DKM: Dense Kernelized Feature Matching for Geometry Estimation

Johan Edstedt, Ioannis Athanasiadis, Mårten Wadenbäck, Michael Felsberg

Computer Vision Laboratory
Linköping University

Abstract

Feature matching is a challenging computer vision task that involves finding correspondences between two images of a 3D scene. In this paper we consider the dense approach instead of the more common sparse paradigm, thus striving to find all correspondences. Perhaps counter-intuitively, dense methods have previously shown inferior performance to their sparse and semi-sparse counterparts for estimation of two-view geometry. This changes with our novel dense method, which outperforms both dense and sparse methods on geometry estimation. The novelty is threefold: First, we propose a kernel regression global matcher. Secondly, we propose warp refinement through stacked feature maps and depthwise convolution kernels. Thirdly, we propose learning dense confidence through consistent depth and a balanced sampling approach for dense confidence maps.

*Through extensive experiments we confirm that our proposed dense method, **D**ense **K**ernelized Feature **M**atching, sets a new state-of-the-art on multiple geometry estimation benchmarks. In particular, we achieve an improvement on MegaDepth-1500 of +4.9 and +8.9 AUC@5° compared to the best previous sparse method and dense method respectively. Our code is provided at the following repository:* `https://github.com/Parskatt/DKM`.

1. Introduction

Two-view geometry estimation is a classical computer vision problem with numerous important applications, including 3D reconstruction [38], SLAM [30], and visual relocalisation [27]. The task can roughly be divided into two steps. First, a set of matching pixel pairs between the images is produced. Then, using the matched pairs, two-view geometry, *e.g.*, relative pose, is estimated. In this paper, we focus on the first step, *i.e.*, feature matching. This task is challenging, as image pairs may exhibit extreme variations in illumination [1], viewpoint [22], time of day [37], and even season [46]. This stands in contrast to small baseline stereo and optical flow tasks, where the changes in viewpoint and illumination are typically small.

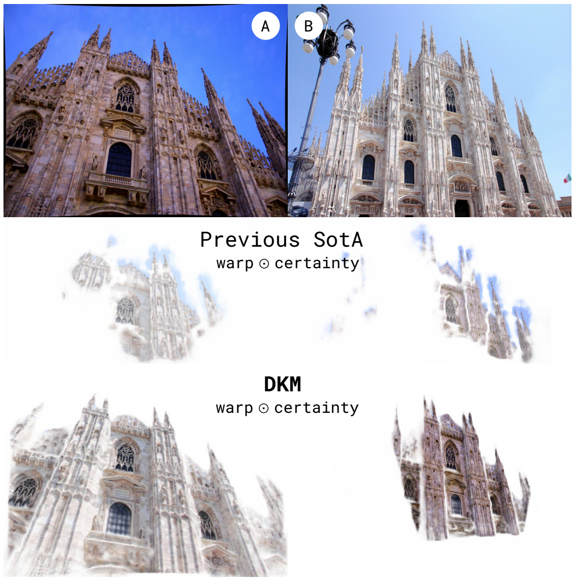

Figure 1. **Qualitative comparison.** We compare our proposed approach **DKM** with the previous SotA method PDC-Net+ [48] on Milan Cathedral. Top row, image $\mathcal{A}$ and $\mathcal{B}$. Middle row and bottom row, forward and reverse warps for PDC-Net+ and DKM weighted by certainty. DKM provides both superior match accuracy and certainty estimation compared to previous methods.

Traditionally, feature matching has been performed by sparse keypoint and descriptor extraction, followed by matching [26, 36]. The main issue with this approach is that accurate localization of reliable and repeatable keypoints is difficult in challenging scenes. This leads to errors in matching and estimation [13, 23]. To tackle this issue, semi-sparse or *detector-free* methods such as LoFTR [41] and Patch2Pix [53] were introduced. These methods do not detect keypoints directly but rather perform global matching at a coarse level, followed by mutual nearest neighbour extraction and sparse match refinement. While those methods degrade less in low-texture scenes, they are still limited by the fact that the sparse matches are produced at a coarse scale, leading to problems with, *e.g.*, repeatability due to grid artifacts [17]. By instead extracting *all* matches between the views, *i.e.*, *dense* feature matching, we face no such issues. Furthermore, dense warps provide affine matches for free, which yield smaller minimal problems for subsequent estimation [3, 4, 15]. While previous dense approaches [39, 47] have achieved good results, they have however failed to achieve performance rivaling that of sparse or semi-sparse methods on geometry estimation.

In this work, we propose a novel dense matching method that outperforms both dense and sparse methods in homography and two-view relative pose estimation. We achieve this by proposing a substantially improved model architecture, including both the global matching and warp refinement stage, and by a simple but strong approach to dense certainty estimation and a balanced dense warp sampling mechanism. We compare qualitatively our method with the previous best dense method in Figure 1.

Our **contributions** are as follows. **Global Matcher:** We propose a kernelized global matcher and embedding decoder. This results in robust coarse matches. We describe our approach in Section 3.2 and ablate the performance gains in Table 5. **Warp Refiners:** We propose warp refinement through large depthwise separable kernels using stacked feature maps as well as local correlation as input. This gives our method superior precision and is described in detail in Section 3.3 with corresponding performance impact ablated in Table 6. **Certainty and Sampling:** We propose a simple method to predict dense certainty from consistent depth and propose a balanced sampling approach for dense matches. We describe our certainty and sampling approach in more detail in Section 3.4 and ablate the performance gains in Table 7. **State-of-the-Art:** Our extensive experiments in Section 4 show that our method significantly improves on the state-of-the-art. In particular, we improve estimation results compared to the best previous dense method by +8.9 AUC@5° on MegaDepth-1500. These results pave the way for dense matching based 3D reconstruction.

2. Related Work

Global Matching: Traditionally, global matching has been performed by computing pair-wise descriptor distances for detected keypoints in the two images, with match extraction performed by mutual nearest neighbours in the distance matrix, see *e.g.* [10, 11, 26]. Instead of directly computing pair-wise distances, one can first condition the descriptors based on the complete set of detections. Sarlin *et al.* [36] proposed a graph neural network approach to condition the descriptors, and optimal transport instead of mutual nearest neighbours for match extraction. Detector-free methods instead perform global matching uniformly over the image grid at a coarse scale [32, 33, 45, 53]. This has the benefit of avoiding the detection problem [41]. These methods typically extract matches by (soft-)mutual-nearest neighbours, or optimal transport [32, 41]. In contrast to detector-free methods, dense methods must produce a dense warp. This warp is typically predicted by regression based on the global 4D-correlation volume [29,47,49]. In this work we propose a Gaussian Process (GP) formulation of the matching problem, as detailed in Section 3.2.

Match Refinement: For detector-free methods, match refinement is typically performed by extracting patches around the sparse matches. Zhou *et al.* [53] propose to refine matches by CNN regression. Sun *et al.* [41] use transformers, with additional improvements by later work [7, 44, 50]. Dense methods in contrast refine matches by dense warp refinement. Troung *et al.* [47, 49] proposed a local-correlation based warp refinement network. In this work, we propose to use stacked feature maps combined with large depth-wise convolution kernels. Our approach to refinement is described in Section 3.3.

Match Certainty and Sampling: Although the dense paradigm provides subpixel-level feature matching capabilities, it also comes with inaccurate correspondences in unmatchable regions, resulting in a need for certainty estimation. Wiles *et al.* [51] and Melekhov *et al.* [29] proposed matchability branches aiming at predicting the presence or the absence of a pixel correspondence. Recently, in PDC-Net [49] and PDC-Net+ [48], the warp estimation was formulated in a probabilistic manner, thus pairing the proposed feature correspondences along with certainty estimates by means of mixture models. We found, however, that their estimated certainty is often confident for unmatchable pairs (Figure 7). In this work, we propose to model certainty as the likelihood of a pixel having a consistent pairwise match in terms of 3D reconstruction, which provides potent certainty maps as illustrated in Figure 1. However, in downstream tasks, *e.g.*, relative pose, the reliability of the extracted correspondence is not the sole factor influencing the performance. For estimation, planar warps are a well known degenerate case [8], and the five-point problem is often ill-conditioned [6, 12]. Hence, well distributed matches are important for estimation [2, 18]. Motivated by this, we propose a balanced sampling mechanism that provides the estimator with diverse matches. We describe the certainty estimation and balanced sampling in more detail in Section 3.4.

3. Method

In the following sections we describe our approach to geometry estimation by dense matching. For an overview, see Figure 2. We first provide a general overview of the dense matching framework (Section 3.1). We then describe our approach for improving the global matcher G_θ (Section 3.2), the warp refiners R_θ (Section 3.3), and certainty estimation along with match sampling (Section 3.4). Lastly, we discuss our loss formulation (Section 3.5).

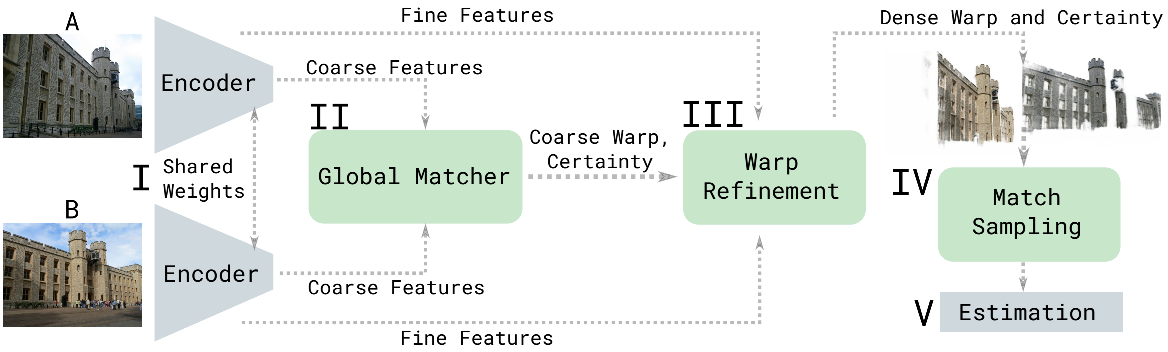

Figure 2. **An overview of geometry estimation by dense matching. I:** In the first stage, a multistride feature pyramid is extracted. We follow previous approaches and use ResNet encoders with shared weights. **II:** In the second stage coarse global matches are established. We improve this stage by viewing it as a embedded probabilistic regression problem combined with a strong embedding decoder. We describe our approach in more detail in Section 3.2. **III:** The coarse warp is then refined. We propose a stacked feature map approach combined with large depthwise kernels, which increases performance. This is detailed in Section 3.3. **IV:** Finally, for geometry estimation a robust certainty estimate is crucial for selecting a set of reliable matches. We find that letting the network learn to classify consistent depth yields a trustworthy certainty estimate. Further combining this with balanced sampling yields even better results. We discuss this in Section 3.4. **V:** Once a set of matches have been selected, we use standard robust solvers for estimation as previous methods.

3.1. Preliminaries

In this paper we consider the task of estimating 3D scene geometry from two images $(I^{\mathcal{A}}, I^{\mathcal{B}})$. For matching we choose the dense feature matching paradigm, *i.e.*, to estimate a dense warp $\mathrm{W}^{\mathcal{A}\to\mathcal{B}}$ and a dense certainty $p^{\mathcal{A}\to\mathcal{B}}$, that is zero for unmatchable pixels. From this complete set of certain and uncertain matches, a subset of matches are sampled (without replacement). Finally, a robust estimation method is used to infer the geometry from the sampled matches. The task can be divided into five stages.

In stage **I**, a feature pyramid is extracted for $\mathcal{A}$ and $\mathcal{B}$,

$$\{\varphi_l^{\mathcal{A}}\}_{l=1}^{L} = F_\theta(I^{\mathcal{A}}) \ , \ \{\varphi_l^{\mathcal{B}}\}_{l=1}^{L} = F_\theta(I^{\mathcal{B}}) \ , \qquad (1)$$

where F_θ is an encoder (we use a ResNet50 [16] pretrained on ImageNet-1K [34]), and $l \in \{1, \ldots, L\}$ are the indices for the multiscale features (in our approach $l = 1$ corresponds to the rgb values of stride 1, and $l = L$ corresponds to deep features of stride $2^{L-1} = 32$). We denote the coarse features as $(\varphi^{\mathcal{A}}_{\text{coarse}}, \varphi^{\mathcal{B}}_{\text{coarse}})$ and fine features as $(\varphi^{\mathcal{A}}_{\text{fine}}, \varphi^{\mathcal{B}}_{\text{fine}})$. In this work the coarse features correspond to stride $\{32, 16\}$ and the fine features to $\{8, 4, 2, 1\}$.

In stage **II**, we estimate a coarse global warp and certainty from the deep features with a global matcher G_θ. Here potential global matches are embedded by the embedder E_θ. We propose to construct the embeddings as solutions to a probabilistic regression problem using a Gaussian Process (GP) formulation. After the embeddings have been computed, an embedding decoder D_θ decodes the embeddings into a dense warp and certainty, *i.e.*,

$$\begin{cases} (\hat{\mathrm{W}}^{\mathcal{A}\to\mathcal{B}}_{\text{coarse}}, \hat{p}^{\mathcal{A}\to\mathcal{B}}_{\text{coarse}}) = G_\theta(\varphi^{\mathcal{A}}_{\text{coarse}}, \varphi^{\mathcal{B}}_{\text{coarse}}), \\ G_\theta(\varphi^{\mathcal{A}}_{\text{coarse}}, \varphi^{\mathcal{B}}_{\text{coarse}}) = D_\theta\big(E_\theta(\varphi^{\mathcal{A}}_{\text{coarse}}, \varphi^{\mathcal{B}}_{\text{coarse}})\big). \end{cases} \qquad (2)$$

We describe the global matching in detail in Section 3.2.

In stage **III**, we refine the coarse warp of G_θ, *i.e.*,

$$\big(\hat{\mathrm{W}}^{\mathcal{A}\to\mathcal{B}}, \hat{p}^{\mathcal{A}\to\mathcal{B}}\big) = R_\theta\big(\varphi^{\mathcal{A}}_{\text{fine}}, \varphi^{\mathcal{B}}_{\text{fine}}, \hat{\mathrm{W}}^{\mathcal{A}\to\mathcal{B}}_{\text{coarse}}, \hat{p}^{\mathcal{A}\to\mathcal{B}}_{\text{coarse}}\big) \ , \qquad (3)$$

where $\hat{\mathrm{W}}$ is the predicted warp, $\hat{p}$ is the predicted certainty, and R_θ is a set of refiners. This is typically done by local correlation volume refinement. In this work we additionally stack the warped feature maps of $\mathcal{B}$, and use large depthwise kernels. We describe this in detail in Section 3.3.

In stage **IV**, reliable and accurate matches need to be selected for estimation of scene geometry. For sparse methods this is done at the coarse level by mutual nearest neighbour matching and certainty thresholding. For dense matching, we are free to choose any method, which is an advantage. In this work we sample the estimated warp using a balanced sampling approach. We describe this in Section 3.4.

Finally, in stage **V**, a robust estimator is used to estimate geometry. We use RANSAC like previous work.

3.2. Constructing the Global Matcher G_θ

For an overview of our global matcher, see Figure 3.

Global Matching as Regression: In this work we construct the global match embeddings as the solution to a (embedded) coordinate regression problem. We phrase this problem as finding a mapping $\varphi \to \chi$ where χ are (embeddings of) spatial coordinates in image $\mathcal{B}$. We can choose any suitable regression framework to infer the mapping for the pixels in $\mathcal{A}$. In this work we consider GP regression. As a general framework for non-parametric regression, it is a natural choice for our formulation of feature matching.

In GP regression, the output (embedded coordinates) $\chi \in \mathbb{R}^{H \cdot W \times C}$, where H, W is the height and width, and C

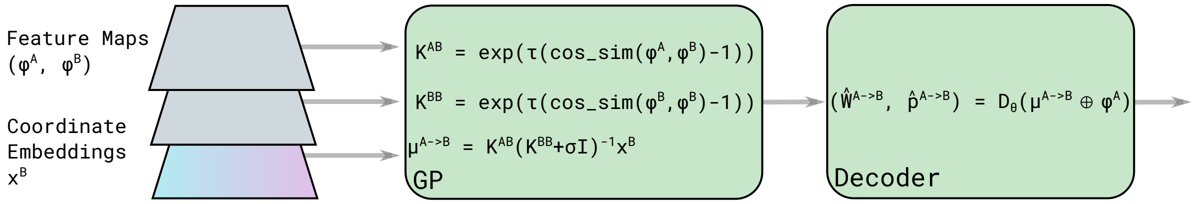
Feature Maps
(φ^A, φ^B)
Coordinate
Embeddings
x^B
K^AB = exp(τ(cos_sim(φ^A,φ^B)-1))
K^BB = exp(τ(cos_sim(φ^B,φ^B)-1))
μ^A->B = K^AB(K^BB+σI)^-1x^B
GP
(Ŵ^A->B, p̂^A->B) = D_θ(μ^A->B ⊕ φ^A)
Decoder

Figure 3. **Illustration of the proposed Global Matcher.** The Gaussian Process (GP) , using an exponential cosine similarity (cos_sim) kernel, and, given features and coordinate embeddings, produces an embedded predicive posterior for the warp. The CNN embedding decoder D_θ decodes the GP output to find the most likely warp and certainty over the grid in image $\mathcal{A}$. For more details, see Section 3.2.

is dimensionality of the coordinate embedding, is regarded as a collection of random variables, with the main assumption being that these are jointly Gaussian. A GP is uniquely[1] defined by its kernel that defines the covariance between outputs, and hence must be a positive-definite function to be admissible. We choose the common assumption [54] that the coordinate embedding dimensions are uncorrelated, which makes the kernel block diagonal. We choose the exponential cosine similarity kernel [24], which is defined by

$$k(\varphi, \varphi') = \exp\left(\tau\left(\frac{\langle \varphi, \varphi' \rangle}{\sqrt{\langle \varphi, \varphi \rangle \langle \varphi', \varphi' \rangle + \varepsilon}} - 1\right)\right), \quad (4)$$

since we empirically found it to work well. We found the squared exponential kernel to perform similarly in early experiments, and other kernels could also be considered. We initialize $\tau = 5$ and keep it fixed and set $\varepsilon = 10^{-6}$.

With the standard assumption [31] that the measurements $(\varphi^{\mathcal{B}}_{\text{coarse}}, \chi^{\mathcal{B}}_{\text{coarse}})$ are observed with i.i.d. noise, the analytic formulae for the posterior conditioned on the features of $\mathcal{B}$ are given by

$$\begin{cases} \mu^{\mathcal{A}\to\mathcal{B}} = K^{\mathcal{AB}}(K^{\mathcal{BB}} + \sigma_n^2 I)^{-1}\chi^{\mathcal{B}}_{\text{coarse}}, \\ \Sigma^{\mathcal{A}\to\mathcal{B}} = K^{\mathcal{AA}} - K^{\mathcal{AB}}(K^{\mathcal{BB}} + \sigma_n^2 I)^{-1}K^{\mathcal{BA}}, \end{cases} \quad (5)$$

where $K^{\mathcal{AA}}, K^{\mathcal{AB}}, K^{\mathcal{BA}}, K^{\mathcal{BB}}$ denotes the kernel matrices, $\mu^{\mathcal{A}\to\mathcal{B}}$ is the posterior mean, $\sigma_n = 0.1$ is the standard deviation of the measurement noise, and $\Sigma^{\mathcal{A}\to\mathcal{B}}$ is the posterior covariance. We refer to Rasmussen [31] for details on GP regression.

Coordinate Embeddings: One challenge with coordinate regression is how to deal with multimodality. GP posteriors are unimodal in the output space, and hence multimodal matches can degrade performance. To deal with this issue we use a cosine embedding

$$B_{\mathcal{F}}(x; A, b) = \cos(Ax + b), \quad (6)$$

where $x \in \mathbb{R}^2$ is the image coordinate, $A_{ij} \sim \mathcal{N}(0, \ell^2)$, $b_i \sim \mathcal{U}_{[0,2\pi]}$, $i \in \{1, \dots, C\}$, $j \in \{1, 2\}$. These types of embeddings preserve multimodality [40]. We illustrate their usefulness in Figure 4.

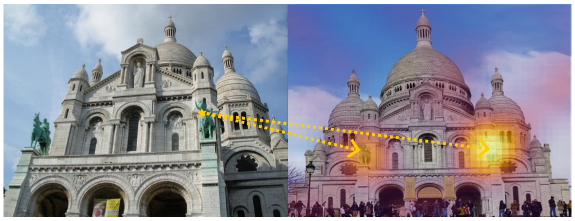

Figure 4. **Coordinate embeddings preserve multimodality.** Real scenes often contain repeating structures, which requires regression capable of handling multimodality. We achieve this through cosine coordinate embeddings. We illustrate the multimodality by correlating the GP posterior with embeddings on the image grid.

Embedding Decoder: While the embedded regression yields a powerful probabilistic representation of the warp, most dense methods require a unimodal warp estimate for the subsequent refinement steps. There are multiple ways of decoding coordinates from the posterior. We use a simple method of reshaping the predictive mean back into grid form $\mu^{\mathcal{A}\to\mathcal{B}}_{\text{grid}} \in \mathbb{R}^{H_{\text{coarse}} \times W_{\text{coarse}} \times C}$ and let

$$G_\theta(\varphi^{\mathcal{A}}_{\text{coarse}}, \varphi^{\mathcal{B}}_{\text{coarse}}) = D_\theta(\mu^{\mathcal{A}\to\mathcal{B}}_{\text{grid}} \oplus \varphi^{\mathcal{A}}_{\text{coarse}}), \quad (7)$$

where D_θ is a CNN embedding decoder. The decoder predicts coordinates in the canonical grid $[-1, 1] \times [-1, 1]$, and additionally logits for the predicted validity of the matches, for each pixel. The architecture of the embedding decoder is inspired by the decoder proposed by Yu *et al.* [52]. We use global matchers on both stride 32 and 16 features of the backbone, and the stride 16 embedding decoder takes in context feature maps from the stride 32 decoder.

3.3. Refining the Warp with R_θ

For an overview of our warp refiners, see Figure 5.

Warp Refinement: Once the embeddings have been decoded, we refine the warp using CNN refiners similarly to previous work [39, 47]. They take as input the feature maps and the previous warp and certainty. The warp and certainty are bilinearly upsampled to match the size of the feature maps. The refiners predict a residual offset for the estimated warp, and a logit offset for the certainty. This is repeated

[1] With the common assumption that the mean function is 0.

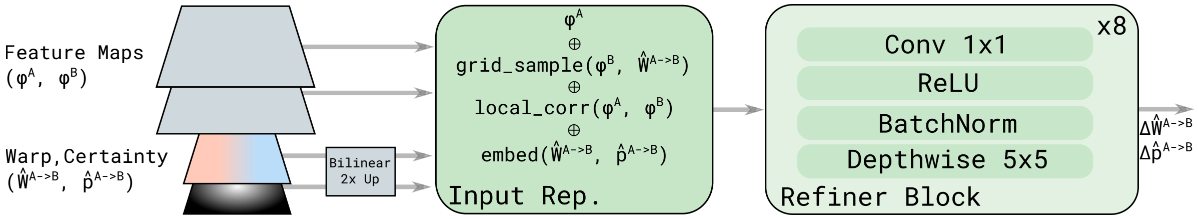

Figure 5. **Illustration of the proposed Warp Refiners.** The Warp Refiners R_θ take in fine features ($\varphi^{\mathcal{A}}_{\text{fine}}, \varphi^{\mathcal{B}}_{\text{fine}}$), and the upsampled coarse warps and certainty estimates . They output a relative offset for the warp and certainty. We use `grid_sample` on $\varphi^{\mathcal{B}}$ to create the stacked feature maps, and `local_corr` to construct a local correlation volume around the warp target in image $\mathcal{B}$. Furthermore, we `embed` the warp (represented as displacement) and certainty linearly. The concatenation constitutes our input representation and is fed into the refiner blocks. For more details, see Section 3.3.

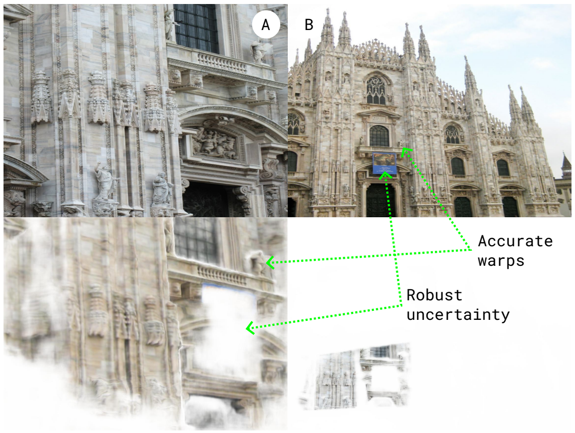

Figure 6. **DKM warps are accurate and robust.** Dense methods often struggle with large viewpoint changes. Our proposed global matcher + refiner architecture is able to produce accurate warps and certainty even for extreme perspective. Top row, image $\mathcal{A}$ and $\mathcal{B}$. Bottom row, forward and reverse warp weighted by certainty.

until reaching full resolution. The process is described recursively by

$$(\hat{W}_l^{\mathcal{A}\to\mathcal{B}}, \ \hat{p}_l^{\mathcal{A}\to\mathcal{B}}) = R_{\theta,l}(\varphi_l^{\mathcal{A}}, \varphi_l^{\mathcal{B}}, \hat{W}_{l+1}^{\mathcal{A}\to\mathcal{B}}, \hat{p}_{l+1}^{\mathcal{A}\to\mathcal{B}}). \quad (8)$$

Input Representation: We make multiple improvements to the input representations of the refiners. Previous work [47–49] uses the warp, the feature maps of $\mathcal{A}$, and local correlation in $\mathcal{A}$ with warped feature maps from $\mathcal{B}$, together with the warp. In contrast, we use all channels of the warped feature maps of $\mathcal{B}$ by concatenation, as well as local correlation in $\mathcal{B}$ instead of $\mathcal{A}$. We investigate the effect of this change of representation in Table 6 and find that it yields improvements in warp accuracy.

Refiner Architecture: Finally, we improve the architecture of the refiner blocks themselves. Previous work [47–49] uses a DenseNet [19] architecture with 3x3 non-separable kernels. We instead propose to use larger 5x5 depthwise separable kernels, followed by 1x1 convolution. We found 8 refiner blocks per scale to give the best results. As we show in Table 6, this improvement significantly increases performance. We qualitatively show the high robustness and accuracy of DKM warps in Figure 6.

3.4. Certainty Estimation and Sampling for Geometry Estimation

Certainty Estimation by Classifying Depth-consistent Matches: We leverage the rich 3D models and densified depth maps in the large scale MegaDepth [22] dataset. We find consistent matches first by warping $\mathcal{A} \to \mathcal{B}$ using the ground truth depth, and then applying a relative depth consistency constraint in image $\mathcal{B}$. This equates to

$$p^{\mathcal{A}\to\mathcal{B}} = |z^{\mathcal{A}\to\mathcal{B}} - z^{\mathcal{B}}| \cdot |z^{\mathcal{B}}|^{-1} < \alpha \quad (9)$$

where z is the depth, $z^{\mathcal{A}\to\mathcal{B}}$ depth projected using the ground truth 3D model, and $\alpha = 0.05$. This approach has similarities to the approach in LoFTR [41], but they indirectly apply the constraint by finding mutual nearest neighbours. We demonstrate the importance of a good certainty estimate in Table 7, and show a qualitative comparison of our certainty estimate compared to the previous best perfoming dense work PDC-Net+ [48] in Figure 7.

Sampling Balanced Matches: For estimation, match sampling is required. A simple approach is to sample using the estimated warp certainty as weight. This approach is written as,

$$\{x_i^{\mathcal{A}}, x_i^{\mathcal{B}}\}_{i=1}^N \sim \hat{p}^{\mathcal{A}\to\mathcal{B}}. \quad (10)$$

Like previous semi-sparse [7, 41] and dense works [48] we threshold the estimated certainty. We use a threshold of 0.05, and sample matches from the thresholded distribution.

While certainty weighted sampling produces good matches, having diverse matches typically improves estimation [6, 8, 12, 18]. To achieve this, we propose a simple method for producing scene balanced matches. First, we sample a large set of matches using the estimated certainty. Secondly, we compute a kernel density estimate (KDE) in the 4-dimensional match space. Finally, we weight each match with the reciprocal of the KDE to produce a balanced set of samples. This produces a balanced distribution in the

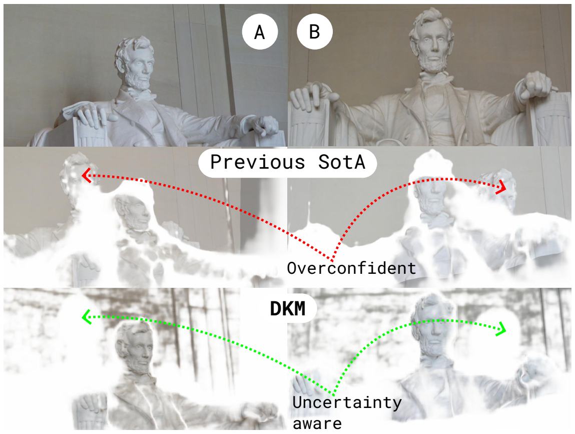

Figure 7. **DKM provides superior uncertainty estimates.** Our certainty estimate compared to PDC-Net+. Top row, image $\mathcal{A}$, image $\mathcal{B}$. Middle row, results for PDC-Net+. Bottom row, results for DKM. DKM places high certainty on repeatable matches, while PDC-Net+ is often overconfident in untextured regions, even predicting high certainty for non-covisible pixel-pairs.

scene. We investigate the impact of the balanced sampling in Table 7, and find that it improves performance.

3.5. Loss Formulation

Like previous work [36, 39, 49] we use separate losses for each stride $l \in \{1, ..., L\}$, and use a combination of regression and certainty [29, 43, 53] losses to train our model. The combined loss is

$$\mathcal{L} = \sum_{l=1}^{L} \mathcal{L}_{\text{warp}}(\hat{W}_l^{\mathcal{A}\to\mathcal{B}}) + \lambda \mathcal{L}_{\text{conf}}(\hat{p}_l^{\mathcal{A}\to\mathcal{B}}), \quad (11)$$

where $\lambda = 0.01$ is a balancing term, similarly to [29, 43].

Specifically, for the warp loss we use the ℓ_2 distance between the predicted and ground truth warp, as in [41]. For the certainty loss we use the unweighted binary cross entropy between the predicted certainty and the ground truth consistent depth mask. Our losses at a given stride l are

$$\mathcal{L}_{\text{warp}}(\hat{W}_l^{\mathcal{A}\to\mathcal{B}}) = \sum_{\text{grid}} p_l \odot \left\| W_l^{\mathcal{A}\to\mathcal{B}} - \hat{W}_l^{\mathcal{A}\to\mathcal{B}} \right\|_2, \quad (12)$$

$$\mathcal{L}_{\text{conf}}(\hat{p}_l) = \sum_{\text{grid}} p_l \log \hat{p}_l + (1 - p_l) \log (1 - \hat{p}_l), \quad (13)$$

where the summation is done over the image grid in $\mathcal{A}$. Like Zhou *et al*. [53] we set p in the fine stride loss to 0 whenever the estimated coarse stride warp is outside a threshold distance from the ground truth. We further found it beneficial to detach the gradients between scales.

Table 1. **SotA comparison.** Homography estimation on HPatches, measured in AUC (higher is better). The top and bottom portions contains sparse methods and dense methods respectively.

Method ↓ AUC →	@3px	@5px	@10px
SuperGlue [36] CVPR'19	53.9	68.3	81.7
LoFTR [41] CVPR'21	65.9	75.6	84.6
SE2-LoFTR [5] CVPRW'22	66.2	76.6	86.0
TopicFM [14] AAAI'23	67.3	77.0	85.7
3DG-STFM [28] ECCV'22	64.7	73.1	81.0
ASpanFormer [7] ECCV'22	67.4	76.9	85.6
PDC-Net+ [48] TPAMI'23	67.7	77.6	86.3
DKM	**71.3**	**80.6**	**88.5**

4. State-of-the-Art Comparison

Similarly to previous approaches [7, 36, 41, 44], we train and evaluate our approach separately on **outdoor** and **indoor** geometry estimation. For evaluation we present the average of 5 benchmark runs. For DKM we sample a maximum of 5000 matches.

4.1. Training Details

We use a batch size of 32 with a learning rate of $4 \cdot 10^{-4}$ for the decoder and refiners, and $2 \cdot 10^{-5}$ for the backbone. We use the AdamW [25] optimizer with a weight-decay factor of 10^{-2}. We train for $250\,000$ steps, decaying the learning rate by a factor 0.2 at step $166\,666$ and $225\,000$. Training takes roughly 5 days on 4 A100fat GPUs, which is comparable to LoFTR that converges in 1 day on 64 1080ti GPUs.

Outdoor Training: We train on the real world dataset MegaDepth [22], using the same training and test split as in previous work [7, 41]. We resize the images to a fixed resolution of 540×720.

Indoor Training: For indoor two-view pose estimation we additionally train on the ScanNet [9] dataset in a similar fashion as previous work [36, 41] and use a resolution of 480×640.

4.2. Outdoor Geometry Estimation

HPatches Homography: HPatches [1] depicts planar scenes divided in sequences, with transformations restricted to homographies. We follow the evaluation protocol proposed LoFTR [41], resizing the shorter side of the images to 480. Table 1 clearly shows the superiority of DKM, showing gains of +3.6 AUC@3px compared to the best previous method.

MegaDepth-1500 Pose Estimation: We use the MegaDepth-1500 test set [41] which consists of 1500 pairs from scene 0015 (St. Peter's Basilica) and 0022 (Brandenburger Tor). We follow the protocol in [7, 41] and use a RANSAC threshold of 0.5 with intrinsics equivalent to a

Table 2. **SotA comparison.** Pose estimation results on the Megadepth-1500 benchmark, measured in AUC (higher is better). The top and bottom portions contains sparse methods and dense methods respectively.

Method ↓ AUC →	@5°	@10°	@20°
SuperGlue [36] CVPR'19	42.2	61.2	76.0
LoFTR [41] CVPR'21	52.8	69.2	81.2
QuadTree [44] ICLR'22	54.6	70.5	82.2
SE2-LoFTR [5] CVPRW'22	52.6	69.2	81.4
MatchFormer [50] ACCV'22	52.9	69.7	82.0
3DG-STFM [28] ECCV'22	52.6	68.5	80.0
ASpanFormer [7] ECCV'22	55.3	71.5	83.1
TopicFM [14] AAAI'23	54.1	70.1	81.6
DenseGAP [21] ICPR'22	41.2	56.9	70.2
ECO-TR [43] ECCV'22	48.3	65.8	78.5
PDC-Net+ [48] TPAMI'23	51.5	67.2	78.5
DKM	**60.4**	**74.9**	**85.1**

longer side of 1200. Our results, presented in Table 2, show that our method sets a new state-of-the-art. Notably, we outperform the current best sparse method ASpanFormer [50] with an improvement of +4.9 AUC@5°. Furthermore, we significantly outperform the best previous dense method PDC-Net+ [48] with an improvement of +8.9 AUC@5°.

Additional Benchmarks: We create a novel benchmark based on 8 diverse MegaDepth scenes, where DKM shows major improvements. We further do comparisons to COTR/ECO-TR [20, 43] on the St. Paul's Cathedral scene, with DKM showing large improvements. Details of these experiments can be found in the supplementary material.

Table 3. **SotA comparison.** Pose estimation results on the ScanNet-1500 benchmark, measured in AUC (higher is better). The top and bottom portions contains sparse methods and dense methods respectively.

Method ↓ AUC →	@5°	@10°	@20°
SuperGlue [36] CVPR'19	16.2	33.8	51.8
LoFTR [41] CVPR'21	22.1	40.8	57.6
QuadTree [44] ICLR'22	24.9	44.7	61.8
MatchFormer [50] ACCV'22	24.3	43.9	61.4
3DG-STFM [28] ECCV'22	23.6	43.6	61.2
ASpanFormer [7] ECCV'22	25.6	46.0	63.3
PDC-Net [49] CVPR'21	18.7	37.0	54.0
DenseGAP [21] ICPR'22	16.9	34.9	53.2
PDC-Net+ [48] TPAMI'23	20.3	39.4	57.1
DKM	**29.4**	**50.7**	**68.3**

4.3. Indoor Geometry Estimation

ScanNet-1500 Pose Estimation: ScanNet [9] is a large scale indoor dataset, composed of challenging sequences with low texture regions and large changes in perspective. We follow the evaluation in SuperGlue [36]. Results are presented in Table 3. Our model achieves a +4.0 AUC@5° gain compared to the previous best sparse method. Compared to the previous best dense method our performance gains are even larger, with gains of +9.3 AUC@5°.

Table 4. **SotA comparison.** Visual localization on the InLoc benchmark using HLoc [35]. Measured in rate (%) of correctly localized queries (higher is better).

Method ↓	DUC1	DUC2
	(0.25m,10°) / (0.5m,10°) / (1.0m,10°)	
SuperGlue [36]	49.0 / 68.7 / 80.8	53.4 / 77.1 / 82.4
LoFTR [41]	47.5 / 72.2 / 84.8	54.2 / 74.8 / 85.5
ASpanFormer [7]	**51.5** / 73.7 / 86.4	55.0 / 74.0 / 81.7
DKM	**51.5 / 75.3 / 86.9**	**63.4 / 82.4 / 87.8**

Visual Localization on InLoc [42]: We follow previous work and use HLoc [35]. Results are presented in Table 4. We find large improvements, particularly on DUC2 where we show a gain of +8.4 % correctly localized queries.

5. Ablation Study

Next, we investigate design choices of our approach.

Global Matcher: Here we investigate the performance impact of replacing a strong baseline correlation volume regressor, similar to the one used in [49] with our proposed kernelized regression and embedding decoder. The results are shown in Table 5. Our proposed method yields an improvement of +1.1 AUC@5°, highlighting the benefits of our proposed global matcher. As expected, the cosine coordinate embeddings outperform the linear embeddings.

Table 5. **Ablation study.** Impact of our proposed global matcher (GM), using either linear or cosine coordinate embeddings, compared to a strong baseline. Measured in AUC (higher is better).

GM ↓ AUC →	@5°	@10°	@20°
Correlation Volume	57.0	72.1	82.9
GM Linear	57.9	72.9	83.7
GM Cosine	**58.1**	**73.2**	**83.8**

Warp Refiners: Here we ablate both the architecture, and the effect of the input representation used. For the architecture we exchange the depthwise convolution blocks for refiners used in previous dense matching work [49]. The results of this ablation are shown in Table 6. Our depthwise refiners significantly outperform the baseline, with a gain of +3.2 AUC@5°. Furthermore, we find that our input representation yields an improvement of +1.6 AUC@5°, highlighting the importance of well chosen representations.

Table 6. **Ablation study.** Impact of removing our proposed depth-wise (DW) warp refiners, or stacked feature maps (FM) from DKM. Measured in AUC (higher is better).

Warp Refiner ↓ AUC →	@5°	@10°	@20°
No DW	54.9	70.0	81.6
No Stacked FM	56.5	71.8	82.7
DW, Stacked FM	**58.1**	**73.2**	**83.8**

Table 7. **Ablation study.** Impact of balanced match sampling for two-view pose estimation, measured in AUC (higher is better).

Sampling ↓ AUC →	@5°	@10°	@20°
No Certainty Sampling	42.9	58.1	70.4
Certainty Sampling	56.1	71.7	83.0
Balanced Sampling	**58.1**	**73.2**	**83.8**

Table 8. **Ablation study.** Impact of changing training resolution for two-view pose estimation, measured in AUC (higher is better).

Resolution ↓ AUC →	@5°	@10°	@20°
384×512	58.1	73.2	83.8
480×640	58.9	73.9	84.4
540×720	**59.4**	**74.0**	**84.5**

Table 9. **Ablation study.** Impact of bidirectional DKM for two-view pose estimation, measured in AUC (higher is better).

Warp ↓ AUC →	@5°	@10°	@20°
Unidirectional	59.4	74.0	84.5
Bidirectional	**60.4**	**74.9**	**85.1**

Match Sampling: We compare a baseline not using the certainty estimate, with either using certainty sampling or our proposed balanced sampling using the reciprocal of the KDE estimate. We present results in Table 7, which shows the need for trustworthy certainty. We find that proposed balanced sampling improves the estimation stage, increasing performance with a gain of +2.0 AUC@5°.

Resolution: Tinchev *et al.* [45] notes the importance of increasing input resolution for estimation performance. To gauge the effect of resolution on estimation performance in the dense paradigm we trained DKM on a set of different resolutions. We present the results of our study in Table 8. We find that high resolution is important for accurate estimation. In particular, comparing 384 × 512 to 540 × 720 we find an increase in performance of +1.3 AUC@5°.

Bidirectionality: Previous dense work [43, 48] has investigated incorporating mutual nearest neighbours in dense matching. Here we propose to instead simply concatenate the reverse warp matches. Results are presented in Table 9. We find an improvement of +1.0 AUC@5°.

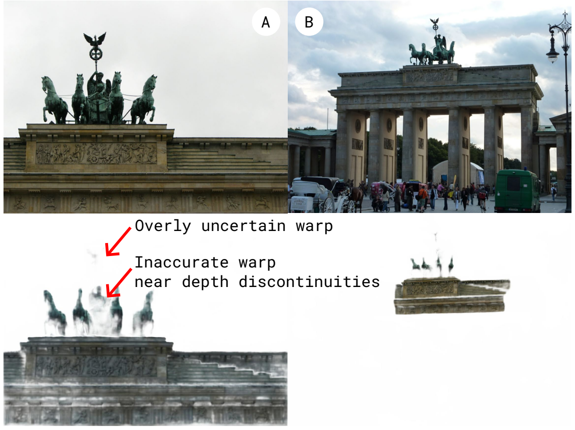

Figure 8. **Representative failure case for DKM.** Our unimodal warp refinement can struggle near depth-discontinuities, and the proposed certainty estimate is occationally overly uncertain.

6. Conclusion

We have presented **DKM**, a novel dense feature matching approach that achieves state-of-the-art two-view geometry estimation results. Three distinct contributions were proposed. We proposed a strong global matcher with a kernelized regressor and embedding decoder. Furthermore, we proposed warp refinement through large depth-wise kernels on stacked feature maps. Finally, we proposed a simple way of learning dense confidence maps by directly classifying consistent depth, and a balanced sampling approach for dense warps. Our extensive experiments clearly showed the superiority of our method, with gains of +8.9 AUC@5° on the MegaDepth-1500 benchmark.

Limitations: While our global matcher can gracefully handle multimodality, the proposed dense warp refinement is unimodal. This poses challenges where the warp is discontinuous, *e.g.*, at depth boundaries. We also found DKM to be overly uncertain for small objects bordering the sky. This could be a limitation of learning to classify consistent depth, instead of predicting model uncertainty as in, *e.g.*, PDC-Net. We illustrate these weaknesses in Figure 8.

Acknowledgements: We thank the anonymous reviewers whose comments helped improve the paper. This work was supported by the Wallenberg Artificial Intelligence, Autonomous Systems and Software Program (WASP), funded by Knut and Alice Wallenberg Foundation; and by the strategic research environment ELLIIT funded by the Swedish government. The computational resources were provided by the National Academic Infrastructure for Supercomputing in Sweden (NAISS), partially funded by the Swedish Research Council through grant agreement no. 2022-06725, and by the Berzelius resource, provided by the Knut and Alice Wallenberg Foundation at the National Supercomputer Centre.

References

[1] Vassileios Balntas, Karel Lenc, Andrea Vedaldi, and Krystian Mikolajczyk. HPatches: A benchmark and evaluation of handcrafted and learned local descriptors. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5173–5182, 2017. 1, 6

[2] Daniel Barath, Luca Cavalli, and Marc Pollefeys. Learning to find good models in ransac. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15744–15753, 2022. 2

[3] Daniel Barath, Michal Polic, Wolfgang Förstner, Torsten Sattler, Tomas Pajdla, and Zuzana Kukelova. Making affine correspondences work in camera geometry computation. In *European Conference on Computer Vision*, pages 723–740. Springer, 2020. 2

[4] Daniel Barath, Tekla Toth, and Levente Hajder. A minimal solution for two-view focal-length estimation using two affine correspondences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6003–6011, 2017. 2

[5] Georg Bökman and Fredrik Kahl. A case for using rotation invariant features in state of the art feature matchers. In *CVPRW*, 2022. 6, 7

[6] Luca Cavalli, Marc Pollefeys, and Daniel Barath. Nefsac: Neurally filtered minimal samples. In *Proc. European Conference on Computer Vision (ECCV)*, 2022. 2, 5

[7] Hongkai Chen, Zixin Luo, Lei Zhou, Yurun Tian, Mingmin Zhen, Tian Fang, David Mckinnon, Yanghai Tsin, and Long Quan. ASpanFormer: Detector-free image matching with adaptive span transformer. In *Proc. European Conference on Computer Vision (ECCV)*, 2022. 2, 5, 6, 7

[8] Ondrej Chum, Tomas Werner, and Jiri Matas. Two-view geometry estimation unaffected by a dominant plane. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2005. 2, 5

[9] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017. 6, 7

[10] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 224–236, 2018. 2

[11] 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 the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019. 2

[12] Hongyi Fan, Joe Kileel, and Benjamin Kimia. On the instability of relative pose estimation and ransac's role. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8935–8943, 2022. 2, 5

[13] Hugo Germain, Guillaume Bourmaud, and Vincent Lepetit. S2DNet: learning image features for accurate sparse-to-dense matching. In *European Conference on Computer Vision (ECCV)*, 2020. 1

[14] Khang Truong Giang, Soohwan Song, and Sungho Jo. TopicFM: Robust and interpretable topic-assisted feature matching. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 2022. 6, 7

[15] Banglei Guan, Ji Zhao, Zhang Li, Fang Sun, and Friedrich Fraundorfer. Minimal solutions for relative pose with a single affine correspondence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1929–1938, 2020. 2

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 3

[17] Xingyi He, Yuang Wang, Jiaming Sun, Zehong Shen, Hujun Bao, and Xiaowei Zhou. Tech details for loftr in the imw challenge. `https://zju3dv.github.io/loftr/files/LoFTR_IMC21.pdf`. 2

[18] Johan Hedborg, Per-Erik Forssén, and Michael Felsberg. Fast and accurate structure and motion estimation. In *International Symposium on Visual Computing*, pages 211–222. Springer, 2009. 2, 5

[19] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017. 5

[20] Wei Jiang, Eduard Trulls, Jan Hosang, Andrea Tagliasacchi, and Kwang Moo Yi. COTR: Correspondence Transformer for Matching Across Images. In *ICCV*, 2021. 7

[21] Zhengfei Kuang, Jiaman Li, Mingming He, Tong Wang, and Yajie Zhao. DenseGAP: Graph-Structured Dense Correspondence Learning with Anchor Points. In *27th International Conference on Pattern Recognition (ICPR)*, 2022. 7

[22] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2041–2050, 2018. 1, 5, 6

[23] Philipp Lindenberger, Paul-Edouard Sarlin, Viktor Larsson, and Marc Pollefeys. Pixel-perfect structure-from-motion with featuremetric refinement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5987–5997, 2021. 1

[24] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, et al. Swin transformer v2: Scaling up capacity and resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12009–12019, 2022. 4

[25] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. 6

[26] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004. 1, 2

[27] Simon Lynen, Bernhard Zeisl, Dror Aiger, Michael Bosse, Joel Hesch, Marc Pollefeys, Roland Siegwart, and Torsten Sattler. Large-scale, real-time visual–inertial localization revisited. *The International Journal of Robotics Research*, 39(9):1061–1084, 2020. 1

[28] Runyu Mao, Chen Bai, Yatong An, Fengqing Zhu, and Cheng Lu. 3DG-STFM: 3d geometric guided student-teacher feature matching. In *Proc. European Conference on Computer Vision (ECCV)*, 2022. 6, 7

[29] Iaroslav Melekhov, Aleksei Tiulpin, Torsten Sattler, Marc Pollefeys, Esa Rahtu, and Juho Kannala. Dgc-net: Dense geometric correspondence network. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1034–1042. IEEE, 2019. 2, 6

[30] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. ORB-SLAM: a versatile and accurate monocular slam system. *IEEE transactions on robotics*, 31(5):1147–1163, 2015. 1

[31] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005. 4

[32] I. Rocco, M. Cimpoi, R. Arandjelović, A. Torii, T. Pajdla, and J. Sivic. Neighbourhood consensus networks. In *Proceedings of the 32nd Conference on Neural Information Processing Systems*, 2018. 2

[33] Ignacio Rocco, Relja Arandjelović, and Josef Sivic. Efficient neighbourhood consensus networks via submanifold sparse convolutions. In *European Conference on Computer Vision*, pages 605–621. Springer, 2020. 2

[34] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015. 3

[35] Paul-Edouard Sarlin, Cesar Cadena, Roland Siegwart, and Marcin Dymczyk. From coarse to fine: Robust hierarchical localization at large scale. In *CVPR*, 2019. 7

[36] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4938–4947, 2020. 1, 2, 6, 7

[37] Torsten Sattler, Will Maddern, Carl Toft, Akihiko Torii, Lars Hammarstrand, Erik Stenborg, Daniel Safari, Masatoshi Okutomi, Marc Pollefeys, Josef Sivic, et al. Benchmarking 6dof outdoor visual localization in changing conditions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8601–8610, 2018. 1

[38] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016. 1

[39] Xi Shen, François Darmon, Alexei A Efros, and Mathieu Aubry. Ransac-flow: generic two-stage image alignment. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV 16*, pages 618–637. Springer, 2020. 2, 4, 6

[40] Herman P Snippe and Jan J Koenderink. Discrimination thresholds for channel-coded systems. *Biological cybernetics*, 66(6):543–551, 1992. 4

[41] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. LoFTR: Detector-free local feature matching with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8922–8931, 2021. 1, 2, 5, 6, 7

[42] 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 the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7199–7209, 2018. 7

[43] Dongli Tan, Jiang-Jiang Liu, Xingyu Chen, Chao Chen, Ruixin Zhang, Yunhang Shen, Shouhong Ding, and Rongrong Ji. ECO-TR: Efficient Correspondences Finding Via Coarse-to-Fine Refinement. In *Proc. European Conference on Computer Vision (ECCV)*, 2022. 6, 7, 8

[44] Shitao Tang, Jiahui Zhang, Siyu Zhu, and Ping Tan. Quadtree attention for vision transformers. In *International Conference on Learning Representations*, 2022. 2, 6, 7

[45] Georgi Tinchev, Shuda Li, Kai Han, David Mitchell, and Rigas Kouskouridas. $\mathbb{X}$resolution correspondence networks. In *Proceedings of British Machine Vision Conference (BMVC)*, 2021. 2, 8

[46] Carl Toft, Will Maddern, Akihiko Torii, Lars Hammarstrand, Erik Stenborg, Daniel Safari, Masatoshi Okutomi, Marc Pollefeys, Josef Sivic, Tomas Pajdla, et al. Long-term visual localization revisited. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 1

[47] Prune Truong, Martin Danelljan, and Radu Timofte. GLU-Net: Global-local universal network for dense flow and correspondences. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6258–6268, 2020. 2, 4, 5

[48] Prune Truong, Martin Danelljan, Radu Timofte, and Luc Van Gool. Pdc-net+: Enhanced probabilistic dense correspondence network. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–18, 2023. 1, 2, 5, 6, 7, 8

[49] Prune Truong, Martin Danelljan, Luc Van Gool, and Radu Timofte. Learning accurate dense correspondences and when to trust them. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5714–5724, 2021. 2, 5, 6, 7

[50] Qing Wang, Jiaming Zhang, Kailun Yang, Kunyu Peng, and Rainer Stiefelhagen. MatchFormer: Interleaving attention in transformers for feature matching. In *Asian Conference on Computer Vision*, 2022. 2, 7

[51] Olivia Wiles, Sebastien Ehrhardt, and Andrew Zisserman. Co-attention for conditioned image matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15920–15929, 2021. 2

[52] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Learning a discriminative feature network for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1857–1866, 2018. 4

[53] Qunjie Zhou, Torsten Sattler, and Laura Leal-Taixe. Patch2pix: Epipolar-guided pixel-level correspondences. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4669–4678, 2021. 1, 2, 6

[54] Mauricio A. Álvarez, Lorenzo Rosasco, and Neil D. Lawrence. Kernels for vector-valued functions: A review. *Foundations and Trends® in Machine Learning*, 4(3):195–266, 2012. 4