25^{th} Computer Vision Winter Workshop
Domen Tabernik, Alan Lukežič, Klemen Grm (eds.)
Rogaška Slatina, Slovenia, February 3–5, 2020

A new semi-supervised method improving optical flow on distant domains

Tomáš Novák, Jan Šochman, Jiří Matas
Center for Machine Perception, Department of Cybernetics, FEE CTU in Prague
Technická 2, 166 27 Prague 6, Czech Republic
{novakt34,jan.sochman}@fel.cvut.cz

Abstract. *We propose a semi-supervised approach to learning by formulating the optimization as constrained gradient descent on a loss function that includes unsupervised terms. The method is demonstrated on semi-supervised optical flow training that promotes photo-consistency and smoothness of the flow. We show that the unsupervised objective significantly improves the estimation on a distant domain while maintaining the performance on the original domain. As a result, we achieve state-of-the-art results on the Creative Flow+ dataset among CNN-based methods that did not train on any samples from the dataset.*

1. Introduction

Supervised learning of CNN methods achieves state of the art results on all major optical flow datasets. However, when presented with samples that are distant to their training set, they often produce inconsistent estimates. We find that unsupervised optical flow methods, possibly due to their less domain-dependent objective, perform better in this setting. However, they fall short of supervised methods when enough labeled training data is available. We aim to combine the performance of the supervised methods with the robustness of the unsupervised methods.

This paper presents a new semi-supervised training approach that combines supervised and unsupervised objectives. The training optimization is formulated as a constraint gradient descent that takes gradients from both losses; however, skips all unsupervised samples that lead to worse performance on the supervised samples i.e., all unsupervised gradients that have a negative dot product with the supervised gradient are omitted. The method is tested on optical flow estimation, and it is shown that it makes the network perform close to the unsupervised methods on data from a distant domain while maintaining the performance on the labeled domain.

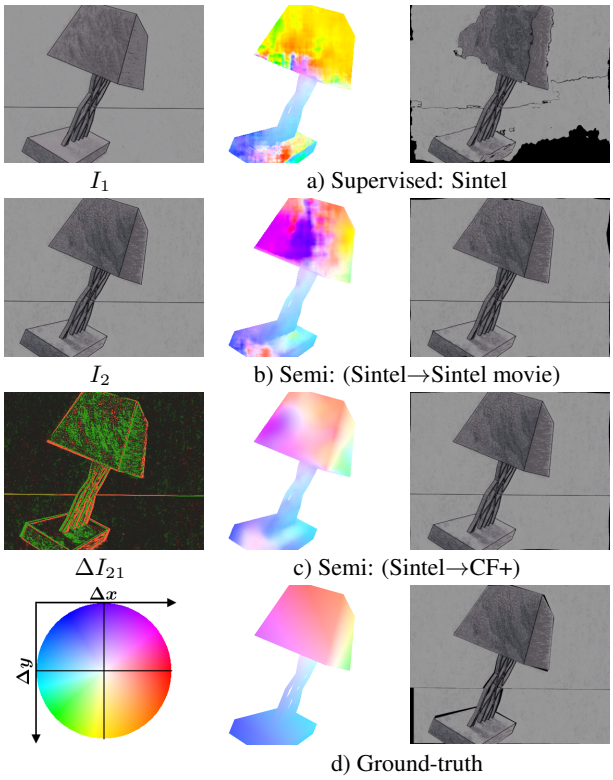

Figure 1: **Optical flow on a distant domain without/with semi-supervision**. Left: A sample pair of images I_1, I_2 from the Creative Flow+ (CF+) dataset, their difference ΔI_{21} and optical flow color coding wheel. Notice texture changes on both the object and background. Middle: Foreground optical flow for supervised and semi-supervised models with GT. Right: I_2 warp to I_1 showing geometric consistency of the flow. The Sintel fine-tuned model (a) corrupts optical flow in major parts. The proposed constrained semi-supervision on Sintel domain alone (b) improves the estimates. Further improvement is achieved by adding unlabeled CF+ samples (c).

More specifically, we demonstrate this behavior on a recently published Creative Flow+ dataset [24]. The dataset features artistic-like scenes with untextured regions or objects with changing texture. All

supervised CNN-based approaches fine-tuned on another domain (Sintel) produce highly inaccurate estimates in this setting. The method is able to retrain a supervised model mitigating this effect. We demonstrate that even without using the distant domain samples, we already get a significant performance gain. Upon introducing the images from the distant domain (with no GT), we are able to bring the error on the distant domain even lower.

The contributions of this paper are the following. First, a novel method to combine supervised and unsupervised objectives is presented. The training is formulated as constrained gradient descent on a loss function that includes terms from unsupervised training - i.e., in the optical flow estimation photoconsistency, smoothness, and forward-backward consistency.

Second, we demonstrate that when supervised training leads to abrupt estimates on a distant domain, introducing the unsupervised objective using the proposed semi-supervised method improves results on the distant domain. However, the model performance on the supervised domain does not drop. This effect is observed even without using any samples from the other domain.

Finally, we show that adding unlabeled samples from the distant domain improves the results on the distant domain even more.

2. Related work

Supervised training. *FlowNet* [7] was the first work to introduce end-to-end supervised training of optical flow. The authors proposed two CNN architectures as well as a large synthetic dataset *FlyingChairs* that was needed to train the network in a supervised fashion. This work demonstrated that neural networks are able to act as an optical flow estimator.

Many other architectures and training techniques were proposed since [12, 10, 28, 21, 27, 11] improving results on standard optical flow benchmarks [4, 8, 20] and surpassing the classical approaches.

Though our method applies to any end-to-end trainable network, we chose to build our experiments on *PWC-Net* [28] architecture, since it is a popular choice among current approaches. It combines a pyramidal approach with correlation cost volume on each level. Furthermore, the correlation is done on encoder features instead of images.

Unsupervised/self-supervised training. There is also a class of unsupervised or self-supervised techniques that aim to train the optical flow network without any ground truth, just from frame pairs (or more frames) themselves [1, 33, 23, 2]. This means they do not rely on any labeling, which in the optical flow context is nontrivial to obtain, and can thus be trained on potentially unlimited size of data.

They apply the same principles from the famous Horn–Schunck method [9] or many related [26] to create a training signal for the network. The main task is to assess the optical flow quality without any ground-truth. This is mostly done by measuring the photometric difference between the source image and the back-warped target image. Other objectives, such as smoothness or consistency between forward and backward flow, are added.

This work is further developed by adding occlusion reasoning [30, 19, 13] and so-called *data distillation* [16, 17]. Furthermore, attempts to train algorithms that combine optical flow with other tasks were done [32, 22, 14].

Fully unsupervised training is, however, not able to compete with the supervised training on the conventional optical flow datasets. They struggle with photometric deviations like occlusions, motion blur, reflections, et cetera. Even the ability to use much more training data than supervised approaches does not compensate.

Semi-supervised training. If we do omit cases of unsupervised pre-training and supervised fine-tuning, there were only a few attempts in the optical flow context to create a combination of supervised and unsupervised training.

A simple supervised and unsupervised loss combination was presented in [31, 34]. Lai et al. [15] present an approach based on a Generative Adversarial Network. The discriminator is trained to recognize the photometric difference map between the source and target image back-warped by either ground truth or estimated optical flow. Further, endpoint error loss is applied alongside the adversarial loss for all labeled data.

3. Method

The goal is to combine supervised and unsupervised training. In this section, the proposed constrained semi-supervision method is first introduced, then the loss terms used throughout the work are listed.

3.1. Semi-supervision: constraint gradient descent

At each iteration during training, the network is evaluated on one pair of frames with ground-truth (supervised sample) and N pairs without (unsupervised samples). The gradient from the supervised sample poses a reasonable (but not optimal) constraint that skips all unsupervised samples leading to worse performance on the supervised samples.

Let Θ be the network parameters. By back-propagation, the gradient

$$\mathbf{G}_s = \nabla L_{sup}(\Theta) \tag{1}$$

is computed for the supervised sample and

$$\mathbf{G}_u^n = \nabla L_{un}(\Theta) \tag{2}$$

for n-th unsupervised sample[1].

$\mathbf{G}_s$ is used as the constraining vector. Positive dot product with the constraining vector ensures that the added $\mathbf{G}_u^i$ does not have an orientation opposite to $\mathbf{G}_s$. Thus, the parameter update vector is defined as:

$$\mathbf{G} = \mathbf{G}_s + \sum_{\forall i: \mathbf{G}_u^i \cdot \mathbf{G}_s > 0} \lambda_M \mathbf{G}_u^i \tag{3}$$

Thus, by updating the parameters by $\mathbf{G}$, the value of L_{sup} linearized at Θ will not rise. However, some updates from unsupervised loss are still considered.

3.2. Loss terms

Let I_1, I_2 be two consecutive frames and $\mathbf{f}_{GT,1\to2}$ ground truth forward flow. Let $l = 1 \dots 5$ be the flow pyramid scale from the largest ¼ to the smallest 1⁄64 of the input image size. Let $\mathbf{f}^l_{1\to2}, \mathbf{f}^l_{2\to1}$ be the estimated forward and backward flow on the scale l. By I^l and $\mathbf{f}^l_{GT}$ we denote an image resp. flow down-sampled to the scale l.

Supervised loss is the standard L2 endpoint-error loss [28]:

$$L_{sup}(\mathbf{f}_{1\to2}) = \sum_{l=1}^{5} \alpha_l \sum_{\mathbf{x}\in P} \left\| \mathbf{f}^l_{1\to2}(\mathbf{x}) - \mathbf{f}^l_{GT,1\to2}(\mathbf{x}) \right\|_2 . \tag{4}$$

Data term. The data term is based on [19]; however, we drop the occlusion-awareness since it has not proven beneficial in our setting. The term is defined as

$$\begin{aligned} L^l_D(\mathbf{f}^l_{1\to2}, \mathbf{f}^l_{2\to1}) = \\ \sum_{\mathbf{x}\in P} \rho\Big(f_D\big(I^l_1(\mathbf{x}), I^l_2(\mathbf{x} + \mathbf{f}^l_{1\to2}(\mathbf{x}))\big)\Big) + \\ \rho\Big(f_D\big(I^l_2(\mathbf{x}), I^l_1(\mathbf{x} + \mathbf{f}^l_{2\to1}(\mathbf{x}))\big)\Big), \end{aligned} \tag{5}$$

where $\rho(x) = (x^2 + \epsilon^2)^\gamma$ (default $\gamma = 0.45$) is the Charbonnier penalty [26] that increases robustness to outliers. f_D measures the photometric difference between two pixels. The experiments are done with both brightness constancy constraint (per channel) [33] and the ternary census transform adjusted for loss function in [19].

Smoothness term. Second order smoothness constraint is employed as in [19], since it has been proved to be beneficial in classical flow estimation methods , [29]. To decrease over-smoothing on object edges, we combine it with edge awareness [13].

$$\begin{aligned} L^l_S(\mathbf{f}^l_{1\to2}, \mathbf{f}^l_{2\to1}) = \\ \sum_{\mathbf{x}\in P} \sum_{(\mathbf{s},\mathbf{r})\in N(x)} \sigma\big(I^l_1, \mathbf{f}^l_{1\to2}, \mathbf{s}, \mathbf{x}, \mathbf{r}\big) + \\ \sigma\big(I^l_2, \mathbf{f}^l_{2\to1}, \mathbf{s}, \mathbf{x}, \mathbf{r}\big), \end{aligned} \tag{6}$$

where $N(\mathbf{x})$ contains horizontal, vertical and both diagonal neighborhoods of $\mathbf{x}$ and σ measures the edge-aware smoothness:

$$\begin{aligned} \sigma(I, \mathbf{f}, \mathbf{s}, \mathbf{x}, \mathbf{r}) = \\ \rho\big(\mathbf{f}(\mathbf{s}) - 2\mathbf{f}(\mathbf{x}) + \mathbf{f}(\mathbf{r})\big) \cdot \\ \cdot \exp\big(- \|I(\mathbf{x}) - I(\mathbf{s})\|_2 \big) \cdot \\ \cdot \exp\big(- \|I(\mathbf{x}) - I(\mathbf{r})\|_2 \big). \end{aligned} \tag{7}$$

We assume $\rho(\cdot)$ computes the average over the penalties from each component.

FW-BW consistency. Adding the forward-backward consistency term also proved to help with learning the flow [19]:

$$\begin{aligned} L^l_C(\mathbf{f}^l_{1\to2}, \mathbf{f}^l_{2\to1}) = \\ \sum_{\mathbf{x}\in P} \rho\Big(\mathbf{f}^l_{1\to2}(\mathbf{x}) - \mathbf{f}^l_{2\to1}\big(\mathbf{x} + \mathbf{f}^l_{1\to2}(\mathbf{x})\big)\Big) + \\ \rho\Big(\mathbf{f}^l_{2\to1}(\mathbf{x}) - \mathbf{f}^l_{1\to2}\big(\mathbf{x} + \mathbf{f}^l_{2\to1}(\mathbf{x})\big)\Big). \end{aligned} \tag{8}$$

Unsupervised loss is defined as a weighted sum over loss terms and pyramid levels:

$$L_{un} = \sum_{l=1}^{5} \alpha_l \big(L^l_D + \lambda_S L^l_S + \lambda_C L^l_C \big) \tag{9}$$

where α_l is the pyramid scale weight.

[1] To ease the notation, we omit some obvious arguments from the loss function.

4. Experiments

This section describes the structure of experiments and the technical details. Results are discussed in the next section.

Overall, the experiments examine the domain transfer ability of supervised, unsupervised and semi-supervised training from Sintel dataset [4] to Creative Flow+ [24]. First, supervised and unsupervised models are tested and their performance is observed. Afterwards, the proposed constrained semi-supervision is put to the test in two settings - limited to samples from Sintel domain or also including unlabeled frames from CF+. For comparison, we try to pose the semi-supervision as a simple loss combination and also test a baseline supervised on both Sintel and CF+. All experiments were done with the popular PWC-Net [28] architecture.

To denote the experiments, a system of abbreviations in the format "[*training method*]: ([*datasets*])" is used. Training method is either supervised (Sup), unsupervised (Unsup) or semi-supervised (Semi). Plus sign "+" denotes the training was done on a combination of two datasets. With semi-supervised training, arrow "$\rightarrow$" separates a dataset serving as the source of supervised samples from a dataset of unsupervised samples.

4.1. Datasets

In the experiments, we use the following datasets. The letter in the bracket next to the dataset name is the abbreviation used in the experiments.

Sintel (S) [4]. To avoid complicated online evaluation, a 90-10 split of the publicly-available data to training and testing parts was created yielding 1562 train and 2×87 test samples (separately clean and final pass). In training, both *Clean* and *Final* passes are combined.

Sintel movie (Sm). All frames from the original movie [5] were extracted for unsupervised and semi-supervised training, similarly to [17]. To cope with compression artifacts, we downscaled the 4K resolution images to 1152×648. Cuts between scenes, where no optical flow exists, were avoided with PySceneDetect [6]. Moreover, too dim (typical for fade ins/outs) or too similar consecutive images were detected using pixel-wise brightness resp. brightness difference and excluded. Altogether, 9372 samples were created.

KITTI 2015 (K) [20]. Testing is done on all 200 annotated samples. Unsupervised methods train on 13K samples from the multiview extensions of KITTI'15 and '12 [8]. Frames from the annotated pairs are excluded.

Creative Flow+ (CF+) [24] is a recently introduced dataset with artistic-like scenes and ground truth optical flow. Tests are done on the 10K sample list provided by the authors. Some of the experiments also use the set of 153K *mixamo* train frames. Full resolution images (1500×1500) are used. Note that it is more meaningful to observe performance on the foreground areas since optical flow on the background is often not well defined.

4.2. Supervised training distant domain performance

First, to establish an overview of how supervised models perform on a distant domain, their performance is tested on CF+, similarly to [24]. The two pre-trained *PWC-Net* models made available by authors [28] are evaluated. One was trained on FlyingChairs (C) [7] and FlyingThings3D (T) [18] datasets, the second was fine-tuned for the Sintel [4] dataset. The experiments are denoted as *Sup: (C,T)* and *Sup: (C,T,S)*.

4.3. Unsupervised training distant domain performance

Next, we make a similar overview for the unsupervised training and its distant domain transfer ability. Two unsupervised models are trained, one with per-channel brightness constancy constraint, another with census transform data term. We name the models *Unsup [brightness]: (C,K+S)* and *Unsup [Census]: (C,K+S)* respectively.

Tests with different parameter settings and training protocols resulted in the following training procedure. To initialize the models, a pre-training phase consisting of 240K iterations on FlyingChairs dataset [12] is performed with unsup. loss L_{un}, regularization $\lambda_S = 3.0$, no forward-backward consistency ($\lambda_C = 0$) and f_D as a brightness or difference. Learning rate starts with $1e-4$ and is halved every 100K iterations. Input image size is 512×384. Fine-tuning is done on all KITTI and Sintel samples with the same setting, apart from activated consistency term $\lambda_C = 0.3$ and f_D as brightness or census difference respectively. With the brightness difference, convergence is reached after 455K iterations, 746K iterations are needed for the census difference. Images are cropped to 896×320.

4.4. Semi-supervision on single domain

The previous overview shows that unsupervised models have a better distant domain transfer ability, but suffer from low accuracy on the original domain. We therefore attempt to introduce the transfer ability of unsupervised methods to a well-performing supervised model using the proposed semi-supervision method.

The *PWC-Net* model trained (supervised) for Sintel dataset by the authors [28] is fine-tuned using the constrained semi-supervision method taking supervised samples from Sintel and unsupervised samples from Sintel movie dataset. We refer to this experiment as *Semi: (S→Sm)*.

In order to establish a control experiment, we also continue training with supervised loss only (labeled as *Sup: (C,T,S) - modif. supervision*).

In the experiments we tested multiple hyperparameter settings and ended with the following one: One supervised and six unsupervised samples are fed to the method at each iteration. We set $\lambda_M = 0.1$, f_D as per-channel brightness constancy constraint. Frames are cropped to 768×384. To warm-up the optimization, first three epochs are performed just with supervised loss and are followed by 2 semi-supervised epochs with small learning rate 1e-7. Afterwards, we perform 133K iterations with learning rate 1e-5 that is halved after 30K, 50K, 70K, 90K, 105K and 120K iterations.

4.5. Semi-supervision including distant domain

Next, the idea of the previous experiment is developed further by taking unlabeled samples from the distant domain.

The network is trained in the same way as in the previous experiment with the only difference that the unsupervised samples are taken from the training part of the CF+ dataset (i.e., frames only, no GT flow). We name the experiment as *Semi: (S→CF)*.

4.6. Unconstrained semi-supervision

To test the need for the constrained semi-supervision method, an experiment without any constraining takes place. The loss is simply defined as a combination of supervised and unsupervised terms

$$L_{comb} = L_{sup} + \lambda_U L_{un} \tag{10}$$

as e.g. in [31].

We refer to this experiment as *Uncons. semi: (S)*. Again, the experiment starts with the Sintel fine-tuned network as in previous sections. The network is trained with L_{comb} as a loss function on the Sintel dataset with $\lambda_S = 3.0, \lambda_C = 0.3$ and f_D as a brightness constancy constraint.

We test three settings of the unsupervised loss weight λ_U = $0.1, 1$ and 2. In all three cases, a CF+ test error drop occurs in the first 30K iterations, however, it is followed by a rise even above the control (*Sup: (C,T,S) - modif. supervision*) experiment. At the same time, with all three λ_U settings, both terms of the loss L_{sup} and L_{un} are decreasing during training. This suggests that L_{comb} leads to an over-fitting on Sintel in unsupervised objective.

In the final results table, we state the situation before the error rise for $\lambda_U = 0.1$ and 1.

4.7. Supervised training

To establish a supervised comparison, we also fine-tune the PWC-Net model for the CF+ dataset in a supervised manner. We refer to the experiment as *Sup: (C,T,S,S+CF)*.

In each training epoch, we train on all Sintel training samples and the same number of randomly chosen CF+ samples. We train for 171K iterations starting with learning rate 1e-5 that is gradually halved.

4.8. Common technical details

This subsection describes the common technical details of the training.

In all experiments, Adam optimizer is used with default $\beta_1 = 0.9, \beta_2 = 0.999$. Batch size is four with the exception of semi-supervised experiments. As in the original PWC-Net paper [28], the pyramid weights are $\alpha_1 = 0.005$, $\alpha_2 = 0.01$, $\alpha_3 = 0.02$, $\alpha_4 = 0.08$, $\alpha_5 = 0.32$.

Census photometric difference is computed on different window sizes at each pyramid scale, from the largest to the smallest scale it is: $7 \times 7, 7 \times 7, 5 \times 5, 3 \times 3, 3 \times 3$.

For data augmentation, both common and relative (between frames in a pair) geometric transforms are used: random rotation, translation, scale, squeeze, flip, and crop. Photometric transforms are also included: random gamma, brightness, contrast, and relative color channel brightness changes.

Error measures. *EPE* refers to an average endpoint error

$$\frac{1}{\sum_{P \in S} |A(P)|} \sum_{P \in S} \sum_{x \in A(P)} \left\| \mathbf{f}^P_{1 \to 2}(\mathbf{x}) - \mathbf{f}^P_{GT,1 \to 2}(\mathbf{x}) \right\|_2, \tag{11}$$

Method	CF+ AEPE [px] ALL	CF+ AEPE [px] median ALL	CF+ AEPE [px] median FG	Sintel AEPE [px] Clean	Sintel AEPE [px] Final	KITTI 2015 [%] Fl-all
Horn-Schunck [9]	8.34	3.49	12.17	8.73∗	9.61∗	–
Classic+NLfast [25]	13.35	7.05	9.27	9.12∗	10.08∗	–
Brox2011 [3]	9.05	3.27	8.28	7.56∗	9.11∗	–
Sup: (C,T) [28]	66.97	41.88	22.77	2.44	3.82	34.3
Sup: (C,T,S) [28]	74.23	33.54	18.21	1.78	2.41	10.6
Sup: (C,T,S) - modif. supervision	30.44	14.73	11.30	1.69	2.22	14.7
Unsup [brightness]: (C,K+S)	10.60	4.80	7.99	5.23	6.18	40.2
Unsup [Census]: (C,K+S)	15.06	9.05	8.65	4.22	5.19	25.1
Uncons. semi: (S) $\lambda_U = 0.1$	25.76	15.19	10.63	1.79	2.19	12.2
Uncons. semi: (S) $\lambda_U = 1$	24.91	15.32	9.95	2.54	3.10	22.0
Semi: (S→Sm)	17.36	8.41	8.91	1.81	2.49	16.9
Semi: (S→CF)	7.88	3.79	6.65	1.79	2.25	18.9
Sup: (C,T,S,S+CF)	8.19	3.54	5.62	1.81	2.24	17.4

Table 1: **Main results table.** All numbers except columns marked median and Fl-all, are mean endpoint errors over all test samples. Fl-all denotes outlier ratio (>3px and >5% EPE), median is computed across individual sample average EPEs. Dataset abbreviations: C: Flying Chairs [12], T: FlyingThings3D [18], S: Sintel [4], Sm: Sintel movie, CF: Creative Flow+[24], K: KITTI unlabeled multiview extension [8, 18]. For classical methods, we list the results from [24]. Results marked with a star (*) come from the official test benchmark.

	Creative Flow+ AEPE [px]									
		median		Style, FG				Speeds, FG		
	ALL	ALL	FG	flat	toon	tex	stylit	<1%	1-3%	>3%
Sup: (C,T) [28]	66.97	41.88	22.77	41.18	10.86	16.09	23.67	23.17	17.84	32.73
Sup: (C,T,S) [28]	74.23	33.54	18.21	24.71	7.03	17.46	21.77	17.50	15.18	30.94
Sup: (C,T,S) - modif. supervision	30.44	14.73	11.30	7.79	6.42	13.62	14.76	8.48	12.36	28.22
Unsup [brightness]: (C,K+S)	10.60	4.80	7.99	7.67	5.90	9.01	8.85	4.90	9.54	25.51
Unsup [Census]: (C,K+S)	15.06	9.05	8.65	7.83	5.93	9.94	9.99	5.47	9.81	27.79
Uncons. semi: (S) $\lambda_U = 0.1$	25.76	15.19	10.63	11.14	5.99	12.35	12.19	8.26	10.99	26.19
Uncons. semi: (S) $\lambda_U = 1$	24.91	15.32	9.95	11.24	5.72	11.75	10.86	7.76	10.24	24.5
Semi: (S→Sm)	17.36	8.41	8.91	7.20	5.66	10.66	10.79	5.95	10.18	26.23
Semi: (S→CF)	7.88	3.79	6.65	6.85	5.32	8.94	6.19	3.47	8.68	23.58
Sup: (C,T,S,S+CF)	8.19	3.54	5.62	5.84	4.61	9.10	4.36	2.94	7.21	20.20

Table 2: **Detailed results of the presented methods on CF+.** We list the same metrics as in the original paper [24]. All numbers except column marked median, are average endpoint errors. Median is computed across individual sample average EPEs. Performance is broken down into All (full frame) and FG (foreground) as well as by style and speed (<1% ground-truth optical flow length less than 1% of the frame size i.e. 15 px, 1-3% between 15 and 45 px, >3% over 45 px).

where S is a set of test samples, $A(P)$ defines the area of interest (whole image, foreground pixels etc.) and $\mathbf{f}^P_{1\to 2}$ is the flow estimated on sample P scaled to original image size.

Fl-all is an error measure proposed for the KITTI'15 dataset, where there is an uncertainty in optical flow measurements. It is defined as the percentage of optical flow outliers i.e., flow end-point error is $> 3px$ and $> 5\%$ of GT flow.

5. Results and discussion

This section discusses the results of the experiments described in the previous section. The results of the experiments are listed in Table 1, qualitative assessment is presented in Figure 2. Extended evaluation on the Creative Flow+ dataset is shown in Table 2.

Supervised training. First, we observe that the supervised methods fail on the CF+ dataset, see *Sup: (C,T)* and *Sup: (C,T,S)* in Table 1. Figure 2 indicates abruptly outlying estimates on constant intensity regions. Problems also occur on object texture changes. We get slightly different results to [24], possibly due to a different framework, however, the conclusion is the same.

Unsupervised training. With unsupervised training, the models do not suffer from the distant domain transfer issues - the performance on the CF+ dataset is significantly better, as shown in Table 1, *Unsup [brightness]: (C,K+S)* and *Unsup [Cen-*

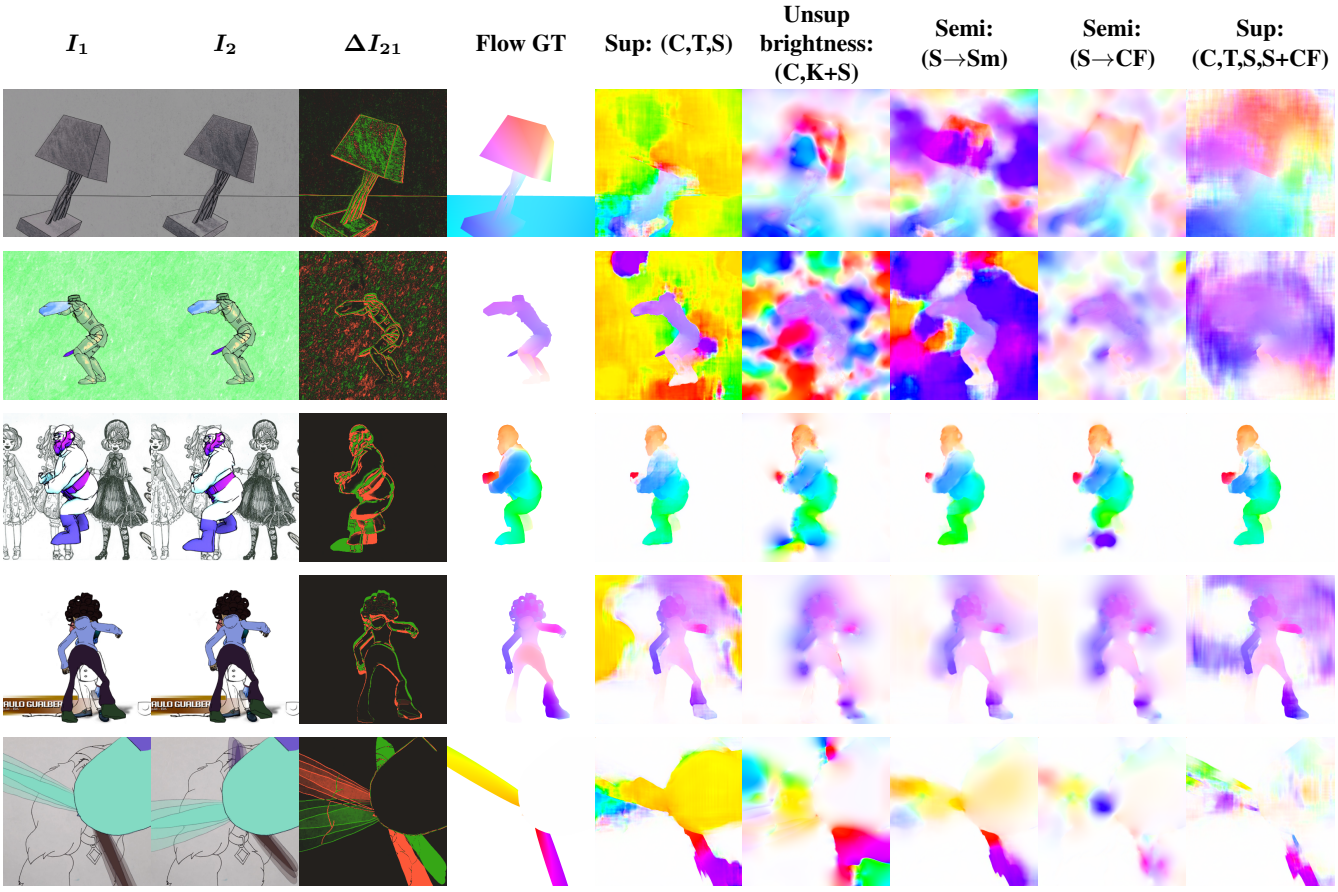

Figure 2: **Qualitative assessment.** Input images (first two columns) with a color coded difference visualization (third column); the ground truth flow and flow estimates for selected methods (following columns).

sus]: (C,K+S). Figure 2 shows that the estimated flow field is smoother, with no abrupt outliers. However, the test errors on Sintel and KITTI dataset stay far behind the supervised models.

We hypothesize that although the unsupervised objective is unable to properly handle the effects of occlusions, motion blur, local ambiguities, etc., yet, it is more universal than training for a supervised objective on a single domain. Therefore, we expect it to perform better on a distant domain.

Semi-supervision on single domain. Semi-supervision attempts to combine the observed distant domain transfer ability of unsupervised models with the accuracy of supervised models on Sintel and KITTI.

Table 1, *Semi: (S→Sm)*, shows that constrained semi-supervision training significantly drops the test error on CF+ while the error on Sintel changes just slightly. Curiously, this is done without introducing any CF+ samples.

We attribute the increased CF+ accuracy partially to our way of supervised training, which seems to decrease the error on CF+ as shown by our control experiment *Sup: (C,T,S) - modif. supervision*. It is most likely caused by differences in augmentations, probably skipping additive white noise in our setting.

However, semi-supervision leads to a significant decrease, suggesting that adding the unsupervised loss with the proposed method makes the model perform closer to unsupervised methods on a distant domain with only minor changes on the Sintel domain.

Semi-supervision including distant domain. When the semi-supervised model is explicitly presented with the samples from CF+, the error on this distant domain drops significantly to the level of the unsupervised methods (Table 1 - *Semi: (S→CF)*). Note that the error is also significantly below the semi-supervision on a single domain. Again, the error on Sintel stays virtually the same.

We hypothesize that since the images from the other domain are presented, the network starts to recognize it and optimize the unsupervised criterion specifically on these samples. However, the super-

vised constraint prevented to apply the same criterion on the supervised samples.

Unconstrained semi-supervision. Unconstrained semi-supervision tested the need for the proposed constrained semi-supervision method by formulating the training as a simple linear combination of supervised and unsupervised losses.

As Table 1 *Uncons. semi: (S)* shows, the performance on CF+ is similar for both λ_U settings, especially on the foreground regions. On Sintel and KITTI, low $\lambda_U = 0.1$ preserves the accuracy of the initial model; however, a significant error rise is observed with higher $\lambda_U = 1$.

The observations correspond to the expectations - with small unsupervised term weight, the training is not able to introduce the unsupervised objective to the model. When we attempt to promote it more with higher λ_U, the accuracy on the supervised domain is lost.

Supervised CF training. Supervised training on CF+ is able to improve the performance on the dataset while maintaining the accuracy on Sintel (see Table 1, *Sup: (C,T,S,S+CF)*). Evaluated on the whole frames, it does not surpass constrained semi-supervision. However, as it was already mentioned, the background flow is often not well defined; thus, this metric is not as relevant.

The performance margin to a constrained semi-supervision on the foreground areas is not as large as e.g., the margin between supervised and unsupervised methods on Sintel, suggesting that CF+ features complicated scenes that are hard to solve even with supervision.

6. Conclusion

In this paper, we propose a semi-supervision method by constraining the unsupervised update by the supervised gradient.

The experiments show that the proposed constrained semi-supervision method leads to a better performance in distant domain transfer while maintaining the performance on the supervised (i.e., Sintel) domain. Some improvement is already observed when introducing the unsupervised objective only on a single domain, even better results are achieved when the unlabeled samples from the distant domain are included. Our control experiment was not able to prove that the same effect is achieved by an unconstrained formulation.

As it could be foreseen, supervised training on the distant domain improves the results even further, but the margin is not as significant as expected.

References

[1] A. Ahmadi and I. Patras. Unsupervised convolutional neural networks for motion estimation. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 1629–1633, Sept. 2016. 2

[2] S. Alletto, D. Abati, S. Calderara, R. Cucchiara, and L. Rigazio. TransFlow: Unsupervised Motion Flow by Joint Geometric and Pixel-level Estimation. *arXiv:1706.00322 [cs]*, June 2017. arXiv: 1706.00322. 2

[3] T. Brox and J. Malik. Large displacement optical flow: descriptor matching in variational motion estimation. *IEEE transactions on pattern analysis and machine intelligence*, 33(3):500–513, 2010. 6

[4] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A Naturalistic Open Source Movie for Optical Flow Evaluation. In A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, editors, *Computer Vision – ECCV 2012*, Lecture Notes in Computer Science, pages 611–625. Springer Berlin Heidelberg, 2012. 2, 4, 6

[5] C. Levy (Director). Sintel. Blender Institute, 2010. 4

[6] B. Castellano. Breakthrough/PySceneDetect, Nov. 2019. [Online] https://github.com/Breakthrough/PySceneDetect. 4

[7] A. Dosovitskiy, P. Fischer, E. Ilg, P. Häusser, C. Hazırbaş, V. Golkov, P. v. d. Smagt, D. Cremers, and T. Brox. FlowNet: Learning Optical Flow with Convolutional Networks. In *IEEE International Conference on Computer Vision (ICCV)*, 2015. 2, 4

[8] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. 2, 4, 6

[9] B. K. Horn and B. G. Schunck. Determining Optical Flow. Technical report, Massachusetts Institute of Technology, Cambridge, MA, USA, 1980. 2, 6

[10] T.-W. Hui, X. Tang, and C. Change Loy. Liteflownet: A lightweight convolutional neural network for optical flow estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8981–8989, 2018. 2

[11] J. Hur and S. Roth. Iterative Residual Refinement for Joint Optical Flow and Occlusion Estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5754–5763, 2019. 2

[12] 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)*, July 2017. 2, 4, 6

[13] J. Janai, F. Guney, A. Ranjan, M. Black, and A. Geiger. Unsupervised Learning of Multi-Frame Optical Flow with Occlusions. pages 690–706, 2018. 2, 3

[14] H.-Y. Lai, Y.-H. Tsai, and W.-C. Chiu. Bridging Stereo Matching and Optical Flow via Spatiotemporal Correspondence. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1890–1899, 2019. 2

[15] W.-S. Lai, J.-B. Huang, and M.-H. Yang. Semi-Supervised Learning for Optical Flow with Generative Adversarial Networks. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 354–364. Curran Associates, Inc., 2017. 2

[16] P. Liu, I. King, M. R. Lyu, and J. Xu. DDFlow: Learning Optical Flow with Unlabeled Data Distillation. *arXiv:1902.09145 [cs]*, Feb. 2019. arXiv: 1902.09145. 2

[17] P. Liu, M. Lyu, I. King, and J. Xu. SelFlow: Self-Supervised Learning of Optical Flow. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4571–4580, 2019. 2, 4

[18] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4040–4048, 2016. 4, 6

[19] S. Meister, J. Hur, and S. Roth. UnFlow: Unsupervised Learning of Optical Flow With a Bidirectional Census Loss. In *Thirty-Second AAAI Conference on Artificial Intelligence*, Apr. 2018. 2, 3

[20] M. Menze and A. Geiger. Object Scene Flow for Autonomous Vehicles. pages 3061–3070, 2015. 2, 4

[21] M. Neoral, J. Šochman, and J. Matas. Continual Occlusion and Optical Flow Estimation. In *Asian Conference on Computer Vision*, pages 159–174. Springer, 2018. 2

[22] A. Ranjan, V. Jampani, L. Balles, K. Kim, D. Sun, J. Wulff, and M. J. Black. Competitive Collaboration: Joint Unsupervised Learning of Depth, Camera Motion, Optical Flow and Motion Segmentation. *arXiv:1805.09806 [cs]*, May 2018. arXiv: 1805.09806. 2

[23] Z. Ren, J. Yan, B. Ni, B. Liu, X. Yang, and H. Zha. Unsupervised Deep Learning for Optical Flow Estimation. In *Thirty-First AAAI Conference on Artificial Intelligence*, Feb. 2017. 2

[24] M. Shugrina, Z. Liang, A. Kar, J. Li, A. Singh, K. Singh, and S. Fidler. Creative Flow+ Dataset. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1, 4, 6

[25] D. Sun, S. Roth, and M. J. Black. Secrets of optical flow estimation and their principles. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2432–2439, June 2010. ISSN: 1063-6919. 6

[26] D. Sun, S. Roth, and M. J. Black. A Quantitative Analysis of Current Practices in Optical Flow Estimation and the Principles Behind Them. *International Journal of Computer Vision*, 106(2):115–137, Jan. 2014. 2, 3

[27] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz. Models Matter, So Does Training: An Empirical Study of CNNs for Optical Flow Estimation. *arXiv:1809.05571 [cs]*, Sept. 2018. arXiv: 1809.05571. 2

[28] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz. PWC-Net: CNNs for Optical Flow Using Pyramid, Warping, and Cost Volume. pages 8934–8943, 2018. 2, 3, 4, 5, 6

[29] W. Trobin, T. Pock, D. Cremers, and H. Bischof. An Unbiased Second-Order Prior for High-Accuracy Motion Estimation. In *DAGM-Symposium*, 2008. 3

[30] Y. Wang, Y. Yang, Z. Yang, L. Zhao, P. Wang, and W. Xu. Occlusion Aware Unsupervised Learning of Optical Flow. pages 4884–4893, 2018. 2

[31] X. Xiang, M. Zhai, R. Zhang, Y. Qiao, and A. El Saddik. Deep Optical Flow Supervised Learning With Prior Assumptions. *IEEE Access*, 6:43222–43232, 2018. 2, 5

[32] Z. Yin and J. Shi. GeoNet: Unsupervised Learning of Dense Depth, Optical Flow and Camera Pose. pages 1983–1992, 2018. 2

[33] J. J. Yu, A. W. Harley, and K. G. Derpanis. Back to Basics: Unsupervised Learning of Optical Flow via Brightness Constancy and Motion Smoothness. In G. Hua and H. Jégou, editors, *Computer Vision – ECCV 2016 Workshops*, Lecture Notes in Computer Science, pages 3–10. Springer International Publishing, 2016. 2, 3

[34] M. Zhai, X. Xiang, R. Zhang, N. Lv, and A. El Saddik. Learning Optical Flow Using Deep Dilated Residual Networks. *IEEE Access*, 7:22566–22578, 2019. 2