Style Transfer for Keypoint Matching Under Adverse Conditions

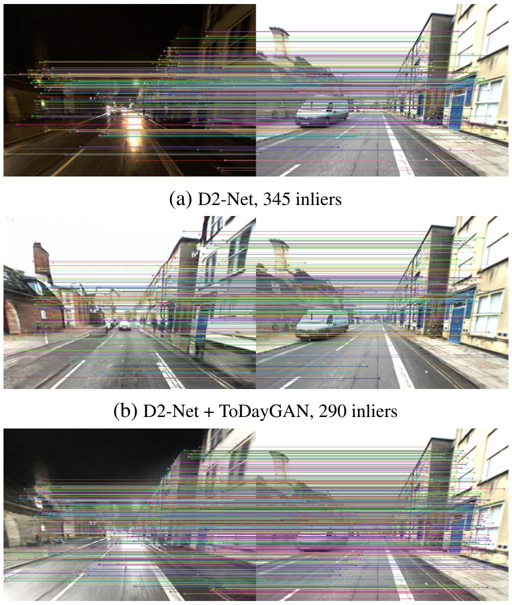
Ali Uzpak[1] Abdelaziz Djelouah[2] Simone Schaub-Meyer[1,3]

[1]Dept. of Computer Science, ETH Zurich, Switzerland
[2]DisneyResearch|Studios, Zurich, Switzerland
[3]Dept. of Computer Science, TU Darmstadt, Germany

uzpaka@ethz.ch, abdelaziz.djelouah@disney.com, simone.schaub@inf.ethz.ch

Abstract

In this work, we address the difficulty of matching local features between images captured at distant points in time resulting in a global appearance change. Inspired by recent neural style transfer techniques, we propose to use an image transformation network to translate night images into day-like appearance, with the objective of better matching performance. We extend traditional style transfer, that optimize for content and style, with a keypoint matching loss function. The joint optimization of these losses allows our model to generate images that can significantly improve the performance of local feature matching, in a self-supervised way. As a result, our approach is flexible and does not require paired training data, which is difficult to obtain in practice. We show how our method can be used as an extension to a state-of-the-art differentiable feature extractor to improve its performance in challenging scenarios. This is demonstrated in our evaluation on day-night image matching and visual localization tasks with night-rain image queries.



(a) D2-Net, 345 inliers

(b) D2-Net + ToDayGAN, 290 inliers

(c) D2-Net + Transformer (ours), 674 inliers

Figure 1: Comparison between different methods on a night-day image pair from RobotCar [17]. The first row shows the results of plain D2-Net [5]. The second row uses the ToDayGAN model [2] to translate the night image into day before computing D2-Net features. Our approach (third row) uses a self-supervised style transfer combined with D2-Net to transform the night image in a way that significantly improves the matching results.

1. Introduction

Visual localization, the computation of the camera pose and orientation from images, is an important part of applications such as 3D reconstruction [6], image-based rendering [30] and augmented reality and robotics [16]. At the core, visual localization often relies on accurately computing and matching image keypoints of a query image with respect to reference images. However, especially in outdoor scenarios, these images can largely vary in terms of illumination, season, scene structure and viewpoint, making matching inherently challenging. An ideal feature extractor should be invariant to these changes while still detecting and accurately matching keypoints.

Sparse local feature extractors such as SIFT [15] have proven to be very suitable for applications requiring precise pixel-correspondences, but have difficulties with extreme illumination changes [36]. An alternative to designing handcrafted invariant feature extractors is to learn this invariance from data [5, 20, 34]. However, this requires corresponding training data and robust features often come with a trade-off with respect to accuracy [5]. Recent advances in image transformations [9] have shown an alternative to address this domain gap. The goal is to modify the query image to resemble reference images to improve the performance of localization [2, 19]. However, these methods require training data from the targeted different conditions and the improvement of the performance is limited to the trained transformation.

In our method we address the challenge of matching images under adverse conditions without the need of additional training data. Instead we infer the transformation directly from the image pair to be matched, which allows handling various appearance transformations. The proposed solution is an extension of state-of-the-art differentiable feature description pipeline designed to increase the number of correct matches in cases of severe differences in appearance, see Figure 1. The goal is to reduce the domain gap of the images by transforming the query image accordingly and performing the image matching on the transformed image. While traditional style transfer methods synthesize visually pleasant images, this does not automatically lead to a better matching performance. To address this problem, we introduce an additional matching loss such that the image transformation targets matching performance.

The contribution of this work is threefold: (1) We propose a novel approach for keypoint matching under adverse conditions, through style transfer based image translation. In particular we propose optimizing an additional matching loss for the translation of query images. (2) The proposed translation is self-supervised and does not require training data. It can be applied to any image pair and there is no explicit constraint on the type of adverse condition (night, rain, etc.). (3) Our evaluation demonstrates the effectiveness of the proposed solution on both image matching and localization in challenging scenarios.

2. Related Work

Handcrafted local feature descriptors such as SIFT [15], are still very common and often used in practice. A comprehensive overview can be found in [18]. These traditional methods perform feature matching by first detecting keypoints and then computing matches between the descriptors. However, they have difficulties in severe illumination changes such as matching day-night images [36].

With the increased success of deep learning in various computer vision areas, learning invariant features from data was also explored [25]. Now, most recent methods suggest to learn to *detect-and-describe* keypoints in a single step [4, 5, 20] to increase the robustness of local features. On top of the detection and descriptor stage, SuperGlue [22] recently proposed a network for learned matching. These methods have achieved state-of-the-art results in visual localization. Another alternative is to circumvent the detection stage for the night image and densely selecting keypoints [8]. The increase in robustness comes with an increase of computation and memory requirements.

One direction to improve long-term localization is to rely on semantic information [26, 31, 32]. For example Schonberger *et al.* [26] require depth maps and rely jointly on 3D geometric and semantic information. Stenborg *et al.* [31] perform semantic segmentation on the query image which is used together with a 3D semantic segmentation of the scene. Finally, Toft *et al.* [32] integrate the semantic segmentation of the images with the standard visual localization pipeline. The proposed solution uses semantic matching score to prioritize more consistent matches during RANSAC-based pose estimation. Despite the improvements, these approaches are not applicable to a wide variety of scenarios and require ground truth segmentation data for training.

A more promising direction is to consider transforming the challenging query image to resemble more the reference images, for improved matching and localization performances. Image-to-image translation methods [9, 14, 37] are among the main methods to bridge this domain gap. ToDayGAN [2] uses unsupervised image-to-image transformation [1, 37] to improve the localization performance of night query images. Before matching, query images are transformed by a network to *day-like* images. The used transformer network is trained on a set consisting of day and night images. By using CycleGAN [37], the method does not require aligned image pairs, however the improved localization is still mainly limited to transformations corresponding to the training dataset. A similar approach, which also uses cycle-consistency of GANs, is presented by Porav *et al.* [19]. Their approach also includes a descriptor specific loss helping to generate images suitable for matching, but requires a fine-tuning stage with aligned image pairs that are difficult to obtain in practice. Moreover, similarly to [2], their transformer is still pretrained to handle a specific domain shift such as night-to-day or rain-to-day. Each transformation requires a different network trained on a corresponding training set containing images from source and target domain.

Our method also uses image translation to improve matching, however the strategy we adopt is based on style transfer [7], which allows to modify an input image (the query) based on the style of a reference image. Originally applied to transfer abstract artistic styles, recent methods also focus on creating photorealistic images [13, 35]. However, the objective is to generate visually pleasant result, which does not automatically result in images suitable for feature extractors. Local low-level images statistics of these artificially created images can be drastically different compared to natural images, making it even more difficult to find matches with existing methods using local features. By integrating a matching loss, our solution ensures that the image translation improves the matching performance. The proposed transformer network is self-supervised, making it more flexible and reduces the necessity of a specific training data.

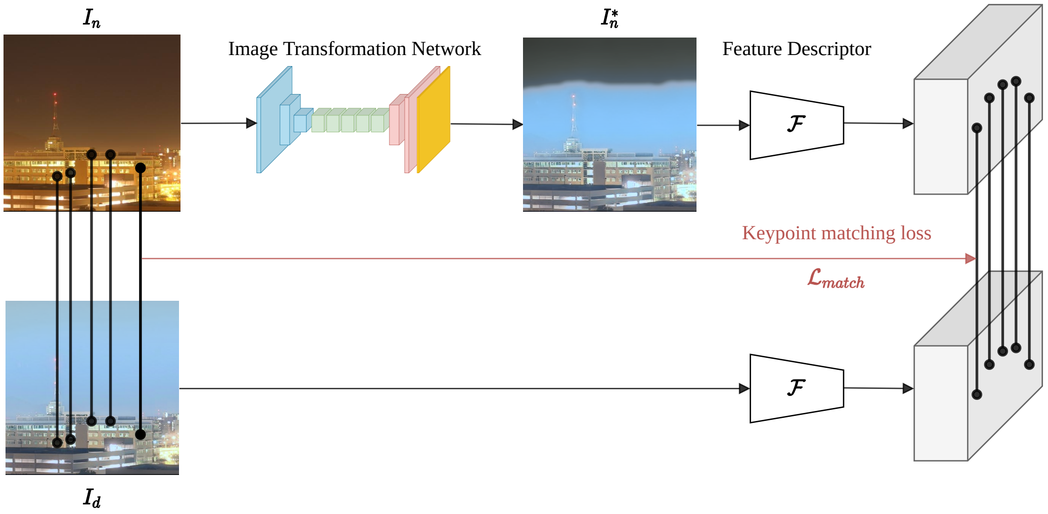


Figure 2: Our transformer network approach. Given day and night images I_d and I_n to be matched, we generate an intermediate image I_n^* by optimizing standard content and style losses (not illustrated here) combined with our proposed keypoint matching loss. Based on initial keypoint matches between I_n and I_d, this formulation helps the transformer network to generate an image I_n^* more suitable for matching with I_d compared with traditional style transfer methods. $\mathcal{F}$ corresponds to a fixed descriptor model such as D2-Net [5].

3. Method

The objective of this work is to propose a transformation of images taken under adverse conditions (nigh, rain, etc.) to day-like conditions, more similar to the reference images used for localization to improve the matching performance. We frame this matching problem under the style transfer framework, where the content image corresponds to the query image (e.g. night) and the style image corresponds to the reference image, i.e. normally day images. Below we start with a description of the classic optimization-based style transfer [7], followed by our proposed solution to improve the matching.

Image transformation network. Let $\mathcal{T}$ be an image transformation network (that we also call *transformer*) and I_n, I_d representing the night and day RGB images, or more generally the query and reference images, that we want to match. Our goal is to translate I_n using $\mathcal{T}$ into a new RGB image $I_n^* = \mathcal{T}(I_n)$ resembling more the appearance style of I_d. The transformation is obtained by iteratively modifying the input image I_n with respect to the style of I_d by minimizing a style loss, $\mathcal{L}_{style}(I_n^*, I_d)$, and a content loss, $\mathcal{L}_{content}(I_n^*, I_n)$:

$$\mathcal{L}_{image} = \mathcal{L}_{style}(I_n^*, I_d) + \mathcal{L}_{content}(I_n^*, I_n)\,, \qquad (1)$$

where we assume that the transferred style representation of I_d accurately captures the day illumination condition. We use the same transformer network as in [11], which is a fully convolutional encoder-decoder model illustrated in Figure 2.

The style representation of an image can be captured with the Gram matrix of CNN features [7] using a fixed VGG-16 network [29] pretrained on ImageNet [21]. Let ϕ be this VGG-16 network, where $\phi_j(x)$ denotes the $H_j \times W_j \times C_j$ feature map at layer j, for the input image $x \in \mathbb{R}^{H\times W\times 3}$. The Gram matrix at layer j can be defined as

$$G_j(x) = \phi'_j(x)^T\ \phi'_j(x) \in \mathbb{R}^{C_j\times C_j}\,, \qquad (2)$$

where $\phi'_j(x)$ is the 2D matrix representation of the j-th feature map with size $H_jW_j \times C_j$, obtained with a simple reshaping operation. The style loss is then defined as

$$\mathcal{L}_{style}(I_n^*, I_d) = \sum_{j\in\mathcal{S}} \left\| \frac{G_j(I_n^*) - G_j(I_d)}{H_jW_jC_j} \right\|_2^2\,, \qquad (3)$$

where $\mathcal{S} = \{relu1\_2,\ relu2\_2,\ relu3\_3,\ relu4\_3\}$ is the considered set of VGG-16 layers.

The objective of the content loss is to maintain similarity to the original I_n image. Because we want this similarity to be only on a higher conceptually level and not perfectly on a per pixel-to-pixel basis we use a *perceptual* content loss [11] defined as

$$\mathcal{L}_{content}(I_n^*, I_n) = \frac{1}{H_jW_jC_j}\,\|\phi_j(I_n^*) - \phi_j(I_n)\|_2^2\,, \qquad (4)$$

where $j = relu3\_3$.

Keypoint matching loss. Unfortunately, the objective functions $\mathcal{L}_{content}$ and $\mathcal{L}_{style}$ do not explicitly encourage the local features of I_n^* and I_d to match better and, as we show in the experiments section, traditional style transfer alone cannot guarantee a superior matching performance in general. We address this issue by additionally using a *keypoint matching loss* $\mathcal{L}_{match}$.

Let $\mathcal{F}$ be a differentiable and fixed local feature model that outputs a dense set of descriptors $\mathcal{F}(I) \in \mathbb{R}^{H \times W \times D}$ for an input image I. A local descriptor at a pixel location (i, j) would then correspond to $\mathcal{F}(I)_{ij} \in \mathbb{R}^D$. Using $\mathcal{F}$ and a keypoint detector, we compute an initial set of keypoint matches between I_n and I_d. We note $\mathcal{K}$ the resulting set of sparse keypoint correspondences $c = (i, j, k, l)$, where pixels (i, j) in I_n match pixels (k, l) in I_d. These matches are obtained with a mutual nearest neighbor criterion on the descriptors and geometrically verified with RANSAC when fitting a fundamental matrix model. We choose $\mathcal{F}$ to be the D2-Net [5] but other differentiable local feature models could be considered.

Our goal is to force the transformer network to preserve the point correspondences from $\mathcal{K}$ when generating I_n^*. To do so, we choose the triplet loss function used for the D2-Net model [5] to minimize the distance between descriptors of positive correspondences from $\mathcal{K}$ while maximizing the distance for the hardest negative correspondences. In particular, for a keypoint correspondence $c = (i, j, k, l) \in \mathcal{K}$, we define the positive distance $\mathcal{P}$ as

$$\mathcal{P}(i, j, k, l) = ||\mathcal{F}(I_n^*)_{ij} - \mathcal{F}(I_d)_{kl}||_2 \tag{5}$$

and the negative distance $\mathcal{N}$ as

$$\begin{aligned}\mathcal{N}(i, j, k, l) = min(&||\mathcal{F}(I_n^*)_{ij} - \mathcal{F}(I_d)_{k'l'}||_2, \\ &||\mathcal{F}(I_n^*)_{i'j'} - \mathcal{F}(I_d)_{kl}||_2)\,,\end{aligned} \tag{6}$$

where the hard negative points (i', j') and (k', l') are picked as follows:

$$\begin{aligned} &(i', j') = \underset{(x,y)}{\arg\min}\, ||\mathcal{F}(I_n^*)_{xy} - \mathcal{F}(I_d)_{kl}||_2\,, \\ &\text{s.t.} \quad max(|x - i|, |y - j|) > t \\ \text{and} \\ &(k', l') = \underset{(x,y)}{\arg\min}\, ||\mathcal{F}(I_n^*)_{ij} - \mathcal{F}(I_d)_{xy}||_2\,, \\ &\text{s.t.} \quad max(|x - k|, |y - l|) > t\,. \end{aligned} \tag{7}$$

The parameter t forces the hard negatives (i', j') and (k', l') to be at least t pixels away from the positive points (i, j) and (k, l), respectively. The matching loss $\mathcal{L}_{match}(I_n^*, I_d)$ is then defined as

$$\mathcal{L}_{match}(I_n^*, I_d) = \sum_{c \in \mathcal{K}} max(0,\, \mathcal{P}(c)^2 - \mathcal{N}(c)^2 + m) \tag{8}$$

for a chosen margin m.

We define our final loss $\mathcal{L}$ for the transformer network $\mathcal{T}$ as a weighted sum:

$$\begin{aligned}\mathcal{L} \ = \lambda_c\, \mathcal{L}_{content}(I_n^*, I_n) + \lambda_s\, \mathcal{L}_{style}(I_n^*, I_d) + \\ \lambda_m\, \mathcal{L}_{match}(I_n^*, I_d)\,,\end{aligned} \tag{9}$$

where $\lambda_c, \lambda_s, \lambda_m$ are hyper parameters that weigh the importance of each loss. For optimizing Equation 9, only the parameters of the transformation network $\mathcal{T}$ are updated while the feature descriptor network $\mathcal{F}$ stays fixed. Once the optimization is done, we use the same local features model $\mathcal{F}$ to compute new features and matches between I_n^* and I_d instead of the original pair I_n and I_d.

Keypoint accuracy. Similarly to most conventional CNN architectures, the D2-Net model uses pooling operations to reduce the spatial resolution of the input while increasing the receptive field. One of the advantages of pooling is to reduce the computation time and memory requirements of the network, which is important when dealing with high resolution images. On the other hand, D2-Net keypoints that are detected in those lower resolution feature maps will be less precise and detrimental to subsequent applications such as 3D reconstruction or image matching, as pointed out by the authors [5].

In this work we propose to compute the keypoint locations on an image $I \in \mathbb{R}^{H \times W \times 3}$ with a full resolution model such as [20] and use the descriptors from D2-Net [5] extracted with $\mathcal{F}$, where $\mathcal{F}(I) \in \mathbb{R}^{H_c \times W_c \times D}$ and $H_c < H$ and $W_c < W$. We can then retrieve the corresponding descriptors in $\mathcal{F}(I)$ by scaling the keypoint position with the factors W_c/W, H_c/H and using bilinear interpolation. Our experiments show that this leads to a good trade-off between robustness and accuracy for the baseline model.

4. Experimental Evaluation

In this experimental section we evaluate the advantage of using the proposed style transfer method for both image matching and localization in adverse conditions. However, before evaluating these experiments, we first also demonstrate the advantage of the proposed combination of keypoint detector on full resolution and the interpolation of D2-Net features to set the baseline. Furthermore, we provide more details regarding the architecture, the optimization process and evaluation procedure.

We use 3 different datasets for our experiments. The first one, referred to as *HPatches*, is used for the baseline evaluation of keypoint/descriptor matching accuracy. It consists of a subset of the original HPatches dataset [3] with 540 pairs of images with illumination or viewpoint changes as described in [5]. The second dataset is a modified version of the Day-Night Image Matching dataset [36]. The DNIM dataset contains 17 sequences of various city scenes captured by a fixed webcam throughout the day resulting in capturing illumination changes that can be drastic. In our modified version (MDNIM), we pick 6 day and 6 night images that we exhaustively match, yielding 36 pairs per scene and 612 pairs in total. Since cameras are fixed, we have trivial ground-truth pixel-to-pixel correspondences to

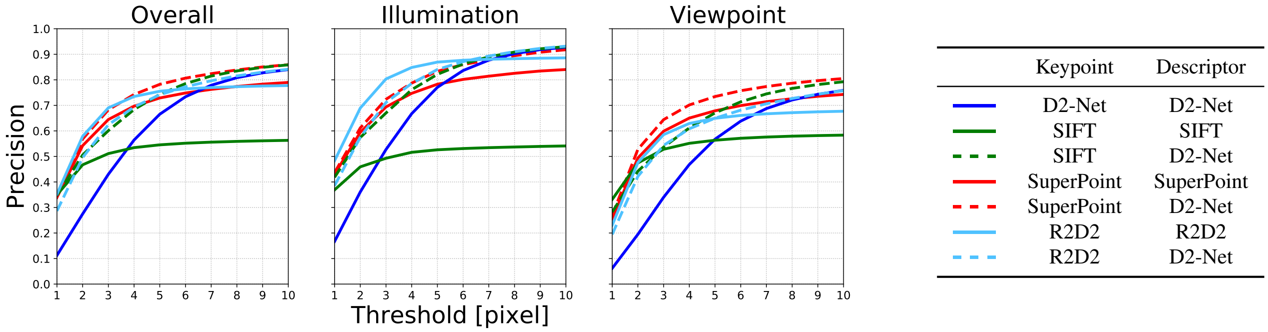


Figure 3: Matching precision on HPatches [3] image pairs for different keypoint-descriptor combinations. D2-Net [5] descriptors are quite robust to illumination and viewpoint changes due to the VGG-16 backbone and extensive fine-tuning on quality data, but they are penalized by imprecise keypoints. To solve this issue, we replace D2-Net keypoints with R2D2 [20], SuperPoint [4] or SIFT keypoints [15] to achieve better than (or close to) state-of-the-art performance.

evaluate our approaches. Finally, localization is evaluated on the visual localization benchmark of the RobotCar Seasons dataset [17, 23].

Similarly to [36] we use *Precision* and *Recall* for the evaluation of keypoint matching at varying pixel thresholds. Precision, also known as the Mean Matching Accuracy (MMA), is defined as $Precision@T = \frac{N_T}{N_m}$ where N_T is the number of correct matches found within T pixels and N_m the number of mutual nearest neighbor matches. Recall is defined as $Recall@T = \frac{N_T}{N}$, where N is the number of correct matches obtainable with the detected keypoints. When evaluating the localization, we use the pose accuracy [23] which corresponds to the percentage of queries correctly localized within a threshold of X meters and Y degrees from the ground-truth.

4.1. Keypoint Accuracy

As D2-Net model uses pooling operations to reduce the spatial resolution, we proposed using a bilinear interpolation of the feature map for increased precision. Different keypoint extractors can be used and we test with R2D2 [20], SuperPoint [4] and SIFT [15]. We also include a comparison with their proposed descriptors. Note that R2D2 and SuperPoint extract descriptors at the same resolution as the input image through upsampling in the network itself or by using dilated convolutions. However, we show that this is actually not needed and our simple interpolation achieves competitive results with D2-Net.

To evaluate the different keypoint/descriptor combinations, we perform an experiment on the image matching task of [5] with the *HPatches* dataset. The results are shown in Figure 3. To illustrate the advantage of using the interpolated D2-Net feature map, we can look at the SIFT/D2-Net combination: we see that D2-Net (blue line) achieves better matching precision at higher pixel thresholds, while SIFT (green line) is better at lower thresholds. Combining SIFT keypoints with D2-Net descriptors (green dashed) surpasses both methods individually. We also observe that R2D2 seems better in the case of illumination changes, while SuperPoint is better for handling viewpoint changes. This experiment also suggests that it is not necessary to learn full resolution feature maps for the descriptors, which means that the descriptor branch in both SuperPoint and R2D2 networks may be reduced to save computation time and memory. Based on these experiments, we use R2D2 for keypoints detection due to their superior performance in case of illumination and use interpolated D2-Net descriptors for the subsequent experiments.

4.2. Implementation Details and Model Analysis

The transformer network $\mathcal{T}$ is implemented as an autoencoder [11]. Instance normalization [33] and ReLU activation are used throughout the network (see supplementary material for more details). The transformer is randomly initialized and trained for 800 gradient descent steps for each considered day and night image pair. In the first 400 steps, the transformer is trained to accurately reconstruct the night image, which can be achieved by only minimizing the content loss. After, the network is trained for the remaining 400 steps with the combination of all losses, where $\lambda_c = 0.2$, $\lambda_s = 2e4$ and $\lambda_m = 1$. The two stage training procedure is necessary to first reconstruct the query image which then get transformed for improved feature matching. We use the Adam [12] optimizer with default parameters and a constant learning rate of $1\mathrm{e}{-3}$. For 800×800 input images, the optimization takes about 5 minutes on a modern GPU.

Ablation study. We evaluate the impact of optimizing the different loss functions of the transformer coupled with R2D2 [20] keypoints and D2-Net [5] descriptors. After the content image reconstruction phase ($\lambda_c = 1, \lambda_s = 0, \lambda_m = 0$), the transformer optimizes in addition to the content loss either the style loss only ($\lambda_c = 0.2, \lambda_s = 2e4, \lambda_m = 0$), the matching loss only ($\lambda_c = 0.2, \lambda_s = 0, \lambda_m = 1$) or both the

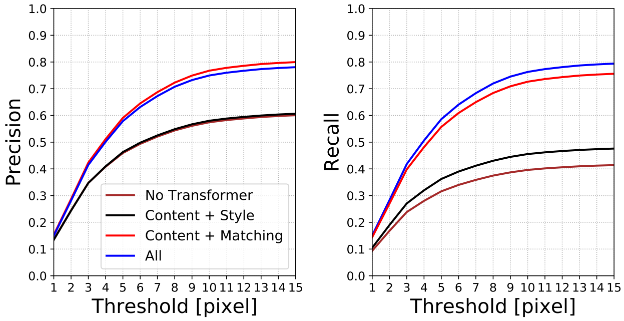


Figure 4: Ablation study on the second optimization phase of the transformer on the MDNIM dataset. Optimizing only the content and style loss (i.e. doing plain style transfer) slightly increases the recall, while optimizing with the matching loss brings a larger increase in overall performance. Optimizing all the losses, content, style and matching, is the preferred method if RANSAC is used to compensate for the lower precision.

style and matching losses ($\lambda_c = 0.2, \lambda_s = 2e4, \lambda_m = 1$). For completeness, we also provide results when no transformer network is used (i.e. plain D2-Net). Figure 4 shows the precision and recall of the different configurations averaged over all the 612 image pairs of the MDNIM dataset. Compared to plain D2-Net, we see that the transformer network optimized only for the style loss brings no benefits to the precision but slightly increases the recall by $5 - 10\%$ after a pixel threshold of 5. On the other hand, optimizing the matching loss proves to be the most beneficial approach, since it is responsible for the largest increase in both precision and recall in comparison to plain D2-Net. An interesting point is that the transformer optimizing both matching and style losses achieves slightly lower precision but higher recall than the transformer optimizing the matching loss only. Since matches are generally filtered with RANSAC, we conclude that optimizing both the matching and style losses is the best configuration since it achieves the highest recall while having high precision.

Optimization analysis. We have used a fixed number (800) of optimization steps for the appearance transfer. Our objective here is to analyze if computations can be reduced and how much effect this would have on the matching performances. Figure 5 shows the evolution of the $Precision\,@\,5$ and $Recall\,@\,5$ with respect to the number of optimization steps on the MDNIM dataset. The results show that the most important improvement comes from the first 50 to 100 steps. This suggest that the number of steps, and hence processing time, could be halved without loss in performance.

4.3. Image Matching

For each pair of day and night images, we compute local features and match them with a mutual nearest neighbor criterion. For our approach, we use R2D2/D2-Net as the keypoint/descriptor combination. The transformer is used to translate the night images before computing new features and matches. We compare our results with SIFT [15], SuperPoint [4], R2D2 [20] and the original D2-Net [5]. We also include a comparison with the ToDayGAN [2] model, where we rely on the same R2D2/D2-Net combination for matching after the translation of night images. To test its generalization capabilities, we choose the ToDayGAN model that was trained on the RobotCar dataset [17].

Figure 6 shows the precision and recall of the different methods averaged over the 612 image pairs of the MDNIM dataset. We see that the transformer improved both metrics by a large margin, surpassing plain D2-Net by $10 - 15\%$ in precision after a threshold of 5 pixels and by $20 - 40\%$ in recall after a threshold of 4 pixels. R2D2 descriptors have better precision at lower pixel thresholds (1 to 3) but have significantly worse recall at every pixel threshold. In practice, a method that exhibits high precision is not necessarily better if it is not able to produce a consequent number of matches in absolute value. In fact, matches are usually verified with RANSAC by fitting a fundamental matrix model, therefore it is not required to have a very high precision since bad matches will be filtered out. On the other hand, having a high recall implies that more matches will be found which can make the applications relying on keypoint matching more robust. The visual qualitative results in Figure 7 show how our transformer can lead to a significant larger amount of inliers in case of severe illumination changes compared to D2-Net [5] and R2D2 [20]. Note that our transformer achieves better results than ToDayGAN, which in fact severely decreases the performance of plain D2-Net, see Figure 6. The reason for this is that ToDayGAN was trained on a completely different dataset and therefore does not achieve good translations for new scenes. This shows the versatility of our transformer approach which does not need any training data and can easily adapt to new scenes.

4.4. Visual Localization

We evaluate the transformer on the visual localization benchmark of the RobotCar Seasons dataset [17, 23]. We use the same Structure-from-Motion pipeline as the one provided for the local features challenge of the Aachen Day-Night dataset [23]. The pipeline is based on COLMAP [24, 27] and reconstructs a sparse reference 3D model of the scene before estimating the query poses with a Perspective-n-Point algorithm, given our custom local features and matches. We compute poses for 185 selected query images from the ‘night-rain’ condition that we manually pair with the 5 spatially nearest reference images from the day ‘overcast’ condition. The 3D model for pose estimation is reconstructed by exhaustively matching the 5 references associated with each query. Additionally, we estimate the query poses by match-

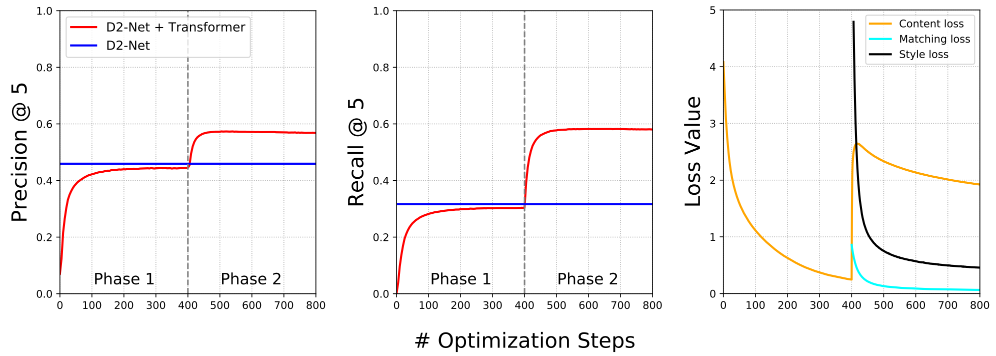


Figure 5: Evolution of the $Precision\,@\,5$, $Recall\,@\,5$ and the losses with respect to the number of optimization steps on the MDNIM dataset. The optimization of the transformer is performed in 2 phases. In the first phase (steps 0-400), the transformer reconstructs the night input image by optimizing only the content loss. In the second phase (steps 400-800), the transformer optimizes all 3 losses (content, style and matching), which significantly improves the precision and recall compared to the D2-Net baseline (with R2D2-keypoints).

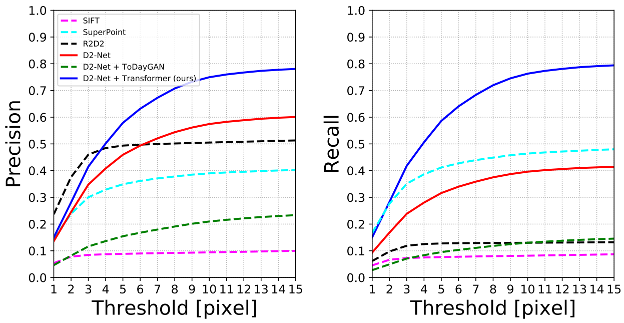


Figure 6: Precision and recall on the MDNIM dataset. The D2-Net descriptors [5] coupled with our transformer achieve the best results almost everywhere, being the preferred approach especially if RANSAC is used to compensate for lower precision. The original ToDayGAN [2] model trained on RobotCar [17] has been used in order to evaluate its generalization performance on new scenes.

ing each query to only **1** of the 5 references, but let the original reference-reference pairs unchanged to reconstruct the same 3D scene as before. The chosen reference image is the spatially furthest one from the query. This additional experiment can be motivated by applications where only a few query-reference matches can be performed, which is the case when there is a lack of reference images or if we want to decrease the computation time of the pose solver.

Table 1 shows the resulting query pose accuracy for both settings. We consider the same local feature configurations as in Section 4.3. Our transformer model is used to translate night images before computing D2-Net descriptors and matches. When using a single reference per query, the advantages of using this transformation are significant. The benefits are however less important when using multiple reference frames. One reason for this is the increased robustness inherent to using multiple cameras.

The results with ToDayGAN show the importance of using the matching loss for the translation of night images. Although it was trained on this particular dataset, and achieves visually plausible results (see Figure 1), it performs poorly as a way of improving localization performance. It even decreases the performance of plain D2-Net.

5. Discussion

It is interesting to understand how the additional optimization of Equation 8 improves on the matching of local features. In fact, the optimization of the matching loss is self-supervised and therefore, at first sight, merely tries to achieve the performance that was already attainable without the transformation. However, the key reason why this objective can still improve the matching performance lies in the fact that we compute the translated image I_n^* with a convolutional image transformation network. As a result, the convolution filters of the transformer $\mathcal{T}$ which are optimized to achieve a low matching loss will be applied to the entire image and can improve the performance of local features in parts of the images that were not covered by the initial set of matches $\mathcal{K}$ used for self-supervision.

Relying on this initial set of *good* matches in order to improve the overall number of matched keypoints has some limitations: the query-reference image pair can be so difficult to match that there are no good matches in the initial set and the transformer will not bring any increase in performance. It is worth mentioning though that our experience suggests that when this happens, it is mostly due to a very large gap in the geometry i.e. a too large scale gap in the common region between the two images. This suggests that further improving matching requires to jointly consider appearance and spatial transformation of the images.

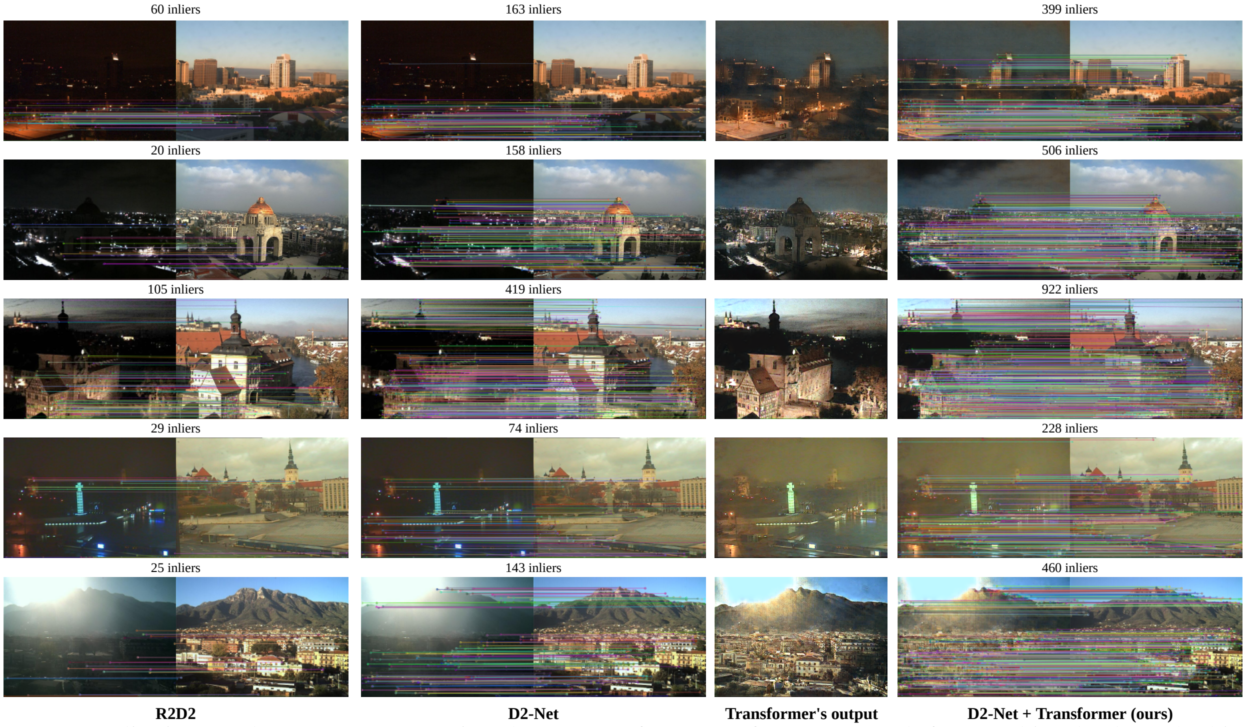


Figure 7: Qualitative results on MDNIM. Using the output of our transformer network for matching leads to the highest number of inliers compared to using the original query image with R2D2 [20] and D2-Net [5].

Method	Pose Accuracy	
	1 reference per query	5 references per query
SIFT [15]	6.4 / 12.4 / 13.6	9.8 / 20.0 / 22.1
SuperPoint [4]	14.5 / 26.4 / 31.9	32.3 / 54.7 / 67.5
R2D2 [20]	8.1 / 16.2 / 16.6	25.4 / 40.0 / 50.9
D2-Net [5]	46.5 / 71.4 / 90.8	50.8 / **81.6 / 100**
D2-Net + ToDayGAN [2]	39.5 / 70.2 / 91.8	46.4 / 77.8 / 99.4
D2-Net + Transformer (ours)	**50.3 / 76.8 / 98.9**	**51.9 / 81.6 / 100**

Table 1: Pose accuracies for our 'night-rain' queries from RobotCar [17, 23]. The localization thresholds for the poses are $(0.25m, 2°)/(0.5m, 5°)/(5m, 10°)$. We see that the transformer significantly improves the pose accuracies when the number of references per query is only 1. All methods based on D2-Net [5] use R2D2-keypoints [20] and therefore do not vary in accuracy due to differences in keypoint precision but soley due to the matching ability of the descriptors.

6. Conclusion and Outlook

In this work, we have presented a novel approach to translate images exhibiting adverse conditions for improving the matching performance of local features. Our transformer approach is self-supervised and can handle any illumination condition without relying on additional training data, which is a significant advantage over appearance transfer models based on Generative Adversarial Networks. We have shown that our model outperforms the state-of-the-art on a day and night image matching task, and achieves a meaningful improvement on visual localization when the number of reference images per query is small.

Nevertheless, the computation time for each image pair remains too slow for practical applications. A possible solution would be to use meta-networks in order to perform real-time neural style transfer [28]. In our case, one would have to incorporate the keypoint matching information into the meta network in order to predict transformers that are also good for matching. On another note, our transformer currently deals with illumination changes only. Spatial transformer networks [10] could further be used to augment our model for closing the viewpoint gap as well.

Acknowledgments. This work was supported by Ringier, TX Group, NZZ, SRG, VSM, and the ETH Foundation.

References

[1] A. Anoosheh, E. Agustsson, R. Timofte, and L. V. Gool. Combogan: Unrestrained scalability for image domain translation. In *CVPR Workshops*, 2018. 2

[2] A. Anoosheh, T. Sattler, R. Timofte, M. Pollefeys, and L. V. Gool. Night-to-day image translation for retrieval-based localization. In *ICRA*, 2019. 1, 2, 6, 7, 8

[3] V. Balntas, K. Lenc, A. Vedaldi, and K. Mikolajczyk. Hpatches: A benchmark and evaluation of handcrafted and learned local descriptors. In *CVPR*, 2017. 4, 5

[4] D. DeTone, T. Malisiewicz, and A. Rabinovich. Superpoint: Self-supervised interest point detection and description. In *CVPR Workshops*, 2018. 2, 5, 6, 8

[5] M. Dusmanu, I. Rocco, T. Pajdla, M. Pollefeys, J. Sivic, A. Torii, and T. Sattler. D2-net: A trainable CNN for joint description and detection of local features. In *CVPR*, 2019. 1, 2, 3, 4, 5, 6, 7, 8

[6] Y. Furukawa and J. Ponce. Accurate, dense, and robust multiview stereopsis. *PAMI*, 2010. 1

[7] L. A. Gatys, A. S. Ecker, and M. Bethge. Image style transfer using convolutional neural networks. In *CVPR*, 2016. 2, 3

[8] H. Germain, G. Bourmaud, and V. Lepetit. Sparse-to-dense hypercolumn matching for long-term visual localization. In *3DV*, 2019. 2

[9] P. Isola, J. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017. 1, 2

[10] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu. Spatial transformer networks. In *NeurIPS*, 2015. 8

[11] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, 2016. 3, 5

[12] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 5

[13] Y. Li, M. Liu, X. Li, M. Yang, and J. Kautz. A closed-form solution to photorealistic image stylization. In *ECCV*, 2018. 2

[14] M. Liu, T. Breuel, and J. Kautz. Unsupervised image-to-image translation networks. In *NeurIPS*, 2017. 2

[15] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004. 1, 2, 5, 6, 8

[16] S. Lynen, T. Sattler, M. Bosse, J. A. Hesch, M. Pollefeys, and R. Siegwart. Get out of my lab: Large-scale, real-time visual-inertial localization. In *Robotics: Science and Systems*, 2015. 1

[17] W. Maddern, G. Pascoe, C. Linegar, and P. Newman. 1 year, 1000 km: The oxford robotcar dataset. *IJRR*, 2017. 1, 5, 6, 7, 8

[18] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *PAMI*, 2005. 2

[19] H. Porav, W. Maddern, and P. Newman. Adversarial training for adverse conditions: Robust metric localisation using appearance transfer. In *ICRA*, 2018. 2

[20] J. Revaud, C. R. de Souza, M. Humenberger, and P. Weinzaepfel. R2D2: reliable and repeatable detector and descriptor. In *NeurIPS*, 2019. 1, 2, 4, 5, 6, 8

[21] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. S. Bernstein, A. C. Berg, and F. Li. Imagenet large scale visual recognition challenge. *IJCV*, 2015. 3

[22] P. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich. Superglue: Learning feature matching with graph neural networks. In *CVPR*, 2020. 2

[23] T. Sattler, W. Maddern, C. Toft, A. Torii, L. Hammarstrand, E. Stenborg, D. Safari, M. Okutomi, M. Pollefeys, J. Sivic, F. Kahl, and T. Pajdla. Benchmarking 6dof outdoor visual localization in changing conditions. In *CVPR*, 2018. 5, 6, 8

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

[25] J. L. Schönberger, H. Hardmeier, T. Sattler, and M. Pollefeys. Comparative evaluation of hand-crafted and learned local features. In *CVPR*, 2017. 2

[26] J. L. Schönberger, M. Pollefeys, A. Geiger, and T. Sattler. Semantic visual localization. In *CVPR*, 2018. 2

[27] J. L. Schönberger, E. Zheng, M. Pollefeys, and J.-M. Frahm. Pixelwise view selection for unstructured multi-view stereo. In *ECCV*, 2016. 6

[28] F. Shen, S. Yan, and G. Zeng. Neural style transfer via meta networks. In *CVPR*, 2018. 8

[29] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 3

[30] S. N. Sinha, D. Steedly, and R. Szeliski. Piecewise planar stereo for image-based rendering. In *ICCV*, 2009. 1

[31] E. Stenborg, C. Toft, and L. Hammarstrand. Long-term visual localization using semantically segmented images. In *ICRA*, 2018. 2

[32] C. Toft, E. Stenborg, L. Hammarstrand, L. Brynte, M. Pollefeys, T. Sattler, and F. Kahl. Semantic match consistency for long-term visual localization. In *ECCV*, 2018. 2

[33] D. Ulyanov, A. Vedaldi, and V. S. Lempitsky. Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. In *CVPR*, 2017. 5

[34] K. M. Yi, E. Trulls, V. Lepetit, and P. Fua. LIFT: learned invariant feature transform. In *ECCV*, 2016. 1

[35] J. Yoo, Y. Uh, S. Chun, B. Kang, and J. Ha. Photorealistic style transfer via wavelet transforms. In *ICCV*, 2019. 2

[36] H. Zhou, T. Sattler, and D. W. Jacobs. Evaluating local features for day-night matching. In *ECCV Workshops*, 2016. 1, 2, 4, 5

[37] J. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017. 2