2020 IEEE International Conference on Robotics and Automation (ICRA)
31 May - 31 August, 2020. Paris, France

Constrained Filtering-based Fusion of Images, Events, and Inertial Measurements for Pose Estimation

Jae Hyung Jung and Chan Gook Park

***Abstract*— In this paper, we propose a novel filtering-based method that fuses events from a dynamic vision sensor (DVS), images, and inertial measurements to estimate camera poses. A DVS is a bio-inspired sensor that generates events triggered by brightness changes. It can cover the drawbacks of a conventional camera by virtual of its independent pixels and high dynamic range. Specifically, we focus on optical flow obtained from both a stream of events and intensity images in which the former is much like a *differential* quantity, whereas the latter is a pixel *difference* in a much longer time interval than events. This nature characteristic motivates us to model optical flow estimated from events directly, but feature tracks for images in the filter design. An inequality constraint is considered in our method since the inverse scene-depth is larger than zero by its definition. Furthermore, we evaluate our proposed method in the benchmark DVS dataset and a dataset collected by the authors. The results reveal that the presented algorithm has reduced the position error by 49.9% on average and comparable accuracy only using events when compared to the state-of-the-art filtering-based estimator.**

I. INTRODUCTION

An event camera or dynamic vision sensor (DVS) is a bio-inspired sensor that outputs asynchronous *events* generated by brightness changes rather than absolute intensities of each pixel. It addresses limitations of a conventional camera featuring a very high dynamic range (130 dB), and more importantly independent pixels that avoid the motion blur due to a rapid ego-motion [1], [2]. A DVS opens a new way to access visual information previously only perceived as an array filled with the absolute amount of light in most of computer vision algorithms. By virtue of these advantages, events could be an attractive solution in target tracking or robotics in which a high dynamic motion frequently arises. However, events from a monocular system still cannot resolve the scale ambiguity.

Visual-inertial fusion is a popular solution because of its well-known complementary characteristics: an image provides rich information for localization, while inertial measurements from an inertial measurement unit (IMU) fill the gap between images that has typically much lower sampling rates. Rapid motion and high dynamic range scenarios are still challenging that obscure visual measurement. Consequently, biases of an IMU are not properly estimated, and then a navigation solution might diverge eventually. A device that outputs intensity for a pixel along with an event from the same photodiode was introduced in [2]. This work foresaw the combination of the standard frame and events would serve as an attractive sensor: standard image provides intensity information even in static, while events are well-triggered in a rapid motion.

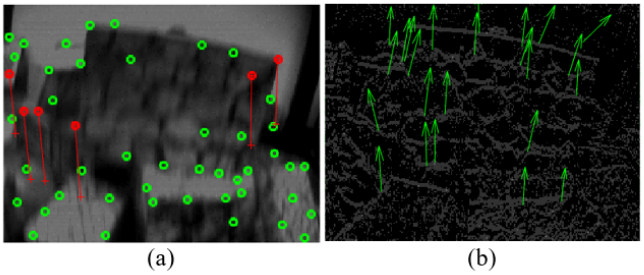

Fig. 1. A sample image (dataset from [25]) during significant angular motion: (a) An intensity image with tracked features (red) and failed tracks (green) during 47ms. Many of them are failed to track due to the motion blur. (b) A synthesized event frame with optical flow measurements (green arrows) with a temporal window 3ms

In our approach, to reliably recover pose even in high dynamics, we focus on optical flow measurements that can be estimated from both an intensity image and an event stream. Optical flow from an image is much like a pixel *difference* among a two-view geometry divided by the time interval, whereas optical flow obtained from an event stream is viewed as a *differential* quantity made from a spatiotemporal window usually much shorter than the sampling frequency of an intensity image in a rapid motion scenario. This motivates us to utilize the optical flow measurements from an event stream directly, but feature tracks for images.

An earlier work of [13] recognized the complementary property of events and images and proposed a method to fuse a visual-inertial system with a stream of events. However, our approach is basically different to the previous work other than the back-end implementation. Specifically, we directly utilize optical flow obtained from a stream of events, whereas the authors synthesized a motion-corrected event frame to obtain feature tracks extracted from sharp frames. Therefore, our method does not require any time synchronization between event and intensity frames: optical flow from events updates the estimator asynchronously as its inherent characteristic.

Fig. 1 shows an example of estimated optical flow measurements from images and a stream of events in which events provide body velocity information from the optical

This work was supported by Samsung Electronics Co.,Ltd. through the Samsung Smart Campus Research Center (0115-20190032) and the Ministry of Science and ICT of the Republic of Korea through the Space Core Technology Development Program (NRF-2018M1A3A3A02065722)

All authors are with the Department of Mechanical and Aerospace Engineering, Automation and Systems Research Institute, Seoul National University, Seoul 08826, Republic of Korea. lastflowers@snu.ac.kr; chanpark@snu.ac.kr

978-1-7281-7395-5/20/$31.00 ©2020 IEEE

flow even in the challenging condition. Moreover, optical flow is much less sensitive to outliers than the multi-view track since it only depends on a two-view measurement. At the best of the authors' knowledge, it is the first time to fuse optical flow from events, feature tracks from images, and inertial measurements in a framework of a filtering-based estimator. The main contribution of our novel method is threefold:

- Filtering-based hybrid estimator using IMU and optical flow computed from events and images, respectively.
- Inequality constraint on the inverse scene-depth is considered in the filtering to more precisely model optical flow.
- The proposed method is evaluated on the event camera dataset [25] quantitatively and a dataset collected by the authors qualitatively.

II. Related Works

Since a DVS conveys different types of output in nature, conventional computer vision techniques should be modified, or a new algorithm has to be designed.

Benosman et al. [6] proposed a method that estimates optical flow of each event by fitting a plane to the surface of active events that is $(x, y) \mapsto t$ where x, y is an event location and t is a timestamp. The optical flow is expressed as the normal vector of the fitted plane. By leveraging [6], Muggler et al. [7] devised the lifetime (time taken to move one pixel) of an event through which one can take a sharp snapshot without accumulating an artificial temporal window. Gallego et al. [8] presented a general framework called contrast maximization to recover ego-motion, depth and, optical flow that are one of the most crucial model parameters in the multiple view geometry. This approach was further analyzed by [9], [10] that presented objective functions in estimating optical flow in which the former more focused on the aperture problem. Zhu et al. [18] presented an expectation-maximization (EM) based algorithm that solves a soft data-association problem to estimate optical flow inspired by EM-ICP [20]. Gehrig et al. [11] posed a maximum likelihood approach for aligning intensity images with events. This corresponds to an image registration problem in which the image acts as a template as similar to the KLT tracker [19].

The back-end of a visual-inertial navigation system can be implemented either based on optimization [3], [4] or filtering method [5], [22]. For an event-based visual-inertial system, [12], [13], [14] presented an optimization-based estimator using a DVS. Rebecq et al. [12] synthesized a motion-corrected event frame and used a conventional vision technique to track a set of features, and Vidal et al. [13] also fused images along with events and inertial measurements based on [12]. However, the work of [13] set the fixed number of event windows that should be appropriately tuned according to the working environment to synchronize event frames to standard frames forcibly. In contrast, our method does not require any synchronizations of events and images since we directly extract the velocity component from optical flow measurements of events.

On the other hand, a filtering-based estimator was proposed in [21] using their previous work on estimating optical flow [18], and the multi-state constraint Kalman filter (MSCKF) [22]. However, they reported that their method could not run in real-time due to the heavy computation of the two EM algorithms. Since our method only considers a two-view geometry of a feature from events, we only utilize the first part of the EM algorithm of [18] to reduce computation while extracting meaningful information.

In a specific application, a state can only be laid in the feasible region due to its definition or a physical constraint. Especially in chemical engineering in which concentration should be larger than or equal to zero, linear and nonlinear constrained filtering has been actively researched [15], [16], [17]. Since one of our state variables, the inverse scene-depth should be larger or equal to zero, we formulated our EKF-based estimator in a framework of the projection approach [24] in the constrained filtering.

III. Notations

In this paper, we denote the global frame as $\{G\}$ that is a local tangent frame aligned with the gravity direction. The body frame $\{B\}$ is coincident with axes of an IMU, while the intensity camera and the event camera frames are designated as $\{C\}$ and $\{E\}$, respectively. Their origins are at the optical centers of each camera, and the XYZ axes follow the right-down-forward convention. A vector and matrix are designated by a small and capital bold letter such as $\mathbf{a}$ and $\mathbf{A}$. A vector is described by a reference, resolved, and object frame. $\mathbf{x}_B^G$ expresses a physical quantity, $\mathbf{x}$ in which the reference/resolved frames are $\{G\}$, and the object frame is $\{B\}$. When a reference and resolved frames are different, we will explicitly mention them in the text. A matrix, $\mathbf{R}_B^G \in \mathbb{SO}(3)$ stands for the direction cosine matrix that transforms a resolved frame from B to G. We denote the corresponding unit quaternion as $\mathbf{q}_{GB}$. For a random vector $\mathbf{x}$, we define the error vector as $\tilde{\mathbf{x}} = \mathbf{x} - \hat{\mathbf{x}}$ where the hat means estimated value.

IV. The proposed method

Our method fuses measurements from a conventional, event camera, and an IMU to track a pose of the body frame $\{B\}$ with respect to the global frame $\{G\}$. The overall flowchart is shown in Fig. 2. Feature points from images are tracked by Kanade-Lucas-Tomasi (KLT) tracker [19] to give a relative constraint to the estimator. Events are tracked by solving a soft data-association problem that was originally proposed in [18]. Since we are only interested in instance optical flow of given features, and estimator does not suffer from a tracking drift, the first part of the EM algorithm of [18] while excluding the affine alignment is employed to estimate the flow. Then, the constrained EKF fuses the likelihood and prior in which the inverse scene-depth (average depth of a scene) is projected into the constrained surface when the constraints are violated.

The below subsections will review the event tracker, and describe the proposed filer in detail.

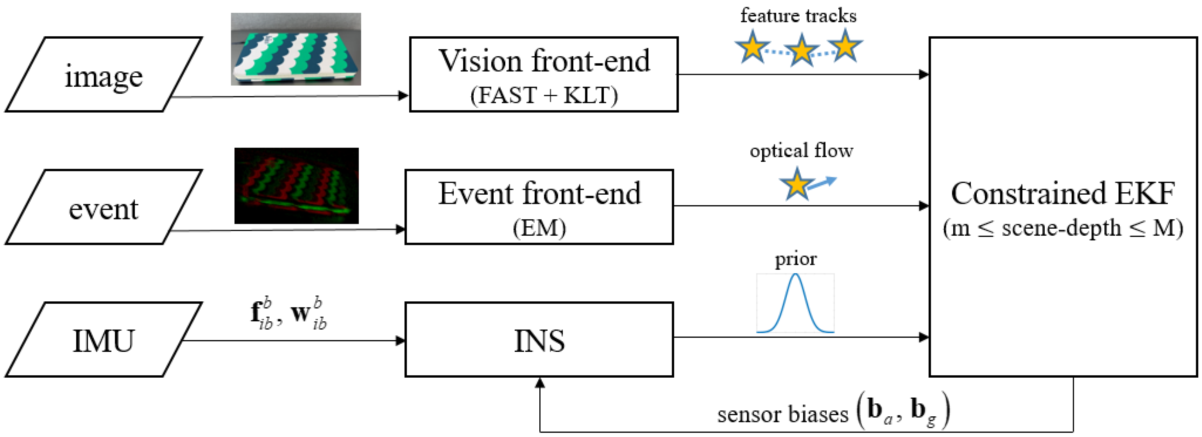

Fig. 2. A flowchart of the proposed algorithm. Intensity images are processed in Vision front-end using a conventional image processing technique to yield feature tracks, while optical flow measurements of events are solved through the expectation-maximization (EM) algorithm in Event front-end. The estimator constraints a range of the scene-depth from prior knowledge of an environment.

A. The event tracker

Events are generated from an edge with an intensity change. It is dissimilar to the conventional images in nature, therefore to extract information for localization, a different method should be employed. An i-th event is a 4-length vector that is,

$$\mathbf{e}_i = \{t_i, \boldsymbol{\xi}_i, p_i\} \tag{1}$$

where t_i is a timestamp, $\boldsymbol{\xi}_i$ is a pixel coordinate in a 2D image plane, and p_i stands for the polarity of an event (-1 for negative, +1 for positive).

By assuming that optical flow $\mathbf{u}$ within a spatiotemporal window is constant, we can estimate it by solving the below minimization problem [18], [21].

$$\hat{\mathbf{u}} = \operatorname*{argmin}_{\mathbf{u}} \sum_{\mathbf{e}_i \in W} \sum_{\mathbf{l}_j \in T} w_{ij} \left\| (\boldsymbol{\xi}_i - \Delta t_i \mathbf{u}) - \mathbf{l}_j \right\|_2^2 \tag{2}$$

In this expression, Δt_i is a time interval between reference time of W and i-th event time. w_{ij} is a probability that i-th event was generated from the j-th template $\mathbf{l}_j$, hence a soft data-association. Also, W is a spatiotemporal window, that is defined as

$$W = \left\{ \mathbf{e}_i \mid \left\| (\boldsymbol{\xi}_i - \Delta t_i \mathbf{u}) - \mathbf{f} \right\|_2^2 \leq \epsilon,\ t_0 \leq t < t_1 \right\} \tag{3}$$

where ϵ and $\mathbf{f}$ are user-defined spatial window size and pixel coordinate of a tracked feature, respectively. Also, t_0 and t_1 are time window. Geometrically, it is a set of events that falls within a circle of radius ϵ given optical flow. Similarly, the template window T is defined as

$$T = \left\{ \mathbf{l}_j \mid \left\| \mathbf{l}_j - \mathbf{f} \right\|_2^2 \leq \epsilon \right\} \tag{4}$$

that is a set of template features that lie within the circle.

Unfortunately, w_{ij} in (2) depends on the parameter $\mathbf{u}$, and can be evaluated up to the expectation. The expectation step estimates w_{ij}, and the maximization step estimates $\mathbf{u}$ by maximizing the likelihood. We model $\boldsymbol{\xi}_i$ generated from $\mathbf{l}_j$ as a Gaussian distribution where its mean is $\mathbf{l}_j$ with a standard deviation as 2 pixels, and set the template as time-shifted events from a previous step as in [18]. In particular, for the firstly tracked feature the template $\{\mathbf{l}_j\}_0$ (subscript means the time step) is taken as events back-propagated by the currently estimated optical flow $\{\mathbf{l}_j\}_0 = \{\boldsymbol{\xi}_i - \Delta t_i \mathbf{u}\}$ where $\boldsymbol{\xi} \in \mathbf{e}_i$. Then for the next epoch, the template is set as the forward-propagated events given the optical flow, that is $\{\mathbf{l}_j\}_1 = \{\boldsymbol{\xi}_i + (\Delta t_e - \Delta t_i)\hat{\mathbf{u}}_0\}$ where $\Delta t_e = t_1 - t_0$.

B. The Filter design

We see that optical flow from a conventional image is much like *difference* of pixel positions in a finite camera sampling time resulted from an image alignment. On the other hand, optical flow obtained from events represents a *differential* element that is estimated in a much shorter time than the former. This motivated us to fuse measurements from two complementary sources using previous works of [22], [23].

The error state vector consists of 15 states of an inertial navigation system, an inverse scene-depth, and sliding windows, that is

$$\begin{aligned} \tilde{\mathbf{x}} &= \begin{bmatrix} \tilde{\mathbf{x}}_I^{\mathrm{T}} & \tilde{\lambda} & \tilde{\mathbf{x}}_S^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} \\ \tilde{\mathbf{x}}_I &= \begin{bmatrix} \tilde{\boldsymbol{\theta}}_{GB}^{\mathrm{T}} & \tilde{\mathbf{p}}_B^{G,\mathrm{T}} & \tilde{\mathbf{v}}_B^{\mathrm{T}} & \tilde{\mathbf{b}}_a^{\mathrm{T}} & \tilde{\mathbf{b}}_g^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} \\ \tilde{\mathbf{x}}_{S_i} &= \begin{bmatrix} \tilde{\boldsymbol{\theta}}_{GB_i}^{\mathrm{T}} & \tilde{\mathbf{p}}_{B_i}^{G,\mathrm{T}} \end{bmatrix} \end{aligned} \tag{5}$$

States in $\tilde{\mathbf{x}}_I$ are the attitude, position, velocity, and biases of an accelerometer and a gyroscope in turn. We define the attitude error that is perturbed from the left side as $\mathbf{R}_B^G \approx \{\mathbf{I}_3 + [\tilde{\boldsymbol{\theta}}_{GB}]_\times\}\hat{\mathbf{R}}_B^G$ where $[\cdot]_\times$ is a skew-symmetric matrix operator for a given vector. Also, the body velocity denoted by $\mathbf{v}_B$ is equal to $\mathbf{R}_G^B \mathbf{v}_B^G$. While sliding windows $\tilde{\mathbf{x}}_S$ are directly related to the feature tracks from the vision front-end, the inverse scene-depth λ that is an inverse of the average feature depth models the optical flow from events.

The nominal states are numerically integrated that provide linearization points for the estimator, whereas error states of the attitude, position, and velocity are governed by following 1st Markov processes,

$$\begin{aligned}\dot{\tilde{\boldsymbol{\theta}}}_{GB} &= -\hat{\mathbf{R}}_B^G\tilde{\boldsymbol{\theta}}_{GB} - \hat{\mathbf{R}}_B^G\mathbf{n}_g \\ \dot{\tilde{\mathbf{p}}}_B^G &= -[\hat{\mathbf{R}}_B^G\hat{\mathbf{v}}_B]_\times\tilde{\boldsymbol{\theta}}_{GB} + \hat{\mathbf{R}}_B^G\tilde{\mathbf{v}}_B \\ \dot{\tilde{\mathbf{v}}}_B &= \hat{\mathbf{R}}_G^B[\mathbf{g}^G]_\times\tilde{\boldsymbol{\theta}}_{GB} - [\mathbf{w}_m - \hat{\mathbf{b}}_g]_\times\tilde{\mathbf{v}}_B \\ &\quad - (\tilde{\mathbf{b}}_a + \mathbf{n}_a) - [\hat{\mathbf{v}}_B]_\times(\tilde{\mathbf{b}}_g + \mathbf{n}_g)\end{aligned} \tag{6}$$

where $\mathbf{n}_a$ and $\mathbf{n}_g$ are white zero-mean Gaussian processes for an accelerometer and a gyroscope, respectively. $\mathbf{g}^G$ is the gravity vector expressed in $\{G\}$, and $\mathbf{w}_m$ is an angular velocity measured from a gyroscope. In addition to (6), biases and the inverse scene-depth are modeled as random walks.

The measurement model for a j-th feature track is given by a pinhole projection $\boldsymbol{\Pi}$ with an additive white zero-mean Gaussian noise $\mathbf{n}_v$ that is uncorrelated to the system model noise. Assuming that intrinsic parameters are calibrated, the pinhole model is

$$\begin{aligned}\mathbf{z}_j &= \boldsymbol{\Pi}(\mathbf{p}_{f_j}^C) + \mathbf{n}_v \\ &= \lambda_j\mathbf{p}_{f_j}^C + \mathbf{n}_v\end{aligned} \tag{7}$$

where λ_j is the j-th inverse depth. The error equation of (7) is projected into the nullspace of the feature-related Jacobian matrix to eliminate feature states [22]. We solve for the feature depth by using inverse parameterization with the Levenberg-Marquardt algorithm.

Taking a time derivative for both sides of (7), an i-th optical flow model estimated from the event front-end is

$$\begin{aligned}\dot{\mathbf{z}}_i &= \dot{\lambda}_i\mathbf{p}_{f_i}^E + \lambda_i\dot{\mathbf{p}}_{f_i}^E \\ &= \frac{\dot{\lambda}_i}{\lambda_i}(\lambda_i\mathbf{p}_{f_i}^E) + [\mathbf{w}_G^E]_\times(\lambda_i\mathbf{p}_{f_i}^E) - \lambda_i\mathbf{v}_{GE}^E\end{aligned} \tag{8}$$

We denote the normalized feature point as $\bar{\mathbf{p}}_{f_i}^E = \lambda_i\mathbf{p}_{f_i}^E$ in the following. To eliminate the dependence on the time derivative of the inverse depth, (8) is projected to the left null space of $\bar{\mathbf{p}}_{f_i}^E$ [23].

$$\begin{gathered}\mathbf{N}_i\left(\dot{\mathbf{z}}_i + [\mathbf{w}_{GE}^E]_\times\bar{\mathbf{p}}_{f_i}^E + \lambda_i\mathbf{v}_{GE}^E\right) = 0 \\ \mathbf{N}_i\bar{\mathbf{p}}_{f_i}^E = \mathbf{0}_{2\times3},\ \mathbf{N}_i\mathbf{N}_i^{\mathrm{T}} = \mathbf{I}_2\end{gathered} \tag{9}$$

Note that $\mathbf{N}_i$ is the left nullspace of $\bar{\mathbf{p}}_{f_i}^E$ that has orthnormal bases. We express (9) with respect to $\{B\}$ and define this measurement as $\mathbf{y}$, that is

$$\mathbf{y}_i = \mathbf{N}_i\left\{\dot{\mathbf{z}}_i + [\mathbf{R}_B^E\mathbf{w}_B]_\times\bar{\mathbf{p}}_{f_i}^E + \lambda_i\mathbf{R}_B^E(\mathbf{v}_B + [\mathbf{w}_B]_\times)\mathbf{p}_E^B\right\} \tag{10}$$

where $\mathbf{w}_B$ is an angular rate of $\{B\}$, and $\mathbf{R}_B^E$ and $\mathbf{p}_E^B$ are extrinsic parameters between $\{B\}$ and $\{E\}$. Ideally, (10) is equal to zero without any noises. Perturbing (10) by the error state up to 1st order yields

$$\begin{aligned}\tilde{\mathbf{y}}_i = \mathbf{N}_i\{&\hat{\lambda}\mathbf{R}_B^E\tilde{\mathbf{v}}_B + ([\bar{\mathbf{p}}_{f_i}^E]_\times\mathbf{R}_B^E + \hat{\lambda}\mathbf{R}_B^E[\mathbf{p}_E^B]_\times)(\tilde{\mathbf{b}}_g + \mathbf{n}_g) \\ &+\mathbf{R}_B^E(\hat{\mathbf{v}}_B + [\mathbf{w}_m - \hat{\mathbf{b}}_g]_\times\mathbf{p}_E^B)(\tilde{\lambda} + \mathbf{n}_{\lambda_i}) + \mathbf{n}_{\dot{z}_i}\}\end{aligned} \tag{11}$$

In this expression, we formulate the i-th inverse-depth as $\lambda_i = \hat{\lambda} + \tilde{\lambda} + \mathbf{n}_{\lambda_i}$ where $\mathbf{n}_{\lambda_i} \sim N(0, \sigma_i^2)$, i.e. this requires only 1 dimensional state $\tilde{\lambda}$. Also, $\mathbf{n}_{\dot{z}_i}$ stands for a measurement noise of optical flow obtained from the event front-end. Note that we assume that the extrinsic parameter is calibrated in advance. According to (11), the projected error equation reveals that the body velocity, gyro bias and scene-depth are directly related to the measurement.

C. Constrained Kalman Filter

It is obviously seen that the inverse scene-depth λ is larger than 0 by its definition. Furthermore, we can reasonably assign the minimum and maximum range of the inverse scene-depth m and M depending on prior knowledge of an environment. Therefore, this is resolved to the quadratic programming (QP) with an inequality constraint.

$$\begin{gathered}\Delta\mathbf{x}^* = \underset{\Delta\mathbf{x}}{\operatorname{argmin}}(\Delta\mathbf{x} - \Delta\hat{\mathbf{x}})^{\mathrm{T}}\mathbf{P}^{-1}(\Delta\mathbf{x} - \Delta\hat{\mathbf{x}}), \\ \text{subject to } m - \hat{\lambda} \leq \mathbf{d}^{\mathrm{T}}\Delta\mathbf{x} \leq M - \hat{\lambda}\end{gathered} \tag{12}$$

Here, $\Delta\hat{\mathbf{x}}$ is the estimated error such that $\Delta\hat{\mathbf{x}} = \mathbf{K}\mathbf{r}$ in which $\mathbf{K}$ and $\mathbf{r}$ are Kalman gain and the innovation of the filter. $\mathbf{P}$ is the covariance matrix of the EKF. In our setting, $\mathbf{d}$ is given as a column vector in which its elements are all zero except for the inverse scene-depth,

$$\mathbf{d} = \begin{bmatrix}0 \cdots 1 \cdots 0\end{bmatrix}^{\mathrm{T}} \tag{13}$$

The solution of (12) is the minimum variance, that is $\operatorname{cov}(\tilde{\mathbf{x}} - \Delta\mathbf{x}^*) \leq \operatorname{cov}(\tilde{\mathbf{x}} - \Delta\mathbf{x})$ [24]. The QP with the scalar constraint of (12) can be easily solved. If the unconstrained solution satisfies the inequality, then finished. Otherwise, the QP with the equality constraint gives the solution. This is equivalent to project the filter state onto the constrained surface whenever the condition is violated.

V. Experiments

In this section, we validate our proposed pose estimator in two datasets: the event-camera dataset [25] and the author-collected dataset. The goal of this section is to analyze the accuracy of the estimator, either using events, images, and both of them in the event-camera dataset. Moreover, we compare our method to EVIO [21] that is a filtering-based estimator as ours quantitatively. Also, we show the qualitative result at the author-collected dataset that exhibits a rapid motion.

A. The event-camera dataset

The event-camera dataset [25] provides 6-DOF IMU readings, intensity images, and events captured by DAVIS [2] along with the ground-truth from a motion-capture system. Specifically, the spatial resolution of the image sensor is 240×180 that outputs events up to 12 Meps, while the IMU readings and images are given at 1000 Hz and around 24 fps, respectively. It offers trajectories of rotating, translating, and both maneuvers in indoor and outdoor environments. Among several sequences, we choose *translation* and *6dof* sets. This is because a feature parallax generated from a pure rotation

TABLE I

MEAN POSITION AND ATTITUDE ERROR IN THE EVENT-CAMERA DATASET

	Image + Event		*Event*		*Image*		EVIO [21]	
	Position [%]	Attitude [deg/m]	Position [%]	Attitude [deg/m]	Position [%]	Attitude [deg/m]	Position [%]	Attitude [deg/m]
boxes_6dof	**0.98**	**0.024**	2.88	0.063	1.07	0.031	3.61	0.34
boxes_translation	**1.24**	0.077	1.50	0.059	6.17	**0.054**	2.69	0.09
hdr_boxes	1.15	0.081	2.45	0.083	**0.90**	0.082	1.23	**0.05**
hdr_poster	**0.57**	0.160	2.38	0.104	0.84	**0.046**	2.63	0.11
poster_6dof	**0.91**	0.183	2.53	**0.099**	1.03	0.272	3.56	0.56
poster_translation	1.83	0.314	3.43	0.122	**0.67**	0.462	0.94	**0.02**
shapes_6dof	**0.59**	0.295	4.91	0.267	2.39	**0.192**	2.69	0.40
shapes_translation	**0.84**	**0.258**	5.25	0.567	1.02	0.765	2.42	0.52
dynamic_6dof	0.98	0.162	6.23	0.234	**0.79**	**0.155**	4.07	0.56
dynamic_translation	0.89	0.149	4.92	0.170	**0.48**	0.160	1.90	**0.02**

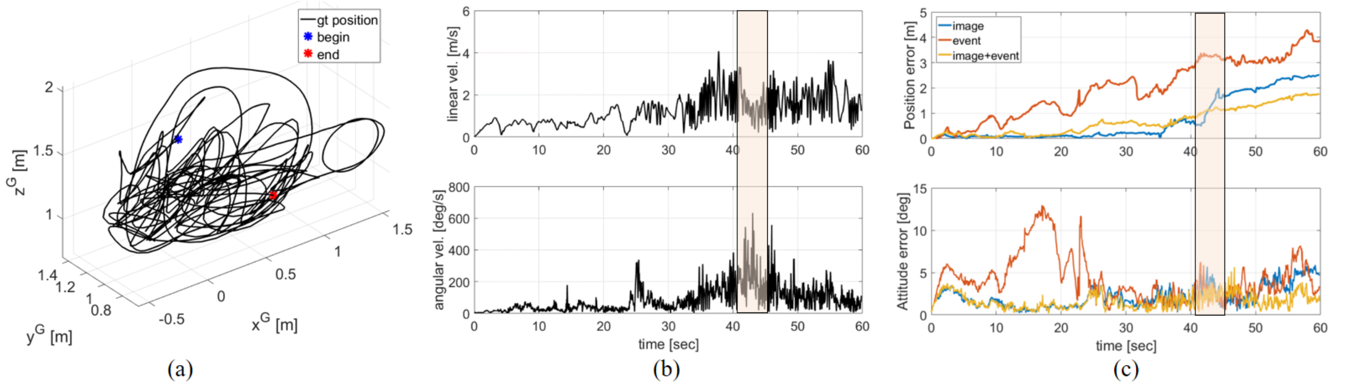

Fig. 3. Representative results in *boxes_6dof*. (a) True 3D position obtained from the motion-capture system, (b) Linear and angular velocity profile computed from numerical differentiation of the position and the gyroscope respectively, (c) Pose errors for 3 cases where vertical bars indicate the rapid rotation around 42 sec.

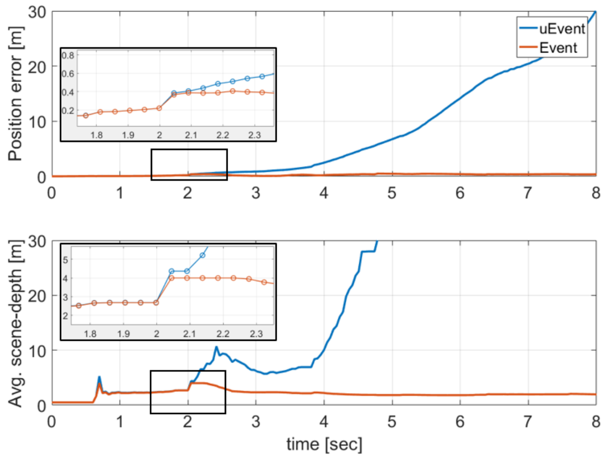

Fig. 4. Position error and estimated average scene-depth $1/\hat{\lambda}$ in the first 8 seconds of *boxes_translation* using only events with the constrained (Event) and unconstrained (uEvent) filtering.

does not convey any information for the reconstruction in a monocular vision setting.

We implement the proposed algorithm in MATLAB. For an event tracker, we process events through the open-source MATLAB script of [21] using only 1st part of the EM algorithm. In particular, Harris corners are detected in a synthesized event frame, namely $\mathbf{I}(\boldsymbol{\xi}) = \sum_i \delta(\boldsymbol{\xi} - \boldsymbol{\xi}_i)$. Throughout all experiments, we set the maximum number of event feature as 30, the upper limit number of the spatiotemporal window as 30,000 events and the next temporal size as 3 times 65% percentile of the lifetime. Chi-squared test rejects optical flow outliers with 95% belief. For intensity feature tracker, 50 of Shi-Tomasi features are tracked by KLT tracker with 8-pt RANSAC. For the inequality constraint in (12), we let $m = 0.25\,\mathrm{m}^{-1}$ and $M = 10\,\mathrm{m}^{-1}$.

We summarize the estimation accuracy in Table I. *Event* means that the filter is updated by the optical flow from events in (11), and *Image* takes only feature tracks from images in (7) that has 5 sliding windows. *Image + Event* represents a hybrid estimator that obtains measurements from the both front-end. Note that *Event* is implemented by using the constrained filtering in Section IV-C. For the error metric, we report the position error $||\mathbf{p} - \hat{\mathbf{p}}||$ and the attitude error $||\mathrm{dcm2rvec}(\mathbf{R}\hat{\mathbf{R}}^{\mathsf{T}})||$ as mean Euclidean distance divided by the total traveled distance. In Table I, it is important to note that not only *Event* alone exhibits comparable accuracy to the filtering-based state-of-the-art EVIO, but also *Image + Event* shows the best accuracy all cases in overall.

The complementary characteristic of events and intensity images is highlighted in Fig. 3. At 42 seconds, the position error of *Image* in Fig. 3c is suddenly increased due to the

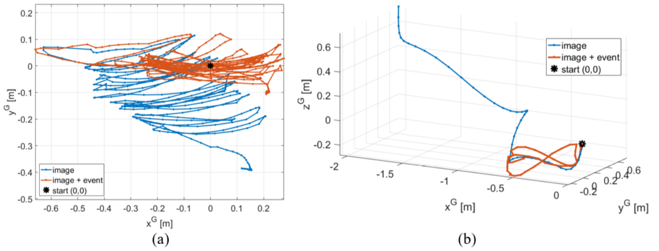

Fig. 5. Estimated positions of (a) *desk_translation* and (b) *desk_inf*.

large angular motion that is observed in Fig. 3b. However, the position error of *Image + Event* slowly increases regardless of the rapid motion. A sample image is shown during that maneuver in Fig. 1 in which most of the intensity features are failed, whereas there are plenty of optical flow measurements from events.

To validate the effect of the inverse scene-depth constraint, we test all sequences in Table I using optical flow from events with (*Event*) and without (*uEvent*) the projection (12). As a result, 3 out of 10 sequences are failed meaning that the filter diverges in *uEvent*. Fig. 4 shows the position error and estimated scene-depth in *boxes_translation*. In *Event*, the filter state is projected onto the constrained surface when $1/\hat{\lambda} > 4m$. This effectively maintains the scene-depth as physically feasible metric. In contrast, the scene-depth in *uEvent* exceeds the upper limit and the position error diverges.

B. The author-collected dataset

We recorded two sequences of dataset called *desk_translation* and *desk_inf* using DAVIS-240C (iniVation AG) in a desk environment. The author excited the hand-held sensor making the rectilinear and ∞ shaped trajectory shown in Fig. 5. They contains challenging sequences exhibiting the maximum angular rate as 294.4 and 297.6 deg/s respectively. Limitations on the standard frames are seen in Fig. 6(a,c): a low light condition and high dynamic motion. However, events is able to capture a set of optical flow in a stream of events as shown in Fig. 6(b,d). Estimated positions are drawn in Fig. 5 in which *Image* does not show clear rectilinear trajectory in Fig. 5(a) nor ∞ shape in Fig. 5(b), but we see the clear shape of trajectories in *Image + event*.

VI. Conclusion

In this paper, we have proposed a filtering-based pose estimator that fuses events, images, and inertial measurements. We directly model optical flow from events with the inequality constraint and feature tracks from intensity images. The complementary characteristic of events and intensity images yields better performance than when fused alone with an IMU in the low light and fast motion scenario. Our experimental results reveal that our method has decreased the position error by 49.9% at the benchmark dataset when compared to the state-of-the-art filtering-based estimator. Moreover, we see that body velocity information obtained from a stream of events could improve the position accuracy in the author-collected dataset.

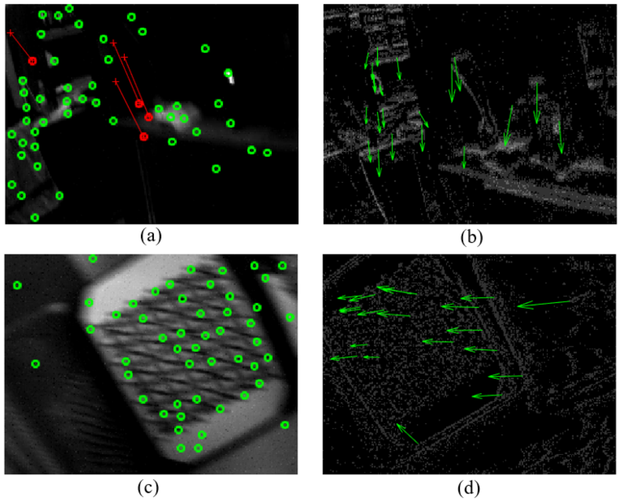

Fig. 6. Synchronized standard and event frame synthesized just for the visualization in a high dynamic range (a,b), and in a huge angular velocity (c,d) of the author-collected dataset. The legend is the same as in Fig. 1.

Our future works include a real-time implementation and evaluation of the algorithm. This would enable us to further analyze the computational time quantitatively of the proposed method. We expect that the proposed fusion scheme can be implemented using an optimization back-end, and would yield comparable pose accuracy to the state-of-the-art optimization counterpart, [13]. In addition to this, we have found that a corner extraction from the naively reconstructed event frame includes even edges that cause the aperture problem due to the blurred frame. As future work, this can be resolved by using a sharp event frame from the lifetime estimation [7], or the event-based corner detector [26] can be employed.

Acknowledgment

We thank the authors in [21] for releasing the event tracker as an open-source.

References

[1] P. Lichtsteiner, C. Posch, and T. Delbruck, ”A 128 × 128 120 dB 15 us Latency Asynchronous Temporal Contrast Vision Sensor,” *IEEE Journal of Soild-State Circuits,* vol.43, no.2, pp.566-576, 2008.

[2] C. Brandli, R. Berner, M. Yang, S. Liu, and T. Delbruck, ”A 240 × 180 130 dB 3 s Latency Global Shutter Spatiotemporal Vision Sensor,” *IEEE Journal of Soild-State Circuits,* vol.49, no.10, pp.2333-2341, 2014.

[3] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, ”Keyframe-based visual–inertial odometry using nonlinear optimization,” *The International Journal of Robotics Research*, vol.34, no.3, pp.314-334, 2015.

[4] T. Qin, P. Li, and S. Shen. ”Vins-mono: A robust and versatile monocular visual-inertial state estimator.” *IEEE Transactions on Robotics*, no.34, no.4, 1004-1020, 2018.

[5] S. Heo, and C. G. Park, ”Consistent EKF-based visual-inertial odometry on matrix Lie group,” *IEEE Sensors Journal*, vol.18, no.9, pp.3780-3788, 2018.

[6] R. Benosman, C. Clercq, X. Lagorce, S. Ieng, and C. Bartolozzi, ”Event-based visual flow,” *IEEE transactions on neural networks and learning systems*, vo.25, no.2, pp.407-417, 2013.

[7] E. Mueggler, C. Forster, N. Baumli, G. Gallego, and D. Scaramuzza, ”Lifetime estimation of events from dynamic vision sensors,” *IEEE international conference on Robotics and Automation (ICRA)*, pp.4874-4881, 2015.

[8] G. Gallego, H. Rebecq, D. Scaramuzza, ”A unifying contrast maximization framework for event cameras, with applications to motion, depth, and optical flow estimation,” *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.3867-3876, 2018.

[9] T. Stoffregen, L. Kleeman, ”Event cameras, contrast maximization and reward functions: an analysis,” *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.12300-12308, 2019.

[10] G. Gallego, M. Gehrig, and D. Scaramuzza. ”Focus Is All You Need: Loss Functions For Event-based Vision,” *2019 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12280-12289, 2019.

[11] D. Gehrig, H. Rebecq, G. Gallego, and D. Scaramuzza, ”EKLT: Asynchronous Photometric Feature Tracking Using Events and Frames,” *International Journal of Computer Vision*, pp.1-18, 2019.

[12] H. Rebecq, T. Horstschaefer, D. Scaramuzza, ”Real-time Visual-Inertial Odometry for Event Cameras using Keyframe-based Nonlinear Optimization,” *In Proc. Brit. Mach. Vis. Conf.*, 2017.

[13] A. R. Vidal, H. Rebecq, T. Horstschaefer, 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.15, no.2 pp.994-1001, 2018.

[14] E. Mueggler, G. Gallego, H. Rebecq, and D. Scaramuzza, ”Continuous-time visual-inertial odometry for event cameras,” *IEEE Transactions on Robotics*, vol.34, no.6, pp.1425-1440, 2018.

[15] D. Simon, ”Kalman filtering with state constraints: a survey of linear and nonlinear algorithms.” *IET Control Theory Applications*, vol.4, no.8, pp.1303-1318, 2010.

[16] P. Vachhani, S. Narasimhan, and R. Rengaswamy, ”Robust and reliable estimation via unscented recursive nonlinear dynamic data reconciliation,” *Journal of process control*, vol.16, no.10, pp.1075-1086, 2006.

[17] B. M. Bell, J. V. Burke, and G. Pillonetto. ”An inequality constrained nonlinear Kalman−Bucy smoother by interior point likelihood maximization,” *Automatica*, vol.45, no.1, pp.25-33, 2009.

[18] A. Z. Zhu, N. Atanasov, and K. Daniilidis., ”Event-based feature tracking with probabilistic data association,” *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp.4465-4470, 2017.

[19] B. D. Lucas, and T. Kanade, ”An iterative image registration technique with an application to stereo vision,” *International Joint Conference on Artificial Intelligence (IJCAI)*, pp.674-679, 1981

[20] S. Granger, and X. Pennec, ”Multi-scale EM-ICP: A fast and robust approach for surface registration,” *European Conference on Computer Vision*, pp.418-432, 2002.

[21] A. Z. Zhu, N. Atanasov, and K. Daniilidis, ”Event-based visual inertial odometry,” *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.5816-5824, 2017.

[22] A. I. Mourikis, and S. I. Roumeliotis, ”A multi-state constraint Kalman filter for vision-aided inertial navigation,” *2007 IEEE International Conference on Robotics and Automation (ICRA)*, pp.3565-3572, 2007.

[23] M. Bloesch, S. Omari, P. Fankhauser, H. Sommer, C. Gehring, J. Hwangbo, M. A. Hoepflinger, M. Hutter, and R. Siegwart, ”Fusion of optical flow and inertial measurements for robust egomotion estimation,” *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp.3102-3107, 2014.

[24] Simon, Dan, ”Optimal state estimation: Kalman, H infinity, and nonlinear approaches,” *John Wiley & Sons*, 2006.

[25] E. Mueggler, H. Rebecq, G. Gallego, T. Delbruck, and D. Scaramuzza, ”The event-camera dataset and simulator: Event-based data for pose estimation, visual odometry, and SLAM,” *The International Journal of Robotics Research*, vol.36, no.2, pp.142-149, 2017.

[26] E. Mueggler, C. Bartolozzi, and D. Scaramuzza, ”Fast Event-based Corner Detection,” *In. Proc. Brit. Mach. Vis. Conf.*, 2017.