GLU-Net: Global-Local Universal Network for Dense Flow and Correspondences

Prune Truong   Martin Danelljan   Radu Timofte
Computer Vision Lab, D-ITET, ETH Zürich, Switzerland
{prune.truong, martin.danelljan, radu.timofte}@vision.ee.ethz.ch

Abstract

Establishing dense correspondences between a pair of images is an important and general problem, covering geometric matching, optical flow and semantic correspondences. While these applications share fundamental challenges, such as large displacements, pixel-accuracy, and appearance changes, they are currently addressed with specialized network architectures, designed for only one particular task. This severely limits the generalization capabilities of such networks to new scenarios, where e.g. *robustness to larger displacements or higher accuracy is required.*

In this work, we propose a universal network architecture that is directly applicable to all the aforementioned dense correspondence problems. We achieve both high accuracy and robustness to large displacements by investigating the combined use of global and local correlation layers. We further propose an adaptive resolution strategy, allowing our network to operate on virtually any input image resolution. The proposed ***GLU-Net*** *achieves state-of-the-art performance for geometric and semantic matching as well as optical flow, when using the* same *network and weights. Code and trained models are available at* https://github.com/PruneTruong/GLU-Net.

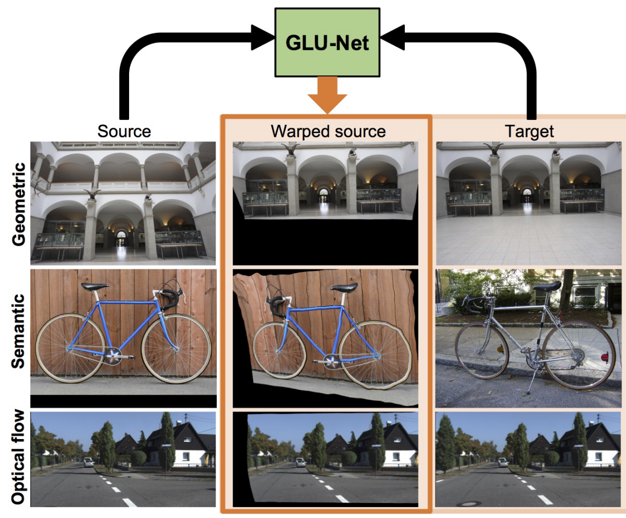


Figure 1. Our GLU-Net estimates dense correspondences between a source (left) and a target (right) image. The estimated correspondences are here used to warp (center) the source image. The warped result (center) accurately matches the target image (right). The *same* network and weights are applied for Geometric matching, Semantic matching and Optical flow tasks.

1. Introduction

Finding pixel-to-pixel correspondences between images continues to be a fundamental problem in Computer Vision [14, 18]. This is due to its many important applications, including visual localization [48, 53], 3D-reconstruction [1], structure-from-motion [47], image manipulation [16, 35], action recognition [50] and autonomous driving [27]. Due to the astonishing performance achieved by the recent developments in deep learning, end-to-end trainable Convolutional Neural Networks (CNNs) are now applied for this task in all the aforementioned domains.

The general problem of estimating correspondences between pairs of images can be divided into several different tasks, depending on the origin of the images. In the *geometric matching* task [18], the images constitute different views of the same scene, taken by a single or multiple cameras. The images may be taken from radically different viewpoints, leading to large displacements and appearance transformations between the frames. On the other hand, *optical flow* [4, 20] aims to estimate accurate pixel-wise displacements between two consecutive frames of a sequence or video. In the *semantic matching* problem [17, 35] (also referred as semantic flow), the task is instead to find semantically meaningful correspondences between different instances of the same scene category or object, such as 'car' or 'horse'. Current methods generally address *one* of these tasks, using specialized architectures that generalize poorly to related correspondence problems. In this work, we therefore set out to design a universal architecture that jointly addresses all aforementioned tasks.

One key architectural aspect shared by a variety of correspondence networks is the reliance on *correlation layers*, computing local similarities between deep features extracted from the two images. These provide strong cues when establishing correspondences. Optical flow methods typically employ *local* correlation layers [13, 22, 23, 25, 51, 52], evaluating similarities in a local neighborhood around an image coordinate. While suitable for small displacements, they are unable to capture large viewpoints changes. On the contrary, geometric and semantic matching architectures utilize *global* correlations [31, 37, 43, 44, 45], where similarities are evaluated between all pairs of locations in the dense feature maps. While capable of handling long-range matches, global correlation layers are computationally unfeasible at high resolutions. Moreover, they constrain the input image size to a pre-determined resolution, which severely hampers accuracy for high-resolution images.

Contributions: In this paper, we propose GLU-Net, a Global-Local Universal Network for estimating dense correspondences. Our architecture is robust to large viewpoint changes and appearance transformations, while capable of estimating small displacements with high accuracy. The main contributions of this work are: **(i)** We introduce a single unified architecture, applicable to geometric matching, semantic matching and optical flow. **(ii)** Our network carefully integrates global and local correlation layers to handle both large and small displacements. **(iii)** To circumvent the fixed input resolution imposed by the global cost volume, we propose an adaptive resolution strategy that enables our network to take *any* image resolution as input, crucial for high-accuracy displacements. **(iv)** We train our network in a self-supervised manner, relying on synthetic warps of real images, thus requiring no annotated ground-truth flow.

We perform comprehensive experiments on the three aforementioned tasks, providing detailed analysis of our approach and thorough comparisons with recent state-of-the-art. Our approach outperforms previous methods for dense geometric correspondences on the HPatches [5] and ETH3D [49] datasets, while setting a new state-of-the-art for semantic correspondences on the TSS [54] dataset. Moreover, our network, without any retraining or fine-tuning, generalizes to optical flow by providing highly competitive results on the KITTI [15] dataset. Both training code and models are available at [55].

2. Related work

Finding correspondences between a pair of images is a classical computer vision problem, uniting optical flow, geometric correspondences and semantic matching. This problem dates back several decades [20], with most classical techniques relying on hand crafted [2, 3, 6, 19, 34, 36, 46] or trained [12, 40, 56] feature detectors/descriptors, or variational formulations [4, 20, 35]. In recent years, CNNs have revolutionised most areas within vision, including different aspects of the image correspondence problem. Here, we focus on Convolutional Neural Network (CNN)-based methods for generating *dense* correspondences or flow fields, as these are most related to our work.

Optical Flow: Dosovitskiy *et al*. [13] constructed the first trainable CNN for optical flow estimation, FlowNet, based on a U-Net denoising autoencoder architecture [57] and trained it on the large synthetic FlyingChairs dataset. Ilg *et al*. [25] stacked several basic FlowNet models into a large one, called FlowNet2, which performed on par with classical state-of-the-art methods on the Sintel benchmark [7]. Subsequently, Ranjan and Black [42] introduced SpyNet, a compact spatial image pyramid network.

Recent notable contributions to end-to-end trainable optical flow include PWC-Net [51, 52] and LiteFlowNet [22], followed by LiteFlowNet2 [23]. They employ multiple constrained correlation layers operating on a feature pyramid, where the features at each level are warped by the current flow estimate, yielding more compact and effective networks. Nevertheless, while these networks excel at small to medium displacements with small appearance changes, they perform poorly on strong geometric transformations or when the visual appearance is significantly different.

Geometric Correspondence: Unlike optical flow, geometric correspondence estimation focuses on *large* geometric displacements, which can cause significant appearance distortions between the frames. Motivated by recent advancements in optical flow architectures, Melekhov *et al*. [37] introduced DGC-Net, a coarse-to-fine CNN-based framework that generates dense 2D correspondences between image pairs. It relies on a global cost volume constructed at the coarsest resolution. However, the input size is constrained to a fixed resolution (240×240), severely limiting its performance on higher resolution images. Rocco *et al*. [45] aim at increasing the performance of the global correlation layer by proposing an end-to-end trainable neighborhood consensus network, NC-Net, to filter out ambiguous matches and keep only the locally and cyclically consistent ones. Furthermore, Laskar *et al*. [33] utilize a modified version of DGC-Net, focusing on image retrieval.

Semantic Correspondence: Unlike optical flow or geometric matching, semantic correspondence poses additional challenges due to intra-class appearance and shape variations among different instances from the same object or scene category. Rocco *et al*. [43, 44] proposed the CNNGeo matching architecture, predicting globally parametrized affine and thin plate spline transformations between image pairs. Other approaches aim to predict richer geometric deformations [10, 29, 30, 45] using *e.g*. Spatial Transformer Networks [26]. Recently, Jeon *et al*. [28] introduced PARN, a pyramidal model where dense affine transformation fields are progressively estimated in a coarse-to-fine manner. SAM-Net [31] obtains better results by jointly learning semantic correspondence and attribute transfer. Huang *et al*. [21] proposed DCCNet, which fuses correlation maps derived from local features and a context-aware semantic feature representation.

3. Method

We address the problem of finding pixel-wise correspondences between a pair of images $I_s \in \mathbb{R}^{H \times W \times 3}$ and $I_t \in \mathbb{R}^{H \times W \times 3}$. In this work, we put no particular assumptions on the origin of the image pair itself. It may correspond to two different views of the same scene, two consecutive frames in a video, or two images with similar semantic content. Our goal is to estimate a dense displacement field, often referred to as flow, $\mathbf{w} \in \mathbb{R}^{H \times W \times 2}$ that warps image I_s towards I_t such that,

$$I_t(\mathbf{x}) \approx I_s(\mathbf{x} + \mathbf{w}(\mathbf{x})) . \quad (1)$$

The flow $\mathbf{w}$ represents the pixel-wise 2D motion vectors in the target image coordinate system. It is directly related to the pixel correspondence map $\mathbf{m}(\mathbf{x}) = \mathbf{x} + \mathbf{w}(\mathbf{x})$, which directly maps an image coordinate $\mathbf{x}$ in the target image to its corresponding position in the source image.

In this work, we design an architecture capable of robustly finding both long-range correspondences and accurate estimation of pixel-wise displacements. We thereby achieve a universal network for predicting dense flow fields, applicable to geometric matching, semantic correspondences and optical flow. The overall architecture follows a CNN feature-based coarse-to-fine strategy, which has proved widely successful for specific tasks [22, 28, 31, 37, 51]. However, contrary to previous works, our architecture combines global and local correlation layers, as discussed in Section 3.1 and 3.2, to benefit from their complementary properties. We further circumvent the input resolution restriction imposed by the global correlation layer by introducing an adaptive resolution strategy in Section 3.3. It is based on a two-stream feature pyramid, which allows dense correspondence prediction for *any* input resolution image. Our final architecture is detailed in Section 3.4 and the training procedure explained in Section 3.5.

3.1. Local and Global Correlations

Current state-of-the-art architectures [21, 22, 28, 37, 51] for estimating image correspondences or optical flow rely on measuring local similarities between the source and target images. This is performed in a deep feature space, which provides a discriminative embedding with desirable invariances. The result, generally referred to as a correlation or cost volume, provides an extremely powerful cue when deriving the final correspondence or flow estimate. The correlation can be performed in a local or global manner.

Local correlation: In a local correlation layer, the feature similarity is only evaluated in the neighborhood of the target image coordinate, specified by a search radius R. Formally, the correlation c^l between the target $F_t^l \in \mathbb{R}^{H_l \times W_l \times d_l}$ and source $F_s^l \in \mathbb{R}^{H_l \times W_l \times d_l}$ feature maps is defined as,

$$c^l(\mathbf{x}, \mathbf{d}) = F_t^l(\mathbf{x})^T F_s^l(\mathbf{x} + \mathbf{d}) , \quad \|\mathbf{d}\|_\infty \leq R , \quad (2)$$

where $\mathbf{x} \in \mathbb{Z}^2$ is a coordinate in the target feature map and $\mathbf{d} \in \mathbb{Z}^2$ is the displacement from this location. The displacement is constrained to $\|\mathbf{d}\|_\infty \leq R$, *i.e.* the maximum motion in any direction is R. We let l denote the level in the feature pyramid. While most naturally thought of as a 4-dimensional tensor, the two displacement dimensions are usually vectorized into one to simplify further processing in the CNN. The resulting 3D correlation volume c^l thus has a dimensionality of $H_l \times W_l \times (2R+1)^2$.

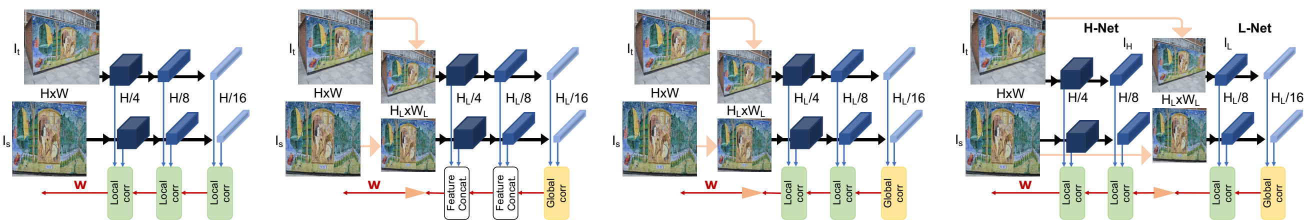
Global correlation: A global correlation layer evaluates the pairwise similarities between all locations in the target and source feature maps. The correlation volume $C^l \in \mathbb{R}^{H_l \times W_l \times H_l \times W_l}$ contains at each target image location $\mathbf{x} \in \mathbb{Z}^2$ the scalar products between corresponding feature vector $F_t^l(\mathbf{x})$ and the vectors $F_s^l(\mathbf{x}') \in \mathbb{R}^d$ extracted from all source feature map coordinates $\mathbf{x}'$,

$$C^l(\mathbf{x}, \mathbf{x}') = F_t^l(\mathbf{x})^T F_s^l(\mathbf{x}') . \quad (3)$$

As for the local cost volume, we vectorize the source dimensions, leading to a 3D tensor of size $H_l \times W_l \times (H_l W_l)$.

Comparison: Local and global correlation layers have a few key contrary properties and behaviors. Local correlations are popularly employed in architectures designed for optical flow [13, 22, 51], where the displacements are generally small. Thanks to their restricted search region, local correlation layers can be applied for high-resolution feature maps, which allows accurate estimation of small displacements. On the other hand, a local correlation based architecture is limited to a certain maximum range of displacements. Conversely, a global correlation based architecture does not suffer from this limitation, encapsulating arbitrary long-range displacements.

The major disadvantage of the global cost volume is that its dimensionality scales with the size of the feature map $H_l \times W_l$. Therefore, due to the quadratic $\mathcal{O}((H_l W_l)^2)$ scaling in computation and memory, global cost volumes are only suitable at coarse resolutions. Moreover, post-processing layers implemented with 2D convolutions expect a fixed channel dimensionality. Since the channel dimension $H_l W_l$ of the cost volume depends on its spatial dimensions $H_l \times W_l$, this effectively constrains the network input resolution to a fixed pre-determined value, referred to as $H_L \times W_L$. The network can thus not leverage the more detailed structure in high-resolution images and lacks precision, since the images require down-scaling to $H_L \times W_L$ before being processed by the network. Architectures with only local correlations (Local-Net) or with a unique global correlation (Global-Net) are visualized in Figure 2a, b.



(a) Local-Net (b) Global-Net (c) GLOCAL-Net (d) GLU-Net (Ours)

Figure 2. Schematic representation of different architectures for dense flow field estimation $\mathbf{w}$. Local-Net (a) and Global-Net (b) employ *only* local and global correlation layers, respectively. Our GLOCAL-Net (c) combines both to effectively handle short and long-range displacements. GLU-Net (d) additionally employs our adaptive resolution strategy, thus capable of processing high-resolution images.

3.2. Global-Local Architecture

We introduce a unified network that leverages the advantages of both global and local correlation layers and which also circumvents the limitations of both. Our goal is to handle any kind of geometric transformations - including large displacements - while achieving high precision for detailed and small displacements. This is performed by carefully integrating global and local correlation layers in a feature pyramid based network architecture.

Inspired by DGC-Net [37], we employ a global correlation layer at the coarsest level. The purpose of this layer is to handle the *long-range correspondences*. Since these are best captured in the coarsest scale, only a single global correlation is needed. In subsequent layers, the dense flow field is refined by computing image feature similarity using local correlations. This allows *precise* estimation of the displacements. Combining global and local correlation layers allows us to achieve robust and accurate prediction of both long and small-range motions. Such an architecture is visualized with GLOCAL-Net in Figure 2c. However, this network is still restricted to a certain input resolution. Next, we introduce a design strategy that circumvents this issue.

3.3. Adaptive resolution

As previously discussed, the global correlation layer imposes a pre-determined input resolution for the network to ensure a constant channel dimensionality of the global cost volume. This severely limits the applicability and accuracy of the correspondence network, since higher resolution images requires down-scaling before being processed by the network, followed by up-scaling of the resulting flow. In this section, we address this key issue by introducing an architecture capable of taking images of *any* resolution, while still benefiting from a global correlation.

Our adaptive-resolution architecture consists of two subnetworks, which operate on two different image resolutions. The first, termed L-Net, takes source and target images downscaled to a fixed resolution $H_L \times W_L$, which allows a global correlation layer to be integrated. The H-Net on the other hand, operates directly on the original image resolution $H \times W$, which is not constrained to any specific value. It refines the flow estimate generated by the L-Net with local correlations applied to a shallow feature pyramid constructed directly from the original images. It is schematically represented in Figure 2d.

Both sub-networks are based on a coarse-to-fine architecture, employing the same feature extractor backbone. In detail, the L-Net relies on a global correlation at the coarsest level in order to effectively handle any kind of geometric transformations, including very large displacements. Subsequent levels of L-Net employ local correlations to refine the flow field. It is then up-sampled to the coarsest resolution of H-Net, where it serves as the initial flow estimate used for warping the source features F_s. Subsequently, the flow prediction is refined numerous times within H-Net, that operates on the full scale images, thus providing a very detailed, sub-pixel accurate final estimation of the dense flow field relating I_s and I_t.

For high-resolution images, the upscaling factor between the finest pyramid level, l_L, of L-Net and the coarsest, l_H, of H-Net (see Figure 2d) can be significant. Our adaptive resolution strategy allows additional refinement steps of the flow estimate between those two levels during inference, thus improving the accuracy of the estimated flow, without training any additional weights. This is performed by recursively applying the l_H layer weights at intermediate resolutions obtained by down-sampling the source and target features from l_H. In summary, our adaptive resolution network is capable of seamlessly predicting an accurate flow field in the original input resolution, while also benefiting from robustness to long-range correspondences provided by the global layer. The entire network is trained end-to-end.

3.4. Architecture details

In this section, we provide a detailed description of our architecture. While any feature extractor backbone can be employed, we use the VGG-16 [8] network trained on ImageNet [32] to provide a fair comparison to previous works in geometric [37] and semantic correspondences [28]. For our L-Net, we set the input resolution to $(H_L \times W_L) = (256 \times 256)$. It is composed of two pyramid levels, using `Conv5-3` (16×16 resolution) and `Conv4-3` (32×32 resolution) respectively. The former employs global correlation,

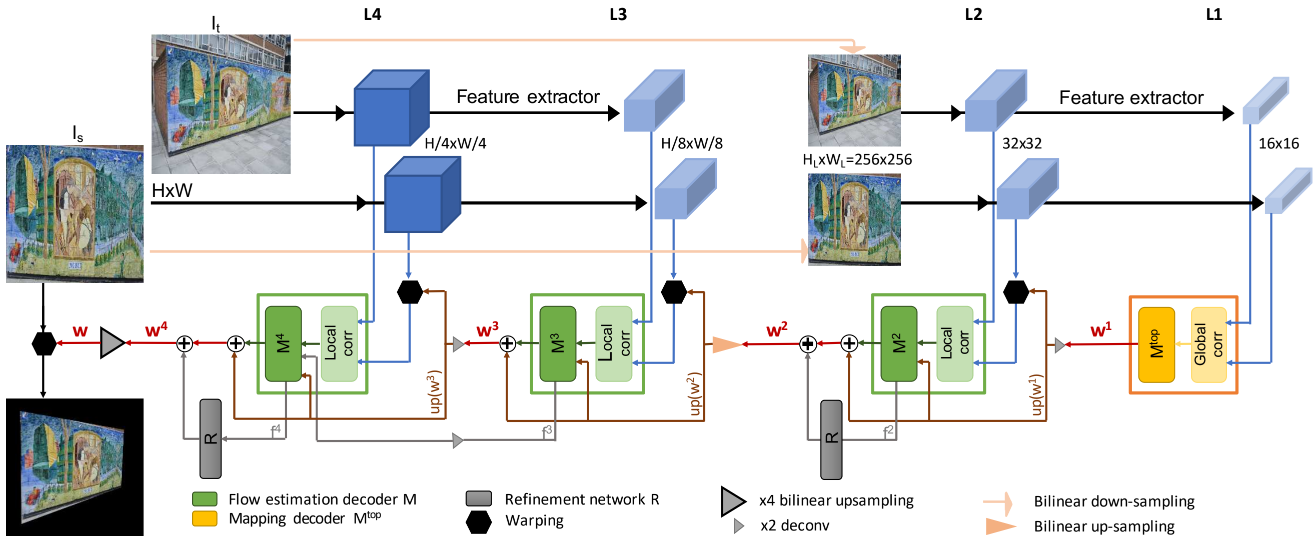


Figure 3. Architectural details of our GLU-Net. It is composed of two modules, operating on two different image resolutions. The L-Net (right) relies on a global correlation for long-range matches, while the H-Net (left) refines the flow estimate with local correlations.

while the latter is based on a local correlation. The H-Net is composed of two feature pyramid levels extracted from the original image resolution $H \times W$. For this purpose, we employ `Conv4-3` and `Conv3-3` having resolutions $\frac{H}{8} \times \frac{W}{8}$ and $\frac{H}{4} \times \frac{W}{4}$ respectively. The H-Net is purely based on local correlation layers. Our final architecture GLU-Net, composed of four pyramid levels in total, is detailed in Figure 3. Next, we describe the various architectural components.

Coarsest resolution and mapping estimation: We compute a global correlation from the L^2-normalized source and target features. The cost volume is further post-processed by applying channel-wise L^2-normalisation followed by ReLU [38] to strongly down-weight ambiguous matches [43]. Similar to DGC-Net [37], the resulting global correlation C is then fed into a correspondence map decoder M_{top} to estimate a 2D dense correspondence map $\mathbf{m}$ at the coarsest level $L1$ of the feature pyramid:

$$\mathbf{m}^1 = M_{\text{top}}\left(C\left(F_{\text{t}}^1, F_{\text{s}}^1\right)\right) . \tag{4}$$

The correspondence map is then converted to a displacement field, as $\mathbf{w}^1(\mathbf{x}) = \mathbf{m}^1(\mathbf{x}) - \mathbf{x}$.

Subsequent flow estimations: The flow is refined by local correlation modules. At level l, the flow decoder M infers the residual flow $\Delta\tilde{\mathbf{w}}^l$ as,

$$\Delta\tilde{\mathbf{w}}^l = M\left(c\left(F_{\text{t}}^l, \widetilde{F}_{\text{s}}^l; R\right), \text{up}\left(\mathbf{w}^{l-1}\right)\right) . \tag{5}$$

c is a local correlation (2) with search radius R and $\widetilde{F}_{\text{s}}^l(\mathbf{x}) = F_{\text{s}}^l\left(\mathbf{x} + \text{up}\left(\mathbf{w}^{l-1}\right)(\mathbf{x})\right)$ is the warped source feature map F_{s} according to the upsampled flow $\text{up}\left(\mathbf{w}^{l-1}\right)$. The complete flow field is computed as $\tilde{\mathbf{w}}^l = \Delta\tilde{\mathbf{w}}^l + \text{up}\left(\mathbf{w}^{l-1}\right)$.

Flow refinement: Contextual information have been shown advantageous for pixel-wise prediction tasks [9, 21]. We thus use a sub-network R, called the refinement network, to post-process the estimated flow at the highest levels of L-Net and H-Net (L2 and L4 in Figure 3) by effectively enlarging the receptive field size. It takes the features f^l of the second last layer from the flow decoder M^l as input and outputs the refined flow $\mathbf{w}^l = R\left(f^l\right) + \tilde{\mathbf{w}}^l$. For the other pyramid level (L3), the final flow field is $\mathbf{w}^l = \tilde{\mathbf{w}}^l$.

Cyclic consistency: Since the quality of the correlation is of primary importance for the flow estimation process, we introduce an additional filtering step on the global cost volume to enforce the reciprocity constraint on matches. We employ the soft mutual nearest neighbor filtering introduced by [45] and apply it to post-process the global correlation.

3.5. Training

Loss: We train our network in a single phase. We freeze the pre-trained backbone feature extractor during training. Following FlowNet [13], we apply supervision at every pyramid level using the endpoint error (EPE) loss with respect to the ground truth displacements.

Dataset: Our network is solely trained on pairs generated by applying random warps to the original images. Since our network is designed to also estimate correspondences between high-resolution images, training data of sufficient resolution is preferred in order to utilize the full potential of our architecture. We use a combination of the DPED [24], CityScapes [11] and ADE-20K [58] datasets, which have images larger than 750×750. On the total dataset of $40,000$ images, we apply the same synthetic transformations as in *tokyo* (DGC-Net [37] training data). The resulting image pairs are cropped to 520×520 for training. We call this dataset *DPED-CityScape-ADE*. We provide additional training and architectural details in the supplementary.

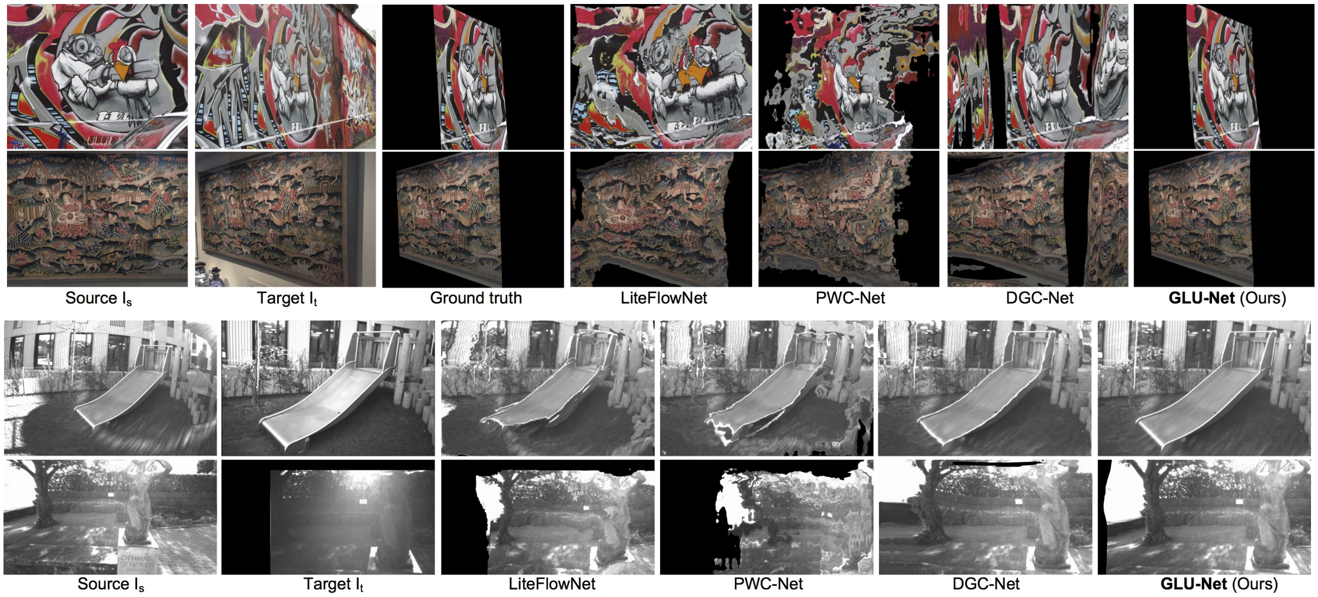


Figure 4. Qualitative comparison with state-of-the-art on geometric correspondence datasets. Top: Pairs of HP images. Bottom: Pairs of images from ETH3D taken by two different cameras. Our approach effectively handles large variations in view-point and appearance.

4. Experimental Validation

In this section, we comprehensively evaluate our approach for three diverse problems: geometric matching, semantic correspondences and optical flow. Importantly, we use the *same* network and model weights, trained on *DPED-CityScape-ADE*, for all three applications. More detailed results are available in the supplementary material.

4.1. Geometric matching

We first apply our GLU-Net for the task of geometric matching. The images thus consist of different views of the same scene and include large geometric transformations.

HP: As in DGC-Net [37], we employ the 59 sequences of the HPatches dataset [5] labelled with `v_X`, which have viewpoint changes, thus excluding the ones labelled `i_X`, which only have illumination changes. Each image sequence contains a source image and 5 target images taken under increasingly larger viewpoints changes, with sizes ranging from 450×600 to 1613×1210. In addition to evaluating on the original image resolution (referred to as **HP**), we also evaluate on downscaled (240×240) images and ground-truths (**HP-240**) following [37].

ETH3D: To validate our approach for real 3D scenes, where image transformations are not constrained to simple homographies, we also employ the Multi-view dataset ETH3D [49]. It contains 10 image sequences at 480×752 or 514×955 resolution, depicting indoor and outdoor scenes. The authors additionally provide a set of sparse geometrically consistent image correspondences (generated by [47]) that have been optimized over the entire image sequence using the reprojection error. We sample image pairs from each sequence at different intervals to analyze varying magnitude of geometric transformations, and use the provided points as sparse ground truth correspondences. This results in about 500 image pairs in total for each selected interval.

Metrics: In line with [37], we employ the Average End-Point Error (AEPE) and Percentage of Correct Keypoints (PCK) as the evaluation metrics. AEPE is defined as the Euclidean distance between estimated and ground truth flow fields, averaged over all valid pixels of the target image. PCK is computed as the percentage of correspondences $\tilde{\mathbf{x}}_j$ with an Euclidean distance error $\|\tilde{\mathbf{x}}_j - \mathbf{x}_j\| \leq \delta$, w.r.t. to the ground truth $\mathbf{x}_j$, that is smaller than a threshold δ.

Compared methods: We compare with DGC-Net [37] trained on *tokyo*, which is the current state-of-the-art for dense geometric matching. For a fair comparison we also train a version, called DGC-Net†, using the same data (*DPED-CityScape-ADE*) as our GLU-Net. We additionally compare with two state-of-the-art optical flow methods, PWC-Net [51] and LiteFlowNet [22], both trained on *Flying-Chairs* [13] followed by *3D-things* [25]. We use the PyTorch [41] implementations [37, 39, 51] of the models and the pre-trained weights provided by the authors.

Results: We first present results on the HP and HP-240 in Table 1. Our model strongly outperforms all others by a large margin both in terms of accuracy (PCK) and robustness (AEPE). It is interesting to note that while our

		HP-240x240			HP		
	Run-time	AEPE	PCK-1px	PCK-5px	AEPE	PCK-1px	PCK-5px
LiteFlowNet [22]	45.10 ms	19.41	28.36 %	57.66 %	118.85	13.91 %	31.64 %
PWC-Net [51, 52]	38.51 ms	21.68	20.99 %	54.19 %	96.14	13.14 %	37.14 %
DGC-Net [37]	138.30 ms	9.07	50.01 %	77.40 %	33.26	12.00 %	58.06 %
DGC-Net†	138.30 ms	9.12	43.09 %	79.35 %	33.47	9.19 %	56.02 %
GLU-Net (Ours)	**38.10** ms	**7.40**	**59.92** %	**83.47** %	**25.05**	**39.55** %	**78.54** %

Table 1. Comparison of state-of-the-art algorithms applied to the task of geometric matching, on the HPatches dataset [5]. Lower AEPE and higher PCK are better.

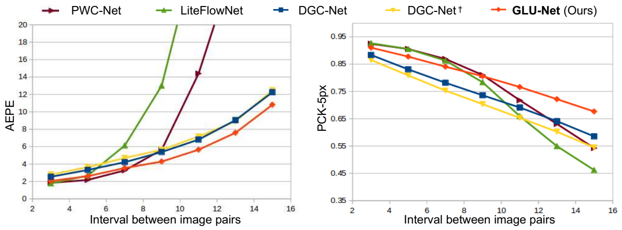


Figure 5. Quantitative results on geometric matching dataset ETH3D [49]. AEPE and PCK-5 are computed on pairs of images sampled from consecutive images of ETH3D at different intervals.

model is already better than DGC-Net on the small resolution HP-240, the gap in performance further broadens when increasing the image resolution. Particularly, GLU-Net obtains a PCK-1px value almost four times higher than that of DGC-Net on HP. This demonstrates the benefit of our adaptive resolution strategy, which enables to process high-resolution images with high accuracy. Moreover, our model achieves a 3.6 times faster inference compared to DGC-Net. Figure 4 shows qualitative examples of different networks applied to HP images and ETH3D image pairs taken by two different cameras. Our GLU-Net is robust to large viewpoints variations as well as drastic changes in illumination.

In Figure 5, we plot AEPE and PCK-5px obtained on the ETH3D scenes for different intervals between image pairs. For small intervals, finding correspondences strongly resembles optical flow task while increasing it leads to larger displacements. Therefore, specialised optical flow methods PWC-Net [51] and LiteFlowNet [22] obtain slightly better AEPE and PCK for low intervals, but rapidly degrade for larger ones. In all cases, our approach consistently outperforms DGC-Net [37] in both metrics by a large margin.

4.2. Semantic matching

Here, we perform experiments for the task of semantic matching, where images depict different instances of the same object category, such as cars or horses. We use the same model and weights as in the previous section.

Dataset and metric: We use the TSS dataset [54], which provides dense flow fields annotations for the foreground object in each pair. It contains 400 image pairs, divided into three groups: FG3DCAR, JODS, and PASCAL, according to the origins of the images. Following Taniai *et al.* [54], we report the PCK with a distance threshold equal to $\alpha \cdot \max(H_s, W_s)$, where H_s and W_s are the dimensions of the source image and $\alpha = 0.05$.

Compared methods: We compare to several recent state-of-the-art methods specialised in semantic matching [21, 28, 29, 31, 44, 45]. In addition to our universal network, we evaluate a version that adopts two architectural details that are used in the semantic correspondence literature. Specifically, we add a consensus network [45] for the global correlation layer and concatenate features from different levels in the L-Net, similarly to [28] (see Section 4.4 for an analysis). We call this version *Semantic-GLU-Net*.

Methods	Feature backbone	FG3DCar	JODS	PASCAL	Avg.
CNNGeo(W) [44]	ResNet-101	90.3	76.4	56.5	74.4
RTNs [29]	ResNet-101	90.1	78.2	63.3	77.2
PARN [28]	VGG-16	87.6	71.6	68.8	76.0
PARN [28]	ResNet-101	89.5	75.9	71.2	78.8
NC-Net [45]	ResNet-101	94.5	81.4	57.1	77.7
DCCNet [21]	ResNet-101	93.5	**82.6**	57.6	77.9
SAM-Net [31]	VGG-19	**96.1**	82.2	67.2	81.8
GLU-Net	VGG-16	93.2	73.3	71.1	79.2
Semantic-GLU-Net	VGG-16	94.4	75.5	**78.3**	**82.8**

Table 2. PCK [%] obtained by different state-of-the-art methods on TSS [54] for the task of semantic matching.

To accommodate reflections, which do not occur in geometric correspondence scenarios, we infer the flow field on original and flipped versions of the target image and output the flow field with least horizontal average magnitude.

Results: We report results on TSS in Table 2. Our universal network obtains state-of-the-art performance on average over the three TSS groups. Moreover, individual results on FG3Dcar and PASCAL are very close to best metrics. This shows the generalization properties of our network, which is not trained on the same magnitude of semantic data. In contrast, most specialized approaches fine-tuned on PASCAL data [17]. Finally, including architectural details specifically for semantic matching, termed Semantic-GLU-Net, further improves our performance, setting a new state-of-the-art on TSS, by improving a substantial 1.0% PCK over the previous best. Interestingly, we outperform methods that use a deeper, more powerful feature backbone. Qualitative examples of our approach are shown in Figure 6.

4.3. Optical flow

Finally, we apply our network, with the same weights as previously, for the task of optical flow estimation. Here, the image pairs stem from consecutive frames of a video.

Dataset and metric: For optical flow evaluation, we use the KITTI dataset [15], which is composed of real road sequences captured by a car-mounted stereo camera rig. The 2012 set only consists of static scenes while the 2015 set is extended to dynamic scenes. For this task, we follow the standard evaluation metric, namely the Average End-Point Error (AEPE). We also use the KITTI-specific F1 metric, which represents the percentage of outliers.

Compared methods: We employ state-of-the-art PWC-

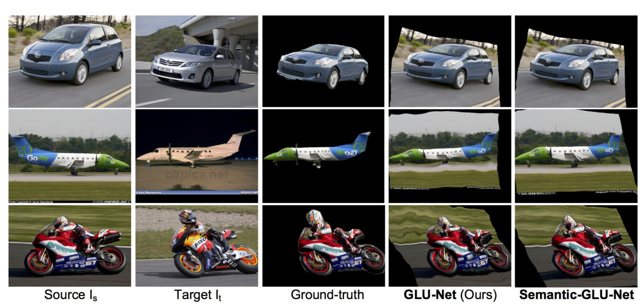


Figure 6. Qualitative examples of GLU-Net (Ours) and Semantic-GLU-Net applied to TSS images [54].

	KITTI-2012		KITTI-2015	
	AEPE-all	F1-all [%]	AEPE-all	F1-all [%]
LiteFlowNet [22]	4.00	**17.47***	10.39	**28.50**
PWC-Net [51, 52]	4.14	20.28*	10.35	33.67
DGC-Net [37]	8.50*	32.28*	14.97*	50.98*
DGC-Net†	7.96	34.35	14.33	50.35
GLU-Net	**3.34**	18.93	**9.79**	37.52

Table 3. Quantitative results on optical flow KITTI training datasets [15]. Fl-all: Percentage of outliers averaged over all pixels. Inliers are defined as $AEPE < 3$ pixels or $< 5\%$. Lower F1 and AEPE are best. * Denotes values which are computed using the trained models provided by the authors.

Net [51, 52] and LiteFlowNet [22] trained on *Flying-Chairs* [13] and *3D-things* [25]. We also compare to DGC-Net [37] (*tokyo*) and DGC-Net† (*DPED-CityScape-ADE*).

Results: Since we do not finetune our model, we only evaluate on the KITTI training sets. For fair comparison, we compare to models not finetuned on the KITTI training data. The results are shown in Table 3 and a qualitative example is illustrated in Figure 7. Our network obtains highest AEPE on both KITTI-2012 and 2015. Nevertheless, we observe that our method achieves a larger F1 on KITTI-2015 compared to approaches specifically trained and designed for optical flow. This is largely due to our self-supervised training data, which currently does not model independently moving objects or occlusions, but could be included to pursue a more purposed optical flow solution. Yet, our approach demonstrates competitive results for this challenging task, without training on any optical flow data. This clearly shows that our network can not only robustly estimate long-range matches, but also accurate small displacements.

4.4. Ablation study

Here, we perform a detailed analysis of our approach.

Local-global architecture: We first analyze the impact of global and local correlation layers in our dense correspondence framework. We compare using only local layers (Local-Net), a global layer (Global-Net) and our combination (GLOCAL-Net), presented in Figure 2. As shown in Table 4, Local-Net fails on the HP dataset, due to its inability to capture large displacements. While the Global-Net can handle large viewpoint changes, it achieves inferior accuracy compared to GLOCAL-Net, which additionaly integrates local correlations layers.

Adaptive resolution: By further adding the adaptive resolution strategy (Section 3.3), our approach (GLU-Net in Table 4) achieves a large performance gain in all metrics compared to GLOCAL-Net. This improvement is most prominent for high resolution images, *i.e*. the original HP data.

Figure 7. Visualization of the flow outputted by different methods for a KITTI-2012 image.

		Local-Net	Global-Net	GLOCAL-Net	GLU-Net (no CC, no it-R)	GLU-Net (no CC, it-R)
HP-240	AEPE	10.62	9.72	**8.77**	7.69	**7.69**
	PCK-1px [%]	35.10	41.28	**48.53**	53.83	**53.83**
	PCK-5px [%]	73.03	72.76	**78.12**	83.17	**83.17**
HP	AEPE	147.96	34.64	**31.64**	25.55	**25.09**
	PCK-1px [%]	7.41	8.86	**10.23**	35.26	**36.81**
	PCK-5px [%]	19.27	50.11	**56.73**	75.79	**77.55**

Table 4. Effect of global and local correlations as well of adaptive resolution strategy. **it-R**: iterative refinement, introduced with our adaptive resolution (Section 3.3), **CC**: cyclic-consistency [45].

		No CC	+ CC (Ours)	+ NC-Net	+ Concat-F
HP	AEPE	25.09	25.05	22.00	**21.40**
	PCK-1px [%]	36.81	**39.55**	37.62	38.49
	PCK-5px [%]	77.55	78.54	79.41	**79.50**
KITTI-2012	AEPE	3.56	**3.34**	3.80	3.85
	F1-all [%]	21.67	**18.93**	23.49	23.84
TSS	PCK [%]	78.97	79.21	82.10	**82.76**

Table 5. Effect of additional architectural details. All models are with iterative refinement. We add **CC**: cyclic-consistency [45], **NC-Net**: Neighborhood Consensus network [45], **Concat-F**: Concatenation of features of L-Net [28].

Iterative refinement: From Table 4, applying iterative refinement (**it-R**) clearly benefits accuracy for high-resolution images (HP). This further allows us to seamlessly add extra flow refinements, without incurring any additional network weights, in order to process images of high resolution.

Global correlation: Lastly, we explore design choices for the global correlation block in our architecture. As shown in Table 5, adding cyclic consistency (**CC**) [45] as a post-processing brings improvements for all datasets. Subsequently adding **NC-Net** and concatenating features of L-Net (**Concat-F**) lead to major overall gain on the HP [5] and TSS [54] datasets. However, we observe a slight degradation in accuracy, as seen on KITTI [15]. We therefore only include these components for the Semantic-GLU-Net version (Section 4.2) and not in our universal GLU-Net.

5. Conclusion

We propose a universal coarse-to-fine architecture for estimating dense flow fields from a pair of images. By carefully combining global and local correlation layers, our network effectively estimates long-range displacements while also achieving high accuracy. Crucially, we introduce an adaptive resolution strategy to counter the fixed input resolution otherwise imposed by the global correlation. Our universal GLU-Net is thoroughly evaluated for the three diverse tasks of geometric correspondences, semantic matching and optical flow. When using the same model weights, our network achieves state-of-the-art performance on all above tasks, demonstrating its universal applicability.

Acknowledgments: This work was partly supported by the ETH Zürich Fund (OK), a Huawei Technologies Oy (Finland) project, an Amazon AWS grant, and Nvidia.

References

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

[2] Alexandre Alahi, Raphael Ortiz, and Pierre Vandergheynst. Freak: Fast retina keypoint. In *CVPR*, pages 510–517. IEEE Computer Society, 2012. 2

[3] Pablo Fernndez Alcantarilla, Adrien Bartoli, and Andrew J. Davison. Kaze features. In Andrew W. Fitzgibbon, Svetlana Lazebnik, Pietro Perona, Yoichi Sato, and Cordelia Schmid, editors, *ECCV (6)*, volume 7577 of *Lecture Notes in Computer Science*, pages 214–227. Springer, 2012. 2

[4] Simon Baker, Daniel Scharstein, J. P. Lewis, Stefan Roth, Michael J. Black, and Richard Szeliski. A database and evaluation methodology for optical flow. *International Journal of Computer Vision*, 92(1):1–31, 2011. 1, 2

[5] Vassileios Balntas, Karel Lenc, Andrea Vedaldi, and Krystian Mikolajczyk. Hpatches: A benchmark and evaluation of handcrafted and learned local descriptors. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 3852–3861, 2017. 2, 6, 8

[6] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. SURF: speeded up robust features. In *Computer Vision - ECCV 2006, 9th European Conference on Computer Vision, Graz, Austria, May 7-13, 2006, Proceedings, Part I*, pages 404–417, 2006. 2

[7] Daniel J. Butler, Jonas Wulff, Garrett B. Stanley, and Michael J. Black. A naturalistic open source movie for optical flow evaluation. In *Computer Vision - ECCV 2012 - 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part VI*, pages 611–625, 2012. 2

[8] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. In *BMVC*, 2014. 4

[9] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *CoRR*, abs/1706.05587, 2017. 5

[10] Christopher Bongsoo Choy, JunYoung Gwak, Silvio Savarese, and Manmohan Krishna Chandraker. Universal correspondence network. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 2406–2414, 2016. 2

[11] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 5

[12] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *2018 IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 224–236, 2018. 2

[13] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Häusser, Caner Hazirbas, Vladimir Golkov, Patrick van der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 2758–2766, 2015. 2, 3, 5, 6, 8

[14] David A. Forsyth and Jean Ponce. *Computer Vision - A Modern Approach, Second Edition.* Pitman, 2012. 1

[15] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *I. J. Robotic Res.*, 32(11):1231–1237, 2013. 2, 7, 8

[16] Yoav HaCohen, Eli Shechtman, Dan B. Goldman, and Dani Lischinski. Non-rigid dense correspondence with applications for image enhancement. *ACM Trans. Graph.*, 30(4):70, 2011. 1

[17] Bumsub Ham, Minsu Cho, Cordelia Schmid, and Jean Ponce. Proposal flow: Semantic correspondences from object proposals. *IEEE Trans. Pattern Anal. Mach. Intell.*, 40(7):1711–1725, 2018. 1, 7

[18] Andrew Harltey and Andrew Zisserman. *Multiple view geometry in computer vision (2. ed.).* Cambridge University Press, 2006. 1

[19] Christopher G. Harris and Mike Stephens. A combined corner and edge detector. In *Proceedings of the Alvey Vision Conference, AVC 1988, Manchester, UK, September, 1988*, pages 1–6, 1988. 2

[20] Berthold K. P. Horn and Brian G. Schunck. "determining optical flow": A retrospective. *Artif. Intell.*, 59(1-2):81–87, 1993. 1, 2

[21] Shuaiyi Huang, Qiuyue Wang, Songyang Zhang, Shipeng Yan, and Xuming He. Dynamic context correspondence network for semantic alignment. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 2010–2019. IEEE, 2019. 3, 5, 7

[22] Tak-Wai Hui, Xiaoou Tang, and Chen Change Loy. Liteflownet: A lightweight convolutional neural network for optical flow estimation. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 8981–8989, 2018. 2, 3, 6, 7, 8

[23] Tak-Wai Hui, Xiaoou Tang, and Chen Change Loy. A Lightweight Optical Flow CNN - Revisiting Data Fidelity and Regularization. 2020. 2

[24] Andrey Ignatov, Nikolay Kobyshev, Radu Timofte, Kenneth Vanhoey, and Luc Van Gool. Dslr-quality photos on mobile devices with deep convolutional networks. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 3297–3305, 2017. 5

[25] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 1647–1655. IEEE Computer Society, 2017. 2, 6, 8

[26] Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 2017–2025, 2015. 2

[27] Joel Janai, Fatma Güney, Aseem Behl, and Andreas Geiger. Computer vision for autonomous vehicles: Problems, datasets and state-of-the-art. *CoRR*, abs/1704.05519, 2017. 1

[28] Sangryul Jeon, Seungryong Kim, Dongbo Min, and Kwanghoon Sohn. PARN: pyramidal affine regression networks for dense semantic correspondence. In *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part VI*, pages 355–371, 2018. 2, 3, 4, 7, 8

[29] Seungryong Kim, Stephen Lin, Sangryul Jeon, Dongbo Min, and Kwanghoon Sohn. Recurrent transformer networks for semantic correspondence. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada.*, pages 6129–6139, 2018. 2, 7

[30] Seungryong Kim, Dongbo Min, Bumsub Ham, Stephen Lin, and Kwanghoon Sohn. FCSS: fully convolutional self-similarity for dense semantic correspondence. *IEEE Trans. Pattern Anal. Mach. Intell.*, 41(3):581–595, 2019. 2

[31] Seungryong Kim, Dongbo Min, Somi Jeong, Sunok Kim, Sangryul Jeon, and Kwanghoon Sohn. Semantic attribute matching networks. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 12339–12348, 2019. 2, 3, 7

[32] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States.*, pages 1106–1114, 2012. 4

[33] Zakaria Laskar, Iaroslav Melekhov, Hamed R. Tavakoli, Juha Ylioinas, and Juho Kannala. Geometric image correspondence verification by dense pixel matching. *CoRR*, abs/1904.06882, 2019. 2

[34] Stefan Leutenegger, Margarita Chli, and Roland Siegwart. BRISK: binary robust invariant scalable keypoints. In *IEEE International Conference on Computer Vision, ICCV 2011, Barcelona, Spain, November 6-13, 2011*, pages 2548–2555, 2011. 2

[35] Ce Liu, Jenny Yuen, and Antonio Torralba. SIFT flow: Dense correspondence across scenes and its applications. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(5):978–994, 2011. 1, 2

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

[37] Iaroslav Melekhov, Aleksei Tiulpin, Torsten Sattler, Marc Pollefeys, Esa Rahtu, and Juho Kannala. DGC-Net: Dense geometric correspondence network. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2019. 2, 3, 4, 5, 6, 7, 8

[38] Vinod Nair and Geoffrey E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10), June 21-24, 2010, Haifa, Israel*, pages 807–814, 2010. 5

[39] Simon Niklaus. A reimplementation of LiteFlowNet using PyTorch. `https://github.com/sniklaus/pytorch-liteflownet`, 2019. 6

[40] Yuki Ono, Eduard Trulls, Pascal Fua, and Kwang Moo Yi. Lf-net: Learning local features from images. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada*, pages 6237–6247, 2018. 2

[41] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in PyTorch. In *NIPS Autodiff Workshop*, 2017. 6

[42] Anurag Ranjan and Michael J. Black. Optical flow estimation using a spatial pyramid network. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 2720–2729, 2017. 2

[43] Ignacio Rocco, Relja Arandjelovic, and Josef Sivic. Convolutional neural network architecture for geometric matching. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 39–48, 2017. 2, 5

[44] Ignacio Rocco, Relja Arandjelovic, and Josef Sivic. End-to-end weakly-supervised semantic alignment. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 6917–6925, 2018. 2, 7

[45] Ignacio Rocco, Mircea Cimpoi, Relja Arandjelovic, Akihiko Torii, Tomás Pajdla, and Josef Sivic. Neighbourhood consensus networks. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada.*, pages 1658–1669, 2018. 2, 5, 7, 8

[46] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary R. Bradski. ORB: an efficient alternative to SIFT or SURF. In *IEEE International Conference on Computer Vision, ICCV 2011, Barcelona, Spain, November 6-13, 2011*, pages 2564–2571, 2011. 2

[47] Johannes L. Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 4104–4113, 2016. 1, 6

[48] Johannes L. Schönberger, Marc Pollefeys, Andreas Geiger, and Torsten Sattler. Semantic visual localization. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 6896–6906. IEEE Computer Society, 2018. 1

[49] Thomas Schöps, Johannes L. Schönberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 2538–2547, 2017. 2, 6, 7

[50] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. *CoRR*, abs/1406.2199, 2014. 1

[51] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 8934–8943, 2018. 2, 3, 6, 7, 8

[52] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Models matter, so does training: An empirical study of cnns for optical flow estimation. *IEEE transactions on pattern analysis and machine intelligence*, 2019. 2, 6, 8

[53] Hajime Taira, Masatoshi Okutomi, Torsten Sattler, Mircea Cimpoi, Marc Pollefeys, Josef Sivic, Tomás Pajdla, and Akihiko Torii. Inloc: Indoor visual localization with dense matching and view synthesis. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 7199–7209. IEEE Computer Society, 2018. 1

[54] Tatsunori Taniai, Sudipta N. Sinha, and Yoichi Sato. Joint recovery of dense correspondence and cosegmentation in two images. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 4246–4255, 2016. 2, 7, 8

[55] Prune Truong. GLU-Net: Github project page. https://github.com/PruneTruong/GLU-Net, 2020. 2

[56] Prune Truong, Stefanos Apostolopoulos, Agata Mosinska, Samuel Stucky, Carlos Ciller, and Sandro De Zanet. Glampoints: Greedily learned accurate match points. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10731–10740, 2019. 2

[57] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Machine Learning, Proceedings of the Twenty-Fifth International Conference (ICML 2008), Helsinki, Finland, June 5-9, 2008*, pages 1096–1103, 2008. 2

[58] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ADE20K dataset. *Int. J. Comput. Vis.*, 127(3):302–321, 2019. 5