A Low Power, High Throughput, Fully Event-Based Stereo System: Supplementary Documentation

Alexander Andreopoulos, Hirak J. Kashyap, Tapan K. Nayak, Arnon Amir, Myron D. Flickner
IBM Research

March 25, 2018

1 Runtime System Architecture

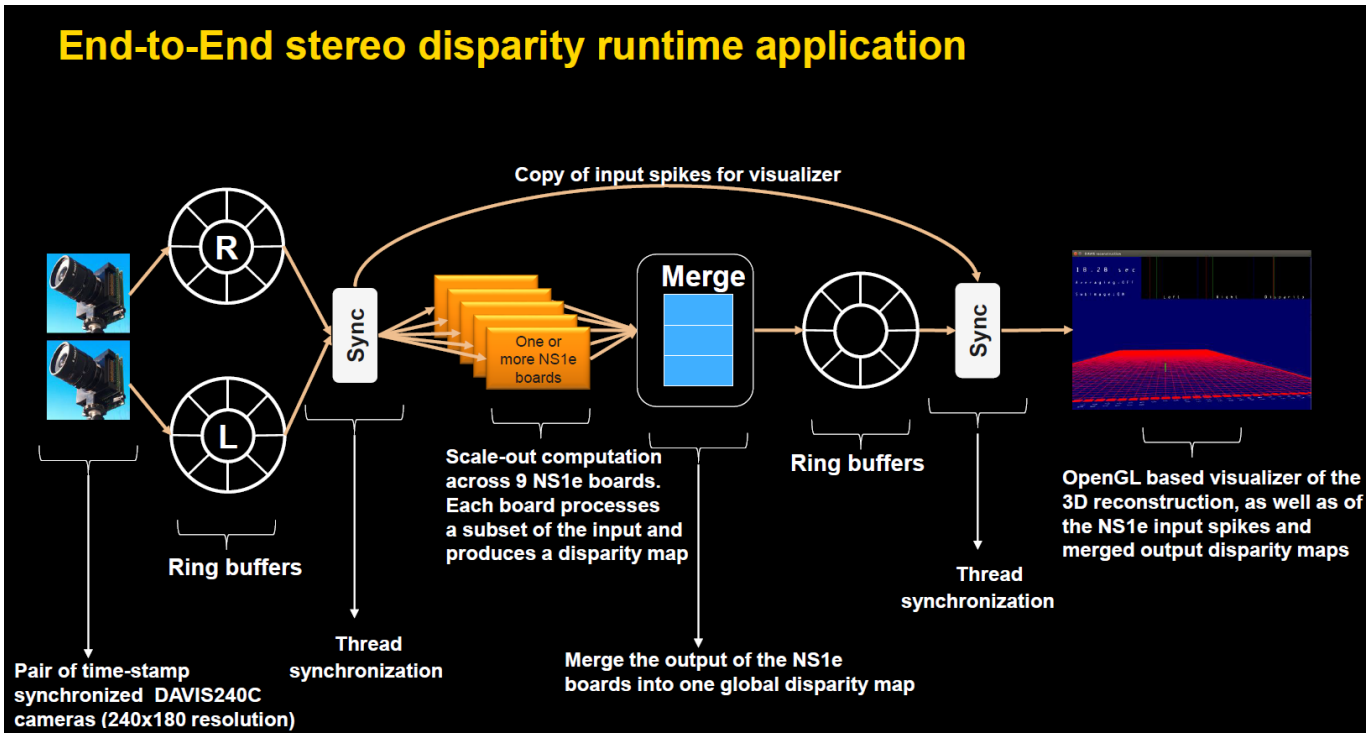


Figure 1: End-to-End system architecture. Each NS1e board processes a cropped subset of the input at a user-specified set of spatio-temporal scales. These cropped outputs are then merged to create a global disparity map which is used by an openGL-based visualizer to do reconstruct the 3D coordinates of the events.

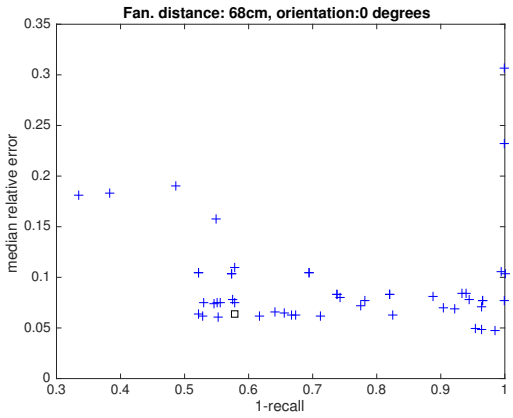
2 Experiments

We give a more detailed break-down of the results presented in the paper for the non-synthetic data. For the Fan sequences we show the results for different distances and different orientations of the fan blades, across all the 60 different models we tested (Figures 2-4). For the Butterfly sequences we show the results at three different distances of the toy Butterfly, across all 60 models tested (Figure 5). We also show the corresponding Kinect based ground truth maps. Notice that the native Kinect depth maps use a different scale, and therefore we had to execute a separate calibration process to regress the native Kinect units of depth to the metric depth units used during evaluation. In Figure 6 we give examples of projecting the 3D coordinates of events extracted using the neuromorphic algorithm, into the Kinect coordinate frame for evaluation purposes. As indicated in the paper, this involved a separate calibration process for finding the rotation and translation parameters that align the undistorted Kinect and DAVIS coordinate frames. The calibration procedures are described in more detail later in this document.

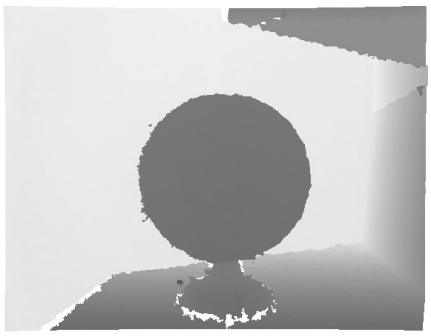
Overall, we notice that distance and orientation does not have a significant effect on the distribution of performances of the sixty models as a whole. Depending on the desired distance and orientation there is a model with performance characteristics similar to those at a different distance and orientation. However the optimal model to use changes depending on the tested sequence characteristics. To investigate this further we tracked the performance of one of the models executed on the TrueNorth simulator, which had the most similar characteristics to the model used as our baseline model in the real-time streaming demo. This baseline model is denoted by a square in the corresponding figures. The model parameters include a single spatial scale (90×120 pixel inputs resulting from subsampling inside of TrueNorth the native 180×240 DAVIS output), a temporal scale of $T = 4$, left-right bidirectional consistency checking enabled, 3×5 windows applied on the 90×120 input, a matching threshold of 4 (i.e., at least 4 pixel events must match in any two templates/windows compared from the left and right sensor) and 21 disparity levels (0-20) plus a 'no disparity' class for pixels where no left-right match was found. On the Fan sequences, we observe that as the distance of the object increases, relative error rates do not increase dramatically, however this comes at the expense of a decreasing recall rate. On the Butterfly sequences we notice similar performance with different depths.

In our preliminary tests with the non-synthetic data we noticed that for negative events there tended not to be many matches, or the matches tended to be of low quality. This is explainable by the fact that negative events tended to either get discarded during the matching process (especially due to bidirectional checks) or they ended up not being a reliable cue. This is because we used a black background and thus negative events tended to correspond to background pixels (and not the foreground object) as the foreground object passed over the background. Therefore for the results presented in this paper on the fan and butterfly sequences we suppressed negative input events and only used the positive events as input (even though the models were created to expect both types of events). For a light/white background we would expect the same effect to be achieved by using the negative instead of the positive events. Notice that for the synthetic test presented in the paper we used both polarities (each input pixel was encoded either as a positive or a negative event).

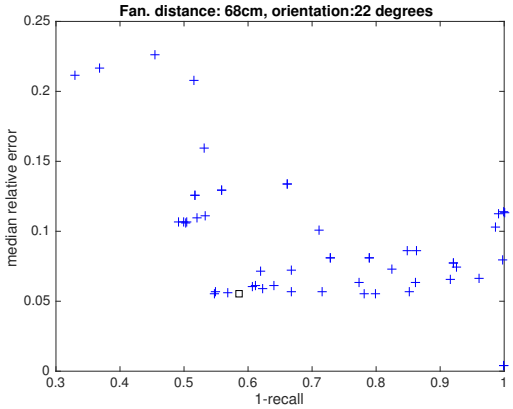
We notice that temporal scale is quite important depending on the speed of the moving object. Recall rates are affected by the distance of the events from the center of the rotating blades (the further from the center, the higher the velocity), implying that temporal scale plays an important role in the best performing models. On the Butterfly sequences we do not notice this effect, likely due to the slower rotation speed of the object. We also observe that morphological erosion and dilation (with 3×3 structuring elements) tends to improve the recall rates (often by over 10%), while the corresponding relative error rates do not tend to increase by more than $\sim 2\%$ (0.02 in absolute terms). Overall we observe that some of the best results occur when at least one of morphological erosion/dilation, or an increase in temporal scale, occurs. The practical effect of both of these operations is to increase the density of the events over which matching occurs. This reinforces the notion that transforming the sparse events into a more dense structural representation will be vital in the design of future neuromorphic systems for event based stereo.



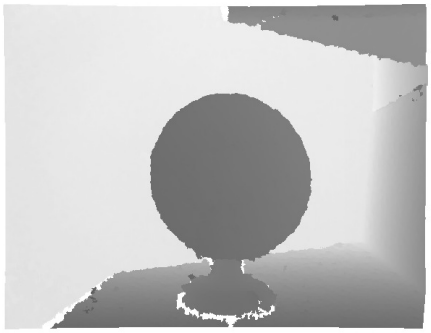
(a)



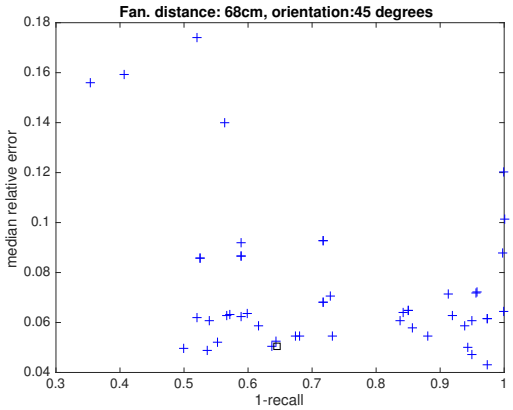
(b)



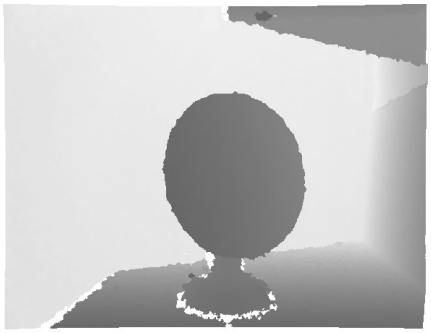
(c)



(d)

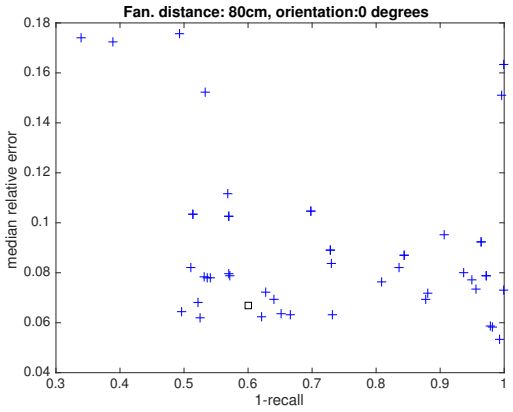


(e)

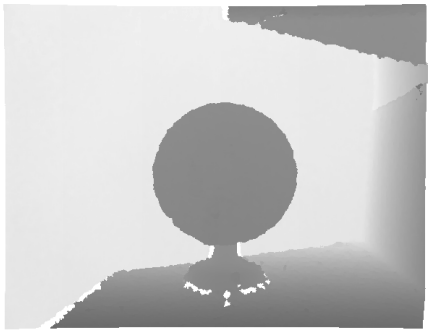


(f)

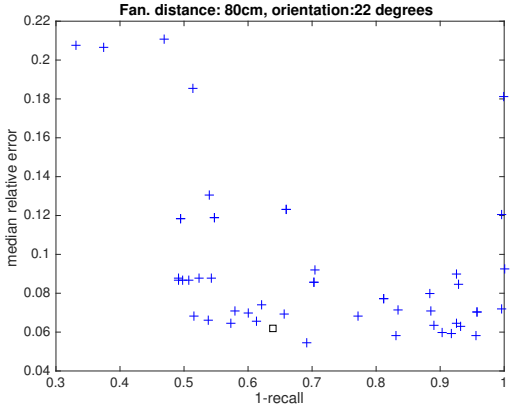
Figure 2: Median relative error vs. 1-recall, for the Fan sequence at the closest distance (Fan center at 68cm in the Kinect coordinate frame), and 3 different orientations ((a),(c),(e)). Orientation refers to the approximate angle of the fan blade plane normal with respect to the Kinect optical axis. We also show the corresponding ground truth depth maps extracted using the Kinect ((b),(d),(f)). Ground truth images created by merging multiple Kinect depth frames. Distances are in the Kinect coordinate frame. In the DAVIS coordinate frame distances are $\sim$ 11cm smaller. The baseline model is denoted with a black square.



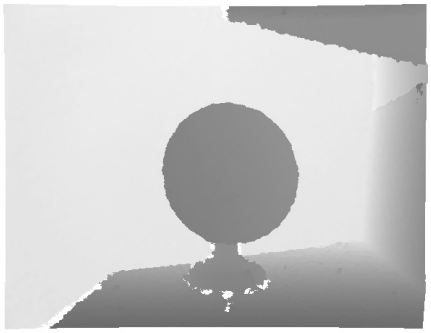
(a)



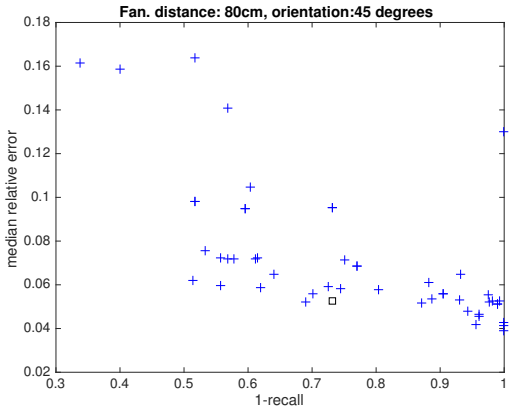
(b)



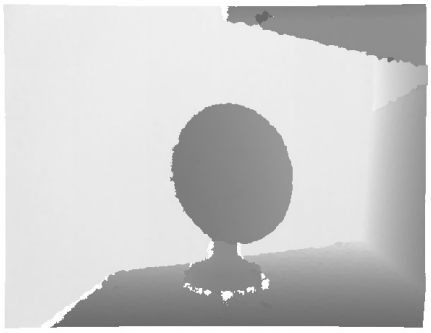
(c)



(d)

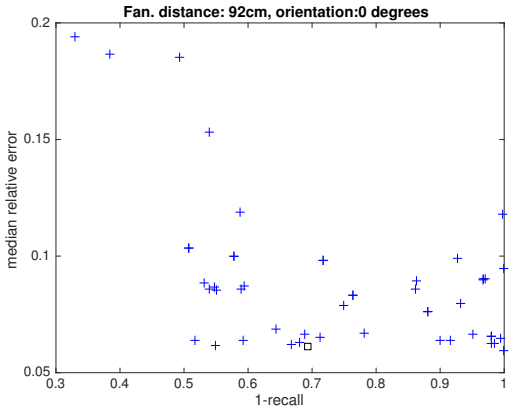


(e)

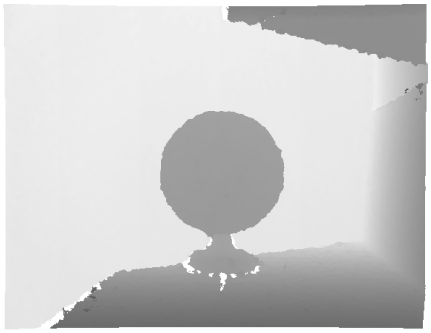


(f)

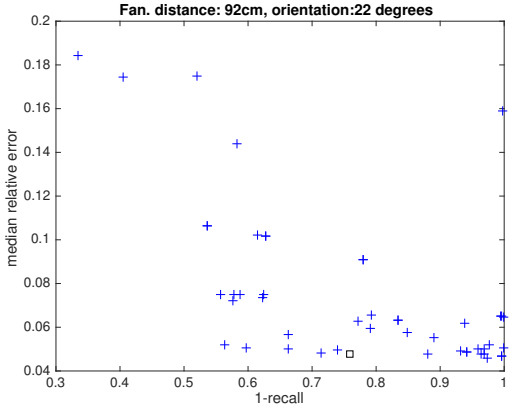
Figure 3: Median relative error vs. 1-recall, for the Fan sequence at the median distance (Fan center at 80cm in the Kinect coordinate frame), and 3 different orientations ((a),(c),(e)). Orientation refers to the approximate angle of the fan blade plane normal with respect to the Kinect optical axis. We also show the corresponding ground truth depth maps extracted using the Kinect ((b),(d),(f)). Ground truth images created by merging multiple Kinect depth frames. Distances are in the Kinect coordinate frame. In the DAVIS coordinate frame distances are $\sim$ 11cm smaller. The baseline model is denoted with a black square.



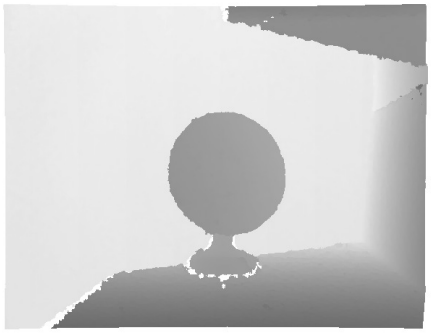
(a)



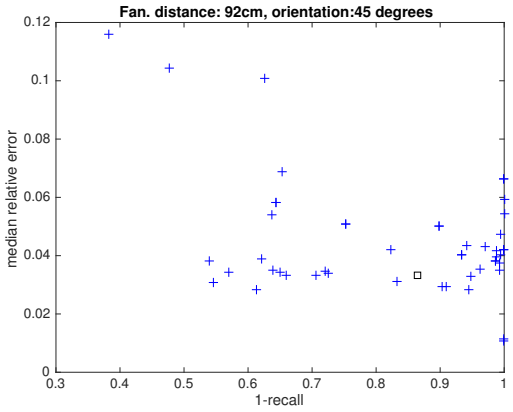
(b)



(c)



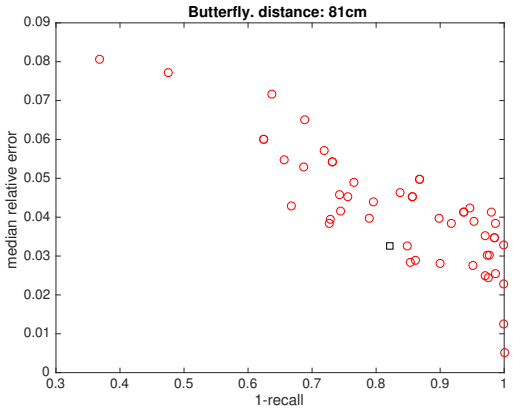
(d)



(e)

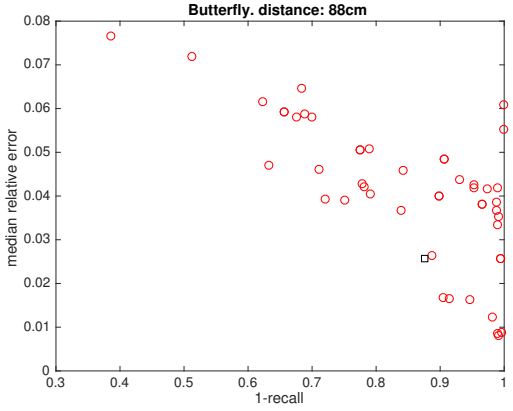
(f)

Figure 4: Median relative error vs. 1-recall, for the Fan sequence at the farthest distance (Fan center at 92cm in the Kinect coordinate frame), and 3 different orientations ((a),(c),(e)). Orientation refers to the approximate angle of the fan blade plane normal with respect to the Kinect optical axis. We also show the corresponding ground truth depth maps extracted using the Kinect ((b),(d),(f)). Ground truth images created by merging multiple Kinect depth frames. Distances are in the Kinect coordinate frame. In the DAVIS coordinate frame distances are $\sim$ 11cm smaller. The baseline model is denoted with a black square.

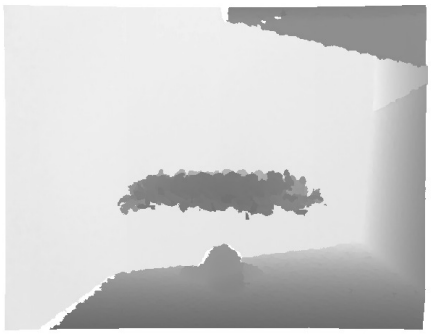


(a)

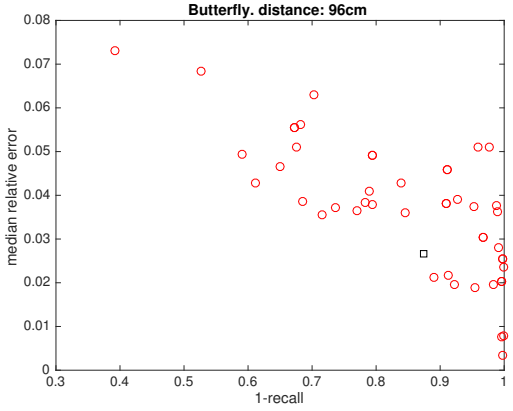
(b)



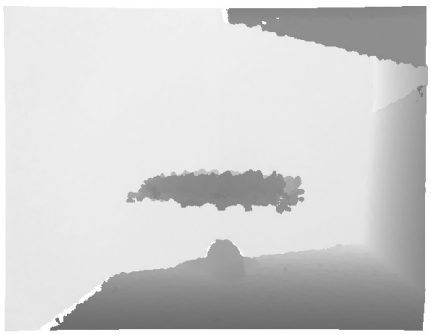
(c)



(d)

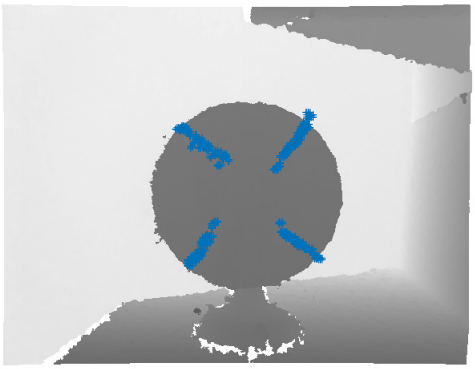


(e)

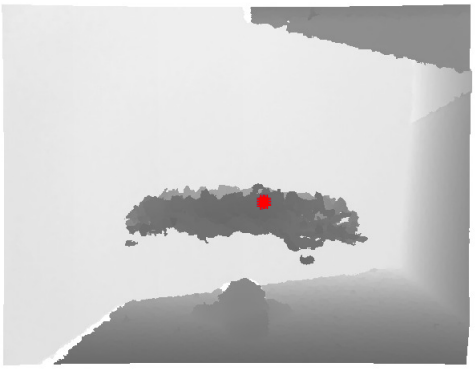


(f)

Figure 5: Median relative error vs. 1-recall, for the Butterfly sequence at three distances —Butterfly base center at 81cm, 88cm, 96cm — ((a),(c),(e)). We also show the union of the corresponding ground truth depth maps extracted using the Kinect, which trace the Butterfly's motion trajectory ((b),(d),(f)). Distances are in the Kinect coordinate frame. In the DAVIS coordinate frame distances are $\sim$ 11cm smaller. The baseline model is denoted with a black square.



(a)



(b)

Figure 6: Examples of projecting the 3D coordinates of events extracted in the DAVIS coordinate frame using the neuromorphic algorithm, to the Kinect coordinate frame. Projected points shown in blue (a) and red (b).

3 Calibration Procedures

There were three calibration phases involved: (i) Calibrating the DAVIS stereo rig, (ii) calibrating the Kinect depth value units to align them with the units (cm) used by the DAVIS stereo rig, and (iii) determining the rotation/translation matrix to align the Kinect depth coordinate frame with the DAVIS stereo rig coordinate frame.

The first calibration phase involved finding the radial distortion parameters, and intrinsic matrices for both DAVIS cameras, as well as the rotation/translation matrices defining their extrinsic parameters. Calibration data was obtained using a slowly moving standard checkerboard calibration pattern. Moving the pattern was necessary to obtain events, due to the nature of the DVS sensor. Given left-right time-synchronized pairs of events, we fused the events into ~ 4 ms time intervals to obtain dense frame reconstructions of the checkerboard pattern edges. We obtained twenty-five such left-right pairs and then manually specified the matching points in the two frames. The Matlab Computer Vision/Calibration toolbox was used subsequently to obtain the calibration parameters. Notice that the DAVIS sensor can also output grayscale images making it possible to do the calibration in grayscale space, which can be simpler since the corner extraction procedure can be automated more easily. However this method is not suitable for all event based sensors.

The second calibration phase involved the Kinect sensor. Using the OpenKinect library we accessed both the RGB and the registered depth map frames. Using the same checkerboard calibration pattern, we performed a single-camera calibration, obtaining radial distortion and intrinsic calibration parameters of the RGB frames. Notice that since the dimensions of the checkerboard pattern are known this suffices to obtain the 3D coordinates of each corner in the calibration pattern appearing in an RGB frame. For each of the RGB frames used in this calibration we also had a corresponding Kinect depth map and the associated depth value at each pixel where a checkerboard corner lied. The goal here was to use the Kinect depth map to obtain (X,Y,Z) depth coordinates in the Kinect frame (post-radial distortion correction). As the Kinect depth values are not necessarily in metric units, we examined the use of a linear regression model to map each each Kinect depth value to its corresponding Z value. The linear regression model was shown to be an excellent predictor of Z values for the maximum ranges of $\sim 2m$ across which we would be doing our evaluation ($R^2 \approx 0.99$). Given the ability to map each Kinect depth value to a Z value, given the Kinect intrinsic parameters and the pixel coordinates of the Z value, it then became possible to also extract the corresponding X,Y values.

The third and final calibration step entailed finding the rotation/translation matrix relating the undistorted Kinect frame and the undistorted left DAVIS sensor coordinate frame. Again a calibration pattern was imaged at different distances and orientations, and corresponding corners in the DAVIS event space frames and the Kinect RGB frames were manually selected. Given that the dimensions of the calibration pattern, and the sensor distortion/intrinsic parameters were known it became possible to extract the 3D coordinates of each corner. At this point we applied a simple closed-form solution to find the best rotation/translation that aligned the two sets of points.

4 Ground Truth and Evaluation Metrics

As indicated previously, the two non-synthetic sequences consist of a rotating fan and a rotating toy butterfly captured using the DAVIS stereo cameras. A Kinect was used to extract ground truth of the scene structure. Note that during DAVIS event acquisition it is not possible to also have the Kinect operating simultaneously, as this results in interference patterns in the events generated, due to the IR sensor used by the Kinect. As discussed below, this affected our testing strategy.

Performance is measured in terms of precision, which is defined as the median relative error $\frac{\|x-x'\|}{\|x'\|}$ between each 3D coordinate x extracted in the DAVIS frame using the neuromorphic algorithm, and the corresponding ground coordinate x' in the aligned Kinect coordinate frame. Performance is also reported in terms of the recall, defined herein as the percentage of DAVIS pixels in rectified space containing events (left sensor), where a disparity estimate was also extracted. We tested a suite of sixty stereo disparity networks generated with ranges of spatiotemporal scales, denoising parameters, kernel match thresholds, with/without left-right consistency constraints etc. The list of parameter valus used is described in detail later in this document.

In the previous section we described the calibration process for transforming the undistorted Kinect coordinate frame to the undistorted DAVIS sensor coordinate frame (and trivially vice-versa). We take advantage of this mapping to map each 3D point extracted using the DAVIS sensors to a corresponding ground truth coordinate extracted using the Kinect. Nine Fan sequences (3 distances $\times$ 3 orientations) and three Butterfly sequences (3 distances) are used.

The fan sequence is useful for testing the ability of the algorithm to operate on rapidly moving objects. Ground truth is extracted in terms of the plane in 3D space representing the blades' plane of rotation. Multiple Kinect depth frames are extracted as the blade is rotating, and we fuse those depth maps into one single depth map for each desired distance and orientation of the blades, as shown in previous figures. Given a 3D point extracted with the DAVIS cameras, we transform that point to the Kinect coordinate frame, project that point to the Kinect image plane and find the corresponding ground truth value which enables us to find the error metric value. Ideally these projected points lie on the fan plane and have nearly identical depth values. The lesser degree that these conditions are satisfied, the higher the error becomes.

The butterfly sequence tests the ability of the algorithm to operate on non-rigid objects which are rapidly rotating in a circular plane approximately perpendicular to the y-axis. Notice that since the butterfly object is non-rigid, this brings certain challenges in the evaluation procedure. Given Kinect depth frames of the rotating object, we first apply a segmentation procedure to keep only the depth values corresponding to the butterfly object. This is accomplished by specifying the range of depth values where the butterfly object lies (notice in the previous figures that butterfly is always superimposed in-front of a distant background wall making this an easy task) and by specifying the pixel/spatial coordinates where the butterfly object may project to. This is a relatively simple procedure that gives us near perfect segmentations of the butterfly in the depth map. The end result is a boolean mask for each Kinect depth map frame. Given a 3D point extracted at a particular time instant using the DAVIS cameras, we need to determine an approximation of the point on the butterfly's rotational circumference where it should lie (the ground-truth). This effectively entails a temporal registration scheme. As indicated previously the difficulty lies in the fact that the butterfly is a non-rigid object, and no two rotations of the butterfly will be exactly identical in terms of the extracted Kinect depth. Also for the reasons indicated previously, it is not possible to simultaneously record from the DAVIS sensors and the Kinect. Also notice that the frame rate of the Kinect is much lower than the DAVIS sensor. Therefore the temporal registration scheme cannot rely on Kinect depth frames extracted simultaneously as the DAVIS events were recorded. Therefore we use the Kinect to record the butterfly rotating, while the DAVIS sensor is not recording. We then extract the corresponding binary masks using the procedure described above. Given a 3D point extracted at a particular time instant using the DAVIS cameras, we find all the Kinect binary masks which overlap the 3D point projection in the Kinect pixel space (providing candidates as to the correct circumference location), and extract the corresponding depths and their associated (X, Y, Z) coordinates. This enables use to find relative error rates for each binary mask. Since we may have more than one matching binary mask, we select the minimum relative error as the error metric for this 3D data point. See the accompanying video for examples of the reconstructed depths.

5 Model Parameters and Results

We present all 60 models parameters used and the respective error/recall results when applied on the two sequences. The model parameter descriptions are as follows:

- **Polarities:** Whether both input event channels are used with their polarity sign preserved ($\{+,-\}$), or whether both input event channels are used but their polarity sign is ignored ($|+,-|$).
- **Spatial scales:** The stride/subsampling factor (for each of the X,Y dimensions) used for each scale. A value of n denotes retaining every n^{th} input pixel.
- **Temporal scales:** The temporal scale assigned to each corresponding spatial scale used above.
- **Disparity ranges:** The mininum and maximum output disparity.
- **Window size:** The $height \times width$ of each spatial window over which matching takes place in each spatial scale.
- **Erosion/Dilation:** Whether erosion/dilation was used.
- **Bidirectional check:** Whether a bidirectional consistency check was enforced.
- **Match threshold:** The minimum threshold to consider two windows (the result of their Hadamard product sum across all spatial scales) as matching.
- **Fan Error:** The median relative error of the model on the fan sequences.
- **Fan Recall:** The recall of the model on the fan sequences.
- **Butterfly Error:** The median relative error of the model on the butterfly sequences.
- **Butterfly Recall:** The recall of the model on the butterfly sequences.

Table 1: Parameters and results for models 1-20

Model	Polarities	Spatial scales	Temporal scales	Disparity ranges	Window size	Erosion/Dilation	Bidirectional check	Match Threshold	Fan Error	Fan Recall	Butterfly Error	Butterfly Recall
1	$\|+,-\|$	1	1	0-40	3×5	No	Yes	4	0.084	0.039	0.030	0.011
2	$\|+,-\|$	1	2	0-40	3×5	No	Yes	4	0.065	0.044	0.036	0.010
3	$\|+,-\|$	1	4	0-40	3×5	No	Yes	4	0.064	0.100	0.042	0.047
4	$\|+,-\|$	1	8	0-40	3×5	No	Yes	4	0.079	0.258	0.046	0.161
5	$\|+,-\|$	1	16	0-40	3×5	No	Yes	4	0.152	0.451	0.059	0.312
6	$\|+,-\|$	1	1	0-40	3×5	No	No	4	0.084	0.040	0.030	0.013
7	$\|+,-\|$	1	2	0-40	3×5	No	No	4	0.071	0.063	0.038	0.012
8	$\|+,-\|$	1	4	0-40	3×5	No	No	4	0.081	0.263	0.048	0.094
9	$\|+,-\|$	1	8	0-40	3×5	No	No	4	0.104	0.483	0.059	0.343
10	$\|+,-\|$	1	16	0-40	3×5	No	No	4	0.181	0.665	0.077	0.615
11	$\|+,-\|$	1	1	0-40	3×5	No	No	8	1	0.000	1	0.000
12	$\|+,-\|$	1	2	0-40	3×5	No	No	8	0.065	0.001	0.033	0.000
13	$\|+,-\|$	1	4	0-40	3×5	No	No	8	0.076	0.150	0.038	0.034
14	$\|+,-\|$	1	8	0-40	3×5	No	No	8	0.101	0.422	0.051	0.225
15	$\|+,-\|$	1	16	0-40	3×5	No	No	8	0.174	0.612	0.072	0.488
16	$\|+,-\|$	1	1	0-40	3×5	No	No	12	1	0.000	1	0.000
17	$\|+,-\|$	1	2	0-40	3×5	No	No	12	1	0.000	1	0.000
18	$\|+,-\|$	1	4	0-40	3×5	No	No	12	0.066	0.026	0.026	0.006
19	$\|+,-\|$	1	8	0-40	3×5	No	No	12	0.098	0.283	0.040	0.102
20	$\|+,-\|$	1	16	0-40	3×5	No	No	12	0.176	0.507	0.065	0.317

Table 2: Parameters and results for models 21-40

Model	Polarities	Spatial scales	Temporal scales	Disparity ranges	Window size	Erosion/Dilation	Bidirectional check	Match Threshold	Fan Error	Fan Recall	Butterfly Error	Butterfly Recall
21	$\{+,-\}$	1	4	0-40	3×5	No	No	4	0.081	0.263	0.048	0.094
22	$\{+,-\}$	1	4	0-40	3×5	No	No	8	0.076	0.150	0.038	0.034
23	$\{+,-\}$	1	4	0-40	3×5	No	No	12	0.066	0.026	0.026	0.006
24	$\{+,-\}$	1	8	0-40	3×5	No	No	4	0.104	0.483	0.059	0.343
25	$\{+,-\}$	1	8	0-40	3×5	No	No	8	0.101	0.422	0.051	0.225
26	$\{+,-\}$	1	8	0-40	3×5	No	No	12	0.098	0.283	0.040	0.102
27	$\|+,-\|$	1,2	1,1	0-40	3×5,3×5	No	Yes	4	0.068	0.157	0.043	0.047
28	$\|+,-\|$	1,2	4,4	0-40	3×5,3×5	No	Yes	4	0.054	0.320	0.040	0.219
29	$\|+,-\|$	1,2	8,8	0-40	3×5,3×5	No	Yes	4	0.066	0.373	0.041	0.289
30	$\|+,-\|$	1,2	16,16	0-40	3×5,3×5	No	Yes	4	0.110	0.413	0.057	0.301
31	$\|+,-\|$	2	1	0-20	3×5	No	Yes	4	0.118	0.003	1	0.000
32	$\|+,-\|$	2	2	0-20	3×5	No	Yes	4	0.063	0.050	0.033	0.010
33	$\|+,-\|$	2	4	0-20	3×5	No	Yes	4	0.055	0.354	0.027	0.125
34	$\|+,-\|$	2	8	0-20	3×5	No	Yes	4	0.060	0.448	0.038	0.280
35	$\|+,-\|$	2	16	0-20	3×5	No	Yes	4	0.074	0.453	0.053	0.324
36	$\|+,-\|$	2	1	0-20	3×5	No	Yes	8	1	0.000	1	0.000
37	$\|+,-\|$	2	2	0-20	3×5	No	Yes	8	1	0.000	1	0.000
38	$\|+,-\|$	2	4	0-20	3×5	No	Yes	8	0.057	0.120	0.020	0.019
39	$\|+,-\|$	2	8	0-20	3×5	No	Yes	8	0.059	0.362	0.026	0.113
40	$\|+,-\|$	2	16	0-20	3×5	No	Yes	8	0.075	0.411	0.049	0.211

Table 3: Parameters and results for models 41-60

Model	Polarities	Spatial scales	Temporal scales	Disparity ranges	Window size	Erosion/Dilation	Bidirectional check	Match Threshold	Fan Error	Fan Recall	Butterfly Error	Butterfly Recall
41	$\|+,-\|$	2	1	0-20	3×5	No	Yes	2	0.073	0.087	0.042	0.026
42	$\|+,-\|$	2	2	0-20	3×5	No	Yes	2	0.063	0.163	0.040	0.073
43	$\|+,-\|$	2	4	0-20	3×5	No	Yes	2	0.059	0.427	0.039	0.264
44	$\|+,-\|$	2	8	0-20	3×5	No	Yes	2	0.062	0.483	0.043	0.367
45	$\|+,-\|$	2	16	0-20	3×5	No	Yes	2	0.076	0.470	0.055	0.378
46	$\|+,-\|$	2	1	0-20	3×5	Yes	Yes	4	1	0.000	1	0.000
47	$\|+,-\|$	2	2	0-20	3×5	Yes	Yes	4	1	0.000	1	0.000
48	$\|+,-\|$	2	4	0-20	3×5	Yes	Yes	4	0.058	0.019	0.008	0.009
49	$\|+,-\|$	2	8	0-20	3×5	Yes	Yes	4	0.061	0.319	0.020	0.086
50	$\|+,-\|$	2	16	0-20	3×5	Yes	Yes	4	0.075	0.445	0.041	0.209
51	$\|+,-\|$	2	1	0-20	3×5	Yes	Yes	2	1	0.000	1	0.000
52	$\|+,-\|$	2	2	0-20	3×5	Yes	Yes	2	1	0.000	1	0.000
53	$\|+,-\|$	2	4	0-20	3×5	Yes	Yes	2	0.056	0.023	0.009	0.010
54	$\|+,-\|$	2	8	0-20	3×5	Yes	Yes	2	0.062	0.327	0.022	0.096
55	$\|+,-\|$	2	16	0-20	3×5	Yes	Yes	2	0.075	0.448	0.043	0.222
56	$\|+,-\|$	2	1	0-20	3×5	Yes	Yes	8	1	0.000	1	0.000
57	$\|+,-\|$	2	2	0-20	3×5	Yes	Yes	8	1	0.000	1	0.000
58	$\|+,-\|$	2	4	0-20	3×5	Yes	Yes	8	0.057	0.007	0.009	0.004
59	$\|+,-\|$	2	8	0-20	3×5	Yes	Yes	8	0.062	0.268	0.019	0.054
60	$\|+,-\|$	2	16	0-20	3×5	Yes	Yes	8	0.078	0.411	0.037	0.161