribution [17, 27, 28, 56]. Particularly, we get the ground truth distribution by converting a 2D key point coordinate into a Probability Density Function (PDF). Specifically, we assign the probability of 4 pixels that are the nearest neighbours of the ground truth key point according to their normalised 2D distance. Then the PDF is further filtered by a 3×3 Gaussian kernel [27]. To keep the ablation test fair, all the networks in Tab. 1 are supervised using the same loss term.

𝕏RCNet Inference We distribute various key modules illustrated in Fig. 2 over multiple GPUs during inference to allow images with much larger resolution to be processed efficiently. Together with the low-cost but relatively accurate $\mathbb{X}\text{RC}_1$, we can address the critical question of how the input image resolution affects the matching accuracy of a DCN. To this end, we evaluate $\mathbb{X}\text{RC}_1$ using various image resolutions ranging from 1280 to 4K (Sec. 4.2). The source code for training and evaluation is attached in the supplementary material.

4 Experiments

Next, we describe the conducted experiments that evaluate the performance of 𝕏RCNet, the training strategy, and the relationship between the input resolution and matching accuracy.

Implementation details: The 𝕏RCNet training and evaluation code is implemented using PyTorch [39]. For the feature backbone, we mainly evaluated the ResNet101/50/18 [19], HRNet64/32/18 [7, 54] and the FPN256/128 [31]. The ResNet and HRNet are pre-trained on ImageNet [26] and kept fixed during all training procedures. The parameters of the FPN layers are trained from scratch. The configuration of ResNet101 is adopted from [42], the ResNet18 is truncated after the 3rd layer, the coarse feature map is extracted from the 3rd layer in the ResNet18 and the coarse layer is taken from the output of layer 1. The FPN architecture is identical to the original work of FPN [31] except that a ReLU layer is inserted before the feature normalisation. For HRNet we tested 18, 32, and 64 channels configurations. We truncated HRNet after the third stage in order to keep the input image ratio identical to ResNet. Here we considered both including and excluding the fusing (transition) stages. In addition, we use the output of the first branch as the fine feature map and the output of the third branch as the coarse feature map in order to be consistent to the fine to coarse ratio we used for ResNet. We train our model using the Adam optimiser with an initial learning rate of 0.01 and momentum 0.9. The batch size is 64. The learning rate is halved for every 5 epoch until 15 and remain constant till the 40th epoch. The model with lowest validation error is adopted for the final evaluation. It is worth pointing out that comparing with the training in Tab. 1 which only runs 15 epochs and uses batch size of 16 as previous work, the training with more epochs and a larger batch size result in a much higher accuracy which can be seen by comparing Fig. 4 b) and the bottom row of Tab. 1.

Training data We adopt the same training protocol as D2Net [12] on the MegaDepth dataset [29]. MegaDepth includes 196 scenes and the corresponding 3D point clouds created using SfM [48, 49]. The camera internal and external parameters are also jointly estimated and provided by the dataset. We follow the same methodology of [12] to extract a sparse set of ground truth correspondent points. Only image pairs with more than 50% of overlapping field of view are selected as training pairs (15,070). The validation image pairs (14,638) are selected from scenes containing more than 500 good pairs. All training pairs are randomly shuffled to avoid over-fitting to specific scenes.

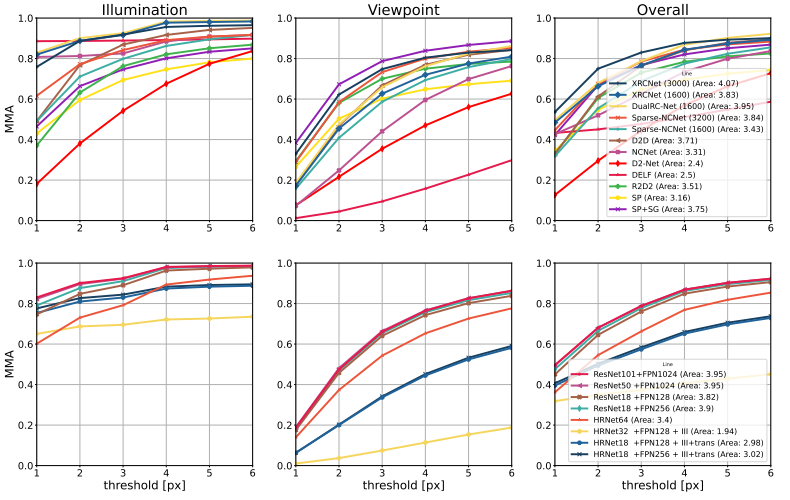


Figure 3: **Top:** Comparison of 𝕏RCNet with state-of-the-art correspondence networks on the HPatches dateset. **Bottom:** Comparison to different backbone architectures.

4.1 Correspondence Evaluation

HPatches is widely used for evaluating sparse feature matching and dense correspondence algorithms [2]. It contains two main challenges — the viewpoint and illumination variations consisting of 56 and 52 sequences of testing images respectively. Each sequence contains 6 images and the first image is matched against the remaining ones. The native image size is reported in Tab 2. Testing images contain both indoor and outdoor scenes. The ground truth homography is provided so that the correspondences can be densely evaluated. The evaluation procedure is adopted from [12, 28, 41, 43] to allow direct comparison with these baseline methods. The evaluation metric used is the Mean Matching Accuracy (MMA) that estimates the average number of correct matches over the total number of matches using top 2000 proposed matches by the testing neural networks, where the correct matches are defined as the distance between the predicted 2D key points to ground truth. 𝕏RCNet sets a new accuracy standard from the comparative evaluation graph shown in Fig. 3, top.

Aachen Day-Night dataset [46, 47] is a challenging outdoor relocalisation dataset. The Day-Night challenge contains 98 night query images to be relocalised with respect to 20 day-time candidate images. The performance of 𝕏RCNet compared to the baselines on the night query images is shown in Tab. 3, while an example qualitative comparison and the produced 2D heatmap in the reference image are shown in Fig. 5 (right).

Table 2: Size statistics for each dataset. The minimum, mean, and maximum size over height and width recorded. HPatches — lowest mean image resolution, InLoc — highest.

	HPatches		InLoc		Aachen	
	h	w	h	w	h	w
min	380	512	1200	1200	1063	1063
mean	780	980	2397	2531	1268	1498
max	1411	1536	4032	4032	1600	1600

We provide 3D reconstruction results of 𝕏RCNet and DualRC in the supplementary material. 𝕏RCNet achieves comparable performance to the state-of-the-art, while having a smaller memory footprint for the used input resolution size and faster inference speed (Tab. 1).

InLoc mainly contains indoor images captured with a different type of sensors [51]. It is a popular benchmark for evaluating the accuracy of camera localisation with respect to large variety of indoor scenes. Reference images are obtained with a 3D scanner and the query images are captured using a mobile phone several months later to introduce extra non-

Table 3: Evaluation on the Aachen dataset. The localization results are reported as the percentage of query images which were localized with in the three error bands during night.

Error Band	ASLFeat+OANet	D2-Net	SparseNC	R2D2	DualRC-Net	SP + SG	𝕏RCNet-1.6k
0.25m & 2°	77.6	74.5	76.5	76.5	79.6	79.6	76.5
0.5m & 5°	89.8	86.7	84.7	90.8	88.8	90.8	85.7
5m & 10°	100.0	100.0	98.0	100.0	100.0	100.0	100.0

Table 4: Evaluation on InLoc. Best result is shown in **bold** and second best is underlined. The used metric is the percentage of query images which were localized successfully.

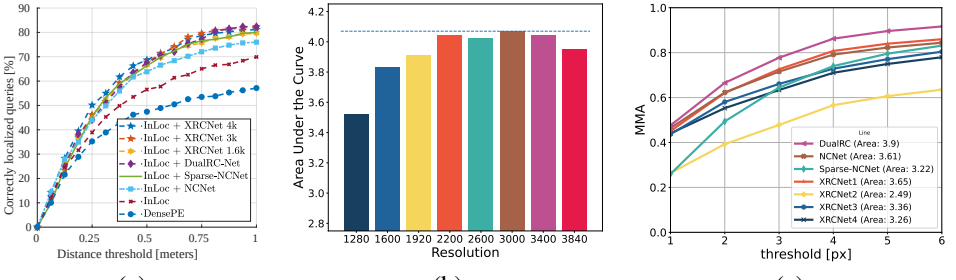
Error Band	DualRC	SparseNC	NCNet	InLoc	DensePE	D2-Net	R2D2	𝕏RCNet-1.6k	𝕏RCNet-3k	𝕏RCNet-4k
0.25m & 10°	44.1	45.6	44.1	38.9	35.3	43.2	47.3	44.7	46.2	**50.2**
0.5m & 10°	67.5	66.3	63.8	56.5	47.4	61.1	67.2	66.6	67.8	**68.7**
1m & 10°	**82.4**	79.9	76.0	69.9	57.1	74.2	73.3	79.6	**82.4**	81.2

static challenges. InLoc contains significant viewpoint changes and illumination variation. We adopt the evaluation procedure of [51] to find the top 10 candidate database images for each query image. 𝕏RCNet is used to calculate the matches between them, and the final 6D camera pose is estimated using P*n*P [14] and dense pose verification [51]. The results are provide in Fig. 4 a) and Tab. 4 which show 𝕏RCNet significantly outperform the others.

4.2 Optimal Resolution 𝕏

A common practice to achieve better accuracy in previous works, is to up-sample the original images to a higher resolution that almost consistently improve the final matching accuracy [28, 42, 43] as long as the network can fit into the GPU memory. However, up-sampling the input image to infinity causes issues because the information contained in the original image is fixed. Increasing the image size implies the receptive field of a deep neural network will reduce and so will the descriptiveness of the feature maps. Therefore, there must be an optimal resolution 𝕏 for a network to achieve its best performance for a given input. Thanks to the light-weight design of 𝕏RCNet and the multi-GPU inference, we ran a series of experiments to confirm the existence of the optimal 𝕏 given a pre-trained 𝕏RCNet by varying the up-sampling rate of the testing images.

Particularly, we resize the image of HPatches from 1280 pixels up to 4K (3840×2160) with fixed aspect ratio and evaluate. Fig. 4 b) shows the total area under the accuracy curve of MMA. We discover that the best matching accuracy increases with respect to the image resolution and gradually saturates around the resolution 3k. Using an image resolution higher than 3k reduces the matching accuracy. The accuracy at 𝕏RCNet-3k surpassed the state-of-the-art DCN performance in the same error band on both HPatches (Fig. 3, top). For InLoc, 𝕏RCNet-4k further surpassed 𝕏RCNet-3k (See Tab. 4 and Fig. 4 a)) possibly because of



(a) (b) (c)

Figure 4: (a) Pose accuracy of InLoc measured by the percentage of correctly localised queries over different distances. (b) Evaluation of the optimal up-sampling resolution. (c) Quantitative evaluation of Neighbourhood Consensus architectures on the HPatches dataset.

Figure 5: **Left quadrant**: top row - ResNet101, bottom row - light-weight ResNet18. Left column - with NC filtering, right column - without NC filtering. Removing the NC affects the matching accuracy. **Right**: the query night image with the chosen keypoint location (red star) and the corresponding heatmap from a ResNet18 model without NC filtering but using a high resolution image. The increased resolution balances out the NC filtering issue.

the relatively larger mean native resolution shown in Tab 2. Unfortunately, the 𝕏RCNet-1.6k gives the best performance on Achen Day-Night which is inconsistent. However, we observe individual cases illustrated in Fig. 5, up-sampling remains effective as the heatmap of the 𝕏RCNet-3k (right) is less ambiguous in the repetitive regions over 𝕏RCNet-1.6k (left). The inconsistent results on Achen Day-Night is possibly due to the much smaller number of testing pairs compared with other datasets (98 pairs in Achen Day-Night vs 108×5 pairs in HPatches and 329×10 pairs in InLoc). Based on these analysis, we suggest a validation set with various image upsampling to define the optimal resolution. In addition, we have evaluated the testing time and compared to [28]. DualRC-1.6k has a sum area under the MMA cuve of 3.95, takes 8.3 seconds and 8.7GB of memory to compute. XRCNet-1.6k has a sum of area 3.83, requires 1.6 seconds and 2.2GB of memory. XRCNet-2.2k has a sum of area 4.02, requires 5.69 seconds and 4.6GB of memory to compute.

4.3 Feature Backbone

In the end, we show experiments with different feature backbone architectures on HPatches. We have evaluated the matching accuracy using variant of both the ResNet and the HRNet backbones. In Fig. 3, bottom, it can be seen that when using ResNet18 and ResNet50, the performance of DualRC is almost identical to the original DualRC with ResNet101. HRNet is another candidate we consider to replace the original feature backbone for DualRC. However, HRNet seems less competitive when integrated with the correspondence network. We also tested different FPN channels - 128 and 256 despite the relatively small cost. Using 128 channels does not affect the accuracy much, and thus we adopt 256 channels.

5 Conclusion

In this paper, we propose the 𝕏Resolution Correspondence Network, which is the result of a systematic study of the state-of-the-art dense correspondence networks. We noticed that a key component of these networks — the learned 4D correlation tensor — does not have a huge impact on the performance. Therefore, removing the 4D filtering with learnable parameters allows 𝕏RCNet to learn quicker and enables it to process input images with resolution up to 4K. The proposed DCN architecture outperforms state-of-the-art on HPatches and InLoc, and enables us to investigate the intriguing question if increasing the input image resolution is always beneficial to matching accuracy. Through extensive experimentation and a thorough ablation study we observe a saturation of the matching performance over the optimal resolution 𝕏. We hope this work can shed light on how to design efficient and effective correspondence networks, while acting as a first step towards the interesting problem how the scale differences in input images affect DCNs.

References

[1] Sameer Agarwal, Yasutaka Furukawa, Noah Snavely, Ian Simon, Brian Curless, Steven M Seitz, and Richard Szeliski. Building rome in a day. *Communications of the ACM*, 54(10):105–112, 2011.

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

[3] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2006.

[4] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. BRIEF: Binary robust independent elementary features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 778–792, 2010.

[5] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected CRFs. In Yoshua Bengio and Yann LeCun, editors, *Proceedings of Intl. Conf. on Learning Representations (ICLR)*, 2015.

[6] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*, pages 1597–1607. PMLR, 2020.

[7] Bowen Cheng, Bin Xiao, Jingdong Wang, Honghui Shi, Thomas S. Huang, and Lei Zhang. HigherHRNet: Scale-Aware Representation Learning for Bottom-Up Human Pose Estimation. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[8] Christopher Choy, Junyoung Gwak, and Silvio Savarese. 4D spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 3070–3079, 2019.

[9] Christopher B Choy and Silvio Savarese. Universal Correspondence Network. In *Proceedings of Conf. on Neural Information Processing Systems (NeurIPS)*, pages 1–9, 2016.

[10] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 3213–3223, 2016.

[11] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[12] 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. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[13] Dorian Galvez-Lopez and Juan D. Tardos. Bags of Binary Words for Fast Place Recognition in Image Sequences. *IEEE Trans. on Robotics (ToR)*, 28(5):1188–1197, 2012.

[14] Xiao-shan Gao, Xiao-rong Hou, Jianliang Tang, and Hang-fei Cheng. Complete solution classification for the perspective-three-point problem. *IEEE Trans. Pattern Anal. Machine Intell. (PAMI)*, 25(8):930–943, 2003.

[15] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[16] Steffen Gauglitz, Tobias Höllerer, and Matthew Turk. Evaluation of interest point detectors and feature descriptors for visual tracking. *Intl. Journal of Computer Vision (IJCV)*, 2011.

[17] Hugo Germain, Guillaume Bourmaud, and Vincent Lepetit. S2DNet: Learning Accurate Correspondences for Sparse-to-Dense Feature Matching. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[18] Kai Han, Rafael S. Rezende, Bumsub Ham, Kwan-Yee K. Wong, Minsu Cho, Cordelia Schmid, and Jean Ponce. Scnet: Learning semantic correspondence. In *Proceedings of Intl. Conf. on Computer Vision (ICCV)*, 2017.

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[20] Tong He, Zhi Zhang, Hang Zhang, Zhongyue Zhang, Junyuan Xie, Mu Li, and Amazon Web Services. Bag of Tricks for Image Classification with Convolutional Neural Networks. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[21] Shuaiyi Huang, Qiuyue Wang, Songyang Zhang, Shipeng Yan, and Xuming He. Dynamic context correspondence network for semantic alignment. In *Proceedings of Intl. Conf. on Computer Vision (ICCV)*, 2019.

[22] Hervé Jégou, Florent Perronnin, Matthijs Douze, Jorge Sánchez, Patrick Pérez, and Cordelia Schmid. Aggregating local image descriptors into compact codes. *IEEE Trans. Pattern Anal. Machine Intell. (PAMI)*, 34(9):1704–1716, 2011.

[23] Diederik P. Kingma and Jimmy Lei Ba. Adam: a Method for Stochastic Optimization. In *Proceedings of Intl. Conf. on Learning Representations (ICLR)*, pages 1–15, 2015.

[24] Junghyup Lee, Dohyung Kim, Jean Ponce, and Bumsub Ham. Sfnet: Learning object-aware semantic correspondence. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[25] Stefan Leutenegger, Margarita Chli, and Roland Y Siegwart. Brisk: Binary robust invariant scalable keypoints. In *Proceedings of Intl. Conf. on Computer Vision (ICCV)*, 2011.

[26] Li-Jia Li, Kai Li, Fei Fei Li, Jia Deng, Wei Dong, Richard Socher, and Li Fei-Fei. ImageNet: a Large-Scale Hierarchical Image Database Shrimp Project View project hybrid intrusion detction systems View project ImageNet: A Large-Scale Hierarchical Image Database. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[27] Shuda Li, Kai Han, Theo W. Costain, Henry Howard-Jenkins, and Victor Prisacariu. Correspondence networks with adaptive neighbourhood consensus. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[28] Xinghui Li, Kai Han, Shuda Li, and Victor Adrian Prisacariu. Dual-Resolution Correspondence Networks. In *Proceedings of Conf. on Neural Information Processing Systems (NeurIPS)*, 2020.

[29] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[30] Jing Liao, Yuan Yao, Lu Yuan, Gang Hua, and Sing Bing Kang. Visual attribute transfer through deep image analogy. *ACM Trans. on Graphics (ToG)*, 36(4):120:1–120:15, July 2017. ISSN 0730-0301. doi: 10.1145/3072959.3073683. URL http://doi.acm.org/10.1145/3072959.3073683.

[31] Tsung-yi Lin, Piotr Doll, Ross Girshick, Kaiming He, Bharath Hariharan, Serge Belongie, and Facebook Ai. Feature pyramid networks for object detection. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[32] Ce Liu, Jenny Yuen, and Antonio Torralba. Sift flow: Dense correspondence across scenes and its applications. *IEEE Trans. Pattern Anal. Machine Intell. (PAMI)*, 2011.

[33] David G Lowe. Distinctive image features from scale-invariant keypoints. *Intl. Journal of Computer Vision (IJCV)*, 2004.

[34] Zixin Luo, Lei Zhou, Xuyang Bai, Hongkai Chen, Jiahui Zhang, Yao Yao, Shiwei Li, Tian Fang, and Long Quan. Aslfeat: Learning local features of accurate shape and localization. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[35] Juhong Min, Jongmin Lee, Jean Ponce, and Minsu Cho. Hyperpixel flow: Semantic correspondence with multi-layer neural features. In *Proceedings of Intl. Conf. on Computer Vision (ICCV)*, 2019.

[36] Juhong Min, Jongmin Lee, Jean Ponce, and Minsu Cho. Learning to Compose Hypercolumns for Visual Correspondence. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

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

[38] Hyeonwoo Noh, Andre Araujo, Jack Sim, Tobias Weyand, and Bohyung Han. Large-Scale Image Retrieval with Attentive Deep Local Features. In *Proceedings of Intl. Conf. on Computer Vision (ICCV)*, pages 3476–3485, 2017.

[39] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Proceedings of Conf. on Neural Information Processing Systems (NeurIPS)*, 2019.

[40] Francesco Pittaluga, Sanjeev J Koppal, Sing Bing Kang, and Sudipta N Sinha. Revealing Scenes by Inverting Structure from Motion Reconstructions. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[41] Jerome Revaud, Philippe Weinzaepfel, César De Souza, Noe Pion, Gabriela Csurka, Yohann Cabon, and Martin Humenberger. R2D2: Repeatable and Reliable Detector and Descriptor. In *Proceedings of Conf. on Neural Information Processing Systems (NeurIPS)*, 2019.

[42] Ignacio Rocco, Mircea Cimpoi, Relja Arandjelović, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Neighbourhood consensus networks. In *Proceedings of Conf. on Neural Information Processing Systems (NeurIPS)*, 2018.

[43] Ignacio Rocco, Relja Arandjelović, and Josef Sivic. Efficient neighbourhood consensus networks via submanifold sparse convolutions. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[44] Edward Rosten, Reid Porter, and Tom Drummond. Faster and better: a machine learning approach to corner detection. *IEEE Trans. Pattern Anal. Machine Intell. (PAMI)*, 2010.

[45] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[46] Torsten Sattler, Tobias Weyand, Bastian Leibe, and Leif Kobbelt. Image Retrieval for Image-Based Localization Revisited. In *British Machine Vision Conference (BMCV)*, 2012.

[47] Torsten Sattler, Will Maddern, Carl Toft, Akihiko Torii, Lars Hammarstrand, Erik Stenborg, Daniel Safari, Masatoshi Okutomi, Marc Pollefeys, Josef Sivic, Fredrik Kahl, and Tomas Pajdla. Benchmarking 6DOF Outdoor Visual Localization in Changing Conditions. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[48] Johannes L. Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[49] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016.

[50] Johannes Lutz Schönberger, Hans Hardmeier, Torsten Sattler, and Marc Pollefeys. Comparative Evaluation of Hand-Crafted and Learned Local Features Comparative Evaluation of Hand-Crafted and Learned Local Features. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[51] Hajime Taira, Masatoshi Okutomi, Torsten Sattler, Mircea Cimpoi, Marc Pollefeys, Josef Sivic, Tomas Pajdla, and Akihiko Torii. InLoc: Indoor visual localization with dense matching and view synthesis. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[52] Yurun Tian, Bin Fan, and Fuchao Wu. L2-Net: Deep learning of discriminative patch descriptor in Euclidean space. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 6128–6136, 2017.

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

[54] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, Wenyu Liu, and Bin Xiao. Deep High-Resolution Representation Learning for Visual Recognition. *IEEE Trans. Pattern Anal. Machine Intell. (PAMI)*, 8828, 2020.

[55] Jamie Watson, Oisin Mac Aodha, Daniyar Turmukhambetov, Gabriel J Brostow, and Michael Firman. Learning Stereo from Single Images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[56] Olivia Wiles, Sebastien Ehrhardt, and Andrew Zisserman. D2D: Learning to find good correspondences for image matching and manipulation. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[57] Nan Yang, Lukas von Stumberg, Rui Wang, and Daniel Cremers. D3VO: Deep Depth, Deep Pose and Deep Uncertainty for Monocular Visual Odometry. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[58] Kwang Moo Yi, Eduard Trulls, Yuki Ono, Vincent Lepetit, Mathieu Salzmann, and Pascal Fua. Learning to Find Good Correspondences. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 2666–2674, 2018.

[59] Andrei Zanfir and Cristian Sminchisescu. Deep Learning of Graph Matching. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 2684–2693, 2018.

[60] Jiahui Zhang, Dawei Sun, Zixin Luo, Anbang Yao, Lei Zhou, Tianwei Shen, Yurong Chen, Hongen Liao, and Long Quan. Learning two-view correspondences and geometry using order-aware network. In *Proceedings of Intl. Conf. on Computer Vision (ICCV)*, 2019.

[61] Zichao Zhang, Torsten Sattler, and Davide Scaramuzza. Reference Pose Generation for Visual Localization via Learned Features and View Synthesis. *Intl. Journal of Computer Vision (IJCV)*, 2020.