Real-world evaluation of Optical Flow on repetitive patterns

Jesse Klijnsma[1]
Supervisor(s): Jan van Gemert, Sander Gielisse
[1]EEMCS, Delft University of Technology, The Netherlands

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
June 22, 2025

Name of the student: Jesse Klijnsma
Final project course: CSE3000 Research Project
Thesis committee: Jan van Gemert, Sander Gielisse, Alexios Voulimeneas

Real-world evaluation of Optical Flow on repetitive patterns

Jesse Klijnsma
EEMCS
Delft University of Technology
The Netherlands

Abstract

1. Introduction

Optical flow estimation refers to the task of predicting the 2-dimensional visual motion between two consecutive images [7]. For each pixel in the first image, the corresponding subpixel in the second image needs to be found. It has several applications in areas such as video processing, medicine, and robotics [1].

In recent years, deep learning has significantly advanced the field, with models such as FlowNet [5] and RAFT [17]. These models are trained on synthetically generated images, as well as a limited amount of real-world data, due to the difficulty of obtaining correct optical flow vectors in real-world settings. Although many models show strong cross-dataset generalization [5, 9, 11, 16–18], showing generalization to proper real-world scenes remains a challenge. For example, one of the few datasets that utilizes real images and is widely used for evaluation is KITTI [13], which derives its ground truth by reconstructing the resulting optical flow from LIDAR scans and not 2D annotations. It predominantly features urban environments where the camera is mounted on a car, with limited diversity of motion as the car only drives forward. A recent re-evaluation [4] of scene flow models on more realistic datasets demonstrates that better performance on KITTI [13] correlates negatively with real-world generalization ability. Although focused on scene flow, it highlights limitations in KITTI [13] that also apply directly to optical flow.

One of the difficulties within optical flow prediction models is repetitive patterns [15], such as tiled floors, fences, textiles, or windows. In these cases, multiple regions in one frame can appear visually identical, making it difficult for a model to determine which regions correspond to each other. Although some studies discuss the challenge of repetitive patterns, or correspondence ambiguity, few studies specifically analyze the models' performance on repetitive patterns using a systematic evaluation. For example, LiteFlowNet3 [8], claims to improve performance on ambiguous correspondences. Even though its benchmark results support this, the model is not evaluated using data consisting of repetitive patterns, leaving it unclear whether the improvements made actually address this particular issue.

This research will investigate this empirical evidence gap. The main research question is: **How does the performance of optical flow prediction models compare on repetitive patterns in real-world footage?** This question is supported by the following subquestions: (1) Which models are most resilient to repetitive patterns in terms of End-Point-Error (EPE), (2) Does a low reconstruction error and a high EPE indicate a failure due to repetitive patterns, and (3) Which models perform best according to the False Correspondence Index?

To answer these questions, a dataset of real-world scenes is collected and annotated in which repetitive patterns are present. Then multiple optical flow models, such as FlowNet [5], RAFT [17], and LiteFlowNet3 [8] are evaluated on this dataset and their performance will be compared to results on existing synthetic benchmarks.

2. Related work

Robustness benchmarks. Other research that tries to investigate areas of difficulty for optical flow models is the work of Yi et al. [19]. They introduce several corruptions into existing datasets, including lighting changes, blur, noise, and digital compression artifacts. It uses a dataset with ground-truth optical flow vectors (KITTI-FC) and one without (GOPRO-FC), and measures the effect of optical and temporal corruptions on the models' performance by comparing the predicted flow to the ground truth or to the prediction on a clean image. They conclude that noise and compression artifacts, which are abundant in videos and images taken in the real world, have a significant negative impact on the accuracy of the optical flow models. Therefore, to properly test models' generalization to real-world data,

these camera and compression artifacts should be present, which calls for more realistic data. Although they show the impact of optical and temporal corruptions, they do not address the effects of scenery that have challenging traits for optical flow models, such as occlusions or repetitive patterns. This highlights the need for further investigation into more realistic confounding factors that challenge optical flow models.

Models with measures against repetitive patterns. One of the models that specifically claim that they have taken measures against errors due to repetitive patterns is LiteFlowNet3 [8] and Ef-RAFT [6].

3. Methodology and Tools

To evaluate and compare the performance of optical flow estimation models on real data containing repetitive patterns, a data set with human-annotated flow vectors is collected. The data set contains image pairs of consecutive frames, between which the optical flow is calculated, and alongside a sparse ground-truth annotation of the optical flow. The data is stored in the same directory structure as the KITTI [13] dataset, for easy loading into existing evaluation frameworks, as KITTI [13] is widely supported and includes a mask to indicate where optical flow vectors are present.

To annotate the image pairs, a tool[1] was collectively developed by the Research Project group. Using the annotation tool, two frames are picked from an input video. With the two frames displayed side by side, multiple pairs of corresponding points are carefully selected by hand with pixel precision. The number of pairs of points per image is around $n = 35$. This number of points covers a large portion of repetitions with at least one annotation point and ensures that a correspondence mismatch by the model is measurable.

For specifically annotating repetitive patterns that lie on a planar surface, we introduce an annotation mode that allows for interpolation of flow vectors within a region of interest. This approach can only be used when the repeated pattern lies entirely within a single 2D plane. A set of point pairs is manually selected between two frames, from which a homography matrix is estimated using OpenCV's homography fitting tools [2]. The transformation matrix is then used to warp all source points within the convex hull of the annotated region, and the optical flow is computed as the displacement between original and warped points.

This annotation method is only valid under three assumptions. First, the annotated area must fully lie on a 2-dimensional plane such that no depth-induced parallax occurs. Secondly, the motion between the frames cannot include large perspective changes, such as tilting or translating from a sharp angle, as this can excessively distort the polygon within which is interpolated. And lastly, the camera must have minimal lens distortion, as this causes nonlinear warping, which violates the assumptions of homographic mapping. If any of these conditions are not met, interpolation between annotated points is not performed.

With the evaluation database in place, an inference pass is done on the complete dataset for each model using a subset of its available checkpoints. To include a large number of models in this evaluation, the PTLFlow [14] framework is used, which provides a collection of deep learning-based optical flow estimation models. It includes the models accompanied by training checkpoints containing pre-trained weights for their specific training dataset. As not all models provide checkpoints for all datasets, a selection of checkpoints needs to be made. Upon reviewing the available checkpoints for each model, we find that including checkpoints trained on FlyingThings [12], Sintel [3], KITTI [13] and checkpoints based on a mix of datasets, cover all models under test. This allows for relative comparison of all models that have the same training data.

The resulting optical flow predictions will be evaluated using the following three metrics:

End-Point-Error (EPE) End-point-error is a widely-used metric to measure the performance of optical flow models. It measures the euclidean distance between the ground-truth and the predicted optical flow vector as shown in Fig. 2. To calculate the EPE for all predicted flow vectors within the image, the mean of the EPEs is taken over all pixels. However, since the collected data is sparsely annotated and the EPE can only be calculated where ground-truth vectors are present, the mean is only taken over pixels where annotations are present. The formal definition of the End-Point-Error (EPE) is:

$$\mathrm{EPE}^{(i)} = \sqrt{(u^{(i)}_{\mathrm{pred}} - u^{(i)}_{\mathrm{gt}})^2 + (v^{(i)}_{\mathrm{pred}} - v^{(i)}_{\mathrm{gt}})^2}, \qquad (1)$$

where $\mathbf{u}^{(i)}_{\mathrm{gt}} = (u^{(i)}_{\mathrm{gt}}, v^{(i)}_{\mathrm{gt}})$ is the ground truth optical flow vector at pixel i, and $\mathbf{u}^{(i)}_{\mathrm{pred}} = (u^{(i)}_{\mathrm{pred}}, v^{(i)}_{\mathrm{pred}})$ is the predicted flow vector.

The mean End-Point-Error over the valid pixels in an image is:

$$\mathrm{EPE}_{\mathrm{mean}} = \frac{1}{\sum_{i=1}^{N} M^{(i)}} \sum_{i=1}^{N} M^{(i)} \cdot \mathrm{EPE}^{(i)}. \qquad (2)$$

Where $M^{(i)} \in \{0, 1\}$ is a binary mask indicating whether the ground truth at pixel i is valid ($M^{(i)} = 1$) or invalid ($M^{(i)} = 0$).

[1] https://github.com/IrisPetre99/RP3000

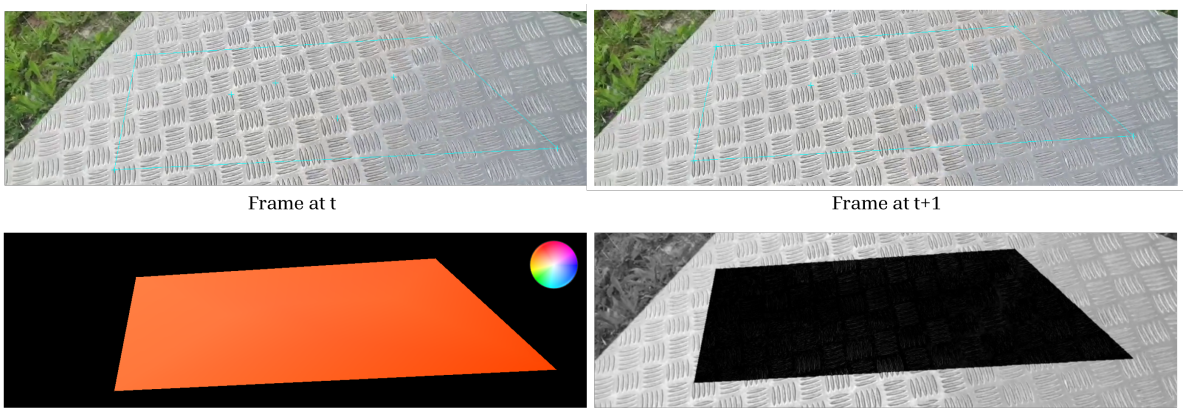

Figure 1. Visualization of homography-based flow annotation and reconstruction error. The top row shows the input frames at times t and $t+1$, with the annotated convex hull region highlighted. The bottom-left image visualizes the ground-truth optical flow, interpolated using a homography fitted to the annotated point pairs (see color wheel for direction/scale reference). The bottom-right image shows the Euclidean distance between the actual frame at $t+1$ and the reconstruction obtained by forward-splatting frame t using the ground-truth flow. This distance represents the pixel-wise RGB error and highlights regions where the homography-based flow fails to reconstruct the appearance of the next frame accurately. In this example, the region with ground-truth flow appears almost entirely black, with only minor speckles, suggesting that the homography-based interpolation is valid in this case.

To compute the overall EPE for a single model, we take the **mean of the per-image mean EPEs** across all images. This ensures that each image contributes equally to the final score. Simply averaging the EPEs over all annotated pixels would overrepresent images with dense or semi-dense flow annotations, such as those generated via homographic interpolation, while underrepresenting sparsely annotated images. By averaging at the image level, a fair contribution of each image is maintained, regardless of annotation density.

End-point-error is suitable for revealing failures due to confusion caused by repetitive patterns, as the predicted flow vector will differ substantially from the ground-truth. It does need sufficient annotations on a repetitive pattern to detect any discrepancies between the prediction and the ground-truth, as a single mismatched repetition could not be captured by the metric otherwise.

F1-All F1-all is another widely used metric that measures the fraction of outlier predictions based on EPE. A flow prediction is considered an outlier when its EPE exceeds a certain threshold. A 3 pixel absolute threshold and a 5% relative threshold are used, which follow from the thresholds used by the KITTI benchmark [13]. Formally, a pixel i is marked an outlier if

$$O^{(i)} = \begin{cases} 1, & \text{if EPE}^{(i)} > \max\left(3\text{ px},\ 0.05\left\|\mathbf{u}_{\text{gt}}^{(i)}\right\|\right), \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

The per-image F1-All score is then the percentage of valid pixels marked as outliers:

$$\text{F1-All}_{\text{img}} = \frac{\sum_{i=1}^{N} M^{(i)}\, O^{(i)}}{\sum_{i=1}^{N} M^{(i)}} \times 100\%. \quad (4)$$

Finally, to obtain an overall F1-All score, these per-image percentages are averaged across all images, ensuring that each scene contributes equally.

Like EPE, F1-All is sensitive to large prediction errors, like those caused by confusion between repeated patterns. Any flow estimate that exceeds the fixed error threshold is

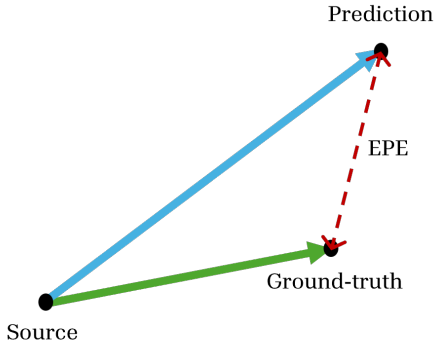

Figure 2. Visualization of End-Point Error (EPE) in optical flow. The source point represents the origin of the flow. The green vector denotes the ground-truth flow, the blue vector denotes the predicted flow, and the red dashed line indicates the EPE, i.e., the Euclidean distance between the predicted and ground-truth endpoints.

counted as an outlier and fully penalized. Unlike EPE, however, it does not account for how far the prediction is from the ground-truth once the threshold is exceeded all outliers are treated equally. As a result, F1-All provides a clear indication of how often a model makes significant mistakes, regardless of their exact magnitude.

False Correspondence Index (FCI) The False Correspondence Index (FCI) is a novel metric introduced in this work to detect a failure mode that occur in scenes with repetitive patterns: visually plausible but incorrect flow predictions, shown in Fig. 3. Optical flow models often rely on visual matching to establish correspondences between features in frames. However, when a visually repetitive pattern is present, the model can mismatch their features. This results in a prediction where the reconstructed second frame looks similar to the actual second frame, but where the flow vectors are incorrect.

To capture such errors, the FCI compares the EPE with the reconstruction error. A high EPE and a low construction error would indicate that a mismatch has occurred. The reconstruction error is calculated by splatting all the pixels of the first image using the predicted flow vectors [10]. Since the vectors can also have decimal components, the color of the source pixel is bilinearly interpolated and splatted across up to four pixels where the flow vector would fall between. To correct for brightness issues, the splats are corrected using a weighted average from multiple splats. Using forward splatting, there are some limitations to keep in mind. Forward splatting cannot be used when the scene contains occlusions, as multiple source pixels can flow to the same point in the resulting image and get averaged, even though one of the pixels might have occluded the other.

4. Results

Tab. 1 shows the resulting EPE values for all models under evaluation. It shows that the best performing models have very similar scores.

False Correspondence Index Fig. 4 shows the relation between visual consistency and the absolute flow error.

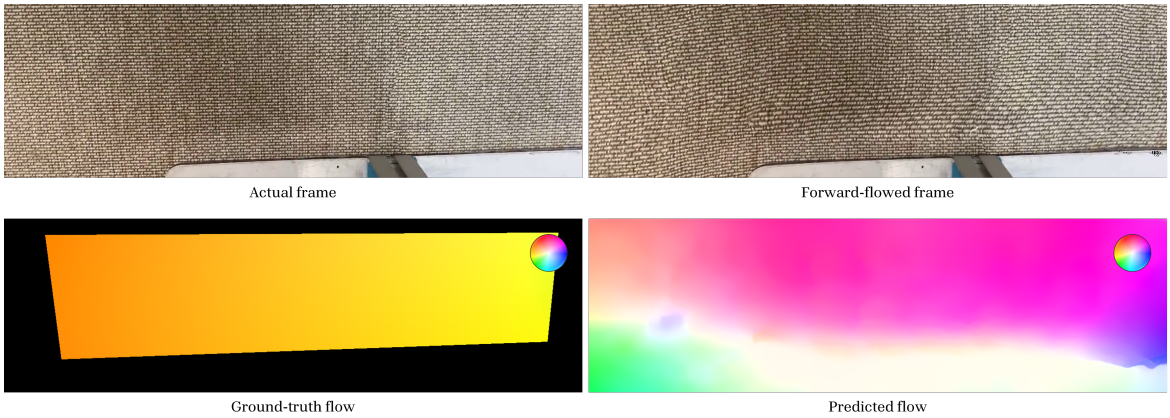

Figure 3. Example of a visually plausible but incorrect forward-warped flow prediction. *Top left:* Actual second frame. *Bottom left:* Annotated ground truth optical flow (interpolated via homography). *Top right:* First frame forward-warped to the second frame using the predicted flow. *Bottom right:* model-predicted optical flow. Although the warped image appears well-aligned with the actual frame, the predicted flow vectors are very incorrect, showing a failure mode due to repetitive patterns

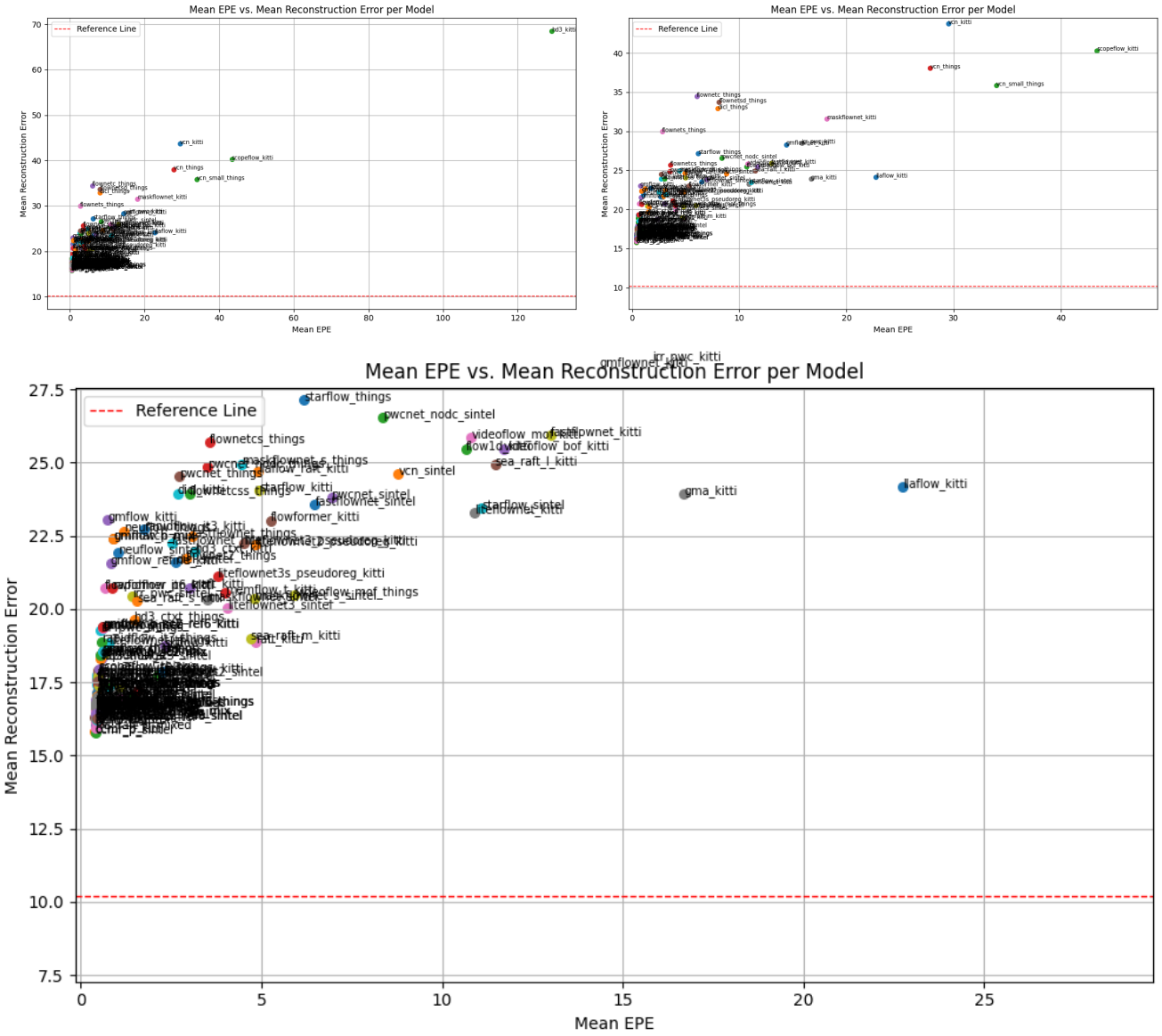

Figure 4. False correspondence index per model. The reference line indicates the reconstruction error achieved by human annotation.

Table 1. Mean End-Point Error (EPE) and F1-All scores for a range of optical flow models across a selection of training checkpoints. Each row represents a single model, and values are reported only for the checkpoints it provides. Models are sorted for convenience by their average EPE across available checkpoints. Lower values indicate better accuracy for both EPE and F1-All

Checkpoint	Things		Sintel		Kitti		Mix	
	EPE	F1-All	EPE	F1-All	EPE	F1-All	EPE	F1-All
Model								
ccmr_p	-	-	0.4007	0.00%	0.3914	0.00%	-	-
ms_raft_p	-	-	-	-	-	-	0.4000	0.07%
dpflow	0.4069	0.07%	0.4021	0.07%	0.3949	0.00%	-	-
splatflow	-	-	-	-	0.4234	0.07%	-	-
ccmr	-	-	0.4521	1.09%	0.4087	0.00%	-	-
rpknet	0.4380	0.07%	0.4315	0.07%	0.4363	0.00%	-	-
csflow	0.4300	0.22%	-	-	0.4672	0.22%	-	-
lcv_raft	0.4525	0.07%	-	-	-	-	-	-
gmflownet_mix	0.4624	0.22%	0.4472	0.30%	-	-	-	-
irr_pwcnet_irr	0.4819	0.07%	-	-	-	-	-	-
liteflownet3s	-	-	0.4850	0.01%	-	-	-	-
flowformer_pp	0.4203	0.00%	0.4474	0.22%	0.6734	1.20%	-	-
unimatch_sc2	0.5061	0.01%	0.5075	0.00%	-	-	0.5329	0.00%
gmflow_p_sc2	0.5061	0.01%	0.5075	0.00%	-	-	0.5329	0.00%
rapidflow	0.4817	0.15%	0.4693	0.37%	0.6068	0.80%	-	-
unimatch_sc2_ref6	0.6606	2.19%	0.4476	0.25%	0.6214	1.11%	0.4111	0.00%
gmflow_p_sc2_ref6	0.6606	2.19%	0.4476	0.25%	0.6214	1.11%	0.4111	0.00%
irr_pwcnet	0.5427	1.06%	-	-	-	-	-	-
rapidflow_it6	0.4894	0.16%	0.4774	0.45%	0.8655	5.20%	-	-
gmflow_refine	0.5061	0.01%	0.5075	0.00%	0.8249	2.27%	-	-
raft_small	0.6471	1.00%	-	-	-	-	-	-
gmflow	0.5985	0.39%	0.6054	0.32%	0.7515	1.16%	-	-
gmflow_p	0.5985	0.39%	-	-	-	-	0.9047	0.94%
unimatch	0.5985	0.39%	-	-	-	-	0.9047	0.94%
neuflow2	0.7615	1.85%	1.1062	1.42%	-	-	0.4440	0.08%
sea_raft_s	0.5395	1.96%	0.7666	2.46%	1.5527	2.11%	-	-
rapidflow_it3	0.5760	0.56%	0.5357	0.39%	1.7480	11.78%	-	-
memflow	0.4070	0.15%	0.4153	0.07%	2.2595	2.59%	-	-
skflow	0.4284	0.45%	0.4218	0.30%	2.3253	2.55%	-	-
neuflow	1.2089	5.63%	1.0353	3.92%	-	-	-	-
dip	0.4779	1.20%	1.6667	3.89%	1.4406	1.62%	-	-
craft	0.4243	0.07%	0.4145	0.07%	3.0311	3.68%	-	-
memflow_t	0.3998	0.00%	0.4190	0.01%	3.9994	7.74%	-	-
hd3_ctxt	1.4730	4.17%	0.4843	0.59%	3.1606	5.12%	-	-
raft	0.4208	0.01%	0.4228	0.22%	4.6698	6.51%	-	-
sea_raft_m	0.5539	2.40%	0.4408	0.31%	4.6927	4.33%	-	-
llaflow_raft	0.4277	0.07%	0.4423	0.15%	4.8990	5.12%	-	-
flowformer	0.4302	0.08%	0.4591	0.15%	5.2715	8.91%	-	-
liteflownet2	-	-	2.1505	2.73%	-	-	-	-
flownets	2.8167	15.73%	-	-	-	-	-	-
flownet2	2.9295	15.00%	-	-	-	-	-	-
flownetcss	3.0046	15.26%	-	-	-	-	-	-
flownetcs	3.5741	16.09%	-	-	-	-	-	-
liteflownet3s_pseudoreg	-	-	-	-	3.7844	10.75%	-	-

Continued on next page

Table 1. Mean End-Point Error (EPE) for different models, sorted ascendingly.

Checkpoint	Things		Sintel		Kitti		Mix	
	EPE	F1-All	EPE	F1-All	EPE	F1-All	EPE	F1-All
Model								
flow1d	0.4443	0.15%	0.4757	0.60%	10.6593	14.52%	-	-
liteflownet3	-	-	4.0702	11.41%	-	-	-	-
liteflownet	0.7697	3.38%	0.6480	2.61%	10.9039	23.09%	-	-
sea_raft_l	0.6604	3.71%	1.0992	1.47%	11.4668	9.34%	-	-
dicl	7.9987	22.91%	2.6321	4.92%	2.6904	14.15%	-	-
liteflownet3_pseudoreg	-	-	-	-	4.5198	12.17%	-	-
maskflownet_s	4.4424	13.34%	4.8010	8.91%	-	-	-	-
liteflownet2_pseudoreg	-	-	-	-	4.8301	12.26%	-	-
pwcnet	2.7295	11.48%	6.9392	8.82%	-	-	-	-
videoflow_mof	5.9263	13.64%	0.4410	0.15%	10.8027	11.65%	-	-
gma	0.4324	0.30%	0.4434	0.30%	16.6945	9.75%	-	-
pwcnet_nodc	3.5060	13.91%	8.3433	15.59%	-	-	-	-
irr_pwc	0.5490	0.42%	1.4158	4.69%	15.8151	22.90%	-	-
flownetc	6.0608	25.51%	-	-	-	-	-	-
videoflow_bof	-	-	0.4668	0.10%	11.7124	12.76%	-	-
fastflownet	3.0698	8.56%	6.4820	14.24%	13.0060	22.94%	2.5197	10.29%
starflow	6.1713	12.44%	11.0985	10.99%	4.9468	14.29%	-	-
gmflownet	0.4535	0.32%	-	-	14.3642	19.28%	-	-
llaflow	0.4252	0.15%	0.4155	0.07%	22.7293	8.33%	-	-
flownetsd	8.1114	42.00%	-	-	-	-	-	-
maskflownet	-	-	3.4979	6.90%	18.1352	25.44%	-	-
scopeflow	0.4950	0.15%	0.4665	0.67%	43.3554	50.93%	-	-
vcn	27.8040	26.19%	8.7887	15.85%	29.4939	50.95%	-	-
vcn_small	33.9843	26.50%	-	-	-	-	-	-
hd3	0.5905	1.46%	1.5735	1.98%	129.1069	57.52%	-	-

5. Responsible Research

6. Discussion

7. Conclusion

References

[1] Andrea Alfarano, Luca Maiano, Lorenzo Papa, and Irene Amerini. Estimating optical flow: A comprehensive review of the state of the art. *Computer Vision and Image Understanding*, 249:104160, Dec. 2024. 2

[2] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000. 3

[3] Daniel J. Butler, Jonas Wulff, Garrett B. Stanley, and Michael J. Black. A Naturalistic Open Source Movie for Optical Flow Evaluation. In Andrew Fitzgibbon, Svetlana Lazebnik, Pietro Perona, Yoichi Sato, and Cordelia Schmid, editors, *Computer Vision – ECCV 2012*, pages 611–625, Berlin, Heidelberg, 2012. Springer. 3

[4] Nathaniel Chodosh, Deva Ramanan, and Simon Lucey. Re-Evaluating LiDAR Scene Flow. In *2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 5993–6003, Waikoloa, HI, USA, Jan. 2024. IEEE. 2

[5] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Häusser, Caner Hazirbas, Vladimir Golkov, Patrick van der Smagt, Daniel Cremers, and Thomas Brox. FlowNet: Learning Optical Flow with Convolutional Networks. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 2758–2766, Dec. 2015. ISSN: 2380-7504. 2

[6] Navid Eslami, Farnoosh Arefi, Amir M. Mansourian, and Shohreh Kasaei. Rethinking RAFT for Efficient Optical Flow. In *2024 13th Iranian/3rd International Machine Vision and Image Processing Conference (MVIP)*, pages 1–7, Mar. 2024. ISSN: 2166-6784. 3

[7] Berthold K. P. Horn and Brian G. Schunck. Determining optical flow. *Artificial Intelligence*, 17(1):185–203, Aug. 1981. 2

[8] Tak-Wai Hui and Chen Change Loy. LiteFlowNet3: Resolving Correspondence Ambiguity for More Accurate Optical Flow Estimation. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 169–184, Cham, 2020. Springer International Publishing. 2, 3

[9] Lingtong Kong and Jie Yang. MDFlow: Unsupervised Optical Flow Learning by Reliable Mutual Knowledge Distillation. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(2):677–688, Feb. 2023. arXiv:2211.06018 [cs]. 2

[10] T. Lin and J. L. Barron. Image Reconstruction Error for Optical Flow. In *Research in Computer and Robot Vision*, pages 269–290. WORLD SCIENTIFIC, Feb. 1995. 5

[11] Liang Liu, Jiangning Zhang, Ruifei He, Yong Liu, Yabiao Wang, Ying Tai, Donghao Luo, Chengjie Wang, Jilin Li, and Feiyue Huang. Learning by Analogy: Reliable Supervision from Transformations for Unsupervised Optical Flow Estimation, Nov. 2020. arXiv:2003.13045 [cs]. 2

[12] Nikolaus Mayer, Eddy Ilg, Philip Häusser, Philipp Fischer, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox. A Large Dataset to Train Convolutional Networks for Disparity, Optical Flow, and Scene Flow Estimation. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4040–4048, June 2016. arXiv:1512.02134 [cs]. 3

[13] Moritz Menze and Andreas Geiger. Object scene flow for autonomous vehicles. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3061–3070, Boston, MA, USA, June 2015. IEEE. 2, 3, 4

[14] Henrique Morimitsu. PyTorch Lightning Optical Flow, 2021. Publication Title: GitHub repository. 3

[15] Syed Tafseer Haider Shah and Xiang Xuezhi. Traditional and modern strategies for optical flow: an investigation. *SN Applied Sciences*, 3(3):289, Mar. 2021. 2

[16] Xiaoyu Shi, Zhaoyang Huang, Dasong Li, Manyuan Zhang, Ka Chun Cheung, Simon See, Hongwei Qin, Jifeng Dai, and Hongsheng Li. FlowFormer++: Masked Cost Volume Autoencoding for Pretraining Optical Flow Estimation, Mar. 2023. arXiv:2303.01237 [cs]. 2

[17] Zachary Teed and Jia Deng. RAFT: Recurrent All-Pairs Field Transforms for Optical Flow. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 402–419, Cham, 2020. Springer International Publishing. 2

[18] Yihan Wang, Lahav Lipson, and Jia Deng. SEA-RAFT: Simple, Efficient, Accurate RAFT for Optical Flow, May 2024. arXiv:2405.14793 [cs]. 2

[19] Zhonghua Yi, Hao Shi, Qi Jiang, Yao Gao, Ze Wang, Yufan Zhang, Kailun Yang, and Kaiwei Wang. Benchmarking the Robustness of Optical Flow Estimation to Corruptions, Nov. 2024. arXiv:2411.14865 [eess] version: 1. 2