A Perturbation-Constrained Adversarial Attack for Evaluating the Robustness of Optical Flow – Supplementary Material –

Jenny Schmalfuss, Philipp Scholze, and Andrés Bruhn

Institute for Visualization and Interactive Systems, University of Stuttgart, Germany
{jenny.schmalfuss,andres.bruhn}@vis.uni-stuttgart.de
philipp.scholze@simtech.uni-stuttgart.de

A Experiment Configurations

Tab. A1 summarizes the experimental configurations for all experiments that were conducted with our PCFA implementation. Different visualizations or representations of the same experiment class are grouped together. If several cells contain multiple choices, the experiment was conducted for all possible combinations of those. The only exception are ε_2 and μ, which should match by line. The attacks are evaluated on the networks FlowNet2 [2] (implementation from [9]), PWCNet [11], SpyNet [7] (implementation from [6]), RAFT [12] and GMA [3]. All network implementations use checkpoints that are not fine-tuned to the KITTI15 [5] data, other checkpoints and networks can easily be evaluated with our implementation (https://github.com/cv-stuttgart/PCFA).

B Additional Material

B.1 Run-time Complexity of PCFA Compared to Other Attacks

In our experiments, the final run-time mainly depends on the tested optical flow network. Hence it is reasonable to compare the number of backward passes for each attack. For image-specific perturbations, PCFA performs 10 backward passes per iteration (for one L-BFGS step), while I-FGSM needs a single backward pass. For universal perturbations trained with the same number of epochs and images and epochs, PCFA still performs 10 backward passes per iteration while the Patch Attack performs two. The more complex optimization of PCFA originates from the constrained optimization, which yields stronger adversarial perturbations than previous optical flow attacks, but requires a more sophisticated optimization routine.

Table A1. Parameters and Configurations for the experimental Results in Section 5. The evaluation- and test splits of the KITTI15 dataset [5] are denoted *K15-te* and *K15-tr* respectively; For the MPI-Sintel dataset [1] we use *S-te* and *S-tr*.

Experiment	Attack	Network	f^t	Loss	Box Constr.	Perturb. Type	ε_2	Penalty μ	Optim. Steps	Batch Size	Epo.	Dataset (train)	Dataset (eval)
Tab. 2,A2	PCFA	RAFT	0, $-f$	AEE	Clip, COV	δ_t, δ_{t+1}	$5 \cdot 10^{-3}$	$5 \cdot 10^5$	20	1	1	K15-te	K15-te
Tab. 2,A2	PCFA	RAFT	0	MSE, CS	Clip, COV	δ_t, δ_{t+1}	$5 \cdot 10^{-3}$	$5 \cdot 10^6$	20	1	1	K15-te	K15-te
Tab. 2,A2	PCFA	RAFT	$-f$	MSE, CS	Clip, COV	δ_t, δ_{t+1}	$5 \cdot 10^{-3}$	$7 \cdot 10^6$	20	1	1	K15-te	K15-te
Fig. 1,3,A1	PCFA	FlowNet2, PWCNet, SpyNet, GMA, RAFT	0	AEE	COV	δ_t, δ_{t+1}	$5 \cdot 10^{-2}$ $5 \cdot 10^{-3}$ $5 \cdot 10^{-4}$	$5 \cdot 10^4$ $5 \cdot 10^5$ $5 \cdot 10^6$	20	1	1	K15-te	K15-te
Fig. 4,A2	PCFA	FlowNet2, PWCNet, SpyNet, GMA, RAFT	0 $-f$ B2-41	AEE	COV	δ_t, δ_{t+1}	$1 \cdot 10^{-1}$ $1 \cdot 10^{-1}$ $1 \cdot 10^{-1}$	$1 \cdot 10^4$ $2 \cdot 10^4$ $2 \cdot 10^4$	20	1	1	K15-te	K15-te
Fig. 5, A3	PCFA	FlowNet2, PWCNet, SpyNet, GMA, RAFT	0	AEE	COV	δ_t, δ_{t+1}	$5 \cdot 10^{-2}$ $1 \cdot 10^{-2}$ $5 \cdot 10^{-3}$ $1 \cdot 10^{-3}$ $5 \cdot 10^{-4}$	$5 \cdot 10^4$ $1 \cdot 10^5$ $5 \cdot 10^5$ $1 \cdot 10^6$ $5 \cdot 10^6$	20	1	1	K15-te	K15-te
Fig. 5, A3	I-FGSM	FlowNet2, PWCNet, SpyNet, GMA, RAFT	0	AEE	COV	δ_t, δ_{t+1}	$5 \cdot 10^{-2}$ $1 \cdot 10^{-2}$ $5 \cdot 10^{-3}$ $1 \cdot 10^{-3}$ $5 \cdot 10^{-4}$	*n.a.*	10-25	1	1	K15-te	K15-te
Tab. 3, Fig. A4	PCFA	FlowNet2, PWCNet, SpyNet, GMA, RAFT	0	AEE	Clip	δ_t, δ_{t+1} $\delta_{t,t+1}$	$5 \cdot 10^{-3}$	$5 \cdot 10^5$	20	1	1	K15-te	K15-te
Tab. 3, Fig. A4	PCFA	FlowNet2, PWCNet, SpyNet, GMA, RAFT	0	AEE	Clip	$\overline{\delta_t, \delta_{t+1}}$ $\overline{\delta_{t,t+1}}$	$5 \cdot 10^{-3}$	$5 \cdot 10^5$	1	4	25	K15-te	K15-te
Tab. A3	PCFA	FlowNet2, PWCNet, SpyNet, GMA, RAFT	$-f$	AEE	Clip	δ_t, δ_{t+1} $\delta_{t,t+1}$	$5 \cdot 10^{-3}$	$1 \cdot 10^6$	20	1	1	K15-te	K15-te
Tab. A3	PCFA	FlowNet2, PWCNet, SpyNet, GMA, RAFT	$-f$	AEE	Clip	$\overline{\delta_t, \delta_{t+1}}$ $\overline{\delta_{t,t+1}}$	$5 \cdot 10^{-3}$	$1 \cdot 10^6$	1	4	25	K15-te	K15-te
Tab. 4, Fig. 6	PCFA	FlowNet2, PWCNet, SpyNet, GMA, RAFT	0	AEE	Clip	$\overline{\delta_{t,t+1}}$	$5 \cdot 10^{-3}$	$5 \cdot 10^5$	1	4	25	K15-tr	K15-te
Tab. A4, Fig. A5	PCFA	FlowNet2, PWCNet, SpyNet, GMA, RAFT	0	AEE	Clip	$\overline{\delta_{t,t+1}}$	$5 \cdot 10^{-3}$	$5 \cdot 10^5$	1	4	25	S-tr	S-te
Fig. A6	PCFA FGSM Patch	FlowNet2, PWCNet, SpyNet, GMA, RAFT	0	AEE	COV Clip Clip	δ_t, δ_{t+1} δ_t, δ_{t+1} $\overline{\delta_{t,t+1}}$	$5 \cdot 10^{-3}$ $5 \cdot 10^{-3}$ *n.a.*	$5 \cdot 10^5$ *n.a.* *n.a.*	20 10-25 1	1	1	K15-tr	K15-te

Table A2. PCFA adversarial robustness AEE($\check{f}, f$) for different *loss functions*, *targets* and *box constraints* on RAFT. Larger values indicate a bigger deviation between adversarial and initial flow.

	$f^t = 0$			$f^t = -f$		
	AEE	MSE	CS	AEE	MSE	CS
Clipping	29.12	22.96	0.00	44.11	29.09	**81.35**
COV	**29.31**	25.88	0.00	47.99	31.94	40.19

B.2 Additional Results for PCFA on Specific Frame Pairs

To complement the configuration study for PCFA that reported the attack strength in Main Tab. 2, we additionally provide the *adversarial robustness* measures for the tested configurations in Tab A2. For the zero-flow target and the cosine similarity (*CS*) loss, no deviation between the initial and the adversarial

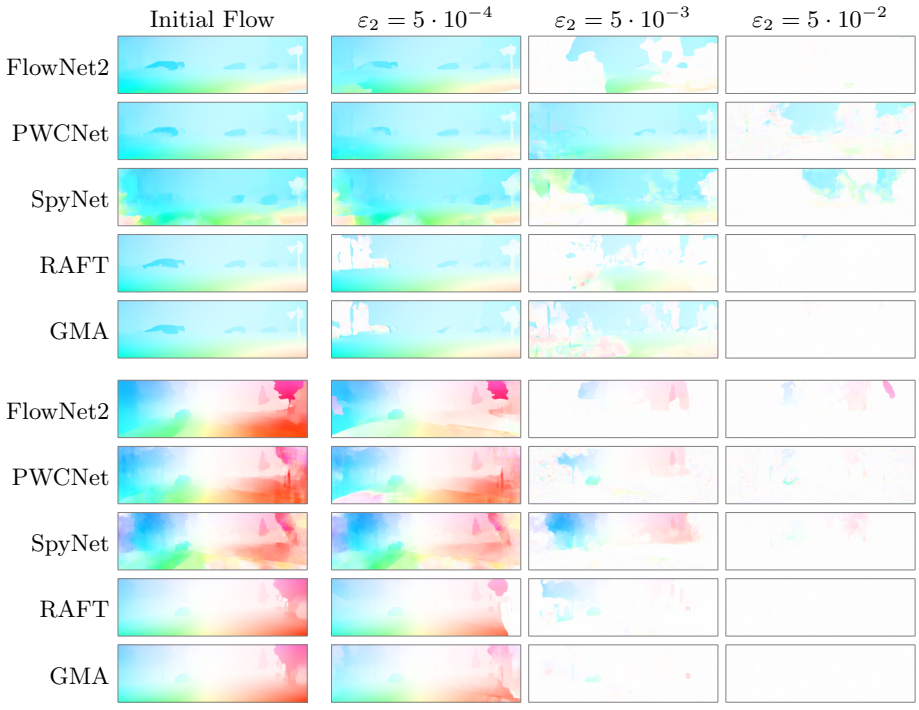


Fig. A1. Visual comparison of PCFA with zero-flow target on different *optical flow methods* for increasing *perturbation sizes* ε_2 on two exemplary scenes from KITTI15. White pixels represent the zero flow.

flow is induced, which shows that the cosine similarity does not train the perturbation towards the zero target. For the negative-flow target, it appears that greater deviations between adversarial and initial flow can be induced by the cosine similarity with clipping. However, a comparison to the target proximity in Main Tab. 2 clearly shows that the high deviation between adversarial and initial flow stems from a non-converging method, i.e. a very large distance to target, rather than from a strong targeted approach.

In the main paper we visualize the effect of increasingly large perturbations on the attacked flow in Main Fig. 1 and Main Fig. 3 for the networks FlowNet2, SpyNet and RAFT. The results for all networks and for an additional input frame pair are shown in Fig. A1. Similarly, Fig. A2 complements the reduced

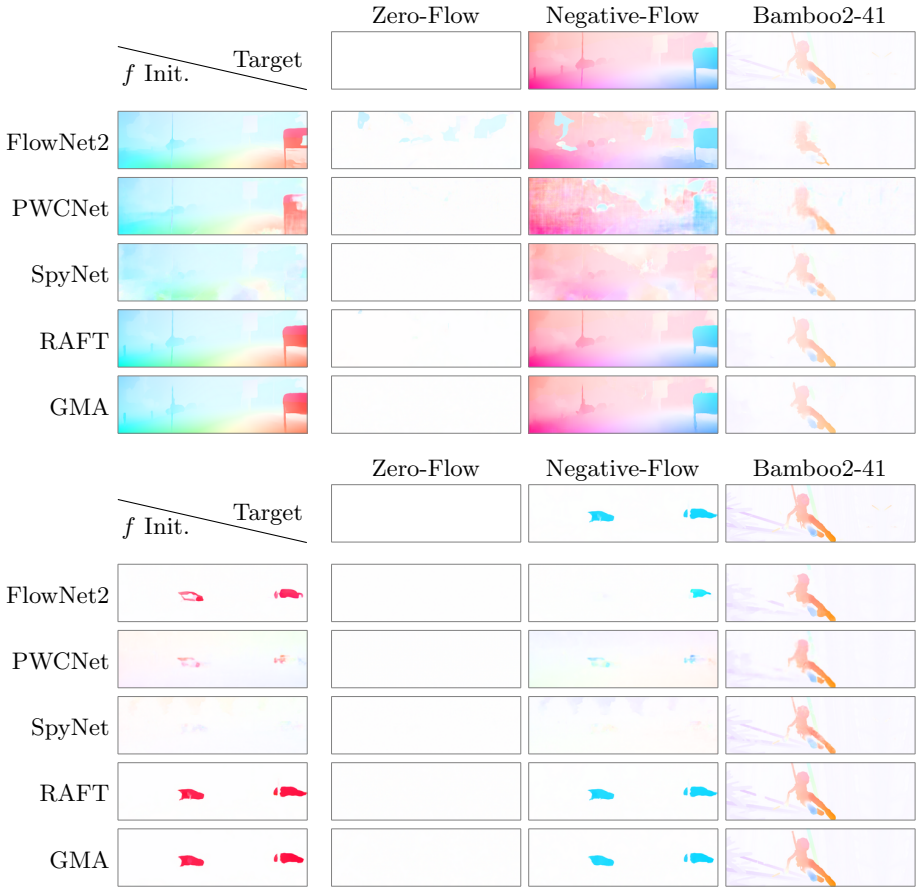
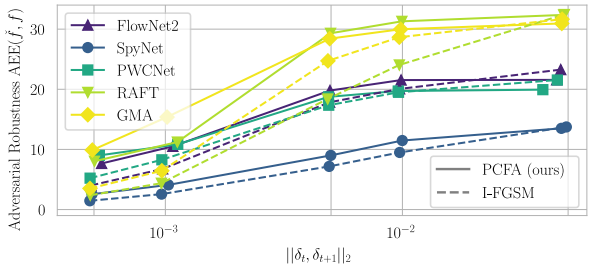


Fig. A2. Visual comparison of the result of PCFA with different *targets* on different *optical flow methods*. Choosing $\varepsilon_2 = 10^{-1}$ allows to come close to any target. *Bamboo2-41* is a ground truth flow from Sintel final's Bamboo2 sequence.



,

Fig. A3. *Adversarial robustness* with zero-flow target over *perturbation size*, for *PCFA* (solid) and *I-FGSM* [4, 10] (dashed) on different flow networks. Larger indicates a greater distance of the adversarial from the initial flow.

selection from Main Fig. 4 with additional illustrations of adversarial flows for the chosen targets on all tested networks.

In addition to Main Fig. 5, Fig. A3 visualizes the distance between adversarial and initial flow that is induced by perturbations with increasing size generated with PCFA and I-FGSM [4, 10]. For large perturbation sizes ($\varepsilon_2 \geq 10^{-2}$), it appears that I-FGSM can cause the adversarial robustness (Fig. A3) to degrade to a similar extent than PCFA. However, comparing to the target proximity in Main Fig. 5 again shows that the target proximity that is reached for I-FGSM is not as good as the one reached by PCFA. Consequently, I-FGSM perturbs the flow *away* from the initial flow, but does so in an untargeted manner as it fails to resemble the target flow. Meanwhile, PCFA has mostly converged to the zero flow for large perturbations and hence does not induce a further strong change, which explains why the distance between adversarial and initial flow does not increase further. Therefore, these results support that PCFA is a stronger method, and better suited than I-FGSM to generate perturbations that induce a desired target flow.

B.3 Additional Results for Joint and Universal Perturbations

Joint and Universal Perturbations. Tab. A3 extends the results in Main Tab. 3 by approximating the negative-flow target instead of the zero-flow. Both tables show the corresponding target proximity for different perturbation types generated with PCFA. Because the trend in Tab. A3 agrees with Main Tab. 3, where joint universal perturbations reach a better target resemblance than disjoint ones, both results suggests that the greater effectiveness of joint over disjoint universal perturbations is not related to the used target. A possible explanation for the better performance of the universal joint perturbations is that the batched training of universal perturbations tends to overfit on the batches, and may therefore not be able to learn better generalizing perturbations. In contrast, training one joint perturbation for both images explicitly incorporates a type of generalization (over input frames) into the optimization.

Table A3. Negative-flow target proximity for different *universal perturbations*.

Perturbation Type		FlowNet2	SypNet	PWCNet	RAFT	GMA
Frame-Specific	δ_t, δ_{t+1}	**14.73**	**15.33**	**13.36**	**19.86**	**16.74**
	$\delta_{t,t+1}$	19.07	18.54	15.60	30.35	20.48
Universal	$\overline{\delta_t, \delta_{t+1}}$	44.97	29.93	40.50	61.04	59.82
	$\overline{\delta_{t,t+1}}$	**43.06**	**29.64**	**39.39**	**60.15**	**58.35**

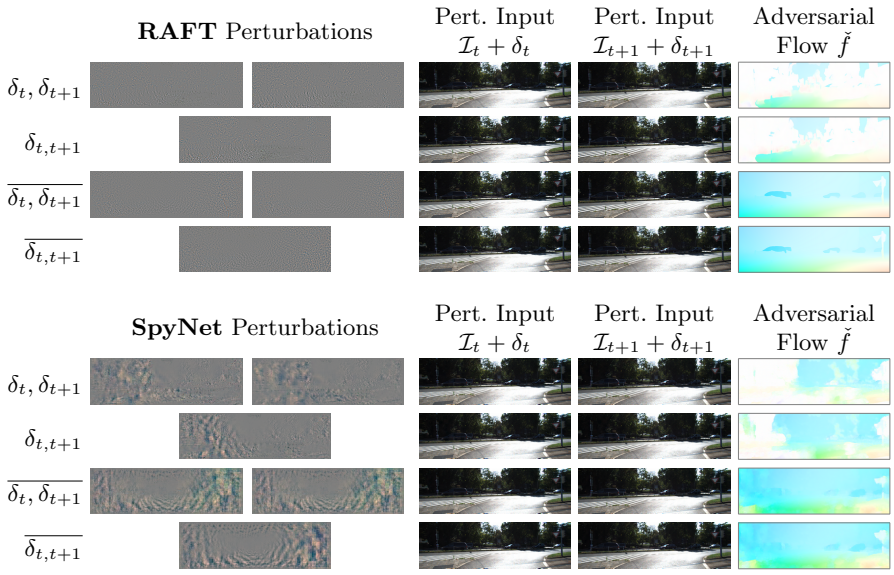


Fig. A4. Different perturbations types (*joint* and *universal*) for *RAFT* and *SpyNet* on an exemplary KITTI frame pair, generated with PCFA ($\varepsilon_2 = 5 \cdot 10^{-3}$, AEE loss, clipping box constraint).

Fig. A4 provides visualizations of the perturbations, perturbed inputs and adversarial flow for the zero-flow attack from Main Tab. 3 on the networks RAFT and SpyNet.

Transferability of Adversarial Perturbations. Tab. A4 shows the transferability of samples for the Sintel final dataset, and complements Main Tab. 4, which shows the same evaluation on KITTI. Again, universal joint perturbations were trained on the training set for a specific network (top row), and then applied to the test set and tested on all available networks (first column). On the Sintel dataset we observe clear trends among the networks in terms of robustness, where state-of-the-art networks like RAFT and GMA exhibit a consistent vulnerability to adversarial perturbations, while SpyNet's output is least distorted;

Table A4. Transferability of Sintel universal perturbations between *training* and *test* dataset and between different *networks*, measured as adversarial robustness AEE($\check{f}, f$). Large values denote a better transferability, smaller values indicate higher robustness.

Eval. \ Train.	FlowNet2	SpyNet	PWCNet	RAFT	GMA
FlowNet2	3.11	2.54	2.47	1.35	1.28
SpyNet	0.99	2.05	0.87	0.79	0.72
PWCNet	2.28	2.25	3.38	1.22	1.11
RAFT	**7.78**	**6.79**	**6.86**	**8.44**	**8.18**
GMA	6.77	5.72	5.79	7.28	7.22

irrespective of the network that was used to train the perturbation. Fig. A5 visualizes best universal perturbations for the Networks PWCNet and GMA, complementing the selection of networks from Main Fig 6.

Comparing the Patch Attack and PCFA. In Main Tab. 5, we compare the Patch Attack by Ranjan *et al.* [8] and our PCFA in terms of distance between the original and perturbed prediction. In both cases, the universal perturbations are trained on the Sintel final training set, and evaluated on test. For the networks listed in the original publication, we use the patches from [8]. As RAFT and GMA are not included in the original publication, we use the official code with standard settings to generate them, i.e. a learning rate of 10^3, 40 epochs, 100 images per epoch, two SGD steps per image. In the following we discuss to what extend and under which assumptions the reported adversarial robustness numbers for PCFA and the Patch Attack [8] are comparable.

Estimation of the per-pixel L_2 norm of Patch Attack. To roughly estimate the per-pixel L_2 norm for Patch Attack [8], we use the following assumptions. First, we assume the patch to introduce an additive distortion in the patch area, while adding zero in every location outside the patch P. Further, we assume the patch is contained in the image area $P \subset I$. And finally, we assume that the distortion adds a fixed value $\bar{b}$ to every location p within the patch P instead of

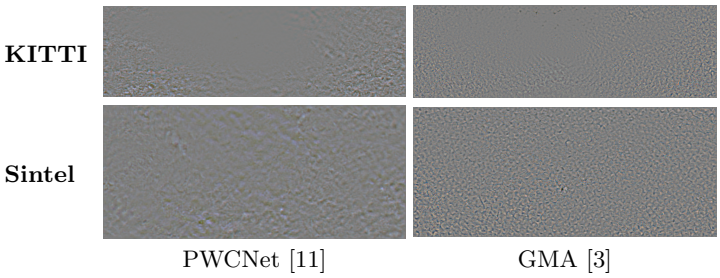


Fig. A5. Normalized *universal perturbations* for different network architectures learned from the respective training datasets. Top row: KITTI. Bottom row: Sintel.

individual values b_p. Please note that this is a very conservative assumption, since among all additive distortions that have a mean absolute value $\bar{b} = \frac{1}{P}\sum_{p\in P}|b_p|$ within the patch P, the constant distortion $b_p = \bar{b}$ has the smallest L_2 norm. Hence, in practice, the L_2 norm of Patch Attack will be larger than our estimate. The three aforementioned assumptions allow us to estimate the per-pixel L_2 norm of the patch distortion as

$$\varepsilon_2 = \frac{\|\delta_P\|_2}{\sqrt{I}} = \sqrt{\frac{\sum_{p\in P} b_p^2 + \sum_{p\notin P} 0^2}{I}} \overset{\substack{b_p=\bar{b}\\ p\in P}}{=} \sqrt{\frac{P\,\bar{b}^2}{I}} = \sqrt{\frac{P}{I}}|\bar{b}|\,. \tag{1}$$

In our comparison to the patch-based method of Ranjan *et al.* [8] we consider a patch of a 102 pixel diameter, which corresponds to approximately 8171 pixels. For a typical KITTI frame with a resolution of $I = 1242\times375$, this results in a perturbation of about 1.75% of all pixels. Given that the patches have colors which are rarely present in a typical KITTI frame, we conservatively estimate that the average additive perturbation $\bar{b}_{joint}$ per patch is about $3-30\%$ of the valid color range. With Eq. (1) this translates to an average color change over the whole frame of $0.40-3.97\%$. We compare the Patch Attack to PCFA with an L_2 bound of $\varepsilon_2 = 5\cdot10^{-3}$, which translates to an average change of 0.50% of the color range per pixel. This is at the lower end of our conservatively estimated range for the Patch Attack. Evidently, comparing two methods with different underlying concepts is difficult. However, with our calculations above, we aimed for a comparison that is as fair as possible under the methodological constraints.

B.4 Additional Results for Quality and Robustness with Multiple Attacks

Finally, we provide additional results for the joint quality and robustness evaluation on the KITTI15 dataset, where the robustness is also evaluated by taking the strongest configurations of FGSM and the Patch Attack in Fig. A6. Please note that the quality scores do not change, since the evaluation strictly separates quality from robustness. From the figure it becomes apparent that the networks

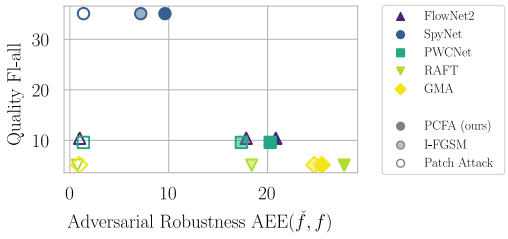


Fig. A6. Joint evaluation of optical flow methods by *prediction quality* and *adversarial robustness* on the KITTI test dataset, where the robustness is evaluated with the Patch Attack, FGSM and our PCFA.

appear more robust if they are evaluated with the weaker Patch Attack (empty markers) or FGSM (semi-transparent markers), hence PCFA (full markers) is a good choice if the robustness should be thoroughly assessed.

References

1. Butler, D.J., Wulff, J., Stanley, G.B., Black, M.J.: A naturalistic open source movie for optical flow evaluation. In: Proc. European Conference on Computer Vision (ECCV). pp. 611–625 (2012)
2. Ilg, E., Mayer, N., Saikia, T., Keuper, M., Dosovitskiy, A., Brox, T.: FlowNet 2.0: Evolution of optical flow estimation with deep networks. In: Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
3. Jiang, S., Campbell, D., Lu, Y., Li, H., Hartley, R.: Learning to estimate hidden motions with global motion aggregation. In: Proc. IEEE/CVF International Conference on Computer Vision (ICCV). pp. 9772–9781 (2021)
4. Kurakin, A., Goodfellow, I., Bengio, S.: Adversarial machine learning at scale. In: arXiv preprint 1611.01236 (2017)
5. Menze, M., Heipke, C., Geiger, A.: Joint 3D estimation of vehicles and scene flow. In: Proc. ISPRS Workshop on Image Sequence Analysis (ISA) (2015)
6. Niklaus, S.: A reimplementation of SPyNet using PyTorch (2018), `https://github.com/sniklaus/pytorch-spynet`
7. Ranjan, A., Black, M.J.: Optical flow estimation using a spatial pyramid network. In: Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
8. Ranjan, A., Janai, J., Geiger, A., Black, M.J.: Attacking optical flow. In: Proc. IEEE/CVF International Conference on Computer Vision (ICCV) (2019)
9. Reda, F., Pottorff, R., Barker, J., Catanzaro, B.: flownet2-pytorch: Pytorch implementation of FlowNet 2.0: Evolution of optical flow estimation with deep networks (2017), `https://github.com/NVIDIA/flownet2-pytorch`
10. Schrodi, S., Saikia, T., Brox, T.: Towards understanding adversarial robustness of optical flow networks. In: Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 8916–8924 (2022)
11. Sun, D., Yang, X., Liu, M.Y., Kautz, J.: PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume. In: Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
12. Teed, Z., Deng, J.: RAFT: Recurrent all-pairs field transforms for optical flow. In: Proc. European Conference on Computer Vision (ECCV). pp. 402–419 (2020)