Real-Time 3D Reconstruction and 6-DoF Tracking with an Event Camera

Hanme Kim, Stefan Leutenegger, and Andrew J. Davison

Department of Computing,
Imperial College London, UK
{hanme.kim,s.leutenegger,a.davison}@imperial.ac.uk

Abstract. We propose a method which can perform real-time 3D reconstruction from a single hand-held event camera with no additional sensing, and works in unstructured scenes of which it has no prior knowledge. It is based on three decoupled probabilistic filters, each estimating 6-DoF camera motion, scene logarithmic (log) intensity gradient and scene inverse depth relative to a keyframe, and we build a real-time graph of these to track and model over an extended local workspace. We also upgrade the gradient estimate for each keyframe into an intensity image, allowing us to recover a real-time video-like intensity sequence with spatial and temporal super-resolution from the low bit-rate input event stream. To the best of our knowledge, this is the first algorithm provably able to track a general 6D motion along with reconstruction of arbitrary structure including its intensity and the reconstruction of grayscale video that exclusively relies on event camera data.

Keywords: 6-DoF Tracking, 3D Reconstruction, Intensity Reconstruction, Visual Odometry, SLAM, Event-Based Camera.

1 Introduction

Event cameras offer a breakthrough new paradigm for real-time vision, with potential in robotics, wearable devices and autonomous vehicles, but it has proven very challenging to use them in most standard computer vision problems. Inspired by the superior properties of human vision [2], an event camera records not image frames but an asynchronous sequence of per-pixel intensity changes, each with a precise timestamp. While this data stream efficiently encodes image dynamics with extremely high dynamic range and temporal contrast, the lack of synchronous intensity information means that it is not possible to apply much of the standard computer vision toolbox of techniques. In particular, the multi-view correspondence information which is essential to estimate motion and structure is difficult to obtain because each event by itself carries little information and no signature suitable for reliable matching.

Approaches aiming at simultaneous camera motion and scene structure estimation therefore need also to jointly estimate the intensity appearance of the scene, or at least a highly descriptive function of this such as a gradient map. So far, this has only been successfully achieved in the reduced case of pure camera rotation, where the scene reconstruction takes the form of a panorama image.

In this paper we present the first algorithm which performs joint estimation of 3D scene structure, 6-DoF camera motion and up to scale scene intensity from a single hand-held event camera moved in front of an unstructured static scene. Our approach runs in real-time on a standard PC. The core of our method is three interleaved probabilistic filters, each estimating one unknown aspect of this challenging Simultaneous Localisation and Mapping (SLAM) problem: camera motion, scene log intensity gradient and scene inverse depth. From pure event input our algorithm generates various outputs including a real-time, high bandwidth 6-DoF camera track, scene depth map for one or multiple linked keyframes, and a high dynamic range reconstructed video sequence at a user-chosen frame-rate.

1.1 Event-Based Cameras

The event camera or silicon retina is gradually becoming more widely known by researchers in computer vision, robotics and related fields, in particular since the release as a commercial device for researchers of the Dynamic Vision Sensor (DVS) [14] shown in Figure 1 (c). The pixels of this device asynchronously report log intensity changes of a pre-set threshold size as a stream of asynchronous events, each with pixel location, polarity, and microsecond-precise timestamp. Figure 1 visualises some of the main properties of the event stream; in particular the almost continuous response to very rapid motion and the way that the output data-rate depends on scene motion, though in practice almost always dramatically lower than that of standard video. These properties offer the potential to overcome the limitations of real-world computer vision applications, relying on conventional imaging sensors, such as high latency, low dynamic range, and high power consumption.

Recently, cameras have been developed that interleave event data with conventional intensity frames (DAVIS [3]), or per-event intensity measurement (ATIS [21]). Our framework could be extended to make use of these image measurements this would surely make joint estimation easier. However, in a persistently dynamic motion, they may not be useful. Also, they partially break the appeal and optimal information efficiency of a pure event-based data stream. We therefore believe that first solving the hardest problem of not relying on standard image frames will be useful on its own and provides the insights to make best use of additional measurements if they are available.

1.2 Related Work

Early published work using event cameras focused on tracking moving objects from a fixed point of view, successfully showing the superior high speed measurement and low latency properties [8, 6]. However, work on tracking and reconstruction of more general, previously unknown scenes with a freely moving

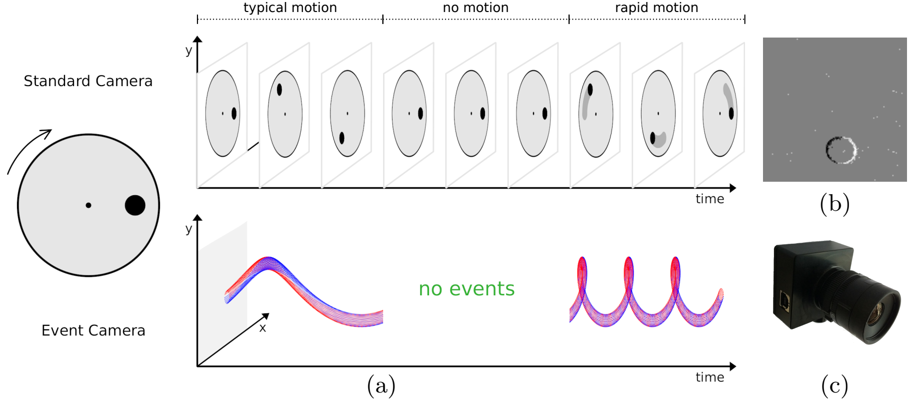


Fig. 1. Event-based camera: (a): in contrast to standard video frames shown in the upper graph, a stream of events from an event camera, plotted in the lower graph, offers no redundant data output, only informative pixels or no events at all. Red and blue dots represent positive and negative events respectively (this figure was recreated inspired by the associated animation of [19]: `https://youtu.be/LauQ6LWTkxM?t=35s`). (b): image-like visualisation by accumulating events within a time interval — white and black pixels represent positive and negative events respectively. (c): the first commercial event camera, DVS128, from iniLabs Ltd.

event camera, which we believe is the best place to take full advantage of its remarkable properties, has been limited. The clear difficulty is that most methods normally used in tracking and mapping, such as feature detection and matching or whole image alignment, cannot be directly applied to its fundamentally different visual measurement stream.

Cook *et al.* [7] proposed an interacting network which interprets a stream of events to recover different visual estimate 'maps' of scenes such as intensity, gradient and optical flow while estimating global rotating camera motion. More recently, Bardow *et al.* [1] presented an optical flow and intensity estimation using an event camera which allows any camera motion as well as dynamic scenes.

An early 2D SLAM method was proposed by Weikersdorfer *et al.* [24] which tracks a ground robot pose while reconstructing a planar ceiling map with an upward looking DVS camera. Mueggler *et al.* [19] presented an onboard 6-DoF localisation flying robot system which is able to track its relative pose to a known target even at very high speed. To investigate whether current techniques can be applied to a large scale visual SLAM problem, Milford *et al.* [16] presented a simple visual odometry system using a DVS camera with loop closure built on top of the SeqSLAM algorithm using events accumulated into frames [17].

In a much more constrained and hardware-dependent setup, Schraml *et al.* [22] developed a special 360° rotating camera that consists of a pair of dynamic vision line sensors which creates 3D panoramic scenes aided by its embedded encoders and stereo event streams. Combined with an active projector, Matsuda *et al.* [15]

showed that high quality 3D object reconstruction can be achievable which is better than for laser scanners or RGB-D cameras in some specific situations.

The most related work to our method is the simplified SLAM system based on probabilistic filtering proposed by Kim *et al.* [12], which estimates spatial gradients which are then integrated to reconstruct high quality and high dynamic range planar scenes while tracking global camera rotation. Their method has a similar overall concept to ours with multiple interacting probabilistic filters, but is limited to pure rotation camera motion and panorama reconstruction. Also it is not completely real-time because of the computational complexity of the particle filter used in their tracking algorithm.

There have been no previous published results on estimating 3D depth from a single moving event camera. Most researchers working with event cameras have assumed that this problem is too difficult, and attempts at 3D estimation have combined an event camera with other sensors: a standard frame-based CMOS camera [5], or an RGB-D camera [23]. These are, of course, possible practical ways of using an event camera for solving SLAM problems. However, we believe that resorting to standard sensors discards many of the advantages of processing the efficient and data-rick pure event stream, as well as introducing extra complication including synchronisation and calibration problems to be solved. One very interesting approach if the application permits is to combine two event cameras in a stereo setup [4]. The nicest part of that method is the way that stereo matching of events can be achieved based on coherent timestamps.

Our work in this paper was inspired by a strong belief that depth estimation from a single moving event camera must be possible, because if the device is working correctly and recording all pixel-wise intensity changes then all of the information present in a standard video stream must be available in principle, at least up to scale. In fact, the high temporal contrast and dynamic range of event pixels means that much more information should be present in an event stream than in standard video at the same resolution. In particular, the results of Kim *et al.* [12] on sub-pixel tracking and super-resolution mosaic reconstruction from events gave a strong indication that the accurate multi-view correspondence needed for depth estimation is possible. The essential insight to extending Kim *et al.*'s approach towards getting depth from events is that once the camera starts to translate, if two pixels have the same intensity gradient, the one which is closer to the camera move past the camera faster and therefore emit more events than the farther one. This is the essential mechanism built into our probabilistic filter for inverse depth.

2 Method

Following many recent successful SLAM systems such as PTAM [13], DTAM [20], and LSD-SLAM [10], which separate the tracking and mapping components based on the assumption that the current estimate from one component is accurate enough to lock for the purposes of estimating the other, the basic structure of our approach relies on three interleaved probabilistic filters. One tracks the global 6-DoF camera motion; the second estimates the log intensity gradients in a keyframe image — a representation which is also in parallel upgraded into a full image-like intensity map. Finally the third filter estimates the inverse depths of a keyframe. It should be noted that we essentially separate the mapping part into two, i.e. the gradient and inverse depth estimations, considering fewer number of events caused by parallax while almost all events carry gradient information. We also build a textured semi-dense 3D point cloud from selected keyframes with their associated reconstructed intensity and inverse depth estimate. We do not use an explicit bootstrapping method as we have found that, starting from scratch, alternating estimation very often lead to convergence.

2.1 Preliminaries

We denote an event as $\mathbf{e}(u,v) = (u,v,p,t)^\top$ where u and v are pixel location, p is polarity and t is microsecond-precise timestamp — our event-based camera has the fixed pre-calibrated intrinsic matrix $\mathtt{K}$ and all event pixel locations are pre-warped to remove radial distortion. We also define two important time intervals τ and τ_c, as in [12], which are the time elapsed since the most recent previous event from *any pixel* and at *the same pixel* respectively.

2.2 Event-Based Camera 6-DoF Tracking

We use an Extended Kalman Filter (EKF) to estimate the global 6-DoF camera motion over time with its state $\mathbf{x} \in \mathbb{R}^6$, which is a minimal representation of the camera pose c with respect to the world frame of reference w, and covariance matrix $\mathtt{P}_\mathbf{x} \in \mathbb{R}^{6\times 6}$. The state vector is mapped to a member of the Lie group $\mathbf{SE}(3)$, the set of 3D rigid body transformations, by the matrix exponential map:

$$\mathtt{T}_{wc} = \exp\left(\sum_{i=1}^{6} \mathtt{x}_i \mathtt{G}_i\right) = \begin{pmatrix} \mathtt{R}_{wc} & \mathbf{t}_w \\ \mathbf{0}^\top & 1 \end{pmatrix} , \tag{1}$$

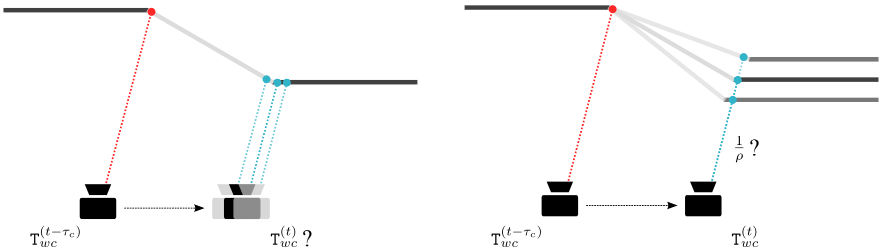
where $\mathtt{G}$ is the Lie group generator for $\mathbf{SE}(3)$, $\mathtt{R}_{wc} \in \mathbf{SO}(3)$, and $\mathbf{t}_w \in \mathbb{R}^3$. The basic idea is to find (assuming that the current log intensity and inverse depth estimates are correct) the camera pose which best predicts a log intensity change consistent with the event just received, as shown in Figure 2 (a).

Motion Prediction We use a 6-DoF (translation and rotation) constant position motion model for motion prediction; the variance of the prediction is proportional to the time interval:

$$\mathbf{x}^{(t|t-\tau)} = \mathbf{x}^{(t-\tau|t-\tau)} + \mathbf{n} , \tag{2}$$

$$\mathtt{P_x}^{(t|t-\tau)} = \mathtt{P}_\mathbf{x}^{(t-\tau|t-\tau)} + \mathtt{P}_\mathbf{n} , \tag{3}$$

where each component of $\mathbf{n}$ is independent Gaussian noise in all six axes i.e. $\mathbf{n}_i \sim \mathcal{N}(0, \sigma_i^2\tau)$, and $\mathtt{P}_\mathbf{n} = \mathrm{diag}(\sigma_1^2\tau, \ldots, \sigma_6^2\tau)$.



(a) camera pose estimation (b) inverse depth estimation

Fig. 2. Camera pose and inverse depth estimation. (a): based on the assumption that the current log intensity estimate (shown as the colour of the solid line) and inverse depth estimate (shown as the geometry of the solid line) are correct, we find current camera pose $\mathtt{T}_{wc}^{(t)}$ most consistent with the predicted log intensity change since the previous event at the same pixel at pose $\mathtt{T}_{wc}^{(t-\tau_c)}$ compared to the current event polarity. (b): similarly for inverse depth estimation, we assume that the current reconstructed log intensity and camera pose estimate are correct, and find the most probable inverse depth consistent with the new event measurement.

Measurement Update We calculate the value of a measurement $z_{\mathbf{x}}$ given an event $\mathbf{e}(u, v)$, the current keyframe pose $\mathtt{T}_{wk}$, the current camera pose estimate $\mathtt{T}_{wc}^{(t)}$, the previous pose estimate $\mathtt{T}_{wc}^{(t-\tau_c)}$, where the previous event was received at *the same pixel*, and a reconstructed image-like log intensity keyframe with inverse depth by taking a log intensity difference between two corresponding ray-triangle intersection points, $\mathbf{p}_w^{(t)}$ and $\mathbf{p}_w^{(t-\tau_c)}$, as shown in Figure 3:

$$z_{\mathbf{x}} = \pm C \ , \tag{4}$$

$$h_{\mathbf{x}}(\mathbf{x}^{(t|t-\tau)}) = \mathtt{I}_l\left(\mathbf{p}_w^{(t)}\right) - \mathtt{I}_l\left(\mathbf{p}_w^{(t-\tau_c)}\right) \ , \tag{5}$$

$$\text{where} \ \ \mathtt{I}_l\left(\mathbf{p}_w\right) = (1-a-b)\mathtt{I}_l\left(\mathbf{v}_0\right) + a\mathtt{I}_l\left(\mathbf{v}_1\right) + b\mathtt{I}_l\left(\mathbf{v}_2\right) \ . \tag{6}$$

Here $\pm C$ is a known event threshold — its sign is decided by the polarity of an event. $\mathtt{I}_l$ is a log intensity value based on a reconstructed log intensity keyframe, and $\mathbf{v}_0$, $\mathbf{v}_1$, and $\mathbf{v}_2$ are three vertices of an intersected triangle. To obtain a corresponding 3D point location $\mathbf{p}_w$ in the world frame of reference, we use ray-triangle intersection [18] which yields a vector $(l, a, b)^\top$ where l is the distance to the triangle from the origin of the ray and a, b are the barycentric coordinates of the intersected point which is then used to calculate an interpolated log intensity.

In the EKF framework, the camera pose estimate and its uncertainty covariance matrix are updated by the standard equations at every event using:

$$\mathbf{x}^{(t|t)} = \mathbf{x}^{(t|t-\tau)} + \mathbf{W}_{\mathbf{x}}\nu_{\mathbf{x}} \ , \tag{7}$$

$$\mathtt{P}_{\mathbf{x}}^{(t|t)} = \left(\mathtt{I}_{6\times6} - \mathbf{W}_{\mathbf{x}}\frac{\partial h_{\mathbf{x}}}{\partial \mathbf{x}^{(t|t-\tau)}}\right)\mathtt{P}_{\mathbf{x}}^{(t|t-\tau)} \ , \tag{8}$$

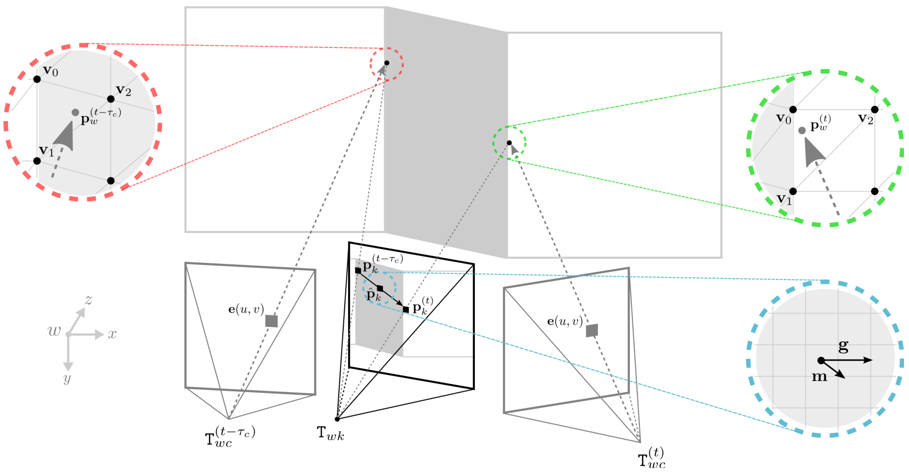


Fig. 3. Basic geometry for; **tracking** and **inverse depth estimation**: we find two corresponding ray-triangle intersection points, $\mathbf{p}_w^{(t)}$ and $\mathbf{p}_w^{(t-\tau_c)}$, in the world frame of reference using a ray-triangle intersection method [18] to compute the value of a measurement — a log intensity difference between two points given an event $\mathbf{e}(u,v)$, the current keyframe pose $\mathtt{T}_{wk}$, the current camera pose estimate $\mathtt{T}_{wc}^{(t)}$, the previous pose estimate $\mathtt{T}_{wc}^{(t-\tau_c)}$, the reconstructed log intensity and inverse depth keyframe, **gradient estimation**: we project two intersection points onto the current keyframe, $\mathbf{p}_k^{(t)}$ and $\mathbf{p}_k^{(t-\tau_c)}$, to find a displacement vector between them, which is then used to calculate a motion vector $\mathbf{m}$ to compute the value of a measurement $(\mathbf{g} \cdot \mathbf{m})$ at a midpoint $\hat{\mathbf{p}}_k$ based on the brightness constancy and the linear gradient assumption.

where the innovation $\nu_{\mathbf{x}}$ and Kalman gain $\mathbf{W}_{\mathbf{x}}$ are defined by the standard EKF definitions. The measurement uncertainty is a scalar variance $\sigma_{\mathbf{x}}^2$, and we omit the Jacobian $\frac{\partial h_{\mathbf{x}}}{\partial \mathbf{x}^{(t|t-\tau)}}$ derivation due to the space limitation.

2.3 Gradient Estimation and Log Intensity Reconstruction

We now use the updated camera pose estimate to incrementally improve the estimates of the log intensity gradient at each keyframe pixel based on a pixel-wise EKF. However, because of the random walk nature of our tracker which generates a noisy motion estimate, we first apply a weighted average filter to the new camera pose estimate. To reconstruct super resolution scenes by harnessing the very high speed measurement property of the event camera, we use a higher resolution for keyframes than for the low resolution sensor. This method is similar to the one in [12], but we model the measurement noise properly to get better gradient estimate, and use a parallelisable reconstruction method for speed.

Pixel-Wise EKF Based Gradient Estimation Each pixel of the keyframe holds an independent gradient estimate $\mathbf{g}(\mathbf{p}_k) = (g_u, g_v)^\top$, consisting of log intensity gradients g_u and g_v along the horizontal and vertical axes in image space respectively, and a 2×2 uncertainty covariance matrix $\mathtt{P}_{\mathbf{g}}(\mathbf{p}_k)$. At initialisation, all gradients are initialised to zero with large variances.

We assume, based on the rapidity of events, a linear gradient between two consecutive events at the same event camera pixel, and update the midpoint $\hat{\mathbf{p}}_k$ of the two projected points $\mathbf{p}_k^{(t)}$ and $\mathbf{p}_k^{(t-\tau_c)}$. We now define $z_{\mathbf{g}}$, a measurement of the instantaneous *event rate* at this pixel, and its measurement model $h_{\mathbf{g}}$ based on the brightness constancy equation $(\mathbf{g} \cdot \mathbf{m})\tau_c = \pm C$, where $\mathbf{g}$ is a gradient estimate and $\mathbf{m} = (m_u, m_v)^\top$ is a motion vector — the displacement between two corresponding pixels in the current keyframe divided by the elapsed time τ_c as shown in Figure 3:

$$z_{\mathbf{g}} = \pm \frac{C}{\tau_c} \ , \tag{9}$$

$$h_{\mathbf{g}} = (\mathbf{g}(\hat{\mathbf{p}}_k) \cdot \mathbf{m}) \ , \tag{10}$$

$$\text{where } \ \mathbf{m} = \frac{\mathbf{p}_k^{(t)} - \mathbf{p}_k^{(t-\tau_c)}}{\tau_c} \ . \tag{11}$$

The current gradient estimate and its uncertainty covariance matrix at that pixel are updated independently in the same way as in the measurement update of our tracker following the standard EKF equations.

The Jacobian $\frac{\partial h_{\mathbf{g}}}{\partial \mathbf{g}(\hat{\mathbf{p}}_k)^{(t-\tau_c)}}$ of the measurement function with respect to changes in gradient is simply (m_u, m_v), and the measurement noise $\mathtt{N}_{\mathbf{g}}$ is:

$$\mathtt{N}_{\mathbf{g}} = \frac{\partial z_{\mathbf{g}}}{\partial C} \mathtt{P}_C \left(\frac{\partial z_{\mathbf{g}}}{\partial C} \right)^\top = \frac{\sigma_C^2}{\tau_c^2} \ , \tag{12}$$

where σ_C^2 is the sensor noise with respect to the event threshold.

Log Intensity Reconstruction Along with the pixel-wise EKF based gradient estimation method, we perform interleaved absolute log intensity reconstruction running on a GPU. We define our convex minimisation function as:

$$\min_{\mathtt{I}_l} \left\{ \int_\Omega ||\mathbf{g}(\mathbf{p}_k) - \nabla \mathtt{I}_l(\mathbf{p}_k)||_{\epsilon_d}^h + \lambda ||\nabla \mathtt{I}_l(\mathbf{p}_k)||_{\epsilon_r}^h \mathrm{d}\mathbf{p}_k \right\} \ . \tag{13}$$

Here the data term represents the error between estimated gradients $\mathbf{g}(\mathbf{p}_k)$ and those of a reconstructed log intensity $\nabla \mathtt{I}_l(\mathbf{p}_k)$, and the regularisation term enforces smoothness, both under a robust Huber norm. This function can be written using the Legendre Fenchel transformation [11] as follows:

$$\begin{aligned} \min_{\mathtt{I}_l} \max_{\mathbf{q}} \max_{\mathbf{p}} \{ \langle \mathbf{p}, \mathbf{g} - \nabla \mathtt{I}_l \rangle - \frac{\epsilon_d}{2} ||\mathbf{p}||^2 - \delta_{\mathbf{p}}(\mathbf{p}) + & \\ \langle \mathbf{q}, \nabla \mathtt{I}_l \rangle - \frac{\epsilon_r}{2\lambda} ||\mathbf{q}||^2 - \delta_{\mathbf{q}}(\mathbf{q}) \} \ , & \end{aligned} \tag{14}$$

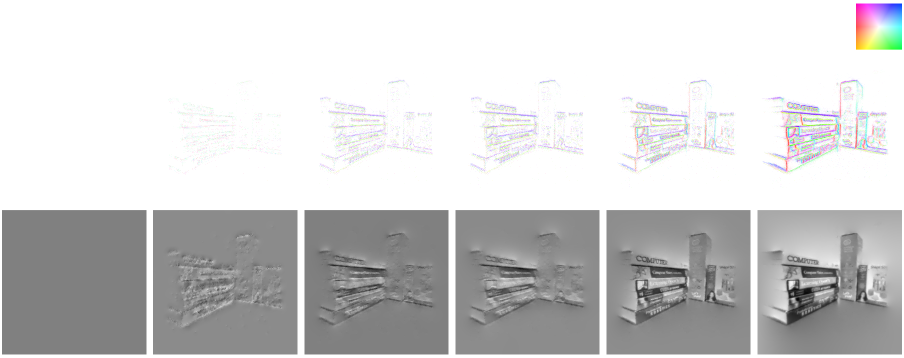


Fig. 4. Typical temporal progression (left to right) of gradient estimation and log intensity reconstruction as a hand-held camera browses a 3D scene. The colours and intensities on the top row represent the orientations and strengths of the gradients of the scene (refer to the colour chart in the top right). In the bottom row, we see these gradient estimates upgraded to reconstructed intensity images.

where we can solve by maximising with respect to $\mathbf{p}$:

$$\mathbf{p}^{(n+1)} = \frac{\frac{\mathbf{p}^{(n)}+\sigma_{\mathbf{p}}(\mathbf{g}-\nabla \mathtt{I}_l)}{1+\sigma_{\mathbf{p}}\epsilon_d}}{\max\left(1, \left|\frac{\mathbf{p}^{(n)}+\sigma_{\mathbf{p}}(\mathbf{g}-\nabla \mathtt{I}_l)}{1+\sigma_{\mathbf{p}}\epsilon_d}\right|\right)} \; , \tag{15}$$

maximising with respect to $\mathbf{q}$:

$$\mathbf{q}^{(n+1)} = \frac{\frac{\mathbf{q}^{(n)}+\sigma_{\mathbf{q}}\nabla \mathtt{I}_l}{1+\frac{\sigma_{\mathbf{q}}\epsilon_r}{\lambda}}}{\max\left(1, \frac{1}{\lambda}\left|\frac{\mathbf{q}^{(n)}+\sigma_{\mathbf{q}}\nabla \mathtt{I}_l}{1+\frac{\sigma_{\mathbf{q}}\epsilon_r}{\lambda}}\right|\right)} \; , \tag{16}$$

and minimising with respect to $\mathtt{I}_l$:

$$\mathtt{I}_l^{(n+1)} = \mathtt{I}_l^{(n)} - \sigma_{\mathtt{I}_l}(\mathrm{div}\mathbf{p}^{(n+1)} - \mathrm{div}\mathbf{q}^{(n+1)}) \; . \tag{17}$$

We visualise the progress of gradient estimation and log intensity reconstruction over time during hand-held event camera motion in Figure 4.

2.4 Inverse Depth Estimation and Regularisation

We now use the same camera pose estimate as in the gradient estimation and a reconstructed log intensity keyframe to incrementally improve the estimates of the inverse depth at each keyframe pixel based on another pixel-wise EKF. As in camera pose estimation, assuming that the current camera pose estimate and reconstructed log intensity are correct, we aim to update the inverse depth estimate to best predict the log intensity change consistent with the current event polarity as shown in Figure 2 (b).

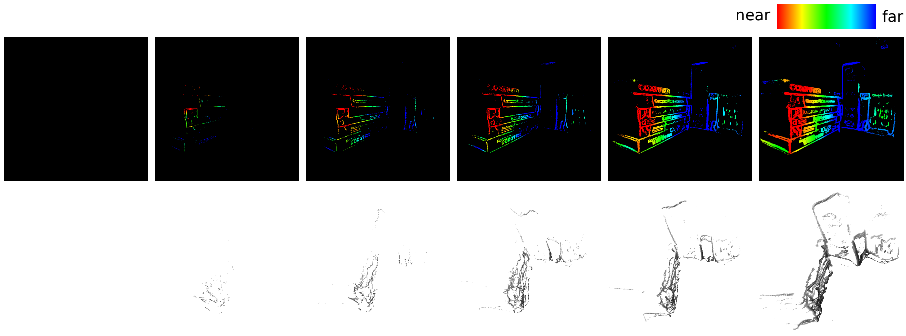


Fig. 5. Typical temporal progression (left to right) of inverse depth estimation and regularisation as a hand-held camera browses a 3D scene. The colours on the top row represent the different depths of the scene (refer to the colour chart in the top right and the associated semi-dense 3D point cloud on the bottom row).

Pixel-Wise EKF Based Inverse Depth Estimation Each pixel of the keyframe holds an independent inverse depth state value $\rho(\mathbf{p}_k)$ with variance $\sigma^2_{\rho(\mathbf{p}_k)}$. At initialisation, all inverse depths are initialised to nominal values with large variances. In the same way as in our tracking method, we calculate the value of a measurement $z_{\boldsymbol{\rho}}$ which is a log intensity difference between two corresponding ray-triangle intersection points $\mathbf{p}_w^{(t)}$ and $\mathbf{p}_w^{(t-\tau_c)}$ as shown in Figure 3:

$$z_{\boldsymbol{\rho}} = \pm C \ , \tag{18}$$

$$h_{\boldsymbol{\rho}} = \mathtt{I}_l\left(\mathbf{p}_w^{(t)}\right) - \mathtt{I}_l\left(\mathbf{p}_w^{(t-\tau_c)}\right) \ . \tag{19}$$

In the EKF framework, we stack the inverse depths of all three vertices $\boldsymbol{\rho} = (\rho_{\mathbf{v}_0}, \rho_{\mathbf{v}_1}, \rho_{\mathbf{v}_2})^\top$ which contribute to the intersected 3D point and update them with their associated 3×3 uncertainty covariance matrix at every event in the same way of the measurement update of our tracker following the standard EKF equations. The measurement noise $\mathtt{N}_{\boldsymbol{\rho}}$ is a scalar variance σ^2_ρ, and we omit the Jacobian $\frac{\partial h_\rho}{\partial \boldsymbol{\rho}^{(t-\tau_c)}}$ derivation due to the space limitation.

Inverse Depth Regularisation As a background process running on a GPU, we perform inverse depth regularisation on keyframe pixels with high confidence inverse depth estimate whenever there has been a large change in the estimates. We penalise deviation from a spatially smooth inverse depth map by assigning each inverse depth value the average of its neighbours, weighted by their respective inverse variances as described in [9]. If two adjacent inverse depths are different more than 2σ, they do not contribute to each other to preserve discontinuities due to occlusion boundaries. We visualise the progress of inverse depth estimation and regularisation over time as event data is captured during hand-held event camera motion in Figure 5.

3 Experiments

Our algorithm runs in real-time on a standard PC with typical scenes and motion speed, and we have conducted experiments both indoors and outdoors. We recommend viewing our video [1] which illustrates all of the key results in a better form than still pictures and in real-time.

3.1 Single Keyframe Estimation

We demonstrate the results from our algorithm as it tracks against and reconstructs a single keyframe in a number of different scenes. In Figure 6, for each scene we show column by column an image-like view of the event streams, estimated gradient map, reconstructed intensity map with super resolution and high dynamic range properties, estimate depth map and semi-dense 3D point cloud. The 3D reconstruction quality is generally good, though we can see that there are sometimes poorer quality depth estimates near to occlusion boundaries and where not enough events have been generated.

3.2 Multiple Keyframes

We evaluated the proposed method on several trajectories which require multiple keyframes to cover. If the camera has moved too far away from the current keyframe, we create a new keyframe from the most recent estimation results and reconstruction. To create a new keyframe, we project all 3D points based on the current keyframe pose and the estimated inverse depth into the current camera pose, and propagate the current estimates and reconstruction only if they have high confidence in inverse depth. Figure 7 shows one of the results in a semi-dense 3D point cloud form constructed based on generated keyframes each consisting of reconstructed super-resolution and high dynamic range intensity and inverse depth map. The bright RGB 3D coordinate axes represent the current camera pose while the darker ones show all keyframe poses generated in this experiment.

3.3 Video Rendering

Using our proposed method, we can turn an event-based camera into a high speed and high dynamic range artificial camera by rendering video frames based on ray-casting as shown in Figure 8. Here we choose to render at the same low resolution as event-based input.

3.4 High Speed Tracking

We evaluated the proposed method on several trajectories which include rapid motion (e.g. shaking hand). The top graph in Fig. 9 shows the estimated camera pose history, and the two groups of the insets below show an image-like event

[1] `https://youtu.be/yHLyhdMSw7w`

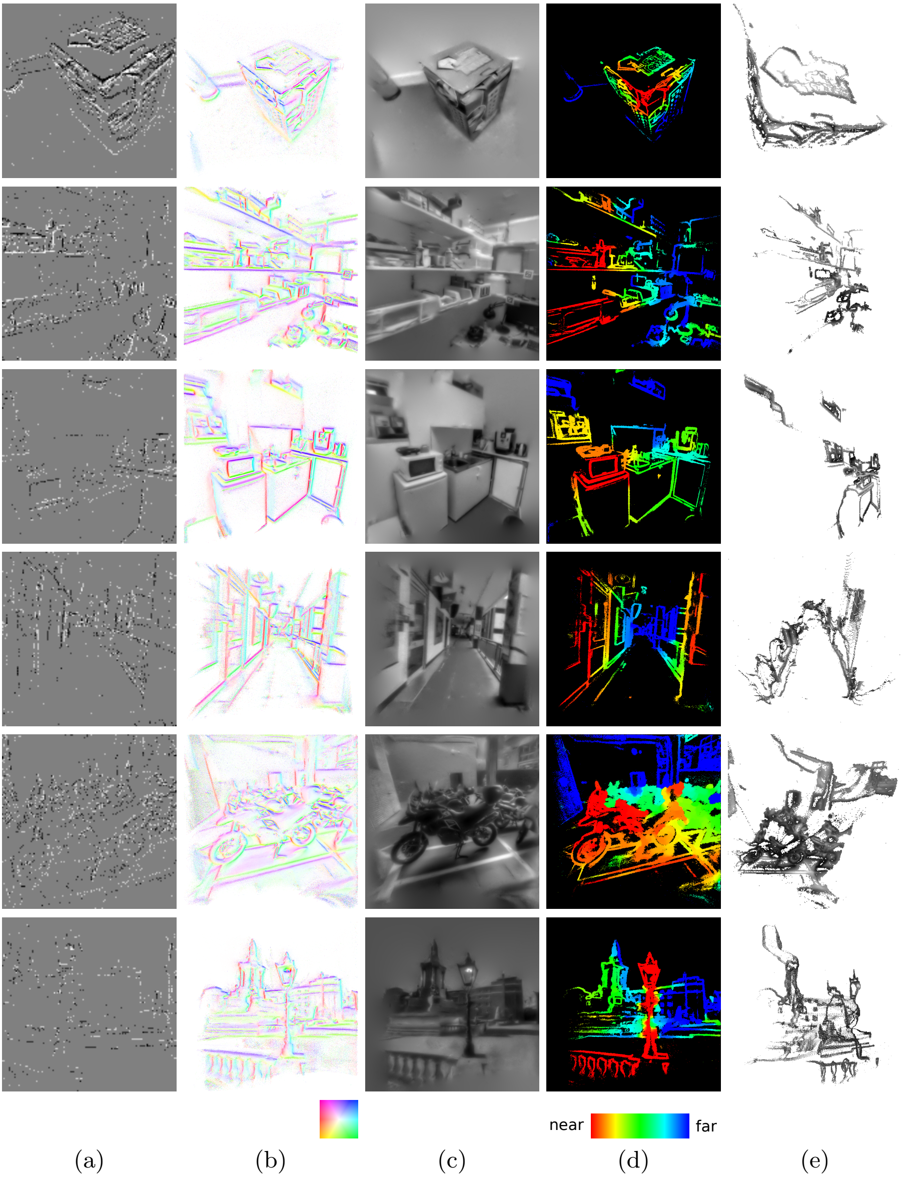


Fig. 6. Demonstrations in various settings of the different aspects of our joint estimation algorithm. (a) visualisation of the input event stream; (b) estimated gradient keyframes; (c) reconstructed intensity keyframes with super resolution and high dynamic range properties; (d) estimated depth maps; (e) semi-dense 3D point clouds.

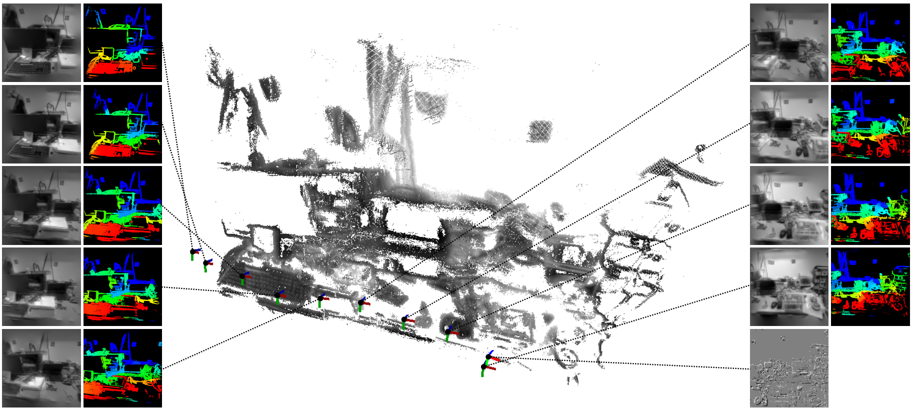


Fig. 7. 3D point cloud of an indoor scene constructed from multiple keyframes, showing keyframe poses with their intensity and depth map estimates.

visualisation, a rendered video frame showing the quality of our tracker, and a motion blurred standard camera video frame as a reference of rapid motion. Our current implementation is not able to process this very high event-rate (up to 1M events per second in this experiment) in real-time, but we believe it is a simple matter of engineering to run at this extremely high rate in real-time in the near future.

3.5 Discussion

Our results so far are qualitative, and we have focused on demonstrating the core novelty of our approach in breaking through to get joint estimation of depth, 6-DoF motion and intensity from pure event data with general motion and unknown general scenes. There are certainly still weakness in our current approach, and while we believe that it is remarkable that our approach of three interleaved filters, each of which operates as if the results of the others are

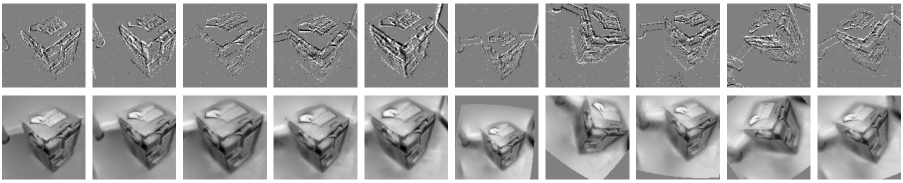


Fig. 8. Our proposed method can render HDR video frames at user-chosen time instances and resolutions by ray-casting the current reconstruction. This is the same scene as in the first row of Figure 6.

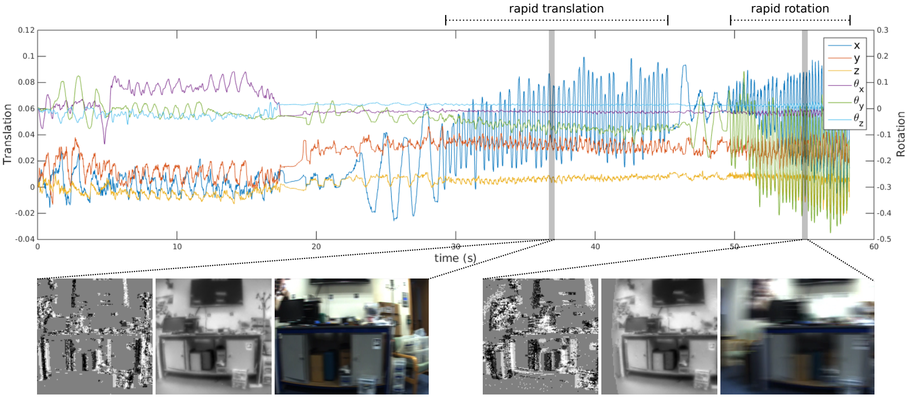


Fig. 9. The top graph shows the estimated camera pose history, and the two groups of the insets below show an image-like event visualisation, a rendered video frame showing the quality of our tracker, and a motion blurred standard camera video frame as a reference of rapid motion (up to 5Hz in this experiment).

correct, works at all, there is plenty of room for further research. It is clear that the interaction of these estimation processes is key, and in particular that the relatively slow convergence of inverse depth estimates tends to cause poor tracking, then data association errors and a corruption of other parts of the estimation process. We will investigate this further, and may need to step back from our current approach of real-time pure event-by-event processing towards a partially batch estimation approach in order to get better results.

4 Conclusions

To the best of our knowledge, this is the first 6-DoF tracking and 3D reconstruction method purely based on a stream of events with no additional sensing, and it runs in real-time on a standard PC. We hope this opens up the door to practical solutions to the current limitations of real-world SLAM applications. It is worth restating that the measurement rate of the event-based camera is on the order of a microsecond, its independent pixel architecture provides very high dynamic range, and the bandwidth of an event stream is much lower than a standard video stream. These superior properties of event-based cameras offer the potential to overcome the limitations of real-world computer vision applications relying on conventional imaging sensors.

Acknowledgements. Hanme Kim was supported by an EPSRC DTA scholarship and the Qualcomm Innovation Fellowship 2014. We thank Jacek Zienkiewicz, Ankur Handa, Patrick Bardow, Edward Johns and other colleagues at Imperial College London for many useful discussions.

References

1. Bardow, P., Davison, A.J., Leutenegger, S.: Simultaneous Optical Flow and Intensity Estimation from an Event Camera. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
2. Boahen, K.: Neuromorphic Chips. Scientific American (2005)
3. Brandli, C., Berner, R., Yang, M., Liu, S.C., Delbruck, T.: A 240×180 130 dB 3 μs Latency Global Shutter Spatiotemporal Vision Sensor. IEEE Journal of Solid-State Circuits (JSSC) 49(10), 2333–2341 (2014)
4. Carneiro, J., Ieng, S., Posch, C., Benosman, R.: Event-based 3D reconstruction from neuromorphic retinas. Journal of Neural Networks 45, 27–38 (2013)
5. Censi, A., Scaramuzza, D.: Low-Latency Event-Based Visual Odometry. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA) (2014)
6. Conradt, J., Cook, M., Berner, R., Lichtsteiner, P., Douglas, R., Delbruck, T.: A pencil balancing robot using a pair of AER dynamic vision sensors. In: IEEE International Symposium on Circuits and Systems (ISCAS) (2009)
7. Cook, M., Gugelmann, L., Jug, F., Krautz, C., Steger, A.: Interacting maps for fast visual interpretation. In: Proceedings of the International Joint Conference on Neural Networks (IJCNN) (2011)
8. Delbruck, T., Lichtsteiner, P.: Fast sensory motor control based on event-based hybrid neuromorphic-procedural system. In: IEEE International Symposium on Circuits and Systems (ISCAS) (2007)
9. Engel, J., Sturm, J., Cremers, D.: Semi-dense visual odometry for a monocular camera. In: Proceedings of the International Conference on Computer Vision (ICCV) (2013)
10. Engel, J., Schoeps, T., Cremers, D.: LSD-SLAM: Large-scale direct monocular SLAM. In: Proceedings of the European Conference on Computer Vision (ECCV) (2014)
11. Handa, A., Newcombe, R.A., Angeli, A., Davison, A.J.: Applications of the Legendre-Fenchel transformation to computer vision problems. Tech. Rep. DTR11-7, Imperial College London (2011)
12. Kim, H., Handa, A., Benosman, R., Ieng, S.H., Davison, A.J.: Simultaneous Mosaicing and Tracking with an Event Camera. In: Proceedings of the British Machine Vision Conference (BMVC) (2014)
13. Klein, G., Murray, D.W.: Parallel Tracking and Mapping for Small AR Workspaces. In: Proceedings of the International Symposium on Mixed and Augmented Reality (ISMAR) (2007)
14. Lichtsteiner, P., Posch, C., Delbruck, T.: A 128×128 120 dB 15 μs Latency Asynchronous Temporal Contrast Vision Sensor. IEEE Journal of Solid-State Circuits (JSSC) 43(2), 566–576 (2008)
15. Matsuda, N., Cossairt, O., Gupta, M.: MC3D: Motion Contrast 3D Scanning. In: Proceedings of the IEEE International Conference on Computational Photography (ICCP) (2015)
16. Milford, M., Kim, H., Leutenegger, S., Davison, A.J.: Towards visual SLAM with event-based cameras. In: The Problem of Mobile Sensors: Setting future goals and indicators of progress for SLAM Workshop in conjunction with Robotics: Science and Systems (RSS) (2015)
17. Milford, M., Kim, H., Mangan, M., Leutenegger, S., Stone, T., Webb, B., Davison, A.J.: Place recognition with event-based cameras and a neural implementation of

SeqSLAM. In: The Innovative Sensing for Robotics: Focus on Neuromorphic Sensors workshop at the IEEE International Conference on Robotics and Automation (ICRA) (2015)

18. Möller, T., Trumbore, B.: Fast , Minimum Storage Ray / Triangle Intersection. Journal of Graphics Tools 2(1), 21–28 (1997)
19. Mueggler, E., Huber, B., Scaramuzza, D.: Event-based , 6-DOF Pose Tracking for High-Speed Maneuvers. In: Proceedings of the IEEE/RSJ Conference on Intelligent Robots and Systems (IROS) (2014)
20. Newcombe, R.A., Lovegrove, S., Davison, A.J.: DTAM: Dense Tracking and Mapping in Real-Time. In: Proceedings of the International Conference on Computer Vision (ICCV) (2011)
21. Posch, C., Matolin, D., Wohlgenannt, R.: A QVGA 143 dB Dynamic Range Frame-Free PWM Image Sensor With Lossless Pixel-Level Video Compression and Time-Domain CDS. IEEE Journal of Solid-State Circuits (JSSC) (2011)
22. Schraml, S., Belbachir, A.N., Bischof, H.: Event-Driven Stereo Matching for Real-Time 3D Panoramic Vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015)
23. Weikersdorfer, D., Adrian, D.B., Cremers, D., Conradt, J.: Event-based 3D SLAM with a depth-augmented dynamic vision sensor. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA) (2014)
24. Weikersdorfer, D., Hoffmann, R., Conradt, J.: Simultaneous Localization and Mapping for event-based Vision Systems. In: International Conference on Computer Vision Systems (ICVS) (2013)