

Event Cameras, Contrast Maximization and Reward Functions: an Analysis

Timo Stoffregen^{1,2}, Lindsay Kleeman¹

¹Dept. Electrical and Computer Systems Engineering, Monash University, Australia.

²Australian Centre of Excellence for Robotic Vision, Australia.

Abstract

Event cameras asynchronously report timestamped changes in pixel intensity and offer advantages over conventional raster scan cameras in terms of low-latency, low redundancy sensing and high dynamic range. In recent years, much of research in event based vision has been focused on performing tasks such as optic flow estimation, moving object segmentation, feature tracking, camera rotation estimation and more, through contrast maximization. In contrast maximization, events are warped along motion trajectories whose parameters depend on the quantity being estimated, to some time t_{ref} . The parameters are then scored by some reward function of the accumulated events at t_{ref} . The versatility of this approach has lead to a flurry of research in recent years, but no in-depth study of the reward chosen during optimization has yet been made. In this work we examine the choice of reward used in contrast maximization, propose a classification of different rewards and show how a reward can be constructed that is more robust to noise and aperture uncertainty. We validate our work experimentally by predicting optical flow and comparing to ground-truth.

1. Introduction

Event cameras, also known as Dynamic Vision Sensors or Neuromorphic Cameras [1], have presented vision and robotics researchers with a new class of visual information. Where traditional frame-based cameras sample the scene at a fixed rate, event cameras capture visual information asynchronously, corresponding to intensity changes at each pixel location. As the intensity at a pixel changes above a certain threshold, an event is generated as a tuple of x, y position, timestamp t and intensity change sign s . Event based cameras offer several advantages over traditional cameras in terms of low latency, high dynamic range (120 dB) and low power consumption (10 mW) [2].

Event data is inherently sparse, because static backgrounds or otherwise slowly changing elements in the scene don't generate events. Since conventional cameras sample the scene based on a fixed clock, they under-sample swiftly

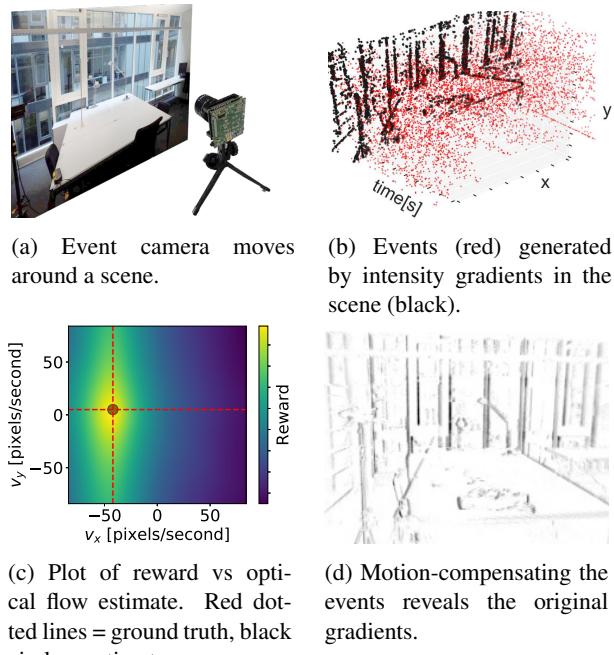


Figure 1: Contrast Maximization: Events generated by scene or camera motion (1a) form a point cloud in a space-time volume (1b). If the events are motion-compensated by some trajectory, the contrast at that point can be evaluated by some reward. Since the resulting reward has gradients with respect to trajectory parameters (1c), the original trajectory can be estimated, giving optic flow and motion-correction (1d) in one step.

changing scenes or redundantly over-sample slowly changing scenes. In contrast, an event camera samples the scene at a rate proportional to the dynamics of the scene.

Events carry little information individually and so are not meaningfully treated in isolation. So far, event based algorithms have been in one of two categories: those which operate on *individual* events to update some previous state and those which operate on a *set* of events to perform a given task or estimate a particular quantity [3]. Those methods which operate on individual events typically require historic

information, such as grayscale images reconstructed from the event stream to make inferences. On the other hand, those which operate on a set of events require no external information. As noted in [3], the former category can be further broken down into those methods which (*a*) discard the temporal information carried by the events, for example by accumulating the events into frames over a temporal window and then performing computations on those frames (such as [4–7]) and those (*b*) which utilize the temporal information of the events (such as [3, 6, 8–18]). This group tends to require more novel techniques, since traditional computer vision algorithms are not well suited to dealing with the continuous time representations that events attempt to approximate.

One such technique is that of *contrast maximization* (CM), whereby events are warped along point trajectories to the image plane. The trajectories can then be optimized with respect to the resulting *image of warped events* (IWE) H to recover the point trajectories that best fit the original set of events.

1.1. Contrast Maximization

Contrast maximization (CM) emerged recently as a promising technique for solving a number of problems in event based vision. Since events are produced by intensity gradients moving over the image plane, CM makes the assumption that if the events are motion compensated by warping them along their point trajectories to some discretized plane at time t_{ref} , events generated by the same point on the intensity gradient will project to the same location at t_{ref} and *accumulate* there, (see Fig. 2) giving a resulting image of warped events H (Fig. 1). While it is possible to generate an IWE with any arbitrary trajectory, certain quantities such as the contrast of the IWE will be maximized by warping the events along the true point trajectories. More formally, given an event defined by its image position, time-stamp and sign of intensity change, $e_n = \{x_n, t_n, s_n\}$, we define the warped location of the event with respect to the warp parameters θ as

$$x'_n = W(x_n, t_n; \theta), \quad (1)$$

[3], where W is the warping function. Thus the image of warped events from N_e events can be formulated as

$$H(x; \theta) = \sum_{n=1}^{N_e} b_n \delta(x - x'_n), \quad (2)$$

[3], where each pixel x sums the warped events x'_n that map to it (indicated by δ since they represent intensity spikes). If b_k is set equal to 1 the number of events are summed, if $b_k = s_k$ the event polarities are summed. This IWE can now be evaluated using a reward function. Since a well parameterized IWE will warp events to the locations of intensity gradients on the image plane, the IWE will seem sharp

and hence the variance of the IWE is commonly used as a measure of contrast. Thus, the steps of the CM method are:

- Collect a set of events generated by gradients moving across the image plane
- Based on a motion assumption, generate image of warped events H
- Use a reward function to evaluate H
- Optimize the reward with respect to the motion parameters

An advantage of this method is that the problem of event associations (which events were produced by the same feature) is solved implicitly. CM is a versatile method and has been recently used to estimate camera rotation on a static scene [12], estimate optical flow [13], track features in the event stream [14], estimate camera motion and depth [3], perform moving object detection [15], motion segmentation [18] and provide a training signal for deep learning using events [6].

1.2. Contributions

Our contributions are:

- To analyze the properties of CM reward functions and provide new insights into existing reward functions
- Show how different reward functions can be used and combined to get improved results on noisy data with large aperture-uncertainty
- Quantify this improvement using optical flow as a benchmark, without loss of generality for the CM method.

An unsolved problem with CM is determining how many events should be processed at once. For the sake of efficiency the warping model used is typically a linearization of some higher dimensional trajectory, thus it is important that the set of events does not span too great a time. However this time is again dependent on the dynamics of the scene. We propose a new solution to this problem which is fully compatible with the general CM framework.

2. Reward Functions

In [3, 12] the total variance of \mathcal{I} ,

$$r_{\sigma^2}(H) = \frac{1}{N_p} \sum_{i,j} (h_{i,j} - \mu_H)^2, \quad (3)$$

was used to evaluate the warp, where N_p is the number of pixels, μ_H is the mean of H and $h_{i,j}$ is the value of pixel i, j in H . In [13] the sum of squares of H (4) was used. These two rewards are essentially equivalent as shown in [12]. The reason these two rewards work is because they disproportionately reward event accumulations of a high magnitude

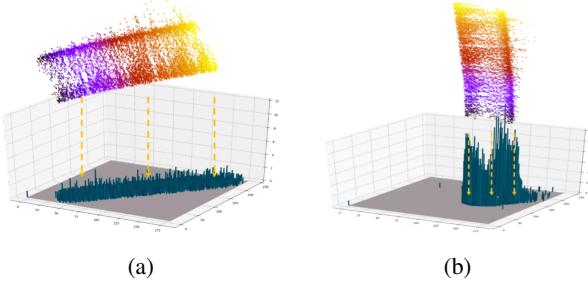


Figure 2: Events from circle moving across the image plane projected along a good (2b) and a poor (2a) estimate of the actual trajectory. In 2a sum of accumulations squared (r_{SoS}) is 5,683 while in 2b $r_{\text{SoS}} = 27,884$.

(see Fig. 2). This occurs since, at the optimal trajectory, events are accumulated onto the small set of locations on the image plane at t_{ref} that was occupied by the original gradients at t_{ref} . In other words any reward will work that rewards high accumulations more than the same total accumulations spread across more locations. At the same time, if most events are accumulated at fewer locations, it means that most locations at t_{ref} contain no events at all. Therefore we propose to explore the benefits of the following rewards for contrast maximization:

Sum of Squares (r_{SoS}):

$$r_{\text{SoS}}(H) = \sum_{i,j} h(i,j)^2. \quad (4)$$

This reward was used in the past [13] and is sufficiently similar to the variance (r_{σ^2}) that we will not consider variance in this work.

Sum of Exponentials (r_{SoE}):

$$r_{\text{SoE}}(H) = \sum_{i,j} e^{h(i,j)}. \quad (5)$$

Exponentials reward higher numbers even more disproportionately than do polynomials (for proof of this note that $\lim_{n \rightarrow \infty} \frac{n^b}{a^n} = 0$). Therefore this reward is more extreme than r_{SoS} or r_{σ^2} .

Max of Accumulations (r_{MoA}):

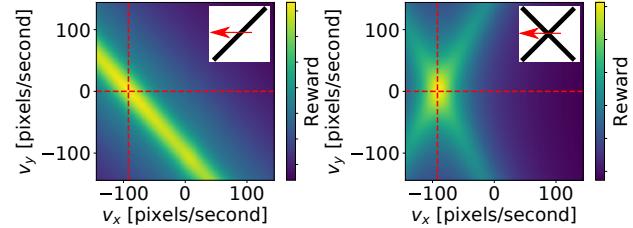
$$r_{\text{MoA}}(H) = \max(h(i,j) \in H). \quad (6)$$

Simply returns the greatest accumulation. This reward takes no other information into account and thus is yet more extreme than exponentiation.

Inverse Sum of Accumulations (r_{ISoA}):

$$r_{\text{ISoA}}(H) = \frac{1}{\sum_{i,j} h(i,j) > 1}. \quad (7)$$

This reward inverts the count of the number of locations with an event accumulated to them. The count will be at a minimum at the correct trajectory, so in order to define a reward we invert the sum. A similar reward was used in [15].



(a) r_{SoS} of events generated by (b) r_{SoS} of events generated by a line (motion vector in red). a cross (motion vector in red).

Figure 3: The reward function (using r_{SoS}) of the events generated by a straight line segment (3a) shows the effect of aperture problem on contrast maximisation techniques when compared to the reward function of a cross moving with the same optic flow velocity indicated by red dotted lines (3b)

Sum of Suppressed Accumulations (r_{SoSA}):

$$r_{\text{SoSA}}(H) = \sum_{i,j} e^{(-h(i,j)*p)}. \quad (8)$$

This reward gives locations with few accumulations in them a higher value than locations with many accumulations in them and saturates for large values of $H(x,y)$. This reward is maximized at the optimal trajectory, since most events are accumulated at few locations and thus at most locations x_l, y_l will return a high value. The factor p is an arbitrary shifting factor which decides the saturation point.

Of these rewards, r_{SoS} , r_{SoE} and r_{MoA} favor trajectories that result in large accumulations (they are *magnitude-rewarding*) and r_{ISoA} and r_{SoSA} favor those that result in many locations having few or no accumulations (they are *sparsity-rewarding*).

2.1. Aperture Problem

The aperture problem arises when optical flow is estimated using only a local region of a moving object. In this case it can happen that only a line feature of the object is visible and thus only the velocity component perpendicular to the local line feature can be estimated. Contrast maximization techniques don't suffer from the aperture problem in the way that local optic flow estimators such as Lucas-Kanade [19] do, since they consider the scene globally.

However, long line segments do introduce uncertainty to the optic flow estimates when using contrast maximization, which can be considered analogous to the aperture problem. A line segment moving over the DVS image plane will generate events which lie on a plane in the space-time volume [20]. Although warping the events along the trajectory of the line segment will generate a large value in the reward function, trajectories which vary slightly but still lie on the event plane will generate large values as well. This can be seen in Fig. 3a; the reward function for the straight

line segment features a long ridge, along which the values are similar to each other. Indeed, there are likely to be two local maxima to either side of the true trajectory in cases of pure line segments and none *at* the true trajectory. This is because it is possible to achieve greater event accumulations when warping over the diagonal of the plane; since greater accumulations are rewarded in r_{SoS} , these trajectories will maximize the reward function (see Fig. 4).

Once the line segment gains features on other axes, this uncertainty is much reduced, the region around the ground truth forming a sharp spike, since changing the trajectory slightly in the direction of one of the image gradient’s principal axes will cause events along the other axis to accumulate less (Fig. 3b).

For sparsity-rewarding rewards the reward will experience much stronger relative change to slightly incorrect trajectories and thus suffer less from aperture problem. The reason for this is demonstrated in Fig. 4; while warping the events parallel to the plane of events diagonally is not likely to influence the r_{SoS} strongly, it will cause the events on the resulting IWE to take up substantially more space and thus strongly affect the r_{ISOA} . This effect is validated experimentally in datasets with dominant line segments (see Section 4.1, 4.3) and is visualized in Fig. 5.

2.2. Noise Tolerance

As sparsity-rewarding methods essentially measure the number of locations containing events in them, they are susceptible to noise. For example, r_{ISOA} becomes entirely meaningless in the worst case, where the event stream becomes so noisy that every location at t_{ref} contains at least one event. As more noise is added, these rewards become more and more uniform, until they are almost entirely flat (see Fig. 8).

2.3. Data Sufficiency

Contrast maximization falls into the category of algorithms that operate on groups of events [3]. This means that events need to be collected over some period of time before a meaningful estimate, such as optic flow can be made. In practice, waiting for movement of at least several pixels is necessary for a reliable estimate (see Fig. 7), which can take a long time for slow moving gradients. Estimating for how long events should be collected before contrast maximization is applied is an important task, since the algorithms presented in [3, 13, 15, 18] make an assumption of constant velocity over small periods of time in order to work. Further, the number of events generated by gradients in the image is dependent on the relative strength of the gradients. In our case it is necessary to know how many events the gradient is producing per pixel moved, since the r_{SOA} reward contains a shifting parameter, which needs to be tuned according to this value.

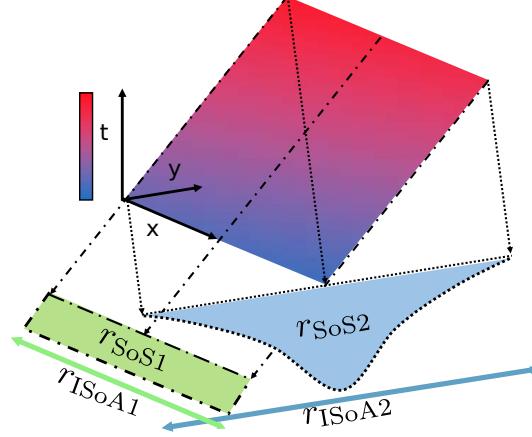


Figure 4: The plane in x, y, t represents a set of events generated by a line segment moving in the direction of the y axis. If the events are projected along the velocity vector (dashed arrows), they accumulate (green double arrow), giving the $r_{\text{ISOA}1}$. The integral of those accumulations (green area) squared is $r_{\text{SoS}1}$. If the events are instead (incorrectly) projected across the diagonal of the plane, the corresponding accumulations (blue arrow) give $r_{\text{ISOA}2}$, which being the inverse of the arrow length becomes smaller. The $r_{\text{SoS}2}$ (blue area) however becomes larger, since it rewards the larger peak in the accumulations. Thus maximizing the r_{ISOA} would give a correct result, the r_{SoS} an incorrect one, showing how sparsity-rewarding rewards are less susceptible to the aperture problem.

We can estimate how many events are produced per pixel moved directly from the r_{SoS} reward, under projection along the zero velocity. As a structure begins to move onto a new pixel, it generates events proportional to the intensity of its gradients. At $v = 0$, these events on the same pixel will accumulate, causing the r_{SoS} reward to grow exponentially. As the structure moves on to the next pixel, exponential growth will have to start anew, since the structure will be entering an empty set of pixels (see Fig. 6a). Thus the rate of change of the r_{SoS} along the $v = (0, 0)$ trajectory should flatten off periodically as the structure moves over the image plane. In real data, this is what happens (see Fig. 6), allowing estimation of whether contrast maximization will be able to give an optic flow estimate given a set of events.

3. Combined Reward Functions

We have identified two classes of reward, sparsity- and magnitude-rewarding, that can be used to optimize the total contrast of an image of warped events and shown that the one class should be much better at dealing with aperture-uncertainty while the other should be more capable of tolerating noise. We wish now to use that knowledge to con-

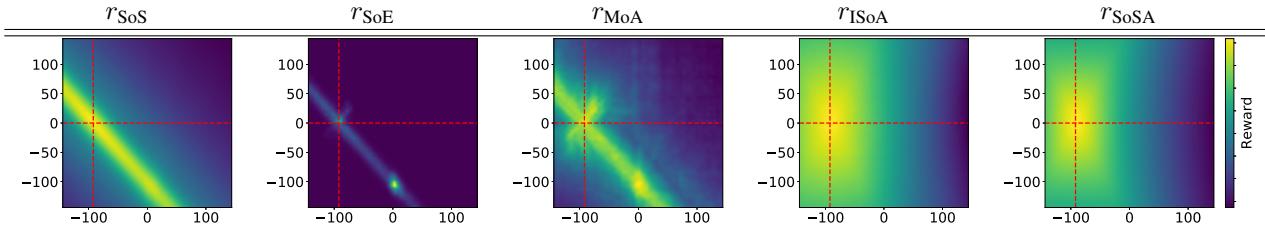


Figure 5: Various rewards sampled vs optical velocities v_x, v_y for a dataset with strong line features (see Fig. 9). Ground truth is indicated by red dotted lines. Note that the magnitude-rewarding r_{SoS} , r_{SoE} and r_{MoA} are much more prone to having incorrect local maxima, for reasons illustrated in Fig. 4 than the sparsity rewarding r_{ISoA} and r_{SoSA} .

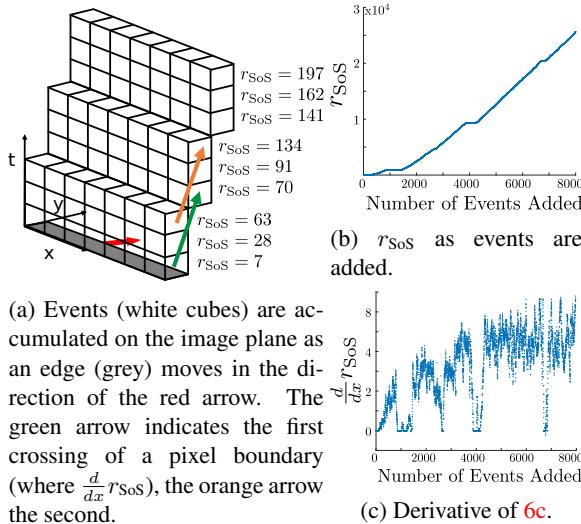


Figure 6: As an edge (grey) moves across the image plane of an idealised event camera, it generates some number of events proportional to the intensity of the edge. These events can be summed and squared to give the r_{SoS} . As the edge moves onto the next pixel, the r_{SoS} grows at a slower rate, so the derivative of r_{SoS} with respect to added events becomes close to zero. The minimum number of events required to compute the optic flow is shown by the green arrow. This effect can be seen in real data - Fig. 6b shows the r_{SoS} as events are added from the sequence in Section 2. In Fig. 6c the temporal derivative is zero when the object crosses pixel boundaries.

struct a new, hybrid reward, which is able to take the best from both classes of reward. Since r_{SoSA} gave better results than r_{ISoA} (since it is not a binary measure it is slightly more noise-tolerant), we combined this with a variety of magnitude-rewarding rewards:

- $r_{\mathbf{R1}} = r_{\text{SoS}} + r_{\text{SoSA}}$ In this reward we use the r_{SoS} reward with the constraint during optimization that successive improved estimates must not decrease the r_{SoSA} .
- $r_{\mathbf{R2}} = r_{\text{SoS}} + r_{\text{SoSA}} + r_{\text{SoE}}$ Here we use the same reward as $r_{\mathbf{R1}}$, except that when we have finally found an es-

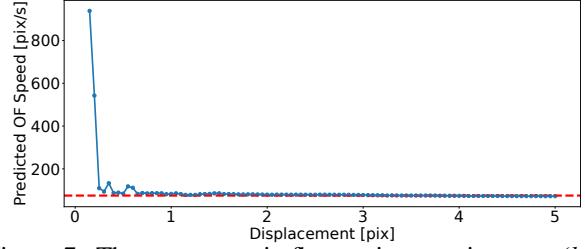


Figure 7: The current optic flow estimate using r_{SoS} (blue line) is noisy and uncertain when the structure has moved less than two pixels, but converges to the ground truth (red line) as the structure moves across more pixels and thus more events are added.

timate, we use it as a starting point for gradient ascent using the r_{SoE} .

The r_{SoE} gives precise and noise-tolerant results, given that the starting point for the optimization used is close to the maximum, which is why we incorporate it in the $r_{\mathbf{R2}}$ reward. Given a bad initial point r_{SoE} performs poorly since it is not sufficiently smooth for most optimization methods.

4. Experimental Results

We tested our rewards on high quality optical flow ground truth data collected from a DAVIS 240C event camera [21], using a linear slider to pan over a variety of scenes. We tested on two simple sequences of a line segment (Fig. 9) and a circle (Fig. 10) moving across the image plane. The line segment sequence introduces a lot of aperture-uncertainty, while the circle has none. We also tested the rewards on a real office scene (Fig. 11) to show that the hypotheses tested on the circle and line segment carry over to real world scenes. We restricted ourselves to relatively simple scenes, since it made it easier to collect ground truth data and control the level of aperture uncertainty. The usefulness of the CM framework on complex scenes has been shown in several other works [3, 6, 12–15, 18].

In the experiments we added random noise to the event stream to give us a signal/noise ratio of 1/10. In this way we were able to show the benefits of different rewards with respect to noise, which can be quite a significant component of current event camera output, as well as the benefits with

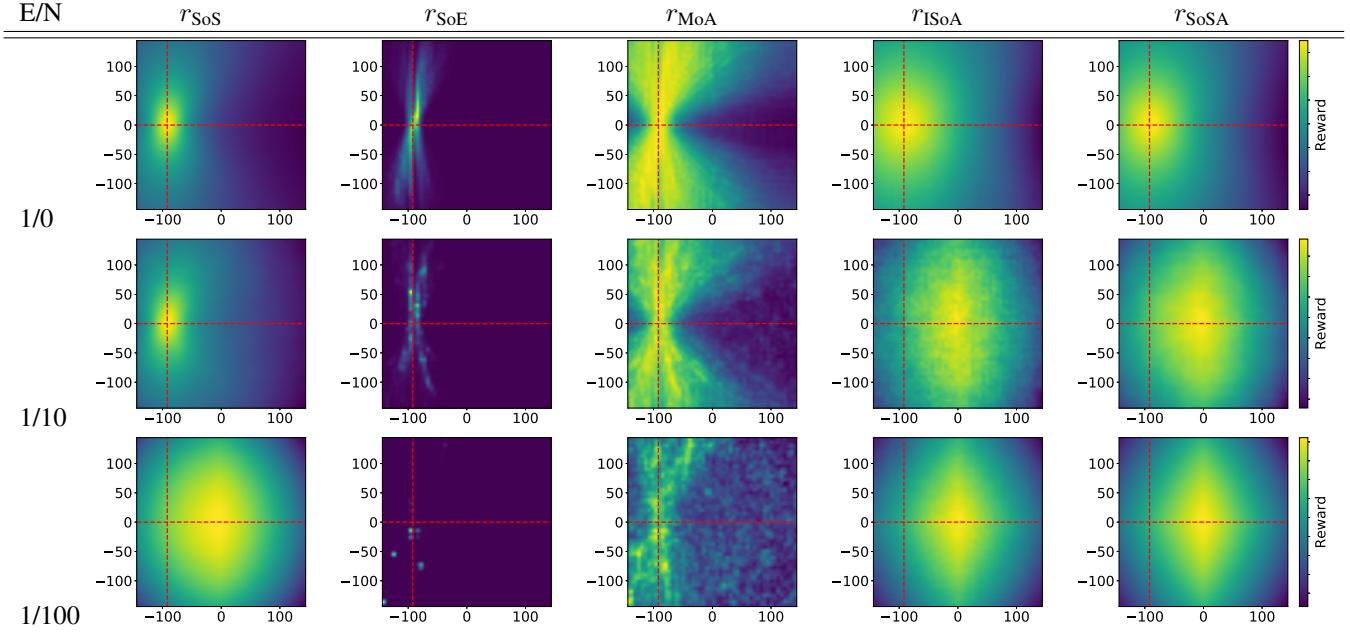


Figure 8: Various rewards sampled vs optical velocities v_x, v_y at various Event-to-Noise (E/N) ratios. Ground truth indicated by red dotted lines. Note that the magnitude-rewarding rewards are much more robust at high event to noise ratios, still giving reasonable estimates at 1/100 E/N. The sparsity-rewarding rewards fail, becoming increasingly flat distributions with the center at $v = 0, 0$.

respect to aperture-uncertainty.

In all of the experiments the same gradient-ascent optimization was used, with numeric gradients derived through the same method. Optimization was performed on consecutive batches of events, using a sliding window with a width of one pixel displacement with regard to the optic flow velocity, with 100 samples taken per sequence.

Optical flow with respect to events is a slightly different creature from the conventional definition; usually optic flow measures the displacement of pixels between consecutive frames, hence the usual way of evaluating optical flow is through looking at the average endpoint error. For events however, this definition makes less sense, since optical flow relates to the velocity of an event on the image plane, not the displacement. Instead we will look at the average absolute magnitude error $\mu(|\text{ME}|)$, so the average error of the magnitudes of the optical flow vector estimates with respect to ground truth, and the average angular error $\mu(\text{AE})$, so the average error of the estimated vector angles. This way we can also look at the standard deviation of the errors, $\sigma^2(|\text{ME}|)$ and $\sigma^2(\text{AE})$.

4.1. Line Segment Sequence

The line-segment sequence (Fig. 9) illustrates the behavior of the different rewards when exposed to data with strong line features (this particular sequence consisting *only* of line features). As discussed in Section 2.1, one would expect sparsity-rewarding rewards to perform better on this

sequence, at least in the case where the event to noise ratio is high. Indeed, of the conventional rewards, the sparsity rewarding ones r_{ISOA} and r_{SoSA} score best under this extreme case of aperture uncertainty (Tab. 1).

However, as hypothesized, once noise is added to the event stream, the sparsity-rewarding rewards perform much worse than the magnitude-rewarding ones. It is worth pointing out here the remarkable robustness of the CM method; even with an order of magnitude more noise polluting the events, the optic flow estimates are still quite reasonable and only slightly deteriorated. The hybrid methods, which are able to take advantage of the best properties of both types of reward, perform best under both normal conditions and with large amounts of noise.

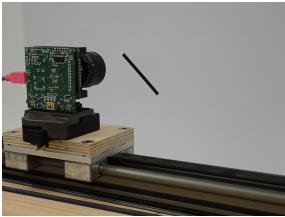
4.2. Circle Sequence

In the circle sequence the event camera slides past an image of a circle generating the events visualized in Fig. 10. This sequence illustrates a scene in which there is no aperture-uncertainty, since there are no dominant line features in the resulting events. In fact, the winner in such a scene is the commonly implemented r_{SoS} , which performs almost three times better than even our hybrid rewards, though these have got slightly better accuracy in the average angular error (Tab. 2).

Once noise is added to the event stream however, the accuracy of r_{SoS} is two entire orders of magnitude worse, whereas the hybrid reward r_{R2} only becomes five times

Line Segment Sequence				
Event/Noise = 1/0 (No Noise)				
r	$\mu(ME)$	$\sigma^2(ME)$	$\mu(AE)$	$\sigma^2(AE)$
r_{SoS}	38.29	1.68	-0.311	0.007
r_{SoE}	137.51	14.02	-0.546	0.019
r_{MoA}	22.41	10.92	-0.280	0.038
r_{ISoA}	11.36	5.41	-0.204	0.006
r_{SoSA}	12.79	1.02	-0.234	0.006
r_{R1}	10.31	3.26	-0.122	0.003
r_{R2}	10.50	3.14	-0.103	0.004
Event/Noise = 1/10				
r	$\mu(ME)$	$\sigma^2(ME)$	$\mu(AE)$	$\sigma^2(AE)$
r_{SoS}	49.14	25.48	-0.611	0.077
r_{SoE}	47.64	30.86	-0.712	0.401
r_{MoA}	82.11	33.63	-0.482	0.051
r_{ISoA}	88.06	11.25	0.559	1.096
r_{SoSA}	67.05	3.60	-0.635	0.010
r_{R1}	55.33	12.56	-0.547	0.012
r_{R2}	37.48	3.52	-0.440	0.007

Table 1: Absolute magnitude error $|ME|$ and angular error AE of flow estimate vectors for line segment sequence



(a) Experimental setup:
Event camera on a linear slider pans past circle.



(b) Timeslice of events to be processed.



(c) Noise added to the events.



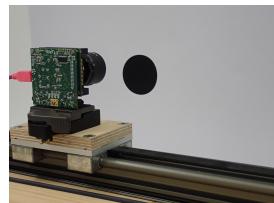
(d) Motion-compensated events using r_{R2} .

Figure 9: Line segment sequence: in the experimental setup, the event camera is moved past a line segment using a linear slider. A slice of the resulting events is processed to extract optical flow estimates, both with noise added to the event stream and once using only the original events.

worse, clearly beating the other methods. Interestingly, the extreme magnitude-rewarding rewards r_{SoE} and r_{MoA} actually improve as the scene becomes noisier. This is because these reward functions have quite strong peaks and are thus prone to local convergence issues; the noise effectively blurs the reward function and thus makes gradient ascent easier. In the interest of a fair comparison, the same amount of smoothing was applied to all rewards during optimization; for more details on the effect of smoothing on different reward types, see the supplementary materials.

Circle Sequence				
Event/Noise = 1/0 (No Noise)				
r	$\mu(ME)$	$\sigma^2(ME)$	$\mu(AE)$	$\sigma^2(AE)$
r_{SoS}	0.49	0.37	-0.153	0.010
r_{SoE}	72.14	11.20	1.517	0.033
r_{MoA}	73.45	18.32	1.066	0.051
r_{ISoA}	2.37	1.91	-0.305	0.034
r_{SoSA}	1.25	0.78	-0.350	0.009
r_{R1}	1.58	1.27	-0.100	0.009
r_{R2}	1.69	1.42	-0.036	0.040
Event/Noise = 1/10				
r	$\mu(ME)$	$\sigma^2(ME)$	$\mu(AE)$	$\sigma^2(AE)$
r_{SoS}	43.41	2.52	-0.512	0.010
r_{SoE}	23.56	27.46	0.354	0.640
r_{MoA}	11.10	3.83	0.132	0.090
r_{ISoA}	89.33	7.76	0.898	1.053
r_{SoSA}	72.79	2.61	-0.737	0.009
r_{R1}	57.95	14.91	-0.531	0.014
r_{R2}	5.29	1.57	0.113	0.053

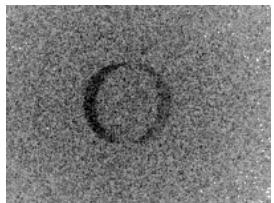
Table 2: Absolute magnitude error $|ME|$ and angular error AE of flow estimate vectors for circle sequence



(a) Experimental setup:
Event camera on a linear slider pans past circle.



(b) Timeslice of events to be processed.



(c) Noise added to the events.



(d) Motion-compensated events using r_{R2} .

Figure 10: Circle sequence: in the experimental setup, the event camera is moved past a circle using a linear slider. A slice of the resulting events is processed to extract optical flow estimates, both with noise added to the event stream and once using only the original events.

4.3. Office Sequence

The office sequence (Fig. 11) consists of panning across an office scene, to which the ground truth optic flow velocities were hand-annotated afterward. The sequence serves to illustrate that the ideas tested in the previous experiments also apply to real scenarios. As is often the case in real world sequences, there are several strong line features visible in the event stream, due to the edges of windows, tables, etc. As such, it is hardly surprising to see that our hybrid approach is able to out-compete the existing methods both with and without added noise (Tab. 3).

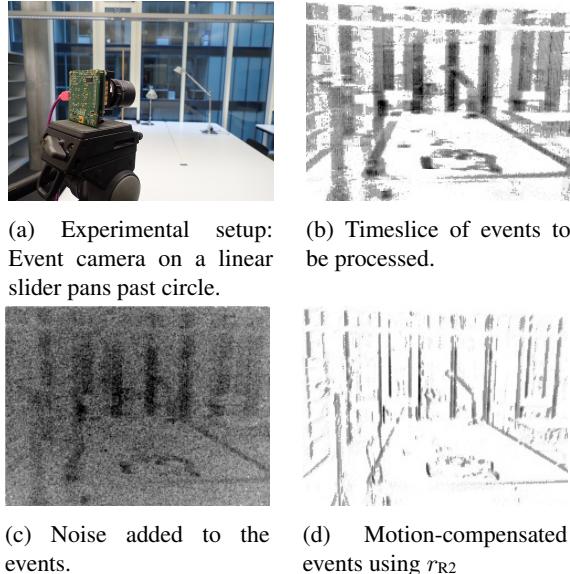


Figure 11: Office sequence: in the experimental setup, the event camera is panned around an office. A slice of the resulting events is processed to extract optical flow estimates, both with noise added to the event stream and once using only the original events.

Office Sequence				
Event/Noise = 1/0 (No Noise)				
r	$\mu(ME)$	$\sigma^2(ME)$	$\mu(AE)$	$\sigma^2(AE)$
r_{SoS}	5.58	3.58	-0.116	0.029
r_{SoE}	17.89	18.43	0.178	0.879
r_{MoA}	19.79	19.68	0.369	0.890
r_{IsOA}	16.28	18.64	-0.064	0.365
r_{SoSA}	5.47	3.28	-0.170	0.041
r_{R1}	9.09	12.93	-0.009	0.041
r_{R2}	4.95	3.23	-0.008	0.112
Event/Noise = 1/10				
r	$\mu(ME)$	$\sigma^2(ME)$	$\mu(AE)$	$\sigma^2(AE)$
r_{SoS}	16.07	5.95	-0.455	0.042
r_{SoE}	15.83	14.98	-0.084	0.766
r_{MoA}	20.26	17.40	-0.139	0.692
r_{IsOA}	46.82	3.55	-0.679	1.121
r_{SoSA}	49.09	0.40	-0.993	0.684
r_{R1}	48.66	1.45	-1.119	0.312
r_{R2}	15.08	14.66	-0.048	0.262

Table 3: Absolute magnitude error $|ME|$ and angular error AE of flow estimate vectors for office sequence

4.4. Real World Noise

Since the previous experiments include synthetic noise, we show a sequence with natural noise. In our experience even if the contrast thresholds are re-calibrated, extreme temperatures, direct sunlight, or very dim scenes tend to generate large amounts of noise in current event cameras. Even in a perfect sensor, phenomena such as lens flare can cause large numbers of “erroneous” events to be generated. In Fig. 12 many noise events are generated as a cyclist rides past while the event camera faces straight into the sun. In this case, noise events are generated by both the sun as well

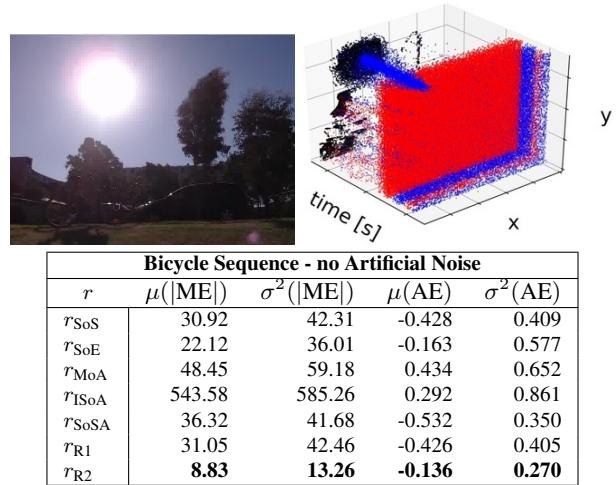


Figure 12: Below: Absolute magnitude error $|ME|$ and angular error AE of flow estimate vectors for bicycle sequence. Above: Conventional frame of the scene, as a cyclist moves past the camera. On the right, the resulting events are shown in the spatiotemporal volume, displaying significant amounts of noise. Figure animates in Adobe Acrobat.

as by lens flaring. Here the r_{R2} reward performs *substantially* better than the other metrics, with r_{SoE} following second, showing that our findings translate well to real-world settings.

5. Conclusion

In this paper we show that the underlying assumption made about contrast maximization is that good trajectory estimates are those where events accumulate in fewer locations. From this observation we are able to devise two categories of reward function, sparsity, and magnitude rewarding functions and from this categorization create four other rewards.

We touch upon the issue of how many events are needed to make good predictions and how this quantity can be estimated. We show which kind of data is likely to cause errors due to aperture uncertainty and that sparsity-rewarding rewards are much less susceptible to this uncertainty. We test various derived reward functions on real data and confirm the hypothesized traits of sparsity and magnitude rewarding functions. Using this knowledge we create the r_{R1} and r_{R2} rewards, which experimentally perform better than previous rewards. Thus, we hope that this work will aid future event-based vision research, providing better reward functions and stimulating discussion about what these rewards fundamentally do.

Acknowledgements

We encourage the reader to look at the supplementary material for additional experiments and proofs. This work was supported by the ARC Centre of Excellence for Robot Vision, project number CE140100016 (www.roboticvision.org).

References

- [1] M. Yang, S.-C. Liu, and T. Delbruck, “A dynamic vision sensor with 1% temporal contrast sensitivity and in-pixel asynchronous delta modulator for event encoding,” *IEEE J. Solid-State Circuits*, vol. 50, no. 9, pp. 2149–2160, 2015. [1](#)
- [2] P. Lichtsteiner, C. Posch, and T. Delbruck, “A 128×128 120 dB $15\ \mu s$ latency asynchronous temporal contrast vision sensor,” *IEEE J. Solid-State Circuits*, vol. 43, no. 2, pp. 566–576, 2008. [1](#)
- [3] G. Gallego, H. Rebecq, and D. Scaramuzza, “A unifying contrast maximization framework for event cameras, with applications to motion, depth, and optical flow estimation,” in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pp. 3867–3876, 2018. [1, 2, 4, 5](#)
- [4] A. Rosinol Vidal, H. Rebecq, T. Horstschafer, and D. Scaramuzza, “Ultimate SLAM? combining events, images, and IMU for robust visual SLAM in HDR and high speed scenarios,” *IEEE Robot. Autom. Lett.*, vol. 3, pp. 994–1001, Apr. 2018. [2](#)
- [5] A. Nguyen, T. Do, D. G. Caldwell, and N. G. Tsagarakis, “Real-time pose estimation for event cameras with stacked spatial LSTM networks,” *arXiv e-prints*, Oct. 2017. [2](#)
- [6] A. Z. Zhu, L. Yuan, K. Chaney, and K. Daniilidis, “EV-FlowNet: Self-supervised optical flow estimation for event-based cameras,” in *Robotics: Science and Systems (RSS)*, 2018. [2, 5](#)
- [7] M. Liu and T. Delbruck, “Adaptive time-slice block-matching optical flow algorithm for dynamic vision sensors,” in *British Mach. Vis. Conf. (BMVC)*, 2018. [2](#)
- [8] E. Mueggler, G. Gallego, and D. Scaramuzza, “Continuous-time trajectory estimation for event-based vision sensors,” in *Robotics: Science and Systems (RSS)*, 2015. [2](#)
- [9] P. Bardow, A. J. Davison, and S. Leutenegger, “Simultaneous optical flow and intensity estimation from an event camera,” in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pp. 884–892, 2016. [2](#)
- [10] H. Rebecq, G. Gallego, and D. Scaramuzza, “EMVS: Event-based multi-view stereo,” in *British Mach. Vis. Conf. (BMVC)*, 2016. [2](#)
- [11] E. Mueggler, G. Gallego, H. Rebecq, and D. Scaramuzza, “Continuous-time visual-inertial odometry for event cameras,” *IEEE Trans. Robot.*, 2018. [2](#)
- [12] G. Gallego and D. Scaramuzza, “Accurate angular velocity estimation with an event camera,” *IEEE Robot. Autom. Lett.*, vol. 2, no. 2, pp. 632–639, 2017. [2, 5](#)
- [13] T. Stoffregen and L. Kleeman, “Simultaneous optical flow and segmentation (SOFAS) using Dynamic Vision Sensor,” in *Australasian Conf. Robot. Autom. (ACRA)*, 2017. [2, 3, 4, 5](#)
- [14] A. Z. Zhu, N. Atanasov, and K. Daniilidis, “Event-based feature tracking with probabilistic data association,” in *IEEE Int. Conf. Robot. Autom. (ICRA)*, pp. 4465–4470, 2017. [2, 5](#)
- [15] A. Mitrokhin, C. Fermuller, C. Parameshwara, and Y. Aloimonos, “Event-based moving object detection and tracking,” in *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, 2018. [2, 3, 4, 5](#)
- [16] C. Scheerlinck, N. Barnes, and R. Mahony, “Continuous-time intensity estimation using event cameras,” in *Asian Conf. Comput. Vis. (ACCV)*, 2018. [2](#)
- [17] C. Scheerlinck, N. Barnes, and R. Mahony, “Asynchronous spatial image convolutions for event cameras,” *IEEE Robot. Autom. Lett.*, vol. 4, pp. 816–822, Apr. 2019. [2](#)
- [18] T. Stoffregen, G. Gallego, T. Drummond, L. Kleeman, and D. Scaramuzza, “Event-based motion segmentation by motion compensation,” *arXiv e-prints*, Apr. 2019. [2, 4, 5](#)
- [19] B. D. Lucas and T. Kanade, “An iterative image registration technique with an application to stereo vision,” in *Int. Joint Conf. Artificial Intell. (IJCAI)*, pp. 674–679, 1981. [3](#)
- [20] E. Mueggler, C. Forster, N. Baumli, G. Gallego, and D. Scaramuzza, “Lifetime estimation of events from dynamic vision sensors,” in *IEEE Int. Conf. Robot. Autom. (ICRA)*, pp. 4874–4881, 2015. [3](#)
- [21] C. Brandli, R. Berner, M. Yang, S.-C. Liu, and T. Delbruck, “A 240×180 130dB 3us latency global shutter spatiotemporal vision sensor,” *IEEE J. Solid-State Circuits*, vol. 49, no. 10, pp. 2333–2341, 2014. [5](#)