From Depth to Flow - Learning Uncertainty Aware Stereo Scene Flow from Partial Synthetic Labels

1st Felix Neumann
Siemens AG
Garching, Germany
neumann.felix@siemens.com

2nd Frederik Deroo
Siemens AG
Garching, Germany
frederik.deroo@siemens.com

3rd Georg von Wichert
Siemens AG
Garching, Germany
wichert.georg@siemens.com

4th Darius Burschka
Technical University of Munich
Garching, Germany
burschka@tum.de

***Abstract*—Estimating both the 3D structure and 3D dynamics of a scene is an important task for autonomous vehicles and mobility systems. Deep learning based methods have shown strong performance on this problem, but are only reliable in their training domain and often do not reason about uncertainty. Training data that contains information about the detailed dynamics of a scene is often difficult and costly to acquire, thus limiting the applicability of such methods. To address these issues, we propose an uncertainty-aware multi-task neural network that jointly estimates disparity, optical flow and 3D scene flow from stereo image pairs in a single model with shared weights for all tasks. Furthermore, we investigate how labels of individual subtasks, e.g. disparity, can be used in combination with self-supervised losses to improve the performance of other subtasks, such as optical flow. We show that a unified model such as ours can leverage these supervision types synergistically, to transfer knowledge even from simpler tasks to more challenging ones. Additionally, we propose the first scene flow approach that estimates uncertainties as variances and multivariate covariance matrices from the cost volume of each respective task and propagates them analytically to the pixel-wise output without any further learned regression. We evaluate the domain adaptability and pixel-wise uncertainty estimations of our model on both synthetic and real datasets, including the KITTI scene flow benchmark, on which our model outperforms prior self-supervised and semi-supervised methods, while estimating representative uncertainties for all tasks.**

***Index Terms*—Scene flow, stereo, uncertainty**

I. INTRODUCTION

Understanding the 3D structure and 3D dynamics of its environment is a crucial task for any vision based autonomous mobile system, because it serves as the foundation for identifying obstacles, self-localization, and other tasks. One approach to address this problem is scene flow estimation, which models the dynamics of a scene as a point-wise 3D motion field [1]. This can be broken down into the subtasks of depth and optical flow estimation from images, where scene flow can be inferred from a depth image as well as the depth change and optical flow between two consecutive frames. These tasks are similar in nature, since they can be expressed as dense correspondence estimation problems, which form the basis of scene flow estimation in our approach. Estimating dense feature correspondences in a pair of stereo images is a subproblem of estimating optical flow, where the correspondences are constrained to only horizontal displacements in the image. Thus, we would intuitively expect that a generic correspondence estimation network can learn from both stereo matching and optical flow training data. However, as indicated by task transfer experiments by Xu *et al.* [2] and further analyzed in our experiments, the transfer from stereo to optical flow matching is not trivial. We investigate this issue and show how self-supervised learning approaches can bridge this gap to transfer knowledge from dense stereo matching to not just 2D optical flow matching, but also to 3D scene flow estimation from partially labeled datasets. A visualization of the resulting 3D structure and scene flow estimated by our network is shown in Fig. 1.

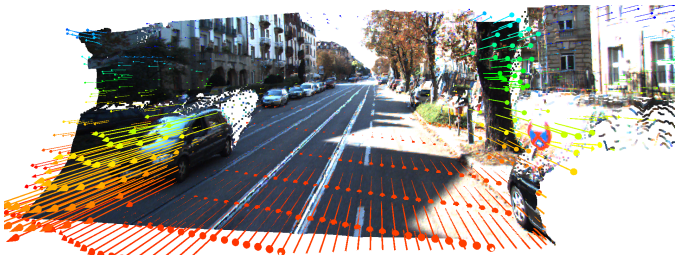


Fig. 1: Visualization of 3D scene flow estimates by our proposed approach. We show a sparse sample of the dense scene flow estimates.

Deep learning based methods have shown very strong results on scene flow estimation and its subtasks, as is evident from the leaderboards of several public benchmarks [3]–[7]. In order to reduce the complexity of the scene flow task, some approaches use a collection of multiple task-specific neural networks to estimate structure and dynamics separately [8], [9]. These methods are parameter inefficient and do not exploit potential synergies between the individual matching tasks, since every task specific network contains an independent feature encoder. On the other hand, some approaches estimate the scene structure and flow jointly [10], [11] using recurrent update operators. A mixture of self-supervision and labeled scene flow data is used for training and fine-tuning in such methods. However, there are few datasets with full scene flow

This research has received funding from the Federal Ministry for Economic Affairs and Climate Action under grant agreements 19I21039A

annotations [5], [7], [12], [13], out of which only KITTI contains real, labeled data. Generally, labelling 3D scene flow data is much more difficult than labeling subcomponents such as depth, which can be done using range sensors such as LiDAR. This lack of real-world labels can pose a problem for the application of deep learning based approaches in safety-critical systems, since neural networks only perform reliably in the data domain covered during training. A further problem for applications in safety-critical systems is that most deep learning based methods for this task do not reason about the uncertainty of their estimations.

To address these issues, we propose a neural network architecture based on Unimatch [2] that efficiently estimates optical flow, disparity, and disparity change using shared weights for all tasks. Our method can learn from labels for individual subtasks of scene flow. We show experimentally that this improves the accuracy of multiple scene flow subtasks on a new data domain using labels of only a single subtask. Notably, we show how disparity (or depth) labels, which are more easily attained than optical flow or scene flow labels, can be used to improve the accuracy of flow estimations, despite flow matching being a much more complex task than disparity estimation. Our proposed network outputs disparity, disparity change, optical flow, and their respective uncertainty maps from a shared set of parameters.

The specific contributions of this paper are:

- We extend the Unimatch framework to a single multi-task model that can estimate scene flow in the form of optical flow, disparity, and disparity change using a single set of weights.
- We furthermore show how established self-supervision techniques can be leveraged to improve all three outputs of the network with optional partial supervision of a network output, thanks to the unified architecture. We are the first to demonstrate that self-supervised losses act as a catalyst to enable the transfer of knowledge from disparity to flow estimation.
- We propose a novel method to output uncertainties in the form of pixel-wise multivariate Gaussians for image correspondence tasks, which propagates the underlying matching uncertainties from cost volumes to the network output without adjusting them using learned components, as is commonly done in other works.

We test and validate our contributions through experimental studies on synthetic and real datasets [4], [12]–[14].

The remainder of this work is structured as follows: Section II reviews existing approaches to self-supervised learning and uncertainty estimation for dense correspondence tasks. We introduce our proposed approach and training scheme in Section III. Section IV presents and discusses experimental results and Section V draws conclusions and outlines promising future work.

II. RELATED WORK

A. Self-supervised Geometric Learning

Self-supervision has shown great promise for geometric learning tasks that are challenging to annotate. This type of supervision for geometric tasks originates from depth estimation methods [15]–[19], which commonly employ a mixture of photometric reconstruction losses and smoothness reguarization during training. We take inspiration from the self-supervised losses proposed by these methods to train our approach. Other works [20], [21] have applied similar losses and additional data augmentation for occluded areas [22] to the task of self-supervised optical flow estimation. Finally, [11], [23]–[27] combine these two classes of self-supervised networks to learn scene flow from unlabeled data. Depth is inferred from either monocular cameras or stereo camera pairs for scene flow estimation. However, these methods do not reason about the uncertainty of their outputs and can be parameter inefficient when using separate networks for depth and optical flow estimation. Furthermore, in contrast to these methods we show that self-supervision has a synergistic effect with partial supervision in a multi-task model, improving performance across multiple tasks, even the task that is partially supervised, more substantially than either individual type of supervision.

B. Uncertainty Aware Geometric Models

Reliable uncertainty estimation is a crucial element for deploying learning based methods in the real world. There are several approaches to estimate confidences and uncertainties for single image depth estimation [17], [28] or multi-view stereo depth estimation [29], however they all directly predict variances or confidences at the output layer of the network, which is prone to shifts in the data domain. More closely related to our work is CFNet [30], which uses the variance of a disparity cost volume to iteratively constrain the search range for stereo matching. Estimation of uncertainties for dynamics such as optical and scene flow using deep learning models is rarely done and when they are estimated, it is in the form of scalar confidences [31]. In contrast, we argue that the matching distribution in cost volumes is a strong indicator of the networks uncertainties. Extending the approach of CFNet, we are the first to estimate the uncertainty of our network outputs as multivariate Gaussian distributions in the context of scene flow estimation. These Gaussians are calculated directly from the cost volume and propagated to the network output to quantify the uncertainty of each task.

C. Multi-Task Scene Flow Models

The basis of our approach is formed by the work of Xu *et al.* [2], who propose a unified architecture named Unimatch for optical flow, disparity, and posed monocular depth estimation. While they demonstrate that training on the optical flow task can serve as a good initialization to train depth/disparity estimators, they do not provide a model that is capable of estimating all tasks jointly. They furthermore show poor transferability from the depth/disparity task to the optical flow estimation task, which we overcome in this work.

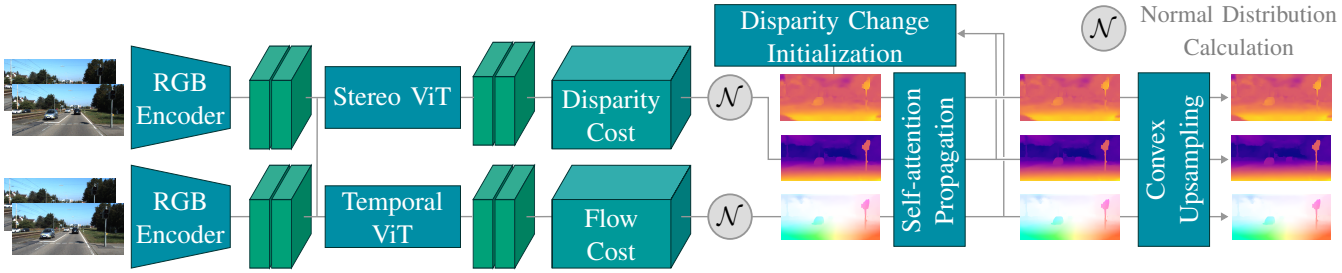


Fig. 2: Overview of our network architecture: The image feature extraction of Unimatch is expanded to include information from temporally and spatially adjacent frames using cross-attention with shared weights in the two ViTs. Up to two cost volumes are constructed, one for optical flow and one for disparity. From each of these cost volumes, initial predictions are extracted as Gaussian distributions, which are propagated using self-attention propagation. Disparity change is initialized from the propagated disparity and flow maps, and then passed through the self-attention propagation module again to account for occlusions resulting from warping by the optical flow. The propagated estimates are all processed by the convex upsampling layer to produce final, pixel-wise outputs.

We build on this work and extend the unified architecture to a single unified model that uses one set of parameters for disparity, disparity change, and optical flow estimation. We exclude the monocular depth estimation component, since initial experimentation showed that the strong reliance on the cost volume causes depth estimation failures in the case of static frames and inaccurate depth values for dynamic objects in the image. Other works such as Manydepth [18] and the works of Guizilini *et al.* [10], [19] have shown how cost volumes can be integrated into monocular depth estimation in such environments. These approaches require additional, task specific network components, which contradicts the unified model approach and increases the reliance on learned biases to cope with degenerate cases, instead of finding correspondences between images.

The works of Hur *et al.* [23], [24] and Guizilini *et al.* [10] have proposed self-supervised and semi-supervised approaches to the task of monocular scene flow estimation. Their approaches employ multi-task models to estimate depth, optical flow, and scene flow. Bendig *et al.* [27] recently proposed a self-supervised training scheme for scene flow estimation from stereo sequences. We take inspiration from their self-supervised loss components, but expand the amount of available training data for semi-supervised learning by including supervision on subtasks of our scene flow network. We show that this partial supervision in combination with self-supervision is much more effective than either type of supervision individually and achieve synergistic effects thanks to our proposed modifications to the Unimatch architecture.

There are several other multi-task approaches to jointly learning the subtasks of scene flow [9], [26], [32]–[34] that employ multiple task-specific models. In contrast to all previous approaches, we propose the first multi-task, image-based scene flow estimation network that explicitly addresses uncertainty estimation.

III. APPROACH

We propose a multi-task neural network for scene flow estimation from two consecutive stereo frames comprised of images $\mathbf{I}_l^t, \mathbf{I}_r^t, \mathbf{I}_l^{t'}, \mathbf{I}_r^{t'} \in \mathbb{R}^{H \times W \times 3}$ in the left and right camera $\{l, r\}$ at two consecutive points in time $\{t, t'\}$ with a given intrinsic matrix $\mathbf{K} \in \mathbb{R}^{3\times3}$. Our goal is to estimate the structure of the scene as disparity maps $\mathbf{D}_l^t, \mathbf{D}_r^t, \mathbf{D}_l^{t'}, \mathbf{D}_r^{t'} \in \mathbb{R}^{H \times W}$. The dynamics of the scene are estimated as optical flow maps $\mathbf{F}_l^t, \mathbf{F}_r^t, \mathbf{F}_l^{t'}, \mathbf{F}_r^{t'} \in \mathbb{R}^{H \times W \times 2}$, where $\mathbf{F}^t, \mathbf{F}^{t'}$ represent the motion of pixels from $\mathbf{I}^t$ to $\mathbf{I}^{t'}$ and from $\mathbf{I}^{t'}$ to $\mathbf{I}^t$, in a camera frame, respectively. To lift optical flow to scene flow, we estimate disparity change maps $\Delta\mathbf{D}_l^{t,t'}, \Delta\mathbf{D}_r^{t,t'}, \Delta\mathbf{D}_l^{t',t}, \Delta\mathbf{D}_r^{t',t} \in \mathbb{R}^{H \times W}$ that model the change in disparity in a given camera frame, e.g., $\Delta\mathbf{D}_l^{t,t'}$ represents the change in disparity from time t to t' in the left camera frame. Together, disparity $\mathbf{D}^t$, the disparity change $\Delta\mathbf{D}^{t,t'}$ and the optical flow $\mathbf{F}^t$ fully parameterize the forward scene flow in an image from t to t'. Estimating only disparity at t and t' and optical flow is not enough, as it does not account for possible occlusions or motions that leave the field of view of the camera between consecutive frames. We model all estimates as Gaussian distributions, where the disparity and optical flow maps defined above represent the means of the distributions. We output the respective variance maps $\mathbf{\Sigma}_D \in \mathbb{R}^{H \times W}$, $\mathbf{\Sigma}_{\Delta D} \in \mathbb{R}^{H \times W}$, and covariance maps $\mathbf{\Sigma}_F \in \mathbb{R}^{H \times W \times 2 \times 2}$ for disparity, disparity change, and optical flow predictions.

Our proposed network determines all estimates for each input frame such that we obtain both forward and backward optical flow and disparity change in both the left and the right camera, as well as left and right disparity at each time frame. Using these outputs, our approach can be trained through supervision if ground truth labels are available and through self-supervision using multiple photometric and consistency losses. An overview of our approach to obtain these outputs is depicted in Fig. 2. Our network architecture is based on Unimatch without refinement. We exclude refinement from this

work, because it is a task-specific learned regression, which has been shown to be effective by several other works [2], [8], [10], [35], [36], at the cost of additional computation. Instead, we focus on developing a lightweight, parameter efficient estimator that can solve multiple tasks with a single set of weights. The main modifications to the network architecture and the training methodology of Unimatch are described in the following.

A. Multi-task Feature Extraction

We follow Unimatch for feature extraction. A shared convolutional neural network encoder extracts feature maps $\boldsymbol{\mathcal{F}} \in \mathbb{R}^{\frac{H}{8} \times \frac{W}{8} \times 128}$ from all input images. These feature maps are further processed by a vision transformer (ViT) consisting of six transformer blocks, each containing a self-attention and a cross-attention layer. The cross-attention layers of the ViT are applied in two configurations that share the same weights. Temporally adjacent feature maps $\boldsymbol{\mathcal{F}}^t$ and $\boldsymbol{\mathcal{F}}^{t'}$ use 2D shifted window attention [37], while spatially adjacent stereo feature maps $\boldsymbol{\mathcal{F}}_l$ and $\boldsymbol{\mathcal{F}}_r$ use horizontal 1D attention along pixel rows.

To construct the downstream cost volumes, the extracted stereo feature maps are used for the disparity cost volume and the temporally adjacent feature maps are used for the optical flow cost volume. This methodology allows our method to estimate only optical flow or only disparity from a pair of input images by selecting the respective ViT configuration without changing the model weights.

B. Uncertainty Estimation from Cost Volumes

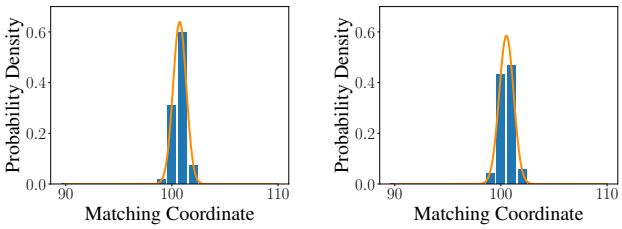
As in Unimatch, we generate task-specific global cost volumes for initial estimates of disparity and optical flow. The optical flow cost volume $\mathbf{C}_F \in \mathbb{R}^{\frac{H}{8} \times \frac{W}{8} \times \frac{H}{8} \times \frac{W}{8}}$ is generated from the feature maps output by the sequential configuration of the ViT, while the disparity cost volume $\mathbf{C}_D \in \mathbb{R}^{\frac{H}{8} \times \frac{W}{8} \times \frac{W}{8}}$ is calculated from the feature maps output by the stereo configuration of the ViT. The normalized matching probability is calculated using the cosine similarity between feature vectors, followed by a softmax activation.

These cost volumes can be seen as discrete probability distributions over matching candidates. We observe that they are close to normally distributed and unimodal, as shown in Fig. 3. For a matching distribution $\mathbf{M}_T$ with task $T \in \{F, D\}$ and corresponding pixel coordinates $\mathbf{G}_T$, the mean match $\bar{\mathbf{G}}_T$ and element i, j of the (co)variance $\boldsymbol{\Sigma}_{T,ij}$ over N matching candidates are given by:

$$\bar{\mathbf{G}}_T = \mathbf{M}_T \mathbf{G}_T \text{ and}$$

$$\boldsymbol{\Sigma}_{T,ij} = \sum_{k=1}^{N} (\mathbf{G}_{T,ik} - \bar{\mathbf{G}}_{T,i})^T \mathbf{M}_{T,k} (\mathbf{G}_{T,jk} - \bar{\mathbf{G}}_{T,j}).$$

The task of disparity estimation only has a single variance term, while the optical flow matching distribution yields a 2×2 covariance matrix for each entry in the feature map.



(a) Cost volume distribution (b) Distribution after upsampling

Fig. 3: Disparity matching distributions from the cost volume (a) and after the propagation and upsampling layers (b). The discrete probability density function is shown in blue, the calculated and analytically propagated normal distribution is visualized in orange. Matching coordinates show a zoomed in region before upsampling and are not adjusted for upsampling in (b) to ensure comparability.

C. Scene Flow Estimation

The 3D scene flow between two consecutive images can be parameterized by the optical flow, disparity in the first frame and change in disparity for each pixel. Disparity change can also be understood as the change in depth over time. Change in disparity is trivial to calculate in non-occluded areas by warping the disparity map $\mathbf{D}^{t'}$ to the same image frame as $\mathbf{D}^t$ using the optical flow $\mathbf{F}^t$ to produce the warped disparity $\mathbf{D}^{t',t}$. Unimatch proposed a component called self-attention-propagation, which leverages feature similarity to fill unmatched areas in disparity and optical flow maps. We leverage this component to also infer disparity change in occluded regions, by warping $\mathbf{D}^{t'}$ by $\mathbf{F}^t$ after self-attention propagation and calculate the difference $\Delta\mathbf{D}^{t,t'}_{\text{init}} = \mathbf{D}^{t',t} - \mathbf{D}^t$. However, this warping step produces unmatched regions in the disparity change map again. Therefore, we then apply the self-attention propagation block again to $\Delta\mathbf{D}^{t,t'}_{\text{init}}$ to account for these gaps. We initialize the disparity change uncertainty as the sum of the uncertainty of disparity at time t and the uncertainty of the disparity map warped from time t' to t: $\boldsymbol{\Sigma}^{t,t'}_{\Delta D} = \boldsymbol{\Sigma}^{t',t}_D + \boldsymbol{\Sigma}^t_D$. Here $\boldsymbol{\Sigma}^{t',t}_D$ is the variance map $\boldsymbol{\Sigma}^{t'}_D$ warped by the optical flow $\mathbf{F}^t$.

D. Uncertainty Propagation and Upsampling

The outputs of the cost volumes are processed by two components, self-attention-propagation and convex upsampling, which both output weighted sums of their respective inputs. The weights of the self-attention propagation block are predicted from global feature similarity and the weights of the convex upsampling layer are predicted directly using a convolutional layer. This allows us to propagate the variances and covariances calculated from the cost volume analytically, instead of regressing them using learned components. A weighted sum of Gaussians does not strictly result in another Gaussian distribution. However, we observe that after each propagation layer, the probability distributions remain approximately Gaussian as shown in Fig. 3. Therefore, we propagate means and (co)variances analytically. For N terms

in the weighted sum, w_k is the predicted weight associated with the distribution $\mathcal{N}_k(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$, with $k \in N$. Here the means $\boldsymbol{\mu}$ are the elements of the matrix $\bar{\mathbf{G}}$. With slight abuse of notation for generality, the propagation of the mean and (co)variance are given by:

$$\hat{\boldsymbol{\mu}} = \sum_{k=1}^{N} w_k \boldsymbol{\mu}_k \text{ and}$$

$$\hat{\boldsymbol{\Sigma}}_{ij} = \sum_{k=1}^{N} w_k (\boldsymbol{\Sigma}_{k,ij} + \boldsymbol{\mu}_{k,i}\boldsymbol{\mu}_{k,j}) - \sum_{k=1}^{N} w_k \boldsymbol{\mu}_{k,i} \sum_{k=1}^{N} w_k \boldsymbol{\mu}_{k,j}.$$

The estimated pixel-wise (co)variances represent heteroscedastic aleatoric uncertainty estimates [28].

E. Supervised Losses

For experiments without supervision of the estimated uncertainties, we use the same supervised losses as Unimatch. Namely, these are the L_1 loss for optical flow, the smooth L_1 loss and an inverse depth gradient loss for disparity, and the L_1 loss for disparity change.

To supervise uncertainty, we modify the L_1 based disparity change/disparity losses and optical flow loss as

$$\mathcal{L}_D(\hat{\mathbf{D}}, \mathbf{D}, \boldsymbol{\Sigma}_D) = \frac{|\hat{\mathbf{D}} - \mathbf{D}|}{\boldsymbol{\Sigma}_D} + \lambda \log(\boldsymbol{\Sigma}_D) \text{ and}$$

$$\mathcal{L}_F(\hat{\mathbf{F}}, \mathbf{F}, \boldsymbol{\Sigma}_F) = (\hat{\mathbf{F}} - \mathbf{F})^\top \boldsymbol{\Sigma}_F^{-1} (\hat{\mathbf{F}} - \mathbf{F}) + \lambda \log(|\boldsymbol{\Sigma}_F|),$$

respectively. These losses are comprised of the 1D and 2D Mahalanobis distance and a regularization term to regularize the magnitude of the uncertainty. The disparity change loss $\mathcal{L}_{\Delta D}$ is calculated the same way as $\mathcal{L}_D$, but with disparity change maps and uncertainties instead. We set the regularization factor λ to 0.02 after hyperparameter tuning.

For supervised training, the total loss is calculated as

$$\mathcal{L}_{\mathsf{sup}} = \lambda_F \mathcal{L}_F + \lambda_D \mathcal{L}_D + \lambda_{\Delta D} \mathcal{L}_{\Delta D}$$

where λ_F, λ_D and $\lambda_{\Delta D}$ are respectively set to 3, 1, and 3 after hyperparameter tuning.

F. Self-supervised Losses

In addition to the supervised losses, we also employ self-supervision for all estimations. The self-supervision loss terms consist of a set of photometric errors based on differentiable grid sampling [38], as well as consistency and smoothness terms for regularization, which we will describe in the following in more detail. In general, we mask all losses that involve multiple frames using occlusion masks derived from a forward-backward optical flow [21] and equivalent left-right disparity consistency check for temporally and spatially adjacent frames, respectively.

Photometric losses: We warp the image $\mathbf{I}^{t'}$ towards the image $\mathbf{I}^t$ using the estimated forward optical flow $\mathbf{F}^t$ to produce the warped image $\mathbf{I}^{t',t}$. Similarly, we warp right images $\mathbf{I}_r$ towards left images $\mathbf{I}_l$ using the estimated disparity map $\mathbf{D}_l$ to produce the warped image $\mathbf{I}_{r,l}$. The warping steps above are conducted for all estimations in all input image frames, such that the forward and backward estimated optical flow in the left and right images, as well as the left and right disparities at time t and t' are supervised using the photometric loss. To supervise points that lie past the image border in one of the frames due to non overlapping fields of view, we train on image crops, but consider the entire original image during grid sampling.

We follow Godard *et al.* [15] and use a weighted average of the photometric loss [39] and structural similarity index (SSIM) [40] with a window size of five pixels

$$\mathcal{L}_p(\mathbf{I}^t, \mathbf{I}^{t',t}) = \alpha \frac{1 - \text{SSIM}(\mathbf{I}^t, \mathbf{I}^{t',t})}{2} + (1 - \alpha)||\mathbf{I}^t - \mathbf{I}^{t',t}||_1$$

with weight $\alpha = 0.85$.

Smoothness losses: As is common practice, we use edge-aware smoothness losses [15] as regularization. These terms encourage similar estimations in textureless image regions and allow discontinuities in regions with high image gradients. We apply edge aware smoothness to stereo disparity maps, as well as the individual x and y components of the optical flow estimates. All inputs to this loss term are mean normalized to discourage shrinking of the estimations [41].

Consistency losses: Since we output the optical flow, disparity, and disparity change between all image pairs, several consistency constraints can provide additional regularization.

For optical flow estimations, we maximize the forward-backward consistency by minimizing the sum of the forward flow $\mathbf{F}^t$ and the backwards warped backward flow $\mathbf{F}^{t',t}$. We apply a similar constraint on the left disparity $\mathbf{D}_l$ and the left warped right disparity $\mathbf{D}_{r,l}$, but minimize their L_1 difference instead of their sum. As with the photometric losses, the consistency losses are applied to all model outputs in all four camera frames.

IV. EXPERIMENTS AND RESULTS

To evaluate the performance and especially the transferability of knowledge between different tasks thanks to our proposed multi-task architecture, we evaluate on both synthetic and real datasets. FlyingThings3D [12] is used for ablations of our model architecture and pretraining for other tasks. Initial experiments regarding cross-task knowledge transfer and uncertainty estimation are conducted on the synthetic VKITTI2 dataset [13] and use the models pretrained on FlyingThings3D. Final evaluations and comparisons to other state-of-the-art (SOTA) models are carried out on the real-world KITTI [4], [5] scene flow benchmark, where models are trained on a collection of synthetic data [12], [13], [42] and self-supervised on the KITTI dataset.

A. Datasets

Sceneflow [12] contains three synthetic datasets; FlyingThings3D, Driving, and Monkaa. FlyingThings3D contains dynamic, synthetic objects from the ShapeNet [43] dataset with random textures, Driving is set in an automotive context and Monkaa contains assets from an open source Blender short film. Labels for all datasets include disparity, optical flow, and

disparity change. We only use FlyingThings3D for ablation studies and use the Driving dataset during the final training of our network for benchmarking. For ablations, we follow Unimatch [2] and split 1024 samples from the training set of FlyingThings3D for validation, and test on the official validation split.

VKITTI2 [13] is a photorealistic, synthetic automotive dataset that reconstructs five sequences from the KITTI [4] dataset. Labels include dense depth, optical flow, scene flow maps, and camera extrinsics. We convert the depth maps to disparity maps using the camera intrinsics and stereo baseline. Disparity change is calculated from the Z component of the scene flow labels. For ablations, we use scenes 1, 6, and 20 for training, scene 18 for validation and scene 2 for testing. For final evaluations we use the entire dataset for training. We use this dataset for initial ablations regarding domain transfer and uncertainty estimation of our approach, since it contains dense, perfect ground truth labels.

KITTI [4] is an automotive dataset that is used for benchmarking optical flow, disparity, depth, and scene flow. It contains sparse annotations for all afforementioned tasks. We use the Eigen split for self-supervised trainings, excluding the training and test frames of the KITTI scene flow benchmark. For testing, we use the 200 annotated training frames from the KITTI scene flow benchmark [5]. We do not use any labels from the KITTI scene flow dataset during training.

Tartanair [42] is a synthetic dataset for simultaneous localization and mapping. It consists of stereo camera data including annotated camera poses, depth and optical flow. We use this dataset for pretraining our model after ablation studies for final evaluation on the KITTI benchmark.

B. Evaluation Metrics

The End-point-error (**Epe**) measures the euclidean distance in pixels between the end points of estimated disparity or optical flow vectors and the corresponding ground-truth vectors. It is calculated as $||\mathbf{F} - \hat{\mathbf{F}}||$ and $||\mathbf{D} - \hat{\mathbf{D}}||$, respectively.

We furthermore use the outlier metrics for disparity, optical flow and scene flow from the KITTI dataset to evaluate. The **D1**, **D2**, **Fl**, and **SF** outlier metrics are the fraction of pixels that are incorrectly estimated in the disparity map at time t, disparity map at time t', the optical flow map between them and the entire scene flow estimate, respectively. Disparity or optical flow estimates for a pixel are classified as correct if their Epe is less than three pixels, or less than five percent of the magnitude of the ground truth vector. The scene flow estimate for a pixel is classified as correct if both disparity estimates and the optical flow between them are classified as correct.

C. Experimental Protocol

All models are implemented using Pytorch [44]. If multiple datasets are used for training, they are randomly sampled to create mixed batches. Network hyperparameters are identical to Unimatch [2] with a channel size of 128 and no refinement steps. Forward and backward flows and left and right disparities are efficiently estimated by transposing the respective cost volume. The entire model has 5.4 million trainable parameters and achieves an inference time of 63 milliseconds, thus achieving a frame rate of 15 FPS to predict disparity, optical flow and disparity change maps for all four cameras in a stereo image pair. Inference time is measured on a Nvidia RTX 6000 Ada GPU. We use the AdamW [45] optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.999$ and a one cycle learning rate policy [46] with a learning rate of $4\mathrm{e}{-4}$ and cosine annealing. All ablation trainings run for $2\mathrm{e}5$ iterations with a batch size of four on a Nvidia RTX 6000 Ada GPU. For our final evaluation we pretrain on VKITTI2 [13], Tartanair [42] and all of Sceneflow [12] for $5\mathrm{e}5$ iterations with a batch size of eight on four GPUs. We continue training using mixed supervision on Driving, VKITTI2, and self-supervision on KITTI for $5\mathrm{e}4$ iterations with a learning rate of $2\mathrm{e}{-5}$ and a batch size of four. The final training takes approximately 24 hours to complete.

D. Ablations

We study the ability of our multi-task network to benefit from individual task supervision and report the results in Table I. We show that multi-task training is feasible and beneficial for transferring knowledge to unlabeled or only partially labeled domains.

We firstly verify that our choice of a multi-task model with shared weights does not harm model performance. For this, we train separate models on only the optical flow and the disparity estimation task, and one model on both tasks simultaneously using the FlyingThings3D dataset. We evaluate on both FlyingThings3D and VKITTI2. The results of this are reported in the first three rows of the table. It is evident that transferring knowledge from the optical flow task to the disparity estimation task is easier than transferring from disparity estimation to optical flow estimation. This is shown in row two by the poor performance on the task of optical flow estimation by the model trained only on disparity estimation, and the comparatively good performance on the task of disparity estimation by the model trained on optical flow estimation in row one. Introducing multi-task training in row three maintains the performance of optical flow estimation both on in-domain (evaluations and training on FlyingThings3D) and out-of-domain data (evaluations on VKITTI2, with training on FlyingThings3D), while causing a minor reduction in the quality of disparity estimation. We attribute this to the model learning a task-specific bias when it is only trained on disparity estimation. This bias only appears for disparity estimation and not for optical flow estimation. This is reasonable, as disparity estimation is a strict subset of the task of optical flow estimation, thus any bias that is learned during optical flow estimation likely also benefits disparity estimation.

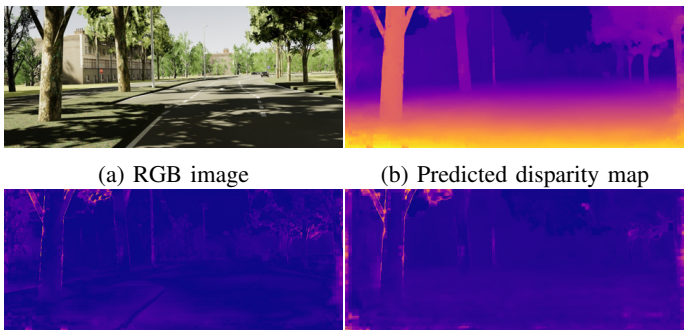
The true strength of the proposed multi-task model comes into play when considering additional data domains, for which training data may be limited. In rows four to six of the table, we investigate how the task transfer capability of the model

TABLE I: Ablation study results on multi-task learning and domain transfer. We train models on different task combinations using supervision and self-supervision on FlyingThings3D (F) and VKITTI2 (V).

Supervised		Self-supervised		FlyingThings3D (final)				VKITTI2			
Flow	Disp	Flow	Disp	Flow Epe	Disp. Epe	Fl	D1	Flow Epe	Disp. Epe	Fl	D1
F	-	-	-	**13.00**	5.84	12.69	11.00	6.06	4.46	**26.07**	56.82
-	F	-	-	77.39	**4.71**	98.89	**3.46**	44.11	**3.63**	98.08	**50.16**
F	F	-	-	**13.00**	4.76	**12.64**	4.10	**6.01**	3.81	26.98	50.68
F+V	F	-	-	13.20	**4.70**	13.02	**3.92**	1.38	3.20	9.21	34.80
F	F+V	-	-	**13.10**	4.74	**12.86**	4.26	6.91	1.82	33.60	14.61
F+V	F+V	-	-	13.25	4.78	13.71	4.37	**1.28**	**1.50**	**7.96**	**8.19**
F	F	F+V	F+V	13.15	4.81	12.69	4.04	5.11	2.96	19.31	27.51
F+V	F	F+V	F+V	13.31	**4.77**	13.17	4.17	**1.31**	2.93	**8.08**	24.59
F	F+V	F+V	F+V	**12.97**	4.80	**12.64**	**4.08**	4.48	**1.30**	17.11	**4.98**

TABLE II: Ablation study of our proposed components on the KITTI training set. (S) Synthetic data, (SSL) Self-supervision on real data during training

Scene Flow	Training	D1-all	D2-all	Fl-all	SF-all
Disparity Change	S	16.95	24.15	34.72	41.65
Warp Output	S + SSL	**4.74**	17.08	**17.70**	23.00
Disparity Change	S + SSL	**4.74**	**11.96**	**17.70**	**20.97**



(a) RGB image (b) Predicted disparity map

(c) Error to GT disparity (d) Predicted variances

Fig. 4: Visualization of disparity and uncertainty estimations of our method compared to the ground truth error. The estimated variance matches the ground truth error and peaks at object boundaries and high frequency areas.

can be used to transfer knowledge to a new data domain, in this case the domain of VKITTI2, shown in the last four columns of the table. For this, all models are supervised on both tasks using FlyingThings3D and the three models are additionally supervised on optical flow estimation, disparity estimation or both tasks using VKITTI2. In row four, we observe that adding optical flow labels from VKITTI2 improves the performance of all tasks, when comparing to training only on FlyingThings3D in row three. Contrary to this, in row five we observe that adding disparity labels from VKITTI2 only improves the performance on the disparity estimation task on the VKITTI2 test set, while the performance of optical flow estimation degrades, increasing the end-point-error by 15% from 6.01 to 6.91. Finally, row six shows that adding supervision for both tasks produces the best results compared to adding labels for each subtask. We note that by adding supervision labels from the VKITTI2 domain, the performance of the model on FlyingThings3D degrades slightly as the total domain covered expands.

While these results show that transferability from the optical flow estimation task to the disparity estimation task exists on a new data domain, they also demonstrate that it is not trivial to transfer knowledge in the opposite direction, from disparity to optical flow estimation. To bridge this gap, in rows seven to nine of the table, we investigate the effects of self-supervised learning, which can be applied in the absence of labels. We focus on the evaluation on the VKITTI2 dataset for this. Firstly, in row seven we report the results of training with FlyingThings3D labels only, while adding self-supervision on both domains. Self-supervision improves all metrics on VKITTI2 compared to row three. The performance on all three tasks further improves by adding optical flow supervision in row eight. Importantly, in row nine we can observe that adding only disparity supervision in combination with self-supervision is able to overcome the gap shown previously, as all three tasks improve on the VKITTI2 domain compared to only training on FlyingThings3D in row three. Notably, the Epe on the optical flow task is reduced by 25%, from 6.01 to 4.48, compared to a reduction in Epe by only 15%, from 6.01 to 5.11 when only the self-supervised losses are applied. At the same time, self-supervision with partial supervision even improves the supervised task more than only using supervision of that task. This is evident comparing row four with row eight, where the optical flow Epe is reduced from 1.38 to 1.31 on VKITTI2, and by comparing row five with row nine, where the disparity Epe is reduced from 1.82 to 1.30 by adding self-supervision.

This ablation study confirms two important aspects of our proposed multi-task model. Firstly, there is no significant loss in performance resulting from the combination of several tasks into one unified model. Secondly, using self-supervised losses in combination with partial labels is very effective in transferring knowledge between the different tasks. The addition of self-supervised losses act as a catalyst for the knowledge transfer from the simpler disparity estimation task to the more complex optical flow estimation task.

Next, we further validate the effectiveness of self-supervised learning and our contributions to the Unimatch network architecture for scene flow estimation in Table II. In contrast

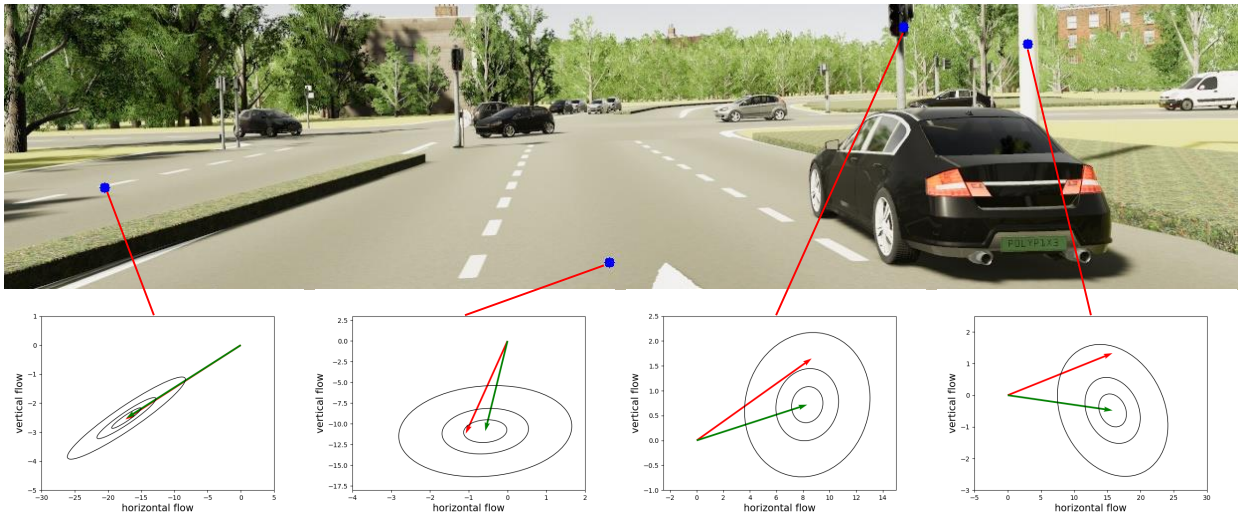


Fig. 5: Examples of the estimated optical flow uncertainty. Flow estimates are shown in green, ground truth flow in red. The estimated covariance is visualized by ellipsoids.

to the previous ablation study, we also evaluate the scene flow estimations of the model. To validate that self-supervision benefits the model, we train a model purely synthetically on FlyingThings3D and VKITTI2 (row one), and one model on the same synthetic data, with self-supervision on real data (row three). We again see a clear performance boost across all metrics. Secondly, we compare explicit disparity change estimation to naively warping the second disparity map to the first image frame using the optical flow estimate to compute scene flow in row two and row three. We observe that the inclusion of disparity change into the estimation provides a clear, significant benefit to the estimation of the disparity at t' and the accuracy of the full scene flow estimate.

E. Uncertainty Estimation

A qualitative example of the uncertainty estimations of our method is provided in Fig. 4, where we compare the ground truth error of the estimated disparity map to the predicted variances from our network. The variance estimates accurately reflect the actual error with peaks at object boundaries and in high frequency image regions.

We furthermore show examples of the covariances of the optical flow estimates in Fig. 5. For the leftmost point, a corner of a lane marking that is easy to track, we observe a low uncertainty perpendicular to the motion of the point, and a larger uncertainty in the direction of motion. The remaining three example points belong to more ambiguous surfaces, in mostly textureless regions. These points are associated with a larger prediction error, which is accurately represented by the larger uncertainty estimated by our method. We note that the direction of the error (the difference between the tips of the green and red arrows) aligns well with the principal component of the error ellipsoid. In general, the model seems to learn a fundamental uncertainty in the direction of motion of the points, but also represents deviations from the ground truth optical flow in the error ellipsoids.

TABLE III: Comparison to SOTA self-supervised and semi-supervised scene flow estimation methods on the KITTI scene flow benchmark. * denotes results on the KITTI training set. Best results on the KITTI test set are denoted in bold.

	Method	D1-all	D2-all	Fl-all	SF-all
Mono	Self-Mono-SF [23]	34.02	36.34	23.54	49.54
	Multi-Mono-SF [24]	30.78	34.41	19.54	44.04
	RAFT-MSF [11]	21.21	27.51	18.37	34.98
	DRAFT* [10]	26.41	28.89	18.71	37.58
Stereo	Self-SuperFlow [27]	8.11	21.57	23.67	28.71
	UnOS [26]	6.67	**12.05**	18.00	22.32
	Ours	**4.96**	13.02	**17.93**	**21.74**
	Ours*	4.74	11.96	17.70	20.97

Both of these examples show that the output uncertainty can be used to identify unreliable regions in the prediction of the output components for downstream tasks.

F. Comparison with Other Methods

We compare the performance of our model on the KITTI scene flow benchmark [14] with published self-supervised and semi-supervised monocular and stereo scene flow estimation methods in Table III. For this, we choose methods that do not use the training dataset of the KITTI scene flow benchmark. We include DRAFT [10] in our evaluations, since their semi-supervised training methodology using labeled synthetic data and unlabeled real data is most similar to ours, but the authors only provide results on the KITTI scene flow training set. We recognize that our method is not fully self-supervised, but semi-supervised due to the use of synthetic training data. However, to the best of our knowledge, no semi-supervised stereo scene flow estimation method has reported their results on the KITTI leaderboard.

We show that our multi-task model outperforms both self-supervised and semi-supervised methods on most metrics. Notably we outperform the stereo scene flow estimation of UnOS [26], which uses estimated camera poses and rigidity

assumptions in a postprocessing step to improve the scene flow estimation in static image regions. Our approach delivers more accurate scene flow estimates without such a postprocessing, but could further benefit from it.

Furthermore, our approach outperforms DRAFT [10] in the optical flow metric. This is notable, because the semi supervised training methodology of DRAFT is most similar to our approach. As DRAFT is a monocular scene flow approach, it is naturally more difficult to estimate disparity, but estimating optical flow is independent of stereo or monocular configurations. Despite DRAFT using proprietary photorealistic synthetic data [47], which can be argued is visually more realistic data than VKITTI2, our approach achieves more accurate optical flow results.

V. CONCLUSION

This paper introduces a unified neural network and training methodology for the estimation of disparity, optical flow, and scene flow from stereo camera images. We parameterize these tasks using a single set of learned, shared weights for each independent task which enables the network to adapt to new data domains through partially or unlabeled data using self-supervised losses to stabilize the knowledge transfer. We rigorously validate this transfer of knowledge between tasks using synthetic and real datasets. Furthermore, we demonstrate that the matching distribution of cost volumes can be used to extract representative estimates of the network uncertainty. Using no real labeled data, we achieve state-of-the-art results on the KITTI scene flow dataset, outperforming prior self-supervised and semi-supervised approaches. One shortcoming of our work is that we only estimate the uncertainty of the separate scene flow tasks, effectively considering each task to be independent of the others. We plan to address this in the future by producing a coherent scene flow output with a full covariance matrix to model the interactions between outputs. Further gains in accuracy and robustness could be achieved by considering longer image sequences, both during self-supervised training and inference. Such approaches have recently found purchase in optical flow estimation [48], [49], but are relatively unexplored in the scene flow domain. Estimating uncertainties in such a setting would be particularly interesting, where an approach similar to ours could facilitate a probabilistic temporal fusion.

REFERENCES

[1] S. Vedula, S. Baker, P. Rander, R. Collins, and T. Kanade, "Three-dimensional scene flow," in *Proceedings of the 7th IEEE International Conference on Computer Vision*, vol. 2, 1999, pp. 722–729.

[2] H. Xu, J. Zhang, J. Cai, H. Rezatofighi, F. Yu, D. Tao, and A. Geiger, "Unifying flow, stereo and depth estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[3] D. J. Butler, J. Wulff, G. B. Stanley, and M. 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 12*. Springer, 2012, pp. 611–625.

[4] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *2012 IEEE conference on computer vision and pattern recognition*. IEEE, 2012, pp. 3354–3361.

[5] M. Menze and A. Geiger, "Object scene flow for autonomous vehicles," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3061–3070.

[6] D. Scharstein, H. Hirschmüller, Y. Kitajima, G. Krathwohl, N. Nešić, X. Wang, and P. Westling, "High-resolution stereo datasets with subpixel-accurate ground truth," in *Pattern Recognition: 36th German Conference, GCPR 2014, Münster, Germany, September 2-5, 2014, Proceedings 36*. Springer, 2014, pp. 31–42.

[7] L. Mehl, J. Schmalfuss, A. Jahedi, Y. Nalivayko, and A. Bruhn, "Spring: A high-resolution high-detail dataset and benchmark for scene flow, optical flow and stereo," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 4981–4991.

[8] Z. Teed and J. Deng, "Raft-3d: Scene flow using rigid-motion embeddings," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 8375–8384.

[9] Z. Yin and J. Shi, "Geonet: Unsupervised learning of dense depth, optical flow and camera pose," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 1983–1992.

[10] V. Guizilini, K.-H. Lee, R. Ambruş, and A. Gaidon, "Learning optical flow, depth, and scene flow without real-world labels," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 3491–3498, 2022.

[11] B. Bayramli, J. Hur, and H. Lu, "Raft-msf: Self-supervised monocular scene flow using recurrent optimizer," *International Journal of Computer Vision*, pp. 1–13, 2023.

[12] N. Mayer, E. Ilg, P. Häusser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox, "A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation," in *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, arXiv:1512.02134. [Online]. Available: http://lmb.informatik.uni-freiburg.de/Publications/2016/MIFDB16

[13] Y. Cabon, N. Murray, and M. Humenberger, "Virtual kitti 2," 2020.

[14] M. Menze and A. Geiger, "Object scene flow for autonomous vehicles," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[15] C. Godard, O. Mac Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 270–279.

[16] C. Godard, O. Mac Aodha, M. Firman, and G. J. Brostow, "Digging into self-supervised monocular depth estimation," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 3828–3838.

[17] N. Yang, L. v. Stumberg, R. Wang, and D. Cremers, "D3vo: Deep depth, deep pose and deep uncertainty for monocular visual odometry," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 1281–1292.

[18] J. Watson, O. Mac Aodha, V. Prisacariu, G. Brostow, and M. Firman, "The temporal opportunist: Self-supervised multi-frame monocular depth," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 1164–1174.

[19] V. Guizilini, R. Ambruș, D. Chen, S. Zakharov, and A. Gaidon, "Multi-frame self-supervised depth with transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 160–170.

[20] R. Jonschkowski, A. Stone, J. T. Barron, A. Gordon, K. Konolige, and A. Angelova, "What matters in unsupervised optical flow," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*. Springer, 2020, pp. 557–572.

[21] S. Meister, J. Hur, and S. Roth, "Unflow: Unsupervised learning of optical flow with a bidirectional census loss," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, no. 1, 2018.

[22] P. Liu, M. Lyu, I. King, and J. Xu, "Selflow: Self-supervised learning of optical flow," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4571–4580.

[23] J. Hur and S. Roth, "Self-supervised monocular scene flow estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 7396–7405.

[24] ——, "Self-supervised multi-frame monocular scene flow," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 2684–2694.

[25] L. Liu, G. Zhai, W. Ye, and Y. Liu, "Unsupervised learning of scene flow estimation fusing with local rigidity." in *IJCAI*, 2019, pp. 876–882.

[26] Y. Wang, P. Wang, Z. Yang, C. Luo, Y. Yang, and W. Xu, "Unos: Unified unsupervised optical-flow and stereo-depth estimation by watching

videos," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 8071–8081.

[27] K. Bendig, R. Schuster, and D. Stricker, "Self-superflow: Self-supervised scene flow prediction in stereo sequences," in *2022 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2022, pp. 481–485.

[28] A. Kendall and Y. Gal, "What uncertainties do we need in bayesian deep learning for computer vision?" *Advances in neural information processing systems*, vol. 30, 2017.

[29] C. Homeyer, O. Lange, and C. Schnörr, "Multi-view monocular depth and uncertainty prediction with deep sfm in dynamic environments," in *International Conference on Pattern Recognition and Artificial Intelligence*. Springer, 2022, pp. 373–385.

[30] Z. Shen, Y. Dai, and Z. Rao, "Cfnet: Cascade and fused cost volume for robust stereo matching," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 13 906–13 915.

[31] A. S. Wannenwetsch and S. Roth, "Probabilistic pixel-adaptive refinement networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11 642–11 651.

[32] H. Jiang, D. Sun, V. Jampani, Z. Lv, E. Learned-Miller, and J. Kautz, "Sense: A shared encoder network for scene-flow estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 3195–3204.

[33] Y. Zou, Z. Luo, and J.-B. Huang, "Df-net: Unsupervised joint learning of depth and flow using cross-task consistency," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 36–53.

[34] F. Aleotti, M. Poggi, F. Tosi, and S. Mattoccia, "Learning end-to-end scene flow by distilling single tasks knowledge," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 07, 2020, pp. 10 435–10 442.

[35] Z. Teed and J. Deng, "Raft: Recurrent all-pairs field transforms for optical flow," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*. Springer, 2020, pp. 402–419.

[36] H. Morimitsu, X. Zhu, R. M. Cesar, X. Ji, and X.-C. Yin, "Rapidflow: Recurrent adaptable pyramids with iterative decoding for efficient optical flow estimation," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 2946–2952.

[37] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 10 012–10 022.

[38] M. Jaderberg, K. Simonyan, A. Zisserman *et al.*, "Spatial transformer networks," *Advances in neural information processing systems*, vol. 28, 2015.

[39] H. Zhao, O. Gallo, I. Frosio, and J. Kautz, "Loss functions for image restoration with neural networks," *IEEE Transactions on computational imaging*, vol. 3, no. 1, pp. 47–57, 2016.

[40] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.

[41] C. Wang, J. M. Buenaposada, R. Zhu, and S. Lucey, "Learning depth from monocular videos using direct methods," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 2022–2030.

[42] W. Wang, D. Zhu, X. Wang, Y. Hu, Y. Qiu, C. Wang, Y. Hu, A. Kapoor, and S. Scherer, "Tartanair: A dataset to push the limits of visual slam," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 4909–4916.

[43] M. Savva, A. X. Chang, and P. Hanrahan, "Semantically-enriched 3d models for common-sense knowledge," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2015, pp. 24–31.

[44] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," in *NIPS-W*, 2017.

[45] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.

[46] L. N. Smith and N. Topin, "Super-convergence: Very fast training of neural networks using large learning rates," in *Artificial intelligence and machine learning for multi-domain operations applications*, vol. 11006. SPIE, 2019, pp. 369–386.

[47] "Parallel domain," https://paralleldomain.com/, November 2020.

[48] Q. Dong and Y. Fu, "Memflow: Optical flow estimation and prediction with memory," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 19 068–19 078.

[49] B. Wang, Y. Zhang, J. Li, Y. Yu, Z. Sun, L. Liu, and D. Hu, "Splatflow: Learning multi-frame optical flow via splatting," *International Journal of Computer Vision*, pp. 1–23, 2024.