2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Skin the sheep not only once: Reusing Various Depth Datasets to Drive the Learning of Optical Flow

Sheng-Chi Huang Wei-Chen Chiu
National Yang Ming Chiao Tung University, Hsinchu, Taiwan

***Abstract*— Optical flow estimation is crucial for various applications in vision and robotics. As the difficulty of collecting ground truth optical flow in real-world scenarios, most of the existing methods of learning optical flow still adopt synthetic dataset for supervised training or utilize photometric consistency across temporally adjacent video frames to drive the unsupervised learning, where the former typically has issues of generalizability while the latter usually performs worse than the supervised ones. To tackle such challenges, we propose to leverage the geometric connection between optical flow estimation and stereo matching (based on the similarity upon finding pixel correspondences across images) to unify various real-world depth estimation datasets for generating supervised training data upon optical flow. Specifically, we turn the monocular depth datasets into stereo ones via synthesizing virtual disparity, thus leading to the flows along the horizontal direction; moreover, we introduce virtual camera motion into stereo data to produce additional flows along the vertical direction. Furthermore, we propose applying geometric augmentations on one image of an optical flow pair, encouraging the optical flow estimator to learn from more challenging cases. Lastly, as the optical flow maps under different geometric augmentations actually exhibit distinct characteristics, an auxiliary classifier which trains to identify the type of augmentation from the appearance of the flow map is utilized to further enhance the learning of the optical flow estimator. Our proposed method is general and is not tied to any particular flow estimator, where extensive experiments based on various datasets and optical flow estimation models verify its efficacy and superiority.**

I. Introduction

Optical flow estimation plays an important role across plenty applications such as robotics, augmented reality, and autonomous vehicles. Although there exist many traditional approaches [1]–[4] which attempt to model such a problem of finding dense pixel-wise displacement across images from different perspectives, their optimization objectives are typically hand-crafted thus being hard to handle various corner cases. Along with the recent advance of deep learning techniques, we have witnessed the magic leap on performance for optical flow estimation brought by deep neural networks (e.g. FlowNet [5] as the seminal work and many others [6]–[10]). While most of these works heavily rely on large-scale datasets with groundtruth annotations (i.e. optical flow maps) to perform the supervised learning, collecting such supervised datasets is highly challenging and expensive in the real world (as there exists no sensor which can directly measure the pixel-wise correspondence between views). To address this problem, many research works [6], [7], [11], [12] have utilized a pre-training approach on large synthetic datasets [11], [13], followed by fine-tuning on limited target datasets [14]–[16]. However, such an approach is still hampered by a lack of groundtruth in the real world (for performing fine-tuning) and suffers from poor generalizability due to the domain shift (i.e. different distributions between synthetic and real-world data). Although there exist several recent attempts to explore the unsupervised learning scenario [17]–[20] (where the photometric consistency is usually adopted to evaluate pixel correspondences across images, and joint learning with other tasks such as depth estimation or camera pose estimation would come into play), their performances are mostly still inferior to those supervised ones.

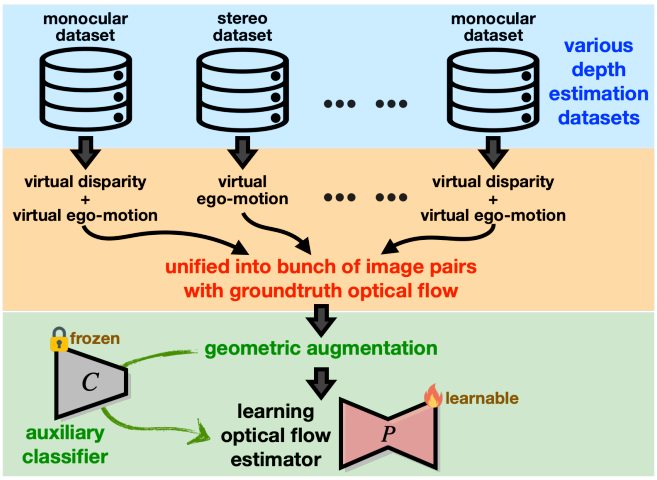

Fig. 1: We unify various depth estimation datasets into supervised data of learning optical flow, via the introduction of virtual disparity and virtual camera motion (i.e. ego-motion) to produce the horizontal and vertical flows. Moreover, the geometric augmentations are applied to not only generate more challenging data samples but also enable an auxiliary classifier to improve the training of optical flow estimator.

To strike a better balance among aforementioned challenges (e.g., lack of real-world supervised dataset, domain shift, and inferior performance for unsupervised learning), [21] recently proposed a novel method to generate an image pair with its optical flow annotations from a single real-world input image. Basically, given the input image, its corresponding 3D point cloud is firstly built with the help of depth estimation (i.e. projecting each pixel in the input image back to 3D space), then a 3D motion (composed of rotation and translation) is applied to the virtual camera of the input image to synthesize a novel view, where the pixel correspondence (i.e. optical flow map) between the original input image and the novel view is naturally available as the entire geometric transformation and projection procedure is under manual

control. In the results, the real-world supervised dataset can be constructed for training the optical flow estimator.

Inspired by [21], we come up with two further considerations: 1) While the accuracy of depth estimation is critical to the quality of synthesized novel views in [21], the monocular depth estimation datasets which contain groundtruth depth maps for their images seem to be a feasible alternative to bypass the uncertainty stemmed from depth estimation; 2) As optical flow estimation has a close relative, i.e. stereo matching, in terms of finding correspondences, the stereo depth estimation datasets ideally ought to be helpful as well for learning optical flow. Moreover, as collecting groundtruth depth maps in real-world scenarios is typically more achievable than the optical flow ones due to the popularity of depth cameras, we are therefore motivated to bring up the following question: *Can optical flow estimation be learned from both monocular and stereo depth datasets, and can the relationship between stereo matching and optical flow be explored beyond treating depth as an intermediate product?*

To this end, we propose a framework to unify both monocular and stereo depth datasets, followed by transforming them into a collection of annotated optical flow data: For an image in the monocular depth dataset, we first translate its groundtruth depth map into the disparity map (named **virtual disparity**), which is used to warp the original image into a novel view, where the original image and the warped one together become a stereo image pair. It is worth noting that the pixel correspondence in such stereo image pair only has the horizontal offset, which can be treated as horizontal optical flow; while for a stereo image pair (obtained from the stereo depth dataset or produced by the previous step), we can apply the same procedure as [21] to employ virtual camera motion (also named as **virtual ego-motion**) on one image of such pair, in which its resultant novel view together with the other image of the stereo pair finally form an optical flow pair containing groundtruth annotation.

Furthermore, as data augmentation has become a widely adopted training strategy to increase the quantity and diversity of training data for further improving the performance of deep models, most works of optical flow estimation also employ data augmentation, in which the used augmentation operations can be categorized into two classes: 1) **photometric augmentations**, which mainly modify the pixel appearance (e.g. contrast, sharpness, brightness, and colors) while preserving the spatial structure, where such property allows them to be applied independently to each image; 2) **geometric augmentations**, which would affect the scene structure (e.g. flipping, cropping, rotating, and scaling), thus they should be applied to both images in a pair and the corresponding optical flow map simultaneously.

In this work, we step further to discover that, once we only perform the geometric augmentations on one image in a pair, its resultant pixel correspondence with respect to another image (i.e. optical flow map) will undergo a drastic distortion and becomes more challenging for the optical flow estimator to predict, thus including such data samples into training would benefit the learning of optical flow estimation model. Moreover, we have another observation that applying different geometric augmentations on only one image in a pair would lead to optical flow maps with distinct characteristics. In the results, we propose to pretrain an auxiliary classifier which takes the optical flow map as input and predicts the type of geometric augmentation being applied, where such a classifier can be later utilized to construct a novel objective for providing additional supervision signals in training optical flow estimator. In summary, our full method is composed of all the designs above (i.e. unification over various depth estimation datasets to create a supervised optical flow dataset, including challenging training samples via applying geometric augmentations on only one image in an optical flow pair, and using the pretrained auxiliary classifier to boost the optical flow learning), in which it can be employed on training any optical flow estimator thus being quite flexible and general.

II. Related Work

In recent years, there has been a growing interest in leveraging the geometric relationship between stereo matching (or depth estimation) and optical flow estimation to jointly learn spatial correspondence and improve flow accuracy [18], [20], [22], [23]. For instance, [20] finds that the optical flow estimation and stereo matching tasks coexist in the temporally adjacent stereo pairs (i.e., stereo video) thus proposes to bridge both tasks via photometric reconstruction in such data format for learning a unified model of finding pixel correspondence across images. Instead of jointly learning optical flow and depth estimation, some works propose to use depth estimation as an intermediate step to generate the training data for learning estimation of pixel correspondence (e.g. stereo matching or optical flow estimation). For instance, [24] and [21] adopt the estimated depth of the input image to perform novel view synthesis for, respectively, constructing the annotated stereo or optical flow pairs. Our proposed framework follows the similar idea as [21] but directly takes the advantage of using groundtruth depth in the supervised depth estimation datasets. Moreover, as our framework has the feature of unifying various depth datasets, it is also conceptually related to [25], in which [25] allows the mixture of multiple datasets with various formats of depth annotations during training the depth estimator. Nevertheless, our target scenario is quite different from [25].

III. Methodology

A. Create Annotated Flow via Unifying Depth Datasets

Producing horizontal optical flow. As previously motivated, we would like to unify various real-world depth estimation datasets with groundtruth annotations, regardless of the monocular or stereo ones, for synthesizing the supervised dataset for learning optical flow estimation. We denote an optical flow training sample as a tuple $(\mathcal{I}_0, \mathcal{I}_1, F_{0\to1})$, where $\mathcal{I}_0$ and $\mathcal{I}_1$ form an image pair, while $F_{0\to1}$ represents the optical flow map between $\mathcal{I}_0$ and $\mathcal{I}_1$.

Given an image $\mathcal{I}_0^m$ and its corresponding groundtruth depth map Z_0^m obtained from a monocular depth dataset,

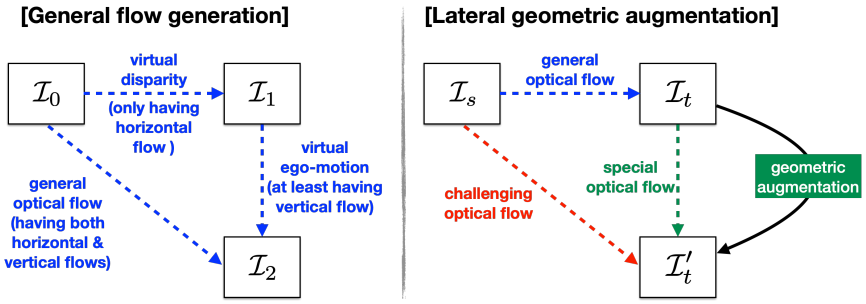

Fig. 2: **Overview of general flow generation (left figure, cf. Section III-A) and lateral geometric augmentation (right figure, cf. Section III-B.** In general flow generation, virtual disparity is firstly adopted to turn an image from the monocular depth dataset into a stereo pair, then the virtual ego-motion is applied on one image of each stereo pair (regardless from monocular or stereo datasets) to include vertical flows, finally we can produce the optical flow training sample with general flow map (i.e. having both horizontal and vertical flows); our lateral geometric augmentation applies geometric augmentation on only one image of a training pair to produce more challenging optical flow cases.

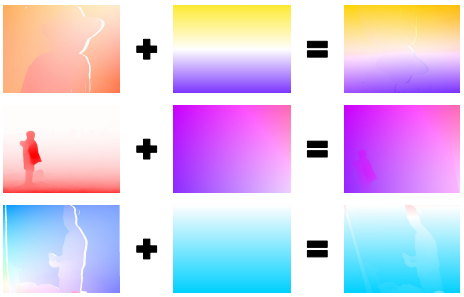

Fig. 3: **Examples of distinct optical flow characteristics from various lateral geometric augmentations (cf. Section III-B).** From top to bottom (zoom in for better view), the rightmost flows are generated by applying flipping, rotation, and shearing operations on the leftmost source flows (following Eq. (14) and (15)).

we start by converting Z_0^m into the disparity map $\tilde{d}_m$ via $\tilde{d}_m = \frac{Bf}{Z_0^m}$ where B and f denote the baseline and focal length, respectively. Following the practice in [24], in order to have a wide range of baselines and focal lengths, we set $Bf = s_c$ in which such a scaling factor s_c is randomly drawn from a uniform distribution. As the disparity represents the pixel-wise displacement along the horizontal direction, we can easily translate it into the form of an optical flow map by $\left\langle \tilde{d}_m, \varnothing \right\rangle$, where $\langle \cdot \rangle$ denotes the channel-wise concatenation operation, and $\varnothing$ is a zero map of the same size as $\tilde{d}_m$. In particular, we can treat $\left\langle \tilde{d}_m, \varnothing \right\rangle$ as a horizontal optical flow $F_{0\to1}^m$. We can now construct a stereo pair $\{\mathcal{I}_0^m, \tilde{\mathcal{I}}_1^m\}$ via

$$\tilde{\mathcal{I}}_1^m = \mathcal{W}(\mathcal{I}_0^m, s_i \cdot F_{0\to1}^m) \tag{1}$$

$$\tilde{Z}_1^m = \mathcal{W}(Z_0^m, s_i \cdot F_{0\to1}^m) \tag{2}$$

where $\mathcal{W}(\alpha, \beta)$ defines the warping function in which α is warped according to β, s_i is randomly set to either 1 or -1 to simulate the respective case where $\mathcal{I}_0^m$ is on the left- or right-hand side, and $\tilde{Z}_1^m$ is the depth map for $\tilde{\mathcal{I}}_1^m$.

Next, we consider an image pair $\{\mathcal{I}_0^s, \mathcal{I}_1^s\}$ obtained from a stereo dataset, where the corresponding groundtruth disparity map is d_s. The depth map $\tilde{Z}_0^s$ of $\mathcal{I}_0^s$ can be simply computed by $\tilde{Z}_0^s = \frac{Bf}{d_s}$ where Bf is set to a constant as d_s contains actual disparity values. And the depth map $\tilde{Z}_1^s$ of $\mathcal{I}_1^s$ is:

$$\tilde{Z}_1^s = \mathcal{W}(\tilde{Z}_0^s, F_{0\to1}^s), \text{ where } F_{0\to1}^s = \langle d_s, \varnothing \rangle . \tag{3}$$

Synthesizing general optical flow. Since $F_{0\to1}^m$ or $F_{0\to1}^s$ only have horizontal displacement, now we advance to utilise virtual camera motion (i.e. virtual ego-motion) as [21] to produce more general flow maps (i.e. having both horizontal and vertical displacements). Given the stereo pair $\{\mathcal{I}_0^s, \mathcal{I}_1^s\}$ and the corresponding flow map $F_{0\to1}^s$, we aim to create a plausible optical flow $F_{1\to2}^s$ that at least has vertical displacement. This operation will enable synthesizing a novel view $\tilde{\mathcal{I}}_2^s$, its corresponding depth map $\tilde{Z}_2^s$, and the target general optical flow $F_{0\to2}^s$ (i.e. the flow map from $\mathcal{I}_0^s$ to $\tilde{\mathcal{I}}_2^s$).

Basically, we first hypothesize an intrinsic matrix K in which its inverse K^{-1} will be used to project the pixels in $\mathcal{I}_1^s$ back into the 3D space, following the common practice of previous approaches [20], [23], [26]. Then we randomly sample a plausible rotation $R_{1\to2}$ and translation $t_{1\to2}$ to obtain a transformation matrix $T_{1\to2} = [R_{1\to2} \mid t_{1\to2}]$. Subsequently, we project each pixel $p_1 \in \mathcal{I}_1^s$ back into 3D space to form a point cloud, perform camera ego-motion, and finally project the 3D point cloud onto the 2D image plane to derive the optical flow map $F_{1\to2}^s$. With such an optical flow map $F_{1\to2}^s$, we can then synthesize the novel view $\tilde{\mathcal{I}}_2^s$ and its corresponding depth map $\tilde{Z}_2^s$.

The overall procedure to reach $F_{0\to2}^s$ is summarized as:

$$F_{1\to2}^s = KT_{1\to2}\tilde{Z}_1^s(p_1)K^{-1}p_1 - p_1 \tag{4}$$

$$\tilde{\mathcal{I}}_2^s = \mathcal{W}(\mathcal{I}_1^s, F_{1\to2}^s), \tilde{Z}_2^s = \mathcal{W}(\tilde{Z}_1^s, F_{1\to2}^s) \tag{5}$$

$$F_{0\to2}^s = F_{0\to1}^s + \mathcal{W}^{-1}(F_{0\to1}^s, F_{1\to2}^s) \tag{6}$$

where $\mathcal{W}^{-1}(\alpha, \beta)$ is a specific warping function to support the backward warping that $\mathcal{W}^{-1}(\alpha, \beta) = \beta(x + \alpha(x))$ and x denote all pixel locations. Note that we can apply the same procedure to the stereo pair $\{\mathcal{I}_0^m, \tilde{\mathcal{I}}_1^m\}$ derived from the monocular dataset to obtain $F_{0\to2}^m$.

Finally, we collect all the produced image pairs and their corresponding groundtruth optical flow maps, in which they are represented as tuples: $(\mathcal{I}_0^m, \tilde{\mathcal{I}}_1^m, F_{0\to1}^m)$, $(\tilde{\mathcal{I}}_1^m, \tilde{\mathcal{I}}_2^m, F_{1\to2}^m)$, $(\mathcal{I}_0^m, \tilde{\mathcal{I}}_2^m, F_{0\to2}^m)$, $(\mathcal{I}_0^s, \mathcal{I}_1^s, F_{0\to1}^s)$, $(\mathcal{I}_1^s, \tilde{\mathcal{I}}_2^s, F_{1\to2}^s)$ and $(\mathcal{I}_0^s, \tilde{\mathcal{I}}_2^s, F_{0\to2}^s)$. The left portion of Figure 2 shows the entire aforementioned procedure to create optical flow training samples, which is named **general flow generation**.

We would like to emphasise that, although the aforementioned procedure of producing horizontal optical flow seems to be redundant when we have monocular depth data (i.e. in which we can directly go for synthesising horizontal and optical flow instead of getting the disparity first, as what [21] does), we intend to have such way of description for putting both stereo and monocular dataset cases into the same framework, and highlighting the common ground (i.e.

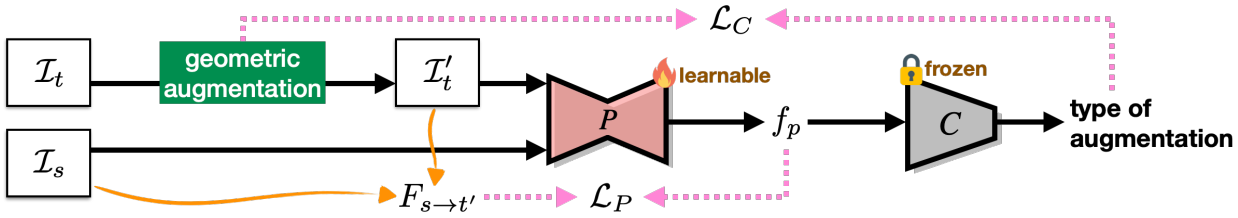

Fig. 4: **Our framework for learning optical flow estimator.** Given an image pair $\{\mathcal{I}_s, \mathcal{I}_t\}$, we firstly apply lateral geometric augmentation on $\mathcal{I}_t$ to obtain $\mathcal{I}'_t$, where the resultant groundtruth optical flow $F_{s\to t'}$ between $\mathcal{I}_s$ and $\mathcal{I}'_t$ can be easily obtained via the computation in Section III-B. Our flow estimator P which takes $\{\mathcal{I}_s, \mathcal{I}'_t\}$ as input is trained to output the flow map f_p that ideally should be identical to $F_{s\to t'}$, where the objective $\mathcal{L}_P$ evaluates the difference between $F_{s\to t'}$ and f_p. Moreover, the auxiliary classifier C, which is trained to identify the type of augmentation used in the lateral geometric augmentation of the appearance of input flow, contributes to define another objective $\mathcal{L}_C$ (cf. Section III-C) to increase the learning of P.

finding correspondence) between optical flow and disparity. Moreover, with respect to [21], in addition to the contribution of our unifying various depth datasets to generate general flows, our framework has novelties from lateral geometric augmentation and the auxiliary classifier as described below (the model variant ablating all our designs, cf. the first row of Table III, reproduces [21] and performs worse than ours).

B. Lateral Geometric Augmentation

In addition to typical augmentation strategies (i.e. applying photometric augmentation on each image independently, or applying the same geometric augmentation on the two images of a training pair), we propose to conduct **lateral geometric augmentation**, where the geometric augmentation is applied on only one image of each training pair to produce more challenging cases to learn the optical flow estimation. Three three geometric augmentation operations are adopted and sequentially introduced in the following.

Flipping operation. Both horizontal and vertical flipping operations are adopted. After applying the horizontal flipping operation, we use the equations below, which are derived from the symmetric attribution of the flipped coordinates.

$$(x_0 + x_1)/2 = H/2, \quad y_0 = y_1 \tag{7}$$

where $p_0 = (x_0, y_0)$ and $p_1 = (x_1, y_1)$ represent the original and flipped pixel coordinates, respectively, and H is the height of the image, we reach the per-pixel flow $f^{hf}_{p_0\to p_1} = (H - 2x_0, 0)$ for horizontal flipping augmentation. Similarly, for the vertical flipping operation, we obtain the per-pixel flow $f^{vf}_{p_0\to p_1} = (0, W - 2y_0)$ where W is the width of the image. When collecting all the per-pixel flow, we can obtain the special flow F^{hf}_a and F^{vf}_a related to horizontal and vertical flipping augmentations, respectively. In particular, the backward flow B resulting from the flipping augmentation is the same as the forward flow F, so we have $B^{hf}_a = F^{hf}_a$ and $B^{vf}_a = F^{vf}_a$.

Rotation operation. We apply rotation operations to images by randomly sampling a Euler angle θ_a and adding a sign factor s_i uniformly sampled from $\{-1, 1\}$ to simulate clockwise and counterclockwise rotations. The rotation matrix $R(\theta_a)$ is then calculated by:

$$R(s_i\theta_a) = \begin{bmatrix} cos(s_i\theta_a) & -sin(s_i\theta_a) \\ sin(s_i\theta_a) & cos(s_i\theta_a) \end{bmatrix} \tag{8}$$

Next, we randomly sample a center point c_0 and rotate every pixel coordinate p_0 about c_0 by applying the rotation matrix. This yields the special flow F^r_a and the backward flow B^r_a resulting from the rotation augmentation:

$$F^r_a = R(s_i\theta_a)(p_0 - c_0) + c_0 - p_0 \tag{9}$$
$$B^r_a = R(-s_i\theta_a)(p_0 - c_0) + c_0 - p_0 \tag{10}$$

Shearing operation. We introduce shearing operations to images along horizontal or vertical directions by randomly sampling a shearing magnitude factor λ_a and a sign factor s_i that is sampled uniformly from $\{-1, 1\}$ to simulate different directions of shearing stress. This creates horizontal and vertical shearing matrices $S^{hs}(\lambda_a)$ and $S^{vs}(\lambda_a)$, respectively:

$$S^{hs}(\lambda_a) = \begin{bmatrix} 1 & s_i\lambda_a \\ 0 & 1 \end{bmatrix}, S^{vs}(\lambda_a) = \begin{bmatrix} 1 & 0 \\ s_i\lambda_a & 1 \end{bmatrix} \tag{11}$$

We then apply these horizontal or vertical shearing matrices to every pixel coordinate p_0 to obtain new coordinates p_1. To obtain the forward flows $\{F^{hs}_a, F^{vs}_a\}$ and backward flows $\{B^{hs}_a, B^{vs}_a\}$ stemmed from shearing augmentation, we use:

$$F^{hs}_a = S^{hs}(\lambda_a)p_0 - p_0, \quad B^{hs}_a = S^{hs}(-\lambda_a)p_0 - p_0 \tag{12}$$
$$F^{vs}_a = S^{vs}(\lambda_a)p_0 - p_0, \quad B^{vs}_a = S^{vs}(-\lambda_a)p_0 - p_0 \tag{13}$$

Lateral geometric augmentation. Given a tuple of training data $(\mathcal{I}_s, \mathcal{I}_t, F_{s\to t})$ which can be mapped to any tuple produced by general flow generation (cf. Section III-A), we can apply geometric augmentation to either the source $\mathcal{I}_s$ or target image $\mathcal{I}_t$ to obtain $\mathcal{I}'_s = A(\mathcal{I}_s)$ or $\mathcal{I}'_t = A(\mathcal{I}_t)$, where $A(\cdot)$ is the augmentation operator. We refer to the forward flow resulting from flipping, rotation, or shearing augmentation as F_a (also named as *special flow*), which is used to warp $\mathcal{I}_s$ to $\mathcal{I}'_s$, and the backward flow as B_a, which is used to warp $\mathcal{I}'_t$ to $\mathcal{I}_t$. We then use the warping operator $\mathcal{W}^{-1}$ to compute the challenging flow cases:

$$F_{s\to t'} = F_{s\to t} + \mathcal{W}^{-1}(F_{s\to t}, F_a) \tag{14}$$
$$F_{s'\to t} = B_a + \mathcal{W}^{-1}(B_a, F_{s\to t}) \tag{15}$$

The resulting tuples of $(\mathcal{I}_s, \mathcal{I}'_t, F_{s\to t'})$ and $(\mathcal{I}'_s, \mathcal{I}_t, F_{s'\to t})$, generated by lateral geometric augmentation, are more challenging and allow us to train the optical flow estimation network more effectively. The right portion of Figure 2 shows an overview of our **lateral geometric augmentation**.

C. Auxiliary Classifier

Optical flow features from different augmentations. As shown by Eq. (14) and (15), both optical flow maps $F_{s\to t'}$ and $F_{s'\to t}$ contain the augmentation-dependent components of forward flow F_a and backward flow B_a, it is thus believed that we should be able to infer the type of augmentation (flipping, rotation, or shearing) being applied for the lateral geometric augmentation. We achieve this by training an auxiliary classifier C that can identify the type of augmentation used in lateral geometric augmentation from the appearance of the input optical flow map. The classifier takes an optical flow F as input and produces a real-valued vector $T_p = C(F)$ composed of four elements, where T_p passed through softmax represents the posterior of four augmentation types (i.e. flipping, rotation, shearing, and none of the above). Our classifier C comprises a feature extractor, an average pooling layer, and a fully connected layer. The feature extractor is based on the small encoder of the RAFT [8] model with input channels set to 2 to support the optical flow input.

Training optical flow estimator with auxiliary classifier. With the auxiliary classifier C pretrained on annotated optical flow data (produced by general flow generation and later geometric augmentation), we can further use it to increase the learning of the flow estimator P, as shown in Figure 4 – *noting that such design is stemmed from the motivation of introducing an additional learning task, which is easier than optical flow estimation but closely related to it, to further boost the training of optical flow estimator via providing more supervision signal*. Given an optical flow training pair $\{\mathcal{I}_s, \mathcal{I}_t\}$, we apply lateral geometric augmentation on $\mathcal{I}_t$ with the augmentation type T_a to produce $\mathcal{I}'_t$. When the flow estimator P takes $\{\mathcal{I}_s, \mathcal{I}'_t\}$ as input, its prediction f_p should not only align with the groundtruth flow map $F_{s\to t'}$ (where the L1 error between f_p and $F_{s\to t'}$ defines the objective $\mathcal{L}_P$), but also to be correctly classified by C to match T_a. The classification difference in terms of cross entropy between $C(f_p)$ and T_a thus forms a novel objective $\mathcal{L}_C$ where the gradients are backpropagated to update P. The idea here is that since our classifier C is pretrained and frozen during the training of flow estimator P, if the optical flow map f_p predicted by P is not accurate enough for the classifier C to distinguish the type of augmentation being applied on $\mathcal{I}'_t$ then the error is all attributed to the model P. The overall loss $\mathcal{L}$ to train the flow estimator P is formulated as:

$$\mathcal{L} = \mathcal{L}_P(f_p, F_{s\to t'}) + \lambda_C \mathcal{L}_C(C(f_p), T_a) \tag{16}$$

where the hyperparameter λ_C balances between $\mathcal{L}_P$ and $\mathcal{L}_C$.

IV. Experiments

A. Datasets

Our approach enables the generation of annotated optical flow data via unifying a variety of depth estimation datasets, where two real-world depth datasets are used in our experiments: **ReDWeb** [27] (a monocular dataset, denoted as `R`) and **DIML** (a stereo dataset, denoted as `D`). For detailedness, RebWeb `R` is an RGB-D dataset composed of 3600 images and their corresponding depth maps, moreover, it contains highly diverse indoor and outdoor scenes; DIML `D` focuses on outdoor scenes, in which it is composed of 1505 stereo pairs and the corresponding disparity maps.

Moreover, as the synthetic datasets, which have groundtruth optical flow maps to enable supervised learning of optical flow estimation, are actually compatible with our proposed framework (cf. Figure 4), we therefore also consider them in our experiments: **FlyingChairs** (denoted as `C`) and **FlyingThings3D** (denoted as `T`), where `T` offers more complex motion patterns than those in `C`.

Based on the four aforementioned datasets, we come up with several settings to leverage them for our training:

- `C` where the flow estimator P is trained on `C` only.
- `C`→`T` where the flow estimator P is firstly trained on `C` then finetuned on the more complex dataset `T`.
- `R+D` where the flow estimator P is trained on the real-world annotated optical flow dataset, which is produced by unifying/mixing across `R` and `D` datasets via our general flow generation procedure (cf. Section III-A).
- `C`→`T`→`R+D` where the flow estimator P is sequentially trained/fine-tuned on dataset `C`, dataset `T`, and our mixed real-world dataset `R+D`.

Furthermore, as our proposed framework for learning optical flow has the flexibility to support arbitrary backbones of flow estimator, here we adopt two well-known optical flow models for P in our experiments: **RAFT** [8] and **GMFlow** [10].

One synthetic and one real-world datasets are adopted for our evaluation: **Sintel** [14] and **KITTI** [15], [16], where both are popular and challenging benchmarks for optical flow estimation. Sintel is generated from a 3D animated short film, which offers two render passes: "clean" and "final", where the latter additionally includes visual variations (e.g. blurring and atmospheric effects); KITTI is collected from real-world street views, where we use both versions of it, namely "KITTI-12" and "KITTI-15". Regarding the evaluation metrics, we adopt **EPE** and **F1-all**, where the former refers to the average endpoint error while the latter refers to the percentage of optical flow outliers over all pixels.

B. Experimental Results

The experimental results of learning flow estimators (i.e. RAFT or GMFlow) upon four training settings are summarized in Table I (for using RAFT model as flow estimator) and Table II (for using GMFlow as flow estimator) respectively. From these results we draw several observations:

1) Regarding the evaluation upon real-world scenarios (i.e. KITTI-12 and KITTI-15), training using real-world datasets `R+D` (where the optical flow annotations in `R+D` are produced by our proposed general flow generation), either only using `R+D` or starting from synthetics ones `C`→`T` then finetuning on `R+D` (i.e. last two rows in both tables), provide better performance than those using only synthetic datasets (i.e. `C` and `C`→`T`), showing our main contribution of providing real-world supervised dataset for optical flow estimation;

2) Leveraging synthetic dataset that has precise optical flows of groundtruth to warm start the training (i.e. `C`→`T`→`R+D`)

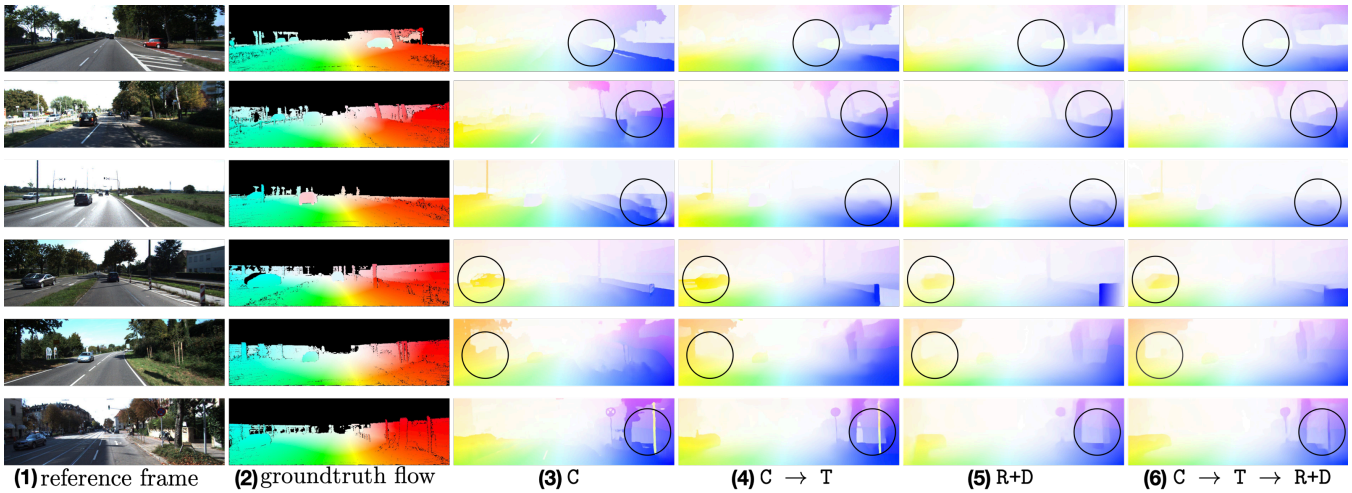

Fig. 5: **Example Results on KITTI-15**: (1) reference frame, (2) groundtruth flow, and flow maps produced by the RAFT models respectively trained on (3) FlyingChairs (C), (4) FlyingChairs→FlyingThings3D (C→T), (5) our real-world dataset mixed across ReDWeb and DIML (R+D), and (6) FlyingChairs→FlyingThings3D→our mixed dataset (C→T→R+D). The circles on the figures highlight the regions where our method improves the flow estimation (zoom in for better visualisation).

often is able to perform the best in real-world scenarios (i.e. KITTI), compared to using only the real-world dataset R+D; **3)** Though having better performance and generalizability in real-world scenes (i.e. KITTI-12 and KITTI-15), training or finetuning on real-world datasets (i.e. R+D and C→T→R+D) would lead to worse performance in the synthetic scenario (i.e. Sintel) due to the domain gap among real and virtual data, while training ended up with synthetic ones (i.e. C and C→T) typically performs better in Sintel, since C and T datasets share the same graphics rendering process as Sintel. Note that [21] also has such an observation similar to ours; **4)** Another important contribution of our proposed method is the capability of unifying various depth estimation datasets, where prior work [21] typically can only utilize a single dataset, thus resulting in worse performance (in which the model variant without any of our proposed designs, i.e. the first row in Table III, actually reproduces [21]'s model trained on D; even when we train [21]'s model based on RAFT flow estimator upon R, it still performs worse to have 2.42 EPE and 9.96 F1-all for KITTI-12 and 5.65 EPE and 18.85 F1-all for KITTI-15) than ours trained on unified dataset R+D.

TABLE I: **Quantitative results upon RAFT flow estimators with different training settings (cf. Section IV-A).**

Datasets	Sintel		KITTI-12		KITTI-15	
	clean	final	EPE	F1-all	EPE	F1-all
C	2.36	4.39	5.14	34.64	10.77	41.08
C → T	**1.64**	**2.83**	2.40	10.49	5.62	18.71
R+D	2.61	3.98	2.16	9.28	4.18	15.03
C → T → R+D	2.44	3.88	**2.12**	**8.28**	**4.06**	**13.58**

TABLE II: **Quantitative results upon GMFlow flow estimators with different training settings (cf. Section IV-A).**

Datasets	Sintel		KITTI-12		KITTI-15	
	clean	final	EPE	F1-all	EPE	F1-all
C	3.23	4.43	8.73	47.10	17.82	56.15
C → T	**1.50**	**2.96**	5.09	25.75	11.60	35.52
R+D	3.58	4.88	**4.21**	21.63	9.80	33.95
C → T → R+D	3.01	4.60	4.33	**19.60**	**8.66**	**28.78**

In Figure 5 we provide some qualitative examples, where we see that the models trained on only synthetic datasets (e.g. C or C→T in the third or fourth columns of Figure 5) though provide correct object shape but may exhibit glaring errors in flow direction, while the models trained on diverse and challenging real-world datasets (attributed to our general flow generation and lateral geometric augmentation; e.g. R+D or C→T→R+D in the fifth or sixth columns of Figure 5) show less artifacts and errors, thus achieving better performance.

C. Ablation Study

Model Designs. We conduct an investigation upon the model designs in our proposed framework: virtual disparity, lateral geometric augmentation, and auxiliary classifier. Please note that here we specifically only adopt DIML dataset to perform model training in order to exclude the benefit of our unifying various depth datasets and better focus on the contributions of our model designs. Moreover, as virtual ego-motion has been introduced in [21], we thus do not consider it in our ablation study. Furthermore, though DIML dataset itself is already a stereo dataset, we can still leverage different scaling factors for baseline B and focal length f to introduce difference virtual disparities. From Table III which summarizes the ablation results on KITTI-15 dataset, we observe that: **1)** the model variant that excludes all our designs (i.e. reproduction of [21]) performs the worse, while introducing our virtual disparity to enrich the depth variance helps to boost the performance; **2)** the further introduction of lateral geometric augmentation provides the significant improvement thanks to its providing more diverse and challenging optical flow samples for learning; **3)** with adopting our proposed auxiliary classifier C, the full model exhibits further advance, thus verifying the contribution and efficacy of the corresponding novel objective $\mathcal{L}_C$.

Accuracy of Auxiliary Classifier C. Moreover, we conduct a study (cf. Table IV) to investigate the impact caused by the accuracy of auxiliary classifier C (noting that the accuracy

TABLE III: **Ablation study for model designs**. Evaluation is based on KITTI-15 dataset with two flow estimator backbones (GMFlow and RAFT) being trained on DIML dataset.

virtual disparity	lateral geometric augmentation	auxiliary classifier	GMFlow		RAFT	
			EPE	F1-all	EPE	F1-all
✗	✗	✗	11.25	39.8	6.08	16.74
✓	✗	✗	12.31	39.91	5.76	16.03
✓	✓	✗	10.97	35.44	4.64	15.83
✓	✓	✓	**10.94**	**34.05**	**4.52**	**15.36**

of the classifier C is evaluated on the ReDWeb dataset, higher accuracy indicates better performance of identifying the augmentation types from the flow maps), in which we observe that a stronger C (with its accuracy to be 0.80) can contribute to better training of flow estimator in comparison to the weaker one (with its accuracy to be 0.69).

TABLE IV: **Study on the impact from the accuracy of auxiliary classifier C.** Evaluation is based on two flow models (RAFT backbone) trained on DIML dataset respectively using two auxiliary classifiers with different accuracy levels.

classifier strength	Sintel		KITTI-15	
	clean	final	EPE	F1-all
weak (accuracy = 0.69)	3.59	**4.86**	5.10	15.61
strong (accuracy = 0.80)	**3.56**	4.91	**4.52**	**15.36**

Please kindly refer to our supplementary video for more ablation studies, implementation details (e.g. the parameter settings and illustrated procedures for generating virtual disparity and virtual ego-motion as well as applying lateral geometric augmentation, and the architecture of auxiliary classifier), and more qualitative results.

V. Conclusion

We propose a novel framework to well unify various supervised depth estimation datasets, including both monocular and stereo ones, for synthesizing the real-world optical flow training set with groundtruth annotations. With further introducing the challenging optical flow training samples by our proposed lateral geometric augmentation and building a novel objective function based on our proposed auxiliary classifier, the learning of optical flow estimator is largely benefited to achieve the superior performance with respect to the state-of-the-art baseline across various experiments. The source code and model are available at https://github.com/AegeanKI/OpticalFlowFromDepth

References

[1] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert, "High accuracy optical flow estimation based on a theory for warping," in *European Conference on Computer Vision (ECCV)*, 2004.

[2] S. Baker, D. Scharstein, J. Lewis, S. Roth, M. J. Black, and R. Szeliski, "A database and evaluation methodology for optical flow," *International Journal of Computer Vision (IJCV)*, 2011.

[3] M. Menze, C. Heipke, and A. Geiger, "Discrete optimization for optical flow," in *German Conference on Pattern Recognition*, 2015.

[4] Q. Chen and V. Koltun, "Full flow: Optical flow estimation by global optimization over regular grids," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[5] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Van Der Smagt, D. Cremers, and T. Brox, "Flownet: Learning optical flow with convolutional networks," in *IEEE International Conference on Computer Vision (ICCV)*, 2015.

[6] A. Ranjan and M. J. Black, "Optical flow estimation using a spatial pyramid network," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[7] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz, "Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[8] Z. Teed and J. Deng, "Raft: Recurrent all-pairs field transforms for optical flow," in *European Conference on Computer Vision (ECCV)*, 2020.

[9] F. Zhang, O. J. Woodford, V. A. Prisacariu, and P. H. Torr, "Separable flow: Learning motion cost volumes for optical flow estimation," in *IEEE International Conference on Computer Vision (ICCV)*, 2021.

[10] H. Xu, J. Zhang, J. Cai, H. Rezatofighi, and D. Tao, "Gmflow: Learning optical flow via global matching," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[11] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox, "Flownet 2.0: Evolution of optical flow estimation with deep networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[12] T.-W. Hui, X. Tang, and C. C. Loy, "Liteflownet: A lightweight convolutional neural network for optical flow estimation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[13] N. Mayer, E. Ilg, P. Hausser, 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 Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[14] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black, "A naturalistic open source movie for optical flow evaluation," in *European Conference on Computer Vision (ECCV)*, 2012.

[15] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

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

[17] S. Meister, J. Hur, and S. Roth, "Unflow: Unsupervised learning of optical flow with a bidirectional census loss," in *AAAI Conference on Artificial Intelligence (AAAI)*, 2018.

[18] Y. Zou, Z. Luo, and J.-B. Huang, "Df-net: Unsupervised joint learning of depth and flow using cross-task consistency," in *European Conference on Computer Vision (ECCV)*, 2018.

[19] P. Liu, M. Lyu, I. King, and J. Xu, "Selflow: Self-supervised learning of optical flow," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[20] H.-Y. Lai, Y.-H. Tsai, and W.-C. Chiu, "Bridging stereo matching and optical flow via spatiotemporal correspondence," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[21] F. Aleotti, M. Poggi, and S. Mattoccia, "Learning optical flow from still images," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[22] C. Chi, Q. Wang, T. Hao, P. Guo, and X. Yang, "Feature-level collaboration: Joint unsupervised learning of optical flow, stereo depth and camera motion," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[23] 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*, 2022.

[24] J. Watson, O. M. Aodha, D. Turmukhambetov, G. J. Brostow, and M. Firman, "Learning stereo from single images," in *European Conference on Computer Vision (ECCV)*, 2020.

[25] R. Ranftl, K. Lasinger, D. Hafner, K. Schindler, and V. Koltun, "Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2022.

[26] Z. Yin and J. Shi, "Geonet: Unsupervised learning of dense depth, optical flow and camera pose," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[27] K. Xian, C. Shen, Z. Cao, H. Lu, Y. Xiao, R. Li, and Z. Luo, "Monocular relative depth perception with web stereo data supervision," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.