Continuous Trajectory Estimation from Depth Augmented Events

Alex Junho Lee[1], Jinyong Jeong[1], Younggun Cho[1], Sungho Yoon[2], Young-Sik Shin[3], and Ayoung Kim[1,2*]

***Abstract*— Event cameras have arisen as an alternative solution to trajectory estimation problem on real-world applications, by its consistent sensor measurement upon environmental variance. In this paper, we present a method of estimating a continuous trajectory from event measurements combined with a depth sensor. We combine sequential depth information with continuous event stream to estimate a 6 DOF trajectory. From accurate depth measurements from the infrared depth sensors or solid-state LiDARs, edge structures are extracted and fitted into the events to estimate a polynomial spline trajectory. We show that to fully utilize the benefits from continuous-time representation of events, camera localization should be solved as a continuous trajectory estimation problem. We also release a public dataset including dynamic lighting conditions and various motion in indoor and outdoor, with ground-truth provided. The dataset can be downloaded from : `https://sites.google.com/view/vivid-kaist`**

I. Introduction

Neuromorphic vision sensors, also known as Dynamic Vision Sensor (DVS) [1] has been proposed as an effective alternative of vision sensor from its several advantages over regular cameras. Frame-based imaging sensors produce an image, which is time-synchronized scene data during exposure time. The sequential output known as image data is intuitive and effective for visual odometry (VO) [2] and simultaneous localization and mapping (SLAM) [3] even with a single camera. And the sensor measurement is produced by integrating a number of photons arrived during the exposure time, requiring specific duration for every measurement. Therefore, image data has limits of the trade-off between frame-rate and dynamic range and is not able to sense changes occurred in exposure time. However, event cameras are free from the problems of classic ones. Neuromorphic vision is not based on the number of photons integrated for a duration but measures the rate of a photon entering each pixel. Therefore, event cameras avoid temporal integration and retain asynchronicity with high dynamic range (up to $130dB$, which is significantly higher than the $60dB$ of conventional cameras).

Since the event camera possesses an asynchronous Address-Event-Representation (AER) representation of illumination change, data acquired from event cameras are not demonstrated with discrete timings. To fully utilize the unique AER of event camera, any information obtained from the event camera should be dealt with as a continuous-time problem. However the number of events may sum up to millions per second, raising temporal sampling resolution does not solve the problem, yielding heavy computation load. Therefore event requires a total new way of data processing.

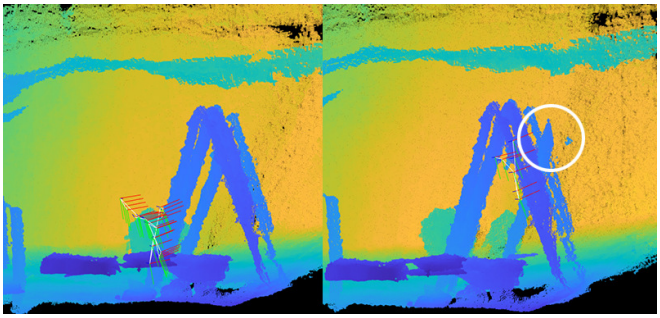


Fig. 1: Estimated trajectory and reconstruction from events with depth (left), Iterative Closest Points (ICP) on depth pointclouds (right). Trajectory estimation using only depth fails on extreme motion, as demonstrated in white circle.

Thus event cameras related researches have focused on finding a way on processing temporal information while less degenerating data. Since pose representation enables global optimization techniques like bundle adjustment or pose-graph, majority of research in event-based VO has been examined on transforming continuous signal into subsequent pose form [4], [5], [6], [7]. However, sequential procedures of classical image processing such as feature extraction, descriptor matching with calculating re-projection error were not necessary for event-based algorithms if the pose is not in a discrete representation [8] [9]. Despite the idea of directly fitting motion from event measurements were in a proper way of utilizing the benefits of event cameras, its relatively lower spatial resolution and noise made it difficult to use events as a single origin pose estimator. To deal with this problem, we suggest combining sparse sensor measurements with events, while maintaining continuous pose representation as a trajectory.

In this paper, we aim to solve the motion estimation problem with the benefits of event sensors by:

- Solving motion estimation problem by estimating polynomial spline trajectory from finding correspondence of continuous events between depth measurements
- Defining correspondence between events and disjointly updated depth, by associating depth discontinuity into event generating edges in the image domain.
- Releasing a first public dataset to cover illumination and motion variances by multi-modal sensor measurements consisted with event, depth, and thermal camera.

[1]Department of Civil and Environmental Engineering, KAIST, 291 Daehak-ro, Yuseong-gu, Daejeon 34141, Republic of Korea `[alex_jhlee,jjy0923,yg.cho,ayoungk]@kaist.ac.kr`

[2]Robotics Program, KAIST, 291 Daehak-ro, Yuseong-gu, Daejeon 34141, Republic of Korea `sungho.yoon@kaist.ac.kr`

[3] Korea Institute of Machinery and Materials Daejeon 34141, Republic of Korea `yshin86@kimm.re.kr`

Manuscript 2190 submitted to 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems. Received March 1, 2020.

II. Related Works

A. Event-based Visual Odometry

The unique advantages of AER and event cameras have encouraged researchers to solve motion estimation aided by event measurements. Since event cameras have both advantages and challenges as described in the previous section, researchers have developed several ways of defining relationship between events, that are continuously measured but asynchronous to typical discrete pose representations.

In the early stages, researchers established probabilistic approaches on event-based VO. A method that first succeeded involves calculating posterior probability with a temporal threshold. Kim et al. [10] has introduced to use Expectation Maximization (EM) schema on fitting events for upon rotational motion. This study was expanded to the optimization of 6-degree of freedom (DOF) trajectory with 3D reconstruction [7]. Their approach has revealed the way of directly optimizing the trajectory from events, yet was filter-based and had less robust due to numbers of assumptions. In [6], authors implemented a particle filter to minimize ray distance through all features in 2D. This study was broadened to include 3D SLAM in [11], but was only effective in planar scenes.

On the other hand, there was another approach reminding of surface of active events (SAE) introduced in [12]. [13] fitted a surface in an $\mathcal{XYT}$ space with SAE to estimate the effective temporal length for each event. These studies suggested finding the effective size of the temporal window and to only filter relevant events with adaptive temporal window size. The proposed algorithm provides a hint for utilizing the high temporal resolution characteristics of events by processing with individual timings.

In [14], a modified version of SAE with exponentially decaying kernels was introduced for VO. In this study, events are accumulated in exponential time-surface and calculated for an optimal camera pose by optimizing re-projection error with other measurement models in stereo-vision. The study focused on directly estimating the camera pose with exponentially decaying kernels, which provided a smoothed representation of SAE. The authors introduced a robust descriptor of SAE in addition to filtering with tree assignment. As a result, the author was able to enhance the mean tracking frequency on the scale. Although their work is based on the complex model and the user-defined descriptor, it has shown that calculating from each event's timestamp could arrange to a convergence.

The next was generating motion-related frame images by temporal thresholding. Numbers of researches had set effective events within a fixed time interval and tried to build event images for discrete trajectory estimation. However, this procedure integrates the asynchronous event signals into synchronous sequences with a fixed temporal window, resulting in the loss of piecewise timing information on each event. Therefore in order to achieve higher levels of accuracy and fully utilize the temporal resolution of event cameras, another solution were required than simple thresholding.

Several studies attempted to utilize the high temporal resolution of event cameras by assigning proper weights to each event. Gallego and Scaramuzza [15] introduced a direct method of accurately estimating angular velocity by compensating events for rotational motion. This approach was the first attempt to build a motion-compensated event frame, that could both conserve asynchronicity and frame-based odometry estimation. In combination with nonlinear optimization, [16] succeeded in integrating inertial measurement unit (IMU) for 6 DOF event VO. This study demonstrated a standard pipeline for event-based visual inertial navigation system (VINS). Although this approach became robust and effective by enabling classical feature-based methods on an event stream, the requirement of generation procedure on the motion-compensated event image for every pose and extracting features demands proper initialization.

Mueggler et al. [17] proposed a method of optimizing the trajectory with events and IMU by introducing cubic spline interpolation into [8]. Ultimate SLAM [9] was nonetheless the state-of-the-art method for integrating sensor measurements with motion-compensated event frames. However, the proposed algorithm required updating the spline parameter upon receiving each event, which was quite heavy in computation.

In this research, we introduce a method to estimate polynomial continuous trajectory in 3D, by directly optimizing spline parameters efficiently with continuous event measurements and sparsely obtained depth measurements.

B. Datasets for Environmental Variance

Numbers of datasets for benchmarking SLAM has been introduced [18], [19] with clear sight and high visibility. Meanwhile, in the real world, lighting conditions are often uncooperative, making computer vision algorithms fail. Moreover, there are few of datasets made to test experimental environments with environmental variations.

NCLT [20] and TUM MonoVO [21] introduced large scale data with huge variance in environments for long-term visual SLAM. These datasets cover challenging indoor and outdoor sequences including natural light and weather changes. However, obtaining the limited information from the classical camera restricts the bandwidth of data from the environment, skipping potentially important information.

By using other types of visual sensors, Choi et al. [22] presented a multi-spectral day/night dataset with the sensor set of stereo RGB, LiDAR and a thermal camera. Their data contains variance along the whole day. Furthermore, numbers of labeled data are provided for autonomous navigation. However, camera movements in the dataset are limited to planar motion because the system is mounted on a car.

In [23], the authors have presented a dataset measuring various lighting and motion sequences with two event cameras aligned with inertial sensors and an IMU. The dataset contains a large variety of condition changes, suggesting utilizing the event camera for low latency and high dynamic range characteristics.

Manuscript 2190 submitted to 2020 IEEE/RSJ International Conference
on Intelligent Robots and Systems. Received March 1, 2020.

Although several datasets have contained environmental variations including movement or lighting variances, unspecified natural environment often possess both of them. In this paper, we release a dataset to record both thermal and events with depth in order to deal with two significant disturbances: luminance conditions and motion. Along the dataset, we also suggest the method to estimate polynomial continuous trajectory in 3D, thereby directly optimizing spline parameters efficiently with given depth measurements.

III. Method

In our method, we start from depth measurement and preprocess the depth image with an edge detector to extract structural edges. As events are only generated upon changes in luminance, measurements occurred from event camera are all from photometric and geometric edges. Thus assuming enough structural than color difference, projected depth edges could be matched to event measurements.

Depth cameras, such as Kinect or Xtion, provide accurate depth information with resolution of millimeters in indoor environments. However it may fail to match between measurements with large baseline, since the sensor rate usually is around tens of hertz. Robotic systems like drones could generate aggressive motion in real world, thus relying on single frame-based sensor may be vulnerable. To overcome this problem, we suggest to find relative motion from another sensor with higher frame, which is event camera. Our algorithm is to utilize the continuous measurement characteristics of event camera, to estimate the motion occurred between frames of the depth sensor.

From measurement obtained with depth camera, we may transform the pointcloud into event camera's frame, and project them into image plane. Then the depth image from event frame is obtained, we use using edge detectors for detecting geometrical discontinuities in the image. Although sensors are placed closely, there are occlusions and discontinuities due to occluded area should be ignored. For transforming depth measurements into event camera frame, we used extrinsic calibration provided from the dataset released.

Algorithm 1: Continuous polynomial spline trajectory estimation with depth image and events

Data: Single depth image, event stream
Result: 6 dof continuous trajectory

X_{depth} = Extracted edge cloud from depth image;
T_e = Tree structure built from events;

repeat
  build polynomial spline trajectory $P(v)$
  for *number of intervals* **do**
    X'_{depth} = transformed X_{depth} with $P(v)$
    I = projected points of X'_{depth} on event frame
    search nearest neighbour of I from T_e
    update loss (1)
  end
  update spline parameters v
until *Converge*;

After extracting geometric discontinuities from event frame with depth camera, we define a polynomial spline and recursively update basis parameters to minimize the loss. We defined loss as weighted sum of three losses : distance between geometric edge and incoming events, error of re-projected depth measurements and parameter regularization term.

$$\begin{aligned} L &= L_{struct} + L_{endpose} + L_{param} \quad (1) \\ L_{struct} &: \text{Structure matching loss} \\ L_{endpose} &: \text{End pose loss} \\ L_{param} &: \text{Regularization loss} \end{aligned}$$

Given spline parameter v with n variables, we construct polynomial trajectory with nth order basis function $P_n(t, v)$, where v is polynomial spline parameter, as

$$P_n(t, v) = \sum_{i=1}^{n} v_i t_i$$

where t is time between two depth poses. Note that there's no zeroth order term from we're solving relative pose problem, so $P(0, v) = 0$. Then with this 6dof polynomial curve, we may transform the geometrical edge point X_{depth} and find nearest neighbour from built tree structure of events T_e.

$$L_{struct} = \pi_0(P(t, v)X_{depth})$$

Then we may calculate structure matching loss from the matched distance. To avoid building too large tree, we divide the intervals by time and do the search only within interval. In the experiment, we used number of intervals significantly more than polynomial order.

The end pose error is obtained from transforming latter depth measurement into prior depth pointcloud by endpoint pose of polynomial.

$$L_{endpose} = X_{depth}(t) - P^{-1}(t, v) \cdot X_{depth}(t + 1)$$

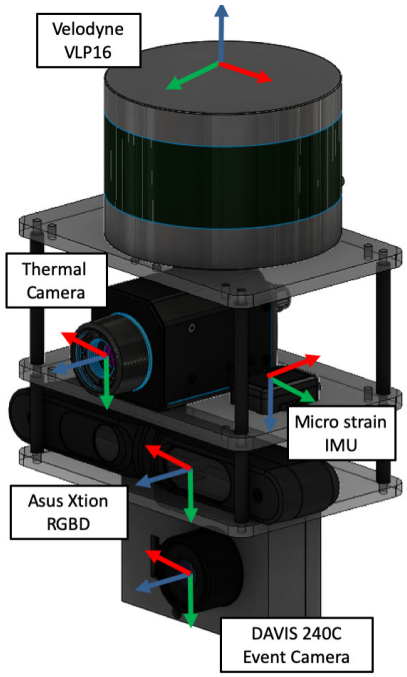
And as the last, we defined regularization loss of spline parameters to avoid overestimating the higher order parameter of the polynomial. Then we optimize the spline parameter v to minimize the loss, using LM method.

IV. Datasets

We tested our algorithm on Vision for Visibility Dataset (ViViD) which is released alongside this paper. In the dataset, variations in motion and lighting are included for testing the robustness of depth sensor and event camera.

A. Sensor configuration

To capture visual information even under various environments, three cameras were installed along with the inertial measurement unit as in Fig. 2. Through this sensor system, we aim to obtain visual information free from external lighting conditions. These unique sensor set collects data from infrared radiation, which is dependent on the temperature of



Sensors	Specifications
Thermal	FLIR A65, 640*542 pixel, 20Hz
RGB-D	Xtion Pro Live, 640*480 pixel, 30Hz
Event	DAVIS 240C, 240*180 pixel, up to 12MEPS
Inertial	On-Board of DAVIS240C : MPU6150 Microstrain IMU 3DM-GX5-25
LiDAR	Velodyne VLP-16

Fig. 2: Sensors hardware configuration.

the object, structured light depth measurement, and relative not absolute intensity changes upon time. The dataset is provided in binary format in rosbag. Note that in the thermal camera, the format of the image is not a typical 8-bit int but 14 bits enclosed in 16 bits.

For calibration, we used general checkerboard and April tags for finding extrinsic parameter between RGB-D and event camera and inertial measurement unit. Since the DAVIS camera also produces intensity-based image, its calibration procedure can be done for image and applied for events. However since the thermal camera only detects differences in temperature, it does not detect normal checkerboard pattern. We used heated printed circuit board (PCB) for calibrating thermal camera to the system.

B. Description of sequences

The sequence list is detailed in Table. I. The sequences are composed of two distinct locations (indoor / outdoor) and four different light conditions (normal / dark / dimmed / varying light) with motion variances for indoor sequences.

1) Environments: The first and second batches of the dataset are recorded in different locations. In indoor sequences, global poses of the platform are captured via motion capture system. The scale of the room is $12.3\,\mathrm{m} \times 8.9\,\mathrm{m} \times 4.5\,\mathrm{m}$ with 12 cameras mounted on the wall for motion capture. The system uses infrared strobes to track the reflected markers of desired platforms. The overview of indoor sequences is provided in Fig. 3.

TABLE I: Environment setting for each sequences

Sequence	Ambient Light	Additional Light	Motion	Pose GT
Indoor	Bright	OFF	Robust	Vicon
	Bright	OFF	Fast	Vicon
	Dark	OFF	Robust	Vicon
	Dark	OFF	Fast	Vicon
	Dark	ON	Robust	Vicon
	Dark	ON	Fast	Vicon
Outdoor	Bright	OFF	Robust	LOAM
	Dark	OFF	Robust	LOAM
	Dark	ON	Robust	LOAM

For outdoor sequences, a pose obtained with LeGO-LOAM [24] was used as a ground truth. The location of the outdoor trajectory is nearly enclosed by buildings, lowering the accuracy of global positioning system (GPS) rather appropriate for LiDAR-based algorithms. The total size of the enclosed region is about $60\,\mathrm{m} \times 40\,\mathrm{m}$, and the trajectory length is around $50\,\mathrm{m}$.

2) Illumination and Ego-motion Variance: In each batch, light and motion variance was applied to create disturbance. In the real world, robots experience both uncooperative luminance and abrupt terrain, which creates aggressive motion. The dataset consists of three sequences from normal, dark and changing illumination conditions. For indoor environments, rapid movements were recorded assuming drone tracking or hand-held scenarios.

V. Evaluation

We evaluate our algorithm along ViViD indoor sequences. Since we aim for robust trajectory estimation even under aggressive motion and dynamic lighting, evaluation sequences included the variances. We calculated absolute trajectory error for each sequences and compared against ORB-SLAM2 [3], using RGB-D configuration. The absolute trajectory error of sequences are listed in Table. II. ORB-SLAM2 shows desirable performance for robust environments, but fails on aggressive motion and dark scenes.

The DAVIS camera produces events from 240×180 pixel array, asynchronously at a timing when the luminescence value of the pixel changes. Therefore, it provides high dynamic range to cover large range of lighting conditions. Since ORB-SLAM2 is based on image features, it ORB-SLAM2 fails to complete estimating the whole trajectory except robust motion and modest lighting conditions. However, our sensor configuration does not fails on given sequences.

TABLE II: Absolute Trajectory Error for ViViD sequences

Sequence	ORB2	Ours
Robust_Global	0.0726	0.1024
Unstable_Global	0.1625	0.3748
Aggressive_Global	Track lost	0.3781
Robust_Dark	Init fail	0.2563
Unstable_Dark	Init fail	0.4041
Aggressive_Dark	Init fail	0.6108

Manuscript 2190 submitted to 2020 IEEE/RSJ International Conference
on Intelligent Robots and Systems. Received March 1, 2020.

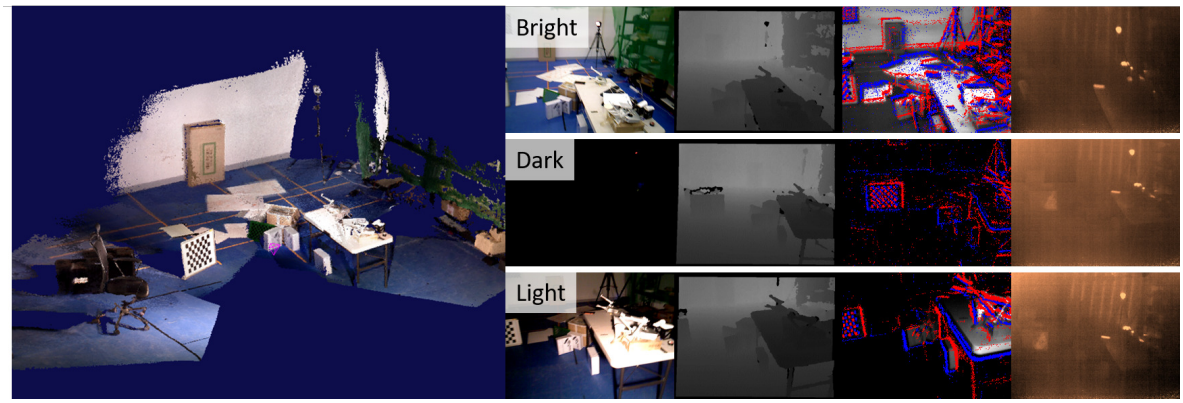


Fig. 3: Scene overview of Vision for Visibility Dataset (left), with data sample from each sensors (right). Images acquired from each sensors are illustrated for each lighting conditions, listed left from RGB, depth, events, thermal camera.

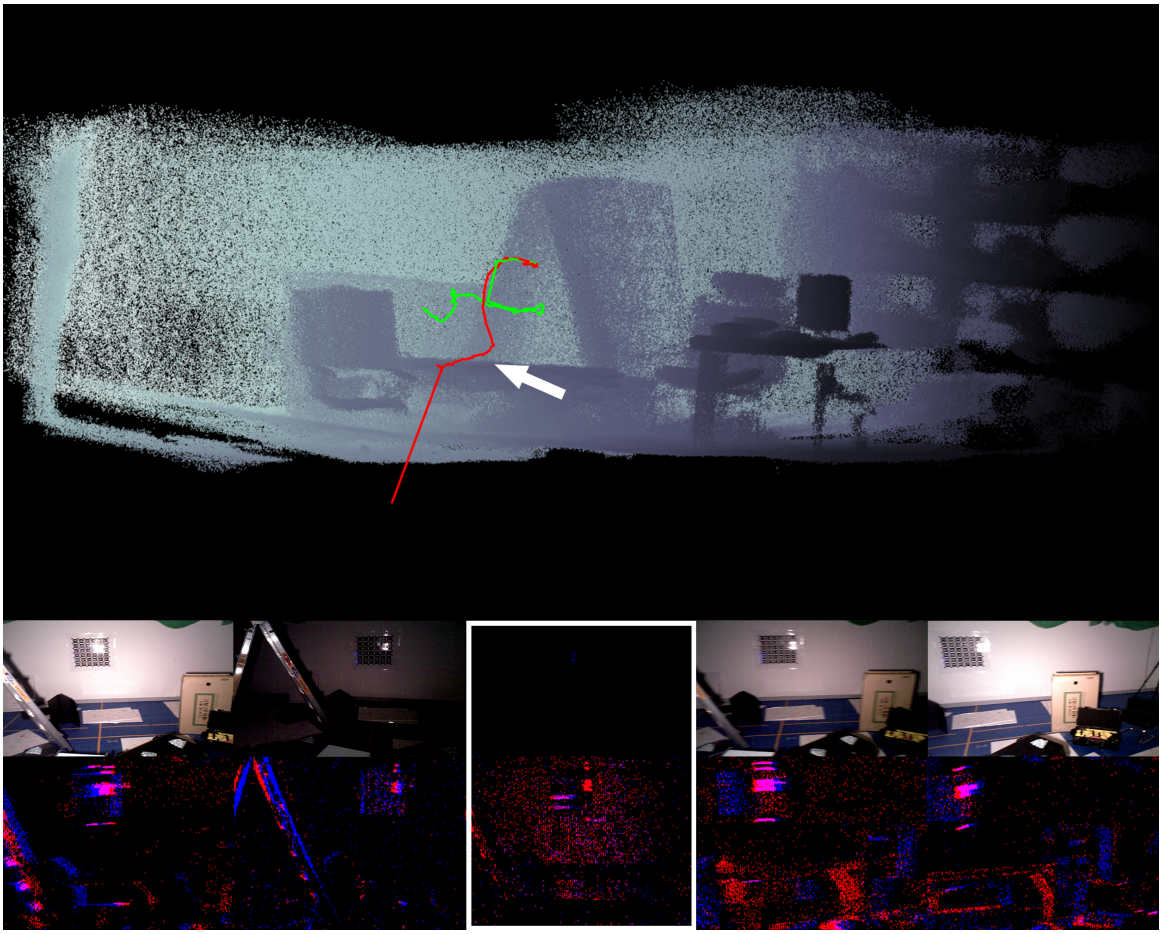


Fig. 4: Estimated trajectory and reconstructed pointcloud for vivid dataset, for varying light sequence. As there is lighting change over time, image feature based ORB-SLAM2 (red path) fails to estimate trajectory when the light turns off. However, Ours (green path) succeeds on estimating the trajectory robustly under circumstances.

Since the resolution of DAVIS camera is relatively lower than other sensors, the output of our algorithm does not gives better results to ORB-SLAM2, on stable environments. In global lighting sequences the room was sufficiently bright with constant ambient light. And for the dark and local light sequences, we ran our experiment with the ambient light turned off. In local light sequences, only an LED installed with sensor rig was turned on, producing unequally configured lighting profile for the scene. As in Fig. 4 local light source changes its brightness, and RGB-based features fail easily, yielding tracking lost.

VI. Conclusion

We have introduced a method to solve pose estimation problem by introducing continuous trajectory fused with event camera, for robust convergence of slow and accurate depth sensors. Our algorithm solves environment undersampling problem by robustly updating time-continuous trajectory parameters to estimate poses between sparsely updated sensor measurements. We also summarized results by showing robustness under lighting and motion variances on publicly released dataset provided with calibration parameters and groundtruth. Our vision for visibility dataset is the first public dataset to include three calibrated multi domain vision sensors, to provide the standard to develop vision for visibility under real-world environments.

References

[1] P. Lichtsteiner, C. Posch, and T. Delbruck, "A 128x128 120 db 15 mu s latency asynchronous temporal contrast vision sensor," *IEEE journal of solid-state circuits*, vol. 43, no. 2, pp. 566–576, 2008.

[2] C. Forster, M. Pizzoli, and D. Scaramuzza, "Svo: Fast semi-direct monocular visual odometry," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014, pp. 15–22.

[3] R. Mur-Artal and J. D. Tardós, "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

[4] X. Lagorce, C. Meyer, S.-H. Ieng, D. Filliat, and R. Benosman, "Asynchronous event-based multikernel algorithm for high-speed visual features tracking," *IEEE transactions on neural networks and learning systems*, vol. 26, no. 8, pp. 1710–1720, 2015.

[5] X. Clady, S.-H. Ieng, and R. Benosman, "Asynchronous event-based corner detection and matching," *Neural Networks*, vol. 66, pp. 91–106, 2015.

[6] D. Weikersdorfer and J. Conradt, "Event-based particle filtering for robot self-localization," in *Robotics and Biomimetics (ROBIO)*. IEEE, 2012, pp. 866–870.

[7] H. Kim, S. Leutenegger, and A. J. Davison, "Real-time 3d reconstruction and 6-dof tracking with an event camera," in *ECCV*. Springer, 2016, pp. 349–364.

[8] H. Rebecq, T. Horstschaefer, G. Gallego, and D. Scaramuzza, "Evo: A geometric approach to event-based 6-dof parallel tracking and mapping in real time," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 593–600, 2017.

[9] A. R. Vidal, H. Rebecq, T. Horstschaefer, and D. Scaramuzza, "Ultimate slam? combining events, images, and imu for robust visual slam in hdr and high-speed scenarios," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 994–1001, 2018.

[10] H. Kim, A. Handa, R. Benosman, S.-H. Ieng, and A. J. Davison, "Simultaneous mosaicing and tracking with an event camera," 2008.

[11] D. Weikersdorfer, R. Hoffmann, and J. Conradt, "Simultaneous localization and mapping for event-based vision systems," in *International Conference on Computer Vision Systems*. Springer, 2013, pp. 133–142.

[12] E. H. Adelson and J. R. Bergen, "Spatiotemporal energy models for the perception of motion," *Josa a*, vol. 2, no. 2, pp. 284–299, 1985.

[13] E. Mueggler, C. Forster, N. Baumli, G. Gallego, and D. Scaramuzza, "Lifetime estimation of events from dynamic vision sensors," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*. IEEE, 2015, pp. 4874–4881.

[14] S. H. Ieng, J. Carneiro, M. Osswald, and R. B. Benosman, "Neuromorphic event-based generalized time-based stereovision," *Frontiers in neuroscience*, vol. 12, p. 442, 2018.

[15] G. Gallego and D. Scaramuzza, "Accurate angular velocity estimation with an event camera," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 632–639, 2017.

[16] H. Rebecq, T. Horstschaefer, and D. Scaramuzza, "Real-time visual-inertial odometry for event cameras using keyframe-based nonlinear optimization," in *British Machine Vis. Conf.(BMVC)*, vol. 3, 2017.

[17] E. Mueggler, G. Gallego, H. Rebecq, and D. Scaramuzza, "Continuous-time visual-inertial odometry for event cameras," *IEEE Transactions on Robotics*, no. 99, pp. 1–16, 2018.

[18] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Conf. on Comput. Vision and Pattern Recog.*, 2016, pp. 3213–3223.

[19] J. Jeong, Y. Cho, Y.-S. Shin, H. Roh, and A. Kim, "Complex urban dataset with multi-level sensors from highly diverse urban environments," in *The International Journal of Robotics Research*, 2019.

[20] N. Carlevaris-Bianco, A. K. Ushani, and R. M. Eustice, "University of Michigan North Campus long-term vision and lidar dataset," *International Journal of Robotics Research*, vol. 35, no. 9, pp. 1023–1035, 2015.

[21] J. Engel, V. Usenko, and D. Cremers, "A photometrically calibrated benchmark for monocular visual odometry," *arXiv preprint arXiv:1607.02555*, 2016.

[22] Y. Choi, N. Kim, S. Hwang, K. Park, J. S. Yoon, K. An, and I. S. Kweon, "KAIST multi-spectral day/night data set for autonomous and assisted driving," *IEEE Trans. Intell. Transport. Sys.*, vol. 19, no. 3, pp. 934–948, 2018.

[23] A. Z. Zhu, D. Thakur, T. Özaslan, B. Pfrommer, V. Kumar, and K. Daniilidis, "The multivehicle stereo event camera dataset: An event camera dataset for 3d perception," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2032–2039, 2018.

[24] T. Shan and B. Englot, "Lego-loam: Lightweight and ground-optimized lidar odometry and mapping on variable terrain," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 4758–4765.