Optical Flow Estimation Using Event-Based Cameras

Improving Optical Flow Estimation Accuracy Using Space-Aware De-Flickering

Student:
Per Magnus Skullerud (P.M.Skullerud@student.tudelft.nl)
Supervisors:
Hesam Araghi (H.Araghi@tudelft.nl)
Nergis Tömen (N.Tomen@tudelft.nl)
TU Delft

1. Introduction

Optical flow

- Apparent visible motion
- Expressed as direction vectors at frames' pixels
- Conventional cameras struggle with low framerate

Event cameras

- Better suited to the optical flow problem?
+ High temporal resolution & pixel bandwidth
– No absolute brightness, unfamiliar representation

Flickering

- High-frequency brightness oscillation
- Is likely to overwhelm event cameras
- Tricky to mend in event data due to lack of absolute brightness

Contrast Maximization (CM)

- A framework for solving motion estimation problems using event data
+ Demonstrated good performance in past works
– Sensitive to flickering

Event-based Flicker Removal (EFR)

- The only existing event data de-flicker algorithm [1]
+ Great results in static conditions
– Not designed for cases where the relative position of the camera and the flickering object is variable

2. Research question

"Can integrating spatial awareness into event-based de-flicker filters be used to improve subsequent optical flow estimation accuracy?"

Motivation

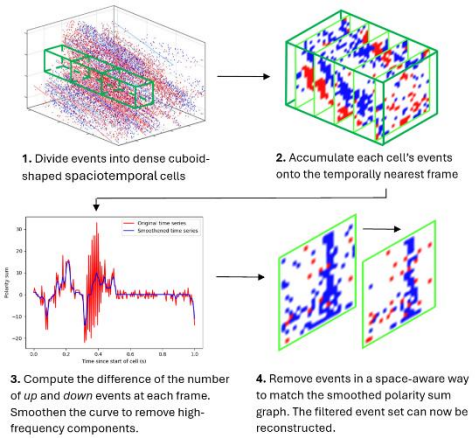
- CM-based methods have been demonstrated to be sensitive to flickering
- EFR filters per pixel, effectively discarding information about the spatial coherence of edges
- Integrating resistance to flickers may not be possible by adjusting the optical flow algorithm itself. By removing flickering via preprocessing, our method is compatible with any optical flow estimator.

3. Method

Assumptions

1. Events caused by mild brightness changes are indistinguishable from noise.
2. The presence of brightness events does not affect the observability of motion events (and vice versa).
3. Motion events are spatially close to other motion events of opposite polarity.

Algorithm



1. Divide events into dense cuboid-shaped spaciotemporal cells

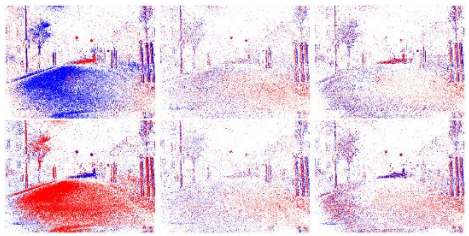
2. Accumulate each cell's events onto the temporally nearest frame

3. Compute the difference of the number of *up* and *down* events at each frame. Smoothen the curve to remove high-frequency components.

4. Remove events in a space-aware way to match the smoothed polarity sum graph. The filtered event set can now be reconstructed.

Example

The DSEC dataset [2] contains night-time sequences with visible flickering due to a 50Hz AC power grid. CM-based methods show a sharp accuracy drop in these sequences.



Two consecutive frames of a flicker-heavy sequence from DSEC.
Left: No filtering – flicker events overwhelm the street.
Center: EFR – flickering is removed, what is left appears to be noise.
Right: Ours – flickering is removed, the noise pattern is different.

4. Evaluation

Filtering is qualitatively evaluated using visualizations. We benchmark optical flow using original sequences, as well as ones filtered by our algorithm and by EFR.

Sequences

- DSEC dataset's *zurich_city_10_a (10a)* sequence, containing heavy flickering
- DSEC dataset's *zurich_city_02_d (02d)* sequence, containing no flickering

Optical flow estimators

- E-RAFT – supervised learning, not CM-based (pre-trained DSEC checkpoint used)
- TamingCM – self-supervised learning, CM-based (pre-trained DSEC checkpoint used)

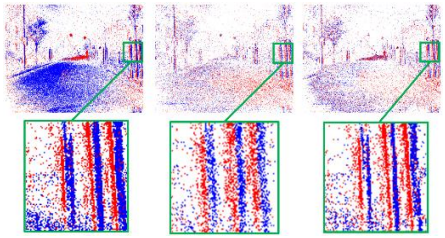
Metrics

- EPE (average Euclidean distance between predicted and ground truth optical flow vectors)
- $\%OUT_3$ (Percentage of optical flow vectors wrong by >3 pixels)

5. Results & Discussion

Filtering results

- EFR removes more events, both flicker and non-flicker events
- EFR's edges are thick but sparse; our filtering's edges are dense but thin
- Concerningly, deleting flickers removes information about edges' presence
- Filtering removes 50-65% of events



Left: No filtering. **Center:** EFR. **Right**: our filtering.

Optical flow results

- Filtering hardly affects optical flow accuracy, despite removing a significant portion of total events
- Filtering often hurts accuracy
- Our filtering works better with E-RAFT, EFR prefers TamingCM
- Curiously, de-flickering a flicker-free sequence improves accuracy
- Limitation: ground truth flow is sparse

Optical flow estimator	Filtering method	Sequence	
		10a	02d
TamingCM	None	3.36	3.25
	EFR	3.22	3.17
	Ours	3.45	3.20
E-RAFT	None	0.44	0.72
	EFR	0.63	0.88
	Ours	0.54	0.75

EPE results (smaller is better)

6. Conclusion & Future Works

- Our algorithm's niche advantages are undermined by de-flickering hardly affecting optical flow in the first place.
- Deleting flickers removes much information about a scene's geometrical features, especially given weak ambient lighting.
- We recommend ignoring polarities to make flickering edges appear normal without losing information about their presence.
- De-flickering may be used as a generic low-pass filter to reduce dataset sizes at a minimal impact to accuracy.

References

[1] Ziwei Wang, Dingran Yuan, Yonhon Ng, and Robert Mahony. A linear comb filter for event flicker removal. In 2022 International Conference on Robotics and Automation (ICRA), pages 398–404. IEEE, 2022.
[2] Mathias Gehrig, Willem Aarents, Daniel Gehrig, and Davide Scaramuzza. Dsec: A stereo event camera dataset for driving scenarios. IEEE Robotics and Automation Letters, 6(3):4947–4954, 2021.