Masked Label Learning for Optical Flow Regression

Guorun Yang, Zhidong Deng, Shiyao Wang and Zeping Li
State Key Laboratory of Intelligent Technology and Systems,
Beijing National Research Center for Information Science and Technology
Department of Computer Science, Tsinghua University, Beijing 100084, China
ygr13@mails.tsinghua.edu.cn, michael@mail.tsinghua.edu.cn,
sy-wang14@mails.tsinghua.edu.cn, li-zp16@mails.tsinghua.edu.cn

***Abstract*—Optical flow estimation is a challenging task in computer vision. Recent methods formulate such task as a supervised-learning problem. But it often suffers from limited realistic ground truth. In this paper, a compact network, embedded with cost volume, residual encoder and deconvolutional decoder, is presented to regress optical flow in an end-to-end manner. To overcome the lack of flow labels, we propose a novel data-driven strategy called masked label learning, where a large amount of masked labels are generated from the FlowNet 2.0 model and filtered by warping calibration for model training. We also present an extended-Huber loss to handle large displacements. With pretraining on massive masked flow data, followed by finetuning on a small number of sparse labels, our method achieves state-of-the-art accuracy on KITTI flow benchmark.**

I. Introduction

Optical flow estimation is a popular task in computer vision. It has a variety of applications, such as object tracking [1], motion detection [2], action recognition [3] and visual odometry [4].

Classical approaches attempt to solve the estimation of optical flow as an energy minimization process with variational methods. These approaches usually fail on the cases of large displacements from fast motion [5], [6]. Later methods introduce descriptor matching algorithms to find matching correspondences on adjacent frames [7], [8], [9] and adopt coarse-to-fine schemes or advanced interpolation techniques [10], [11], [12] to adapt for the scenarios of large displacements. Convolutional neural networks (CNNs) are utilized to describe image patches and bring further improvements on flow estimation. Nevertheless, the whole process of these approaches is time-consuming because they often involve multiple steps, including feature extraction, matching selection, interpolation, and flow refinement.

Recently, fully-convolutional network (FCN) [13] is introduced to optical flow estimation that enables the end-to-end learning of dense flow maps [14], [15], [16]. Generally, the deep models with FCN structure for flow regression require a large number of labels for training. However, it is high-cost to annotate enough flow data if we use manual selections or extra equipments. To solve the lack of labels, computer graphical techniques are employed to synthesize flow datasets, such as FlyingChairs [14] and FlyingThings3D [17], but there remains a gap between synthetic virtual images and realistic scenes, which limits the adaptation of models. In addition, several approaches begin to use an unsupervised fashion to train models by photometric differences between source images and reconstructed images [18], [19]. Although unsupervised approaches can overcome the drawback of insufficient labeled data, such models are difficult to behave well on the regions of object boundaries, local ambiguities, and textureless areas, etc.

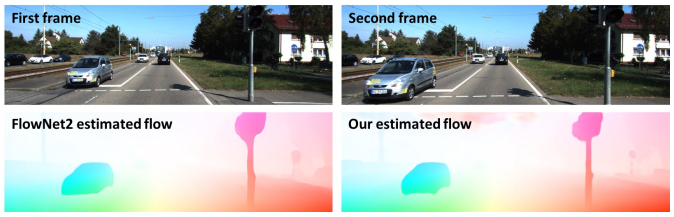

Fig. 1. An example of predicted flow map on KITTI Raw dataset [20]. Top: input frames. Bottom left: FlowNet 2.0 [16] estimated flow map. Bottom right: Our estimated flow map. We colorize the flow maps with the tool provided by Sintel [21].

In this paper, we argue that sufficient data is still a prerequisite for deep model training and therefore propose a novel data-driven strategy named as masked label learning. Instead of using synthetic data or unsupervised scheme, we employ an existing method FlowNet 2.0 [16] to generate a large amount of flow maps on target scenes. To reduce the potential errors in the generated maps, we add an extra filtering step by warping calibration. Each pixel in the flow map is checked by the photometric distance between referenced image and warped image. The flow values that cannot pass the verification will be masked out and excluded from subsequent model training.

For the network architecture, we design an encoder-decoder model. In this model, the correlation layer [14] is introduced as the head part to compute cost volume on feature pairs. The residual network (ResNet) [22] is embedded as the main body to learn image features and encode matching information. Three deconvolutional layers are appended at the end of structure to upsample feature maps and regress the final dense flow map. Without complex cascaded networks like FlowNet 2.0 [16] or coarse-to-fine patterns like SpyNet [15], our model can predict favorable results after feeding mass data.

Different from the classification task, the regression of optical flow needs to estimate real values and it often adopts ℓ_1, ℓ_2 or Charbonnier norm as loss function [23]. However, these functions are easily disturbed by possible outliers in labels or large deviations between predicted values and ground-truth. Here, we extend traditional Huber loss [24] with a square-root term to alleviate this problem. Comparative results

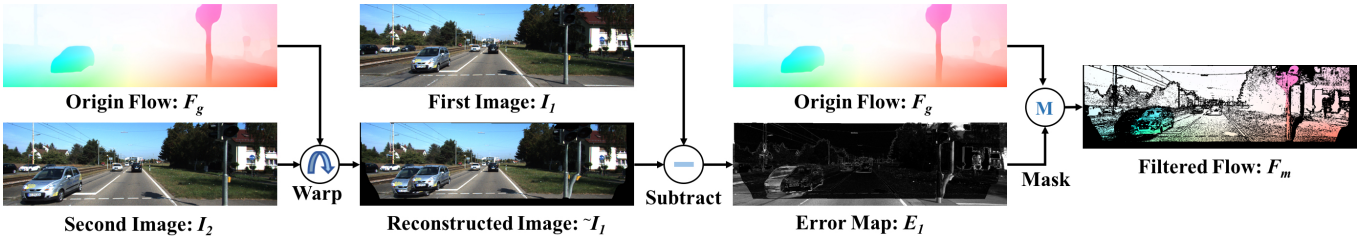

Fig. 2. Schematic diagram of masked label preparation. The origin flow map is generated from Flownet 2.0 [16], followed by label filtering by warping calibration.

illustrate that the extended-Huber loss is more robust to large displacements in road scenes.

In addition to pretraining on the masked data, we also finetune the model based on a few sparse labels provided by KITTI flow dataset [20], [25]. As shown in Fig. 1, after the finetuning, our model can estimate finer result than FlowNet 2.0 [16]. Our method also outperforms other domain-agnostic approaches (distinguished from the approaches with extra stereo or multiview information) on KITTI flow 2012 benchmark [20], which demonstrates the effectiveness of our strategy. Furthermore, the qualitative results on video sequences on Cityscapes dataset [26] illustrate the adaptability of our model. The contributions of this work are summarized below:

- We propose a data-driven strategy of masked label learning where a large amount of flow labels are generated by the FlowNet 2.0 [16] model and filtered by warping calibration.
- We develop a compact model integrated with a cost volume, residual blocks, and deconvolutional layers, which enables the end-to-end optical flow regression.
- An extended-Huber loss function is presented to train the model, which is more robust to large displacements.
- Our method achieves state-of-the-art results on KITTI Flow 2012 dataset [20]. The results on Cityscapes dataset [26] also show its adaptability.

II. Related Work

Research on optical flow could be traced back to basic variational approach proposed by Horn and Schunck [5]. Such pioneer work couples the brightness constancy and global smoothness assumption to an energy-minimization process. Based on the variational method, Black and Ananda present a robust framework to deal with outliers in both the data and spatial terms [6]. Subsequent works explore more robust functions [27], [28] or introduce better constraints [29], [30]. However, most variational methods are difficult to handle the case of large displacements, so that feature matching algorithms are introduced to the variational framework [7], [8], [9], [31]. In addition, some approaches focus on interpolation methods to obtain dense optical flow maps from sparse matchings [10], [11], [12].

With CNN models that show great capability on high-level vision tasks [32], researchers attempt to adopt CNN models to represent image features and learn the optical flow. For example, Gadot and Wolf suggest using a siamese CNN to compute the descriptors of input pair of images [33]. Bailer et al. calculate CNN-based features on different scales combined with a thresholded hinge loss for training [8]. Xu et. al compute the cost volume on compact features extracted from CNN model and adapt semi-global matching for accurate flow results [34]. Besides, several methods leverage extra constraints. Bai et al. utilize instance-level segmentation with epipolar prior to improve flow results in traffic scenes [35]. Hur and Roth exploit forward-backward consistency and occlusion symmetry to estimate optical flow [36].

Inspired by the success of FCN model applied in semantic segmentation [13], Dosovitskiy et al. [14] design two FCN models, FlowNetS and FlowNetC, to regress the flow map in an end-to-end manner. To capture the motion between frames, the models are pretrained on a synthetic dataset. Ranjan and Black [15] embed spatial-pyramid formulation into deep network and make similar performance with FlowNetC model. IIg et al. [16] give an upgraded version called FlowNet 2.0 which cascades the basic models and significantly improves the predicted results. Here, we employ FlowNet 2.0 to generate flow labels and eventually our developed model outperforms the guided model on target benchmark.

Another family of research focus on the unsupervised learning. Yu et al. [18] train the FlowNet-based model with an unsupervised loss function measured via brightness constancy and smoothness assumptions. Meister et al. [19] define a bidirectional census loss to train network model and achieves competitive results with supervised methods on synthetic datasets.

Similar to our approach, Zhu et al. [37] present a guided flow learning method, where FlowFields [8] is employed to generate proxy flow labels for the training of CNN-based estimators. In our work, we further add warping calibration to obtain masked labels and design a new integrated model to learn flow maps, which leads to more accurate results.

III. Methods

In this section, we introduce the method of masked label learning for optical flow regression. First, we explain the label preparation including generation and filtration in details. Second, we describe the model architecture for flow learning. Third, the definition and characteristics of extended-Huber loss is discussed. Given a pair of consecutive images I_1 and I_2, our goal is to estimate a dense flow field $F_p : I_1 \rightarrow I_2$ between I_1 and I_2.

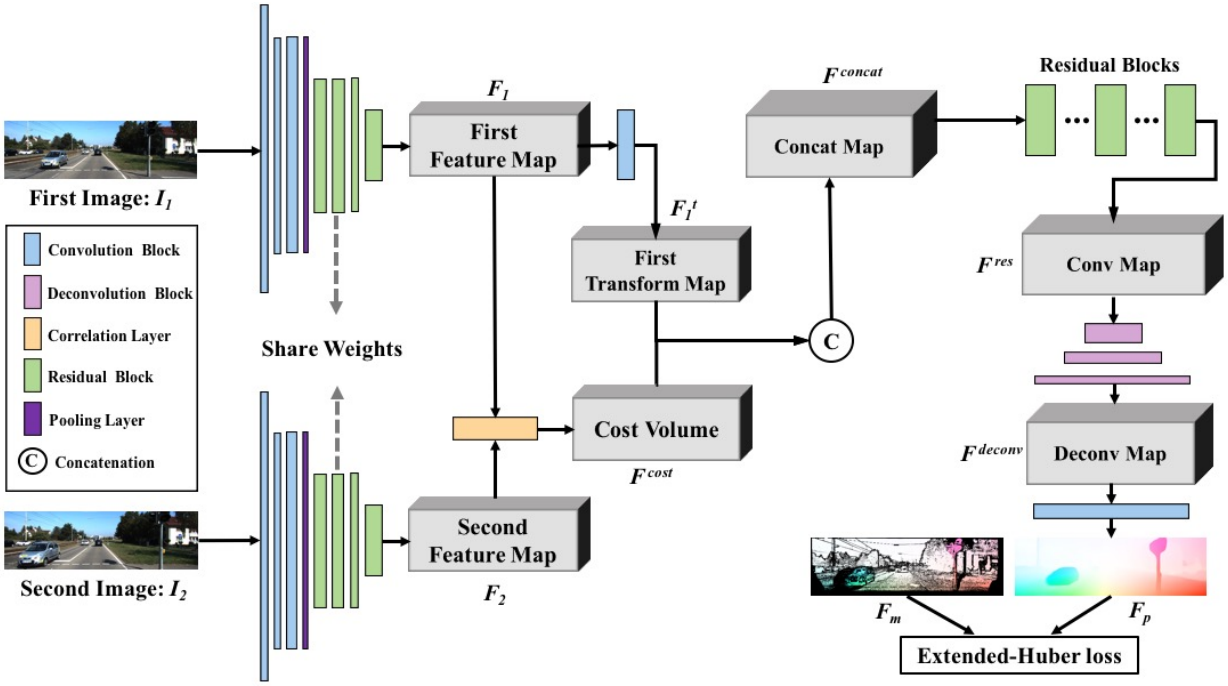

Fig. 3. Our model architecture.

A. Masked Label Preparation

It is difficult to collect flow labels by means of manual annotation on realistic scenes. Here, we employ FlowNet 2.0 [16] as the guided model and select raw KITTI dataset [20] as the target scene to generate flow labels. Unlike those synthetic datasets that provide entirely accurate labels [14], [17], there remain a few errors in our generated data. To reduce the adverse effects, we add a warping calibration where the reconstructed image is exploited to detect errors and conduct filtering. As shown in Fig. 2, the reconstructed image $\tilde{I}_1$ is inversely warped from the source image of second frame I_2 based on generated flow map F_g. Then we subtract the warped image $\tilde{I}_1$ and the source image of first frame I_1 to get error map E_1. If the errors exceed a pre-specified threshold δ, we mask out the flow labels on the original map F_g and obtain the final filtered flow map F_m. During training time, the loss calculated on the masked areas will not be backward propagated.

B. Model Architecture

The model architecture is illustrated in Fig. 3. Our model is composed of various components where we use distinct colors to indicate different blocks. The convolution block generally contains a convolutional layer followed by batch normalization (BN) and rectified linear unit (ReLU) layer, and the deconvolutional block replaces the convolutional layer with the deconvolutional layer. We introduce the correlation layer from FlowNet [14] to encode the matching cues between image pairs. The residual block is the basic unit of ResNet [22] which comprises three or four consecutive convolution blocks with split-transform-merge strategy.

Actually, a modified version of ResNet-50 network [22] is embedded into our model as the main body. Instead of computing cost volume on raw pixels of image pairs I_1 and I_2, we adopt feature descriptors from CNNs which are more robust with local context information. Specifically, we utilize the bottom part of ResNet-50, which contains three convolution blocks, a pooling layer and four residual blocks, to learn the feature maps F_1 and F_2. The spatial scale of F_1 and F_2 is downsampled to $1/8$ of the original image due to pooling and strided-convolution, with the shape of $h \times w \times c$, where h, w and c stand for feature height, width and channels. The cost volume F^{cost} is computed on F_1 and F_2 by correlation layer [14] to encode matching cues. Here, the max displacement parameter in correlation layer is set to d and the consequent cost volume F^{cost} has the shape of $h \times w \times (2d+1)^2$. The feature map on referenced frame (first frame) F_1 should not be abandoned due to its pixel-level information for preserving details. To this end, we apply another convolution block with kernel size 1×1 on feature map F_1 and obtain transformed feature ${F_1}^t$. We concatenate it together with the cost volume F^{cost} and get hybrid feature representation F^{concat}.

After concatenation, the feature F^{concat} is fed into the rest part of ResNet-50 structure, and then the feature map F^{res} is learned. To recover the spatial size, we append three deconvolution block to upsample the feature map accompanied with the reduction of channels. Behind F^{deconv}, we apply a convolutional layer with kernel size $3 \times 3 \times 2$ to regress the final flow map F_p with two channels. The extended-Huber loss is computed on the predicted flow map F_p and the flow label F_m.

C. Extended-Huber Loss

In flow regression task, large displacements or occlusions in source images, and potential outliers in labels easily result in excessive deviations between predict values and ground truth, which affects the convergence of training and the final performance of the model. Here, we extend the traditional Huber loss [24] with square root function for large deviations. We hope that the loss function can be more sensitive to small shifts and more robust to large disturbance. The whole regression loss L_{reg} is normalized as:

$$L_{reg}(y, \hat{y}) = \frac{1}{|N_v|} \sum_{i \in N_v} l_e(y_i - \hat{y}_i) \tag{1}$$

where y represents the predict flow value, $\hat{y}$ denotes the label value, N_v is the set of valid pixels, and the extended-Huber function $l_e(.)$ is defined as:

$$l_e(x) = \begin{cases} \frac{1}{2}x^2 & \text{if } |x| < 1, \\ |x| - \frac{1}{2} & \text{if } 1 \leq |x| < 4, \\ 4\sqrt{|x|} - \frac{9}{2} & \text{otherwise.} \end{cases} \tag{2}$$

IV. Experimental Results

In this section, we first train the model on KITTI raw dataset and then evaluate the performance on KITTI Flow datasets [20], [25]. The extended-Huber loss is compared with $L1$ loss and Huber loss. Both qualitative and quantitative results are given. We also submit the test images to KITTI Flow 2012 benchmark and test our model on Cityscapes dataset [26].

A. Implementation Details

a) Datasets: The KITTI dataset [20] is composed of real road scenes captured by vehicle-mounted cameras and laser scanners. It provides a small number of accurate yet sparse optical flow ground truth. In addition, a large amount of raw image sequences are provided without ground truth. For flow label preparation, we select 21,179 pairs of consecutive images from the city, residential, and road categories of the raw dataset. The "FlowNet2-ft-kitti" model [16] is employed to generate flow maps on the selected pairs, and these original flow maps are filtered as Fig. 2 to obtain masked flow labels, where the error threshold δ is set to 10. We further finetune our model on the 394 pairs of sparse labels, including 194 pairs from KITTI 2012 [20] and 200 pairs from KITTI 2015 [25]. We further select a sequence of "Bielefeld" city from Cityscapes [26] to test our model.

b) Evaluation Metrics: We mainly utilize the average end-point error (AEE) and the flow error (Fl) to evaluate the performance of models. The Fl represents the percentage of optical flow outliers which is more identifiable to the large flow values. The Fl errors on non-occluded regions (Noc) and all areas (All) are evaluated separately.

c) Training Details: Our implementation of the model is based on a customized Caffe version [38]. We use the "poly" learning rate policy where the momentum and weight decay are set to 0.9 and 0.0001 respectively. When pretraining on masked labels, the base learning rate is set to 0.01. When finetuning on sparse labels, we turn down the base learning rate to 0.001. For data augmentation, we randomly resize input images with a scale between 0.5 and 2.0 and crop them into 512×320. The batch size is set to 16 due to the limitation of GPU memory. At training time, the image list is shuffled to avoid similar samples. For the parameters in correlation layer, the max displacement and padding size are set to 32, which supports the maximum encoding range up to 256 on the original scale of input image.

B. Comparison Based on Different Loss Functions

TABLE I
RESULTS YIELDED ACCORDING TO DIFFERENT LOSS FUNCTIONS.

Loss Function	KITTI 2012			KITTI 2015		
	EPE	Fl-Noc	Fl-All	EPE	Fl-Noc	Fl-All
ℓ_1	1.01	4.39	6.88	2.01	9.67	13.86
Huber [24]	1.03	4.44	**6.74**	**1.95**	9.57	13.62
Extended-Huber	**0.99**	**4.14**	6.76	1.96	**8.92**	**13.09**

We compare the extended-Huber loss with normal ℓ_1 loss and Huber loss. Here, we use these three loss to pretrain the model on masked flow labels respectively. We set the max iterations to $200K$ so that about 150 epochs are conducted. In Table I, the EPE and Fl error are evaluated on both KITTI 2012 and 2015 datasets. We find that our presented extend-Huber loss is able to reduce the rate of bad predictions (Fl error), especially the "Noc" areas, with an average 6% improvement compared to ℓ_1 loss and an average 4% improvement compared to Huber loss. Meanwhile, the EPE of the three loss functions are at the same level. The result proves that the extended-Huber loss is more robust to the large displacements.

In Fig. 4, we give several examples of the model which is pretrained with extend-Huber loss. Here, the number of max iterations is set to $400K$ to fully exploit the potential of masked label data. Our model can handle challenging scenarios including fast motion, narrow street and traffic intersection. The regions such as shadows, occlusions and strong illuminations also have reliable estimates. There remain some local areas like object boundaries that need to be improved.

C. Results on KITTI 2012 Benchmark

Based on the pretrained model, we finetune it on the training samples of KITTI 2012 and 2015 datasets. The parameter of max iterations is set to $90K$. As shown in Fig. 5, some local details, such as poles, handrails and object boundaries, are refined by finetuning.

We submit the test images to the benchmark of KITTI Flow 2012. In Table II, we list the test results of recent domain-agnostic methods. The "$> x$ pixels" is also the error rate where x indicates the threshold to determine bad pixels. Our method outperforms other approaches on most tests. Especially on the

TABLE II

COMPARISON WITH OTHER MONOCULAR METHODS ON THE KITTI 2012 TEST DATASET[20]. OUR STRATEGY ACHIEVES STATE-OF-ART ACCURACY AND OUTPERFORMS OTHER METHODS BASED ON MOST EVALUATION METRICS.

Methods	$>$ 2 pixels		$>$ 3 pixels		$>$ 4 pixels		$>$ 5 pixels		EPE		Runtime
	Noc	All	Noc	All	Noc	All	Noc	All	Noc	All	
LDOF [7]	24.43	33.89	21.93	31.39	20.22	29.58	18.83	28.07	5.6 px	12.4 px	1 min
FlowNet [14]	49.33	55.34	37.05	44.49	29.36	37.56	24.11	32.67	5.0 px	9.1 px	**0.08 s**
SPyNet [15]	16.54	25.75	12.31	20.97	9.97	17.96	8.39	15.76	2.0 px	4.1 px	0.16 s
EpicFlow [10]	10.83	20.88	7.88	17.08	6.35	14.65	5.36	12.86	1.5 px	3.8 px	15 s
DeepFlow [39]	9.31	20.44	7.22	17.79	6.08	16.02	5.31	14.69	1.5 px	5.8 px	17 s
DiscreteFlow [9]	9.24	20.37	6.23	16.63	4.77	14.24	3.89	12.46	1.3 px	3.6 px	3 min
PatchBatch [33]	7.73	17.80	5.29	14.17	4.18	11.95	3.52	10.36	1.3 px	3.3 px	50 s
FlowFields [8]	7.33	16.69	5.57	14.01	4.02	10.98	3.95	10.21	1.4 px	3.5 px	23 s
RicFlow [12]	7.34	16.78	4.96	13.04	3.99	10.88	3.42	9.38	1.3 px	3.2 px	5 s
InterpoNet [11]	7.23	17.58	5.28	14.57	3.84	11.87	3.16	10.18	1.0 px	2.4 px	3 min
FlowField CNN [40]	7.42	16.87	4.89	13.01	3.72	10.68	3.04	9.06	1.2 px	3.0 px	23 s
FlowNet2 [16]	7.84	12.68	4.82	8.80	3.51	6.88	2.78	5.69	1.0 px	1.8 px	0.12s
CNNF + PMBP [41]	8.50	19.02	4.70	14.87	3.22	12.73	2.45	11.23	1.1 px	3.3 px	30 min
MirrorFlow [36]	6.10	10.70	4.38	8.20	3.55	6.88	3.02	6.02	1.2 px	2.6 px	11 min
UnFlow [19]	6.84	11.92	4.28	8.42	3.10	6.61	2.41	5.44	0.9 px	1.4 px	0.12 s
SDF [35]	**5.52**	**10.20**	3.80	7.69	3.03	6.40	2.56	5.56	1.0 px	2.3 px	–
Ours	6.51	10.80	**3.63**	**6.89**	**2.40**	**5.04**	**1.77**	**3.99**	**0.8 px**	**1.4 px**	0.8 s

Fig. 4. Qualitative results of our pretrained model on training pairs of KITTI Flow 2015 dataset. From left to right: input image of first frame, input image of second image, colorized flow prediction, error map. The error map is drawn by development kits provided by KITTI dataset, where the blue regions represent correct predictions and the red areas indicate incorrect estimates.

Fig. 5. Qualitative results of our final fine-tuned model on testing pairs of KITTI Flow 2012 dataset. From left to right: input image of first frame, input image of second image, colorized flow prediction, error map. The error map is captured from our submission, which scales linearly between 0 (black) and $>=$ 5 (white) pixels error. Red denotes all occluded pixels, falling outside the image boundaries.

index of "$>$ 5 pixels", our model achieves the Fl error of 1.77 on non-occluded regions and the Fl error of 3.99 on all regions, which gets an improvement of about 30% compared to SDF method [35]. The results demonstrate the effectiveness of our training strategy and model capability. The leaderboard can be seen on the website of KITTI Flow 2012 benchmark[1], and our method is abbreviated as "MLL".

[1]http://www.cvlibs.net/datasets/kitti/eval_stereo_flow.php?benchmark=flow

D. Results on Cityscapes

The Cityscapes dataset is a popular dataset of road scenes [26]. We select a sequence of "Bielefeld" city in this dataset to test our model. Compared to KITTI dataset [20], the images of Cityscapes dataset have distinct differences on scale and brightness. As shown in Fig. 6, our method can also estimate reasonable flow maps, which illustrates the adaptability of our model.

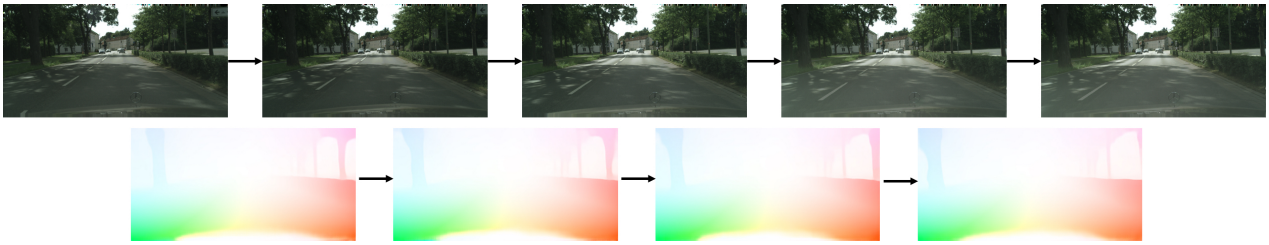

Fig. 6. Qualitative results of our model on a sequence of Cityscapes dataset. Top: a sequence of input images. Down: predicted flow maps.

V. Conclusion

In this paper, we address the task of optical flow estimation. Considering the lack of flow labels, a novel strategy of masked label learning is proposed to conduct label generation and warping filtering together to acquire a large amount of labels for model training. We design a compact network including cost volume, residual blocks, and deconvolutional blocks to learn optical flow map. Moreover, an extended-Huber loss is given to cope with large displacements. Experimental results on both KITTI Flow and Cityscapes datasets demonstrate the effectiveness of our method. In the future, we attempt to use unsupervised loss to further improve the performance of models.

Acknowledgment

This work was supported in part by the National Key Research and Development Program of China under Grant No. 2017YFB1302200, by research fund of Tsinghua University - Tencent Joint Laboratory for Internet Innovation Technology, and by the National Science Foundation of China (NSFC) under Grant Nos. 91420106, 90820305, and 60775040.

References

[1] J. D. L. Y. Y. W. Xizhou Zhu, Yuwen Xiong, "Deep feature flow for video recognition," in *CVPR*, 2017.
[2] I. Cohen and G. Medioni, "Detecting and tracking moving objects for video surveillance," in *CVPR*, 1999.
[3] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," 2014.
[4] A. Behl, O. H. Jafari, S. K. Mustikovela, H. A. Alhaija, C. Rother, and A. Geiger, "Bounding boxes, segmentations and object coordinates: How important is recognition for 3d scene flow estimation in autonomous driving scenarios?" in *ICCV*, 2017.
[5] B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artificial Intelligence*, 1981.
[6] M. J. Black and P. Anandan, "The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields," *Computer vision and image understanding*, 1996.
[7] T. Brox and J. Malik, "Large displacement optical flow: Descriptor matching in variational motion estimation," *TPAMI*, 2011.
[8] C. Bailer, B. Taetz, and D. Stricker, "Flow fields: Dense correspondence fields for highly accurate large displacement optical flow estimation," in *ICCV*, 2015.
[9] M. Menze, C. Heipke, and A. Geiger, "Discrete optimization for optical flow," in *GCPR*, 2015.
[10] J. Revaud, P. Weinzaepfel, Z. Harchaoui, and C. Schmid, "Epicflow: Edge-preserving interpolation of correspondences for optical flow," in *CVPR*, 2015.
[11] S. Zweig and L. Wolf, "Interponet, a brain inspired neural network for optical flow dense interpolation," in *CVPR*, 2017.
[12] Y. Hu, Y. Li, and R. Song, "Robust interpolation of correspondences for large displacement optical flow," in *CVPR*, 2017.
[13] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *CVPR*, 2015.
[14] 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 *ICCV*, 2015.
[15] A. Ranjan and M. J. Black, "Optical flow estimation using a spatial pyramid network," in *CVPR*, 2017.
[16] 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 *CVPR*, 2017.
[17] 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 *CVPR*, 2016.
[18] J. Y. Jason, A. W. Harley, and K. G. Derpanis, "Back to basics: Unsupervised learning of optical flow via brightness constancy and motion smoothness," in *ECCV Workshop*, 2016.
[19] S. Meister, J. Hur, and S. Roth, "Unflow: Unsupervised learning of optical flow with a bidirectional census loss," in *AAAI*, 2018.
[20] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *CVPR*, 2012.
[21] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black, "A naturalistic open source movie for optical flow evaluation," in *ECCV*, 2012.
[22] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.
[23] D. Sun, S. Roth, and M. J. Black, "Secrets of optical flow estimation and their principles," in *CVPR*, 2010.
[24] P. J. Huber, *Robust Estimation of a Location Parameter*. Springer New York, 1992.
[25] M. Menze and A. Geiger, "Object scene flow for autonomous vehicles," in *CVPR*, 2015.
[26] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *CVPR*, 2016.
[27] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert, "High accuracy optical flow estimation based on a theory for warping," in *ECCV*, 2004.
[28] D. Sun, J. P. Lewis, J. P. Lewis, and M. J. Black, "Learning optical flow," in *ECCV*, 2008.
[29] T. Nir, A. M. Bruckstein, and R. Kimmel, "Over-parameterized variational optical flow," *IJCV*, 2008.
[30] A. Wedel, D. Cremers, T. Pock, and H. Bischof, "Structure- and motion-adaptive regularization for high accuracy optic flow," in *ICCV*, 2009.
[31] Q. Chen and V. Koltun, "Full flow: Optical flow estimation by global optimization over regular grids," in *CVPR*, 2016.
[32] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NIPS*, 2012.
[33] D. Gadot and L. Wolf, "Patchbatch: A batch augmented loss for optical flow," in *CVPR*, 2016.
[34] J. Xu, R. Ranftl, and V. Koltun, "Accurate optical flow via direct cost volume processing," in *CVPR*, 2017.
[35] M. Bai, W. Luo, K. Kundu, and R. Urtasun, "Exploiting semantic information and deep matching for optical flow," in *ECCV*, 2016.
[36] J. Hur and S. Roth, "Mirrorflow: Exploiting symmetries in joint optical flow and occlusion estimation," in *ICCV*, 2017.
[37] Y. Zhu, Z. Lan, S. Newsam, and A. G. Hauptmann, "Guided Optical Flow Learning," in *CVPR Workshop*, 2017.
[38] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. B. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *ACM MM*, 2014.
[39] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid, "Deepflow: Large displacement optical flow with deep matching," in *ICCV*, 2014.
[40] C. Bailer, K. Varanasi, and D. Stricker, "Cnn-based patch matching for optical flow with thresholded hinge embedding loss," in *CVPR*, 2017.
[41] F. Zhang and B. W. Wah, "Fundamental principles on learning new features for effective dense matching." *TIP*, 2017.