Event-Based, 6-DOF Camera Tracking from Photometric Depth Maps

— Source link ⧉

Guillermo Gallego, Jon E. A. Lund, Elias Mueggler, Henri Rebecq ...+2 more authors

Institutions: University of Zurich

Topics: Depth map, Image sensor, Motion blur, Video game and Pose

Related papers:

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

- A 128 $\times$ 128 120 dB 15 μ s Latency Asynchronous Temporal Contrast Vision Sensor

- Ultimate SLAM? Combining Events, Images, and IMU for Robust Visual SLAM in HDR and High-Speed Scenarios

- The event-camera dataset and simulator: Event-based data for pose estimation, visual odometry, and SLAM:

- A Unifying Contrast Maximization Framework for Event Cameras, with Applications to Motion, Depth, and Optical Flow Estimation

Year: 2017

Event-based, 6-DOF Camera Tracking from Photometric Depth Maps

Gallego, Guillermo ; Lund, Jon E A ; Müggler, Elias ; Rebecq, Henri ; Delbruck, Tobi ; Scaramuzza, Davide

Abstract: Event cameras are bio-inspired vision sensors that output pixel-level brightness changes instead of standard intensity frames. These cameras do not suffer from motion blur and have a very high dynamic range, which enables them to provide reliable visual information during high-speed motions or in scenes characterized by high dynamic range. These features, along with a very low power consumption, make event cameras an ideal complement to standard cameras for VR/AR and video game applications. With these applications in mind, this paper tackles the problem of accurate, low-latency tracking of an event camera from an existing photometric depth map (i.e., intensity plus depth information) built via classic dense reconstruction pipelines. Our approach tracks the 6-DOF pose of the event camera upon the arrival of each event, thus virtually eliminating latency. We successfully evaluate the method in both indoor and outdoor scenes and show that—because of the technological advantages of the event camera—our pipeline works in scenes characterized by high-speed motion, which are still unaccessible to standard cameras.

Event-based, 6-DOF Camera Tracking from Photometric Depth Maps

Guillermo Gallego, Jon E.A. Lund, Elias Mueggler, Henri Rebecq, Tobi Delbruck, Davide Scaramuzza

Abstract—Event cameras are bio-inspired vision sensors that output pixel-level brightness changes instead of standard intensity frames. These cameras do not suffer from motion blur and have a very high dynamic range, which enables them to provide reliable visual information during high-speed motions or in scenes characterized by high dynamic range. These features, along with a very low power consumption, make event cameras an ideal complement to standard cameras for VR/AR and video game applications. With these applications in mind, this paper tackles the problem of accurate, low-latency tracking of an event camera from an existing photometric depth map (i.e., intensity plus depth information) built via classic dense reconstruction pipelines. Our approach tracks the 6-DOF pose of the event camera upon the arrival of each event, thus virtually eliminating latency. We successfully evaluate the method in both indoor and outdoor scenes and show that—because of the technological advantages of the event camera—our pipeline works in scenes characterized by high-speed motion, which are still unaccessible to standard cameras.

Index Terms—Event-based vision, Pose tracking, Dynamic Vision Sensor, Bayes filter, Asynchronous processing, Conjugate priors, Low Latency, High Speed, AR/VR.

✦

SUPPLEMENTARY MATERIAL

Video of the experiments: https://youtu.be/iZZ77F-hwzs.

1 INTRODUCTION

THE task of estimating a sensor's ego-motion has important applications in various fields, such as augmented/virtual reality (AR/VR), video gaming, and autonomous mobile robotics. In recent years, great progress has been achieved using visual information to fulfill such a task [1], [2], [3]. However, due to some well-known limitations of traditional cameras (motion blur and low dynamic-range), current visual odometry pipelines still struggle to cope with high-speed motions or high dynamic range scenarios. Novel types of sensors, called event cameras [4, p.77], offer great potential to overcome these issues.

Unlike standard cameras, which transmit intensity frames at a fixed framerate, event cameras, such as the Dynamic Vision Sensor (DVS) [5], only transmit *changes of intensity*. Specifically, they transmit per-pixel intensity changes at the time they occur, in the form of a set of asynchronous *events*, where each event carries the space-time coordinates of the brightness change (with microsecond resolution) and its sign.

Event cameras have numerous advantages over standard cameras: a latency in the order of microseconds, a very high dynamic range (140 dB compared to 60 dB of standard cameras), and very low power consumption (10 mW vs 1.5 W of standard cameras). Most importantly, since all pixels capture light *independently*, such sensors do not suffer from motion blur.

It has been shown that event cameras transmit, in principle, all the information needed to reconstruct a full video stream

Fig. 1: Sample application: 6-DOF tracking in AR/VR (Augmented or Virtual Reality) scenarios. The pose of the event camera (rigidly attached to a hand or head tracker) is tracked from a previously built photometric depth map (RGB-D) of the scene. Positive and negative events are represented in blue and red, respectively, on the image plane of the event camera.

[6], [7], [8], [9], which clearly points out that an event camera alone is sufficient to perform 6-DOF state estimation and 3D reconstruction. Indeed, this has been recently shown in [9], [10]. However, currently the quality of the 3D map built using event cameras does not achieve the same level of detail and accuracy as that of standard cameras.

Although event cameras have become commercially available only since 2008 [11], the recent body of literature on these new sensors[1] as well as the recent plans for mass production claimed by companies, such as Samsung and Chronocam[2], highlight that

1. https://github.com/uzh-rpg/event-based_vision_resources
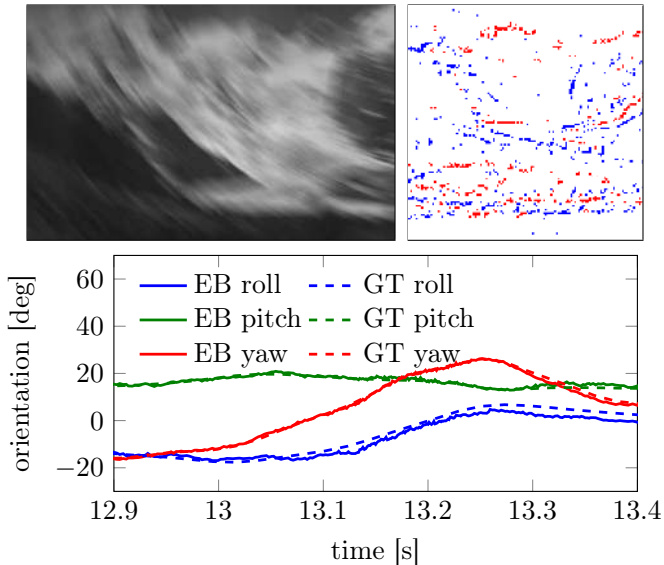2. http://rpg.ifi.uzh.ch/ICRA17_event_vision_workshop.html

Fig. 2: High-speed motion sequence. Top left: image from a standard camera, suffering from blur due to high-speed motion. Top right: set of asynchronous events from a DVS in an interval of 3 milliseconds, colored according to polarity. Bottom: estimated poses using our event-based (EB) approach, which provides low latency and high temporal resolution updates. Ground truth (GT) poses are also displayed.

there is a big commercial interest in exploiting these new vision sensors as an ideal complement to standard cameras for mobile robotics, VR/AR, and video game applications.

Motivated by these recent developments, this paper tackles the problem of tracking the 6-DOF motion of an event camera from an RGB-D (i.e., photometric depth) map that has been previously built via a traditional, dense reconstruction pipeline using standard cameras or RGB-D sensors (cf. Fig. 1). This problem is particularly important in both AR/VR and video game applications, where low-power consumption and robustness to high-speed motion are still unsolved. In these applications, we envision that the user would first use a standard sensor to build a high resolution and high quality map of the room, and then the hand and head trackers would take advantage of an event camera to achieve robustness to high-speed motion and low-power consumption.

The challenges we address in this paper are two: *i*) event-based 6-DOF pose tracking from an existing photometric depth map; *ii*) tracking the pose during very fast motions (still unaccessible to standard cameras because of motion blur), as shown in Fig. 2. We show that we can track the 6-DOF motion of the event camera with comparable accuracy as that of standard cameras and also during high-speed motion.

Our method is based on Bayesian filtering theory and has three key contributions in the way that the events are processed: *i*) event-based pose update, meaning that the 6-DOF pose estimate is updated every time an event is generated, at *micro*second time resolution, *ii*) the design of a sensor likelihood function using a mixture model that takes into account both the event generation process and the presence of noise and outliers (Section 4.3), and *iii*) the approximation of the posterior distribution of the system by a tractable distribution in the exponential family,

which is obtained by minimizing the Kullback-Leibler divergence (Section 4.4). The result is a filter adapted to the asynchronous nature of the event camera, which also incorporates an outlier detector that weighs measurements according to their confidence for improved robustness of the pose estimation. The approximation of the posterior distribution allows us to obtain a closed-form solution to the filter update equations and has the benefit of being computationally efficient. Our method can handle arbitrary, 6-DOF, high-speed motions of the event camera in natural scenes.

The paper is organized as follows: Section 2 reviews related literature on event-based ego-motion estimation. Section 3 describes the operating principle of event cameras. Our proposed event-based, probabilistic approach is described in Section 4, and it is empirically evaluated on natural scenes in Section 5. Conclusions are highlighted in Section 6.

2 RELATED WORK ON EVENT-BASED EGO-MOTION ESTIMATION

The first work on pose tracking with a DVS was presented in [12]. The system design, however, was limited to slow planar motions (i.e., 3 DOF) and planar scenes parallel to the plane of motion consisting of artificial B&W line patterns. The particle filter pose tracker was extended to 3D in [13], where it was used in combination with an external RGB-D sensor (depth estimation) to build a SLAM system. However, a depth sensor introduces the same bottlenecks that exist in standard frame-based systems: depth measurements are outdated for very fast motions, and the depth sensor is still susceptible to motion blur.

In our previous work [14], a standard grayscale camera was attached to a DVS to estimate the small displacement between the current event and the previous frame of the standard camera. The system was developed for planar motion and artificial B&W striped background. This was due to the sensor likelihood being proportional to the magnitude of the image gradient, thus favoring scenes where large brightness gradients are the source of most of the event data. Because of the reliance on a standard camera, the system was again susceptible to motion blur and therefore limited to slow motions.

An event-based algorithm to track the 6-DOF pose of a DVS alone and during very high-speed motion was presented in [15]. However, the method was developed specifically for artificial, B&W line-based maps. Indeed, the system worked by minimizing the point-to-line reprojection error.

Estimation of the 3D orientation of an event camera was presented in [6], [16], [17], [18]. However, such systems are restricted to rotational motions, and, thus, do not account for translation and depth.

Contrarily to all previous works, the approach we present in this paper tackles full 6-DOF motions, does not rely on external sensors, can handle arbitrary fast motions, and is not restricted to specific texture or artificial scenes.

Other pose tracking approaches have been published as part of systems that address the event-based 3D SLAM problem. [10] proposes a system with three interleaved probabilistic filters to perform pose tracking as well as depth and intensity estimation. The system is computationally intensive, requiring a GPU for real-time operation. The parallel tracking-and-mapping system in [9] follows a geometric, semi-dense approach. The pose tracker is based on edge-map alignment and the scene depth is estimated without intensity reconstruction, thus allowing the system to run

(a) The Dynamic Vision Sensor (DVS) from iniLabs.
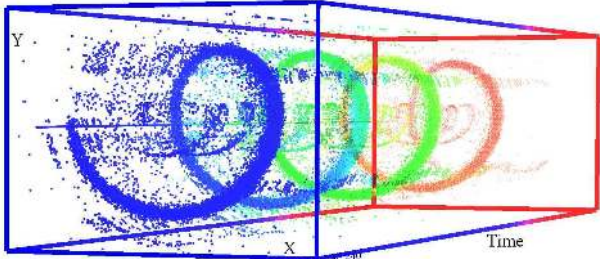
(b) Visualization of the output of a DVS (event stream) while viewing a rotating scene, which generates a spiral-like structure in space-time. Events are represented by colored dots, from red (far in time) to blue (close in time). Event polarity is not displayed. Noise is visible by isolated points.

Fig. 3: An event camera and its output.

in real-time on the CPU. More recently, visual inertial odometry systems based on event cameras have also been proposed, which rely on point features [19], [20], [21].

3 EVENT CAMERAS

Event-based vision constitutes a paradigm shift from conventional (e.g., frame-based) vision. In standard cameras, pixels are acquired and transmitted simultaneously at fixed rates; this is the case of both global-shutter or rolling-shutter sensors. Such sensors provide little information about the scene in the "blind time" between consecutive images. Instead, event-based cameras such as the DVS [11] (Fig. 3a) have independent pixels that respond asynchronously to relative contrast changes. If $I(\mathbf{u}, t)$ is the intensity sensed at a pixel $\mathbf{u} = (x, y)^\top$ of the DVS, an event is generated if the temporal visual contrast (in log scale) exceeds a nominal threshold C_{th}:

$$\Delta \ln I := \ln I(\mathbf{u}, t) - \ln I(\mathbf{u}, t - \Delta t) \gtrless C_{\text{th}}, \qquad (1)$$

where Δt is the time since the last event was generated at the same pixel. Different thresholds may be specified for the cases of contrast increase (C_{th}^+) or decrease (C_{th}^-). An event $e = (x, y, t, p)$ conveys the spatio-temporal coordinates and sign (i.e., polarity) of the brightness change, with $p = +1$ (ON-event: $\Delta \ln I > C_{\text{th}}^+$) or $p = -1$ (OFF-event: $\Delta \ln I < C_{\text{th}}^-$). Events are time-stamped with microsecond resolution and transmitted asynchronously when they occur, with very low latency. A sample output of the DVS is shown in Fig. 3b. Another advantage of the DVS is its very high dynamic range ($140\,\text{dB}$), which notably exceeds the $60\,\text{dB}$ of high-quality, conventional frame-based cameras. This is a consequence of events triggering on

log-intensity changes (1) instead of absolute intensity. The spatial resolution of the DVS is 128×128 pixels, but newer sensors, such as the Dynamic and Active-pixel VIsion Sensor (DAVIS) [22], the color DAVIS (C-DAVIS) [23], and the Samsung DVS [24] have higher resolution (640×480 pixels), thus overcoming current limitations.

4 PROBABILISTIC APPROACH

Consider an event camera moving in a known static scene. The map of the scene is described by a sparse set of reference images $\{I_l^r\}_{l=1}^{N_r}$, poses $\{\boldsymbol{\xi}_l^r\}_{l=1}^{N_r}$, and depth map(s). Suppose that an initial guess of the location of the event camera in the scene is also known. The problem we face is that of exploiting the information conveyed by the event stream to track the pose of the event camera in the scene. Our goal is to handle arbitrary 6-DOF, high-speed motions of the event camera in realistic (i.e., natural) scenes.

We design a robust filter combining the principles of Bayesian estimation, posterior approximation, and exponential family distributions with a sensor model that accounts for outlier observations. In addition to tracking the kinematic state of the event camera, the filter also estimates some sensor parameters automatically (e.g., event triggering threshold C_{th}) that would otherwise be difficult to tune manually. [3]

The outline of this section is as follows. First, the problem is formulated as a marginalized posterior estimation problem in a Bayesian framework. Then, the motion model and the measurement model (a robust likelihood function that can handle both good events and outliers) are presented. Finally, the filter equations that update the parameters of an approximate distribution to the posterior probability distribution are derived.

4.1 Bayesian Filtering

We model the problem as a time-evolving system whose state s consists of the kinematic description of the event camera as well as sensor and inlier/outlier parameters. More specifically,

$$s = (\boldsymbol{\xi}_c, \boldsymbol{\xi}_i, \boldsymbol{\xi}_j, C_{\text{th}}, \pi_m, \sigma_m^2)^\top, \qquad (2)$$

where $\boldsymbol{\xi}_c$ is the current pose of the sensor (at the time of the event, t in (1)), $\boldsymbol{\xi}_i$ and $\boldsymbol{\xi}_j$ are two poses along the sensor's trajectory that are used to interpolate the pose of the last event at the same pixel (time $t - \Delta t$ in (1)), C_{th} is the contrast threshold, and π_m and σ_m^2 are the inlier parameters of the sensor model, which is explained in Section 4.3.2.

Let the state of the system at time t_k be s_k, and let the sequence of all past observations (up to time t_k) be $o_{1:k}$, where o_k is the current observation (i.e., the latest event).

Our knowledge of the system state is contained in the posterior probability distribution $p(s_k|o_{1:k})$, also known as *belief* [25, p.27], which is the marginalized distribution of the smoothing problem $p(s_{1:k}|o_{1:k})$. The Bayes filter recursively estimates the system state from the observations in two steps: prediction and correction. The correction step updates the posterior by:

$$p(s_k|o_{1:k}) \propto p(o_k|s_k)p(s_k|o_{1:k-1}), \qquad (3)$$

3. Today's event-based cameras, such as the DVS [11] or the DAVIS [22], have almost a dozen tuning parameters that are neither independent nor linear.
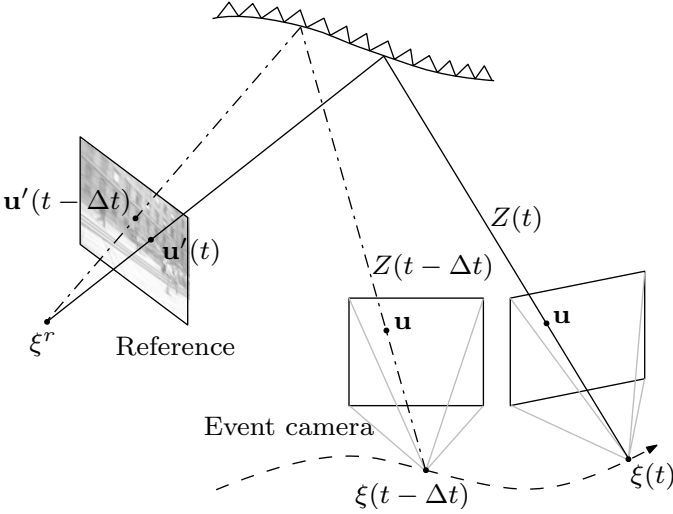
Fig. 4: Computation of the contrast (measurement function) by transferring events from the event camera to a reference image. For each event, the predicted contrast (13), $\Delta \ln I$, used in the measurement function (7) is computed as the log-intensity difference (as in (1)) at two points on the reference image I^r: the points (12) corresponding to the same pixel $\mathbf{u}$ on the event camera, at times of the event (t_k and $t_k - \Delta t$).

where $p(o_k|s_k)$ is the likelihood function (sensor model) and we used independence of the events given the state. The prediction step, defined by

$$p(s_k|o_{1:k-1}) = \int p(s_k|s_{k-1})p(s_{k-1}|o_{1:k-1})ds_{k-1}, \quad (4)$$

incorporates the motion model $p(s_k|s_{k-1})$ from t_{k-1} to t_k.

We incorporate in our state vector not only the current event camera pose $\boldsymbol{\xi}_c^k$ but also the other relevant poses for contrast calculation (poses $\boldsymbol{\xi}_i^k, \boldsymbol{\xi}_j^k$ in (2)), so that we may use the filter to partially correct errors of already estimated poses. Past events that are affected by the previous pose are not re-evaluated, but future events that reference back to such time will have better previous-pose estimates.

To have a computationally feasible filter, we approximate the posterior (3) by a tractable distribution with parameters η_{k-1} that condense the history of events $o_{1:k-1}$,

$$p(s_k|o_{1:k}) \approx q(s_k; \eta_k). \quad (5)$$

Assuming a motion model with slowly varying zero-mean random diffusion, so that most updates of the state are due to the events, the recursion on the approximate posterior becomes, combining (3)-(5),

$$q(s_k; \eta_k) \approx C\, p(o_k|s_k) q(s_k; \eta_{k-1}) \quad (6)$$

for some normalizing constant C. The approximate posterior q is computed by minimization of the Kullback-Leibler (KL) divergence between both sides of (6). As tractable distribution we choose one in the exponential family because they are very flexible and have nice properties for sequential Bayes estimation. The KL minimization gives the update equations for the parameters of the approximate posterior.

4.2 Motion Model

The diffusion process leaves the state mean unchanged and propagates the covariance. How much process noise is added to the evolving state is determined by the trace of the covariance matrix (sum of the eigenvalues): each incoming event adds white noise to the covariance diagonal, thus increasing its trace, up to some allowed maximum. This works gracefully across many motion speeds. More specifically, we used a maximum standard deviation of 0.03 for poses parametrized in normalized coordinates (with translation in units relative to the mean scene depth), to factor out the metric scale in the diffusion process.

4.3 Measurement Model

Here we elaborate on the choice of likelihood function $p(o_k|s_k)$ in (6) that is used to model the events. Our contributions are, starting from an ideal sensor model, i) to define a dimensionless implicit function based on the contrast residual to measure how well the event camera pose and the a priori information (e.g., a map of the scene) explain an event (Section 4.3.1), and ii) to build upon such measurement function taking into account noise and outliers, yielding a mixture model for the likelihood function (Section 4.3.2).

4.3.1 Ideal Sensor Model

In a noise-free scenario, an event is triggered as soon as the temporal contrast reaches the threshold (1). Such a measurement would satisfy $\Delta \ln I - C_{\text{th}} = 0$. For simplicity, let us assume that the polarity has already been taken into account to select the appropriate threshold $C_{\text{th}}^+ > 0$ or $C_{\text{th}}^- < 0$. Defining the measurement function by

$$M := \frac{\Delta \ln I}{C_{\text{th}}} - 1, \quad (7)$$

the event-generation condition becomes $M = 0$ in a dimensionless formulation. Assuming a prediction of the temporal contrast is generated using the system state, $\Delta \ln I(s_k)$, then (7) depends on both the system state and the observation, $M(o_k, s_k)$. More precisely, denoting by

$$\tilde{s} = (\boldsymbol{\xi}_c, \boldsymbol{\xi}_i, \boldsymbol{\xi}_j, C_{\text{th}})^\top, \quad (8)$$

the part of the state (2) needed to compute (7), we have $M(o_k, \tilde{s}_k)$. The likelihood function that characterizes such an ideal sensor model is

$$p(o_k|s_k) = \delta(M(o_k, \tilde{s}_k)), \quad (9)$$

where δ is the Dirac delta distribution.

All deviations from ideal conditions can be collectively modeled by a noise term in the likelihood function. Hence, a more realistic yet simple choice than (9) that is also supported by the bell-shaped form of the threshold variations observed in the DVS [11] is a Gaussian distribution,

$$p(o_k|s_k) = \mathcal{N}(M(o_k, \tilde{s}_k); 0, \sigma_m^2). \quad (10)$$

Most previous works in the literature do not consider an implicit measurement function (7) or Gaussian model (10) based on the contrast residual. Instead, they use explicit measurement functions that evaluate the goodness of fit of the event either in the spatial domain (reprojection error) [12], [15] or in the temporal domain (event-rate error), e.g., image reconstruction thread of [16], assuming Gaussian errors. Our measurement function (7)

is based on the event-generation process and combines in a scalar quantity all the information contained in an event (space-time and polarity) to provide a measure of its fit to a given state and a priori information. However, models based on a single Gaussian distribution (10) are very susceptible to outliers. Therefore, we opt for a mixture model to explicitly account for them, as explained next.

4.3.2 Resilient Sensor Model. Likelihood Function

Based on the empirical observation that there is a significant amount of outliers in the event stream, we propose a likelihood function consisting of a normal-uniform mixture model. This model is typical of robust sensor fusion problems [26], where the output of the sensor is modeled as a distribution that mixes a good measurement (normal) with a bad one (uniform):

$$p(o_k|s_k) = \pi_m \mathcal{N}(M(o_k, \tilde{s}_k); 0, \sigma_m^2) \qquad (11)$$
$$+ (1 - \pi_m) \mathcal{U}(M(o_k, \tilde{s}_k); M_{\min}, M_{\max}),$$

where π_m is the inlier probability (and $(1 - \pi_m)$ is the outlier probability). Inliers are normally distributed around 0 with variance σ_m^2. Outliers are uniformly distributed over a known interval $[M_{\min}, M_{\max}]$. The measurement parameters σ_m^2 and π_m are considered unknown and are collected in the state vector s_k to be estimated.

To evaluate $M(o_k, \tilde{s}_k)$, we need to compute the contrast $\Delta \ln I(\tilde{s}_k)$ in (7). We do so based on a known reference image I^r (and its pose) and both relevant event camera poses for contrast calculation, as explained in Fig. 4. Assuming the depth of the scene is known, the point $\mathbf{u}'$ in the reference image corresponding to the event location $(\mathbf{u}, t)$ in the event camera satisfies the following equation (in calibrated camera coordinates):

$$\mathbf{u}'(t) = \pi\big(T_{RC}(t)\,\pi^{-1}\big(\mathbf{u}, Z(t)\big)\big), \qquad (12)$$

where $T_{RC}(t)$ is the transformation from the event camera frame at time t to the frame of the reference image, $Z(t)$ represents the scene structure (i.e., the depth of the map point corresponding to $\mathbf{u}$ with respect to the event camera), $\pi : \mathbb{R}^3 \to \mathbb{R}^2$, $(X, Y, Z) \mapsto (X/Z, Y/Z)$ is the canonical perspective projection, and π^{-1} is the inverse perspective projection. The transformation $T_{RC}(t_k)$ at the time of the current event depends on the current estimate of the event camera pose $\boldsymbol{\xi}_c \equiv \boldsymbol{\xi}(t_k)$ in (8); the poses $\boldsymbol{\xi}_i \equiv \boldsymbol{\xi}(t_i)$ and $\boldsymbol{\xi}_j \equiv \boldsymbol{\xi}(t_j)$ along the event camera trajectory $\boldsymbol{\xi}(t)$ enclosing the past timestamp $t_k - \Delta t$ are used to interpolate the pose $\boldsymbol{\xi}(t - \Delta t)$, which determines $T_{RC}(t_k - \Delta t)$. For simplicity, separate linear interpolations for position and rotation parameters (exponential coordinates) are used, although a Lie Group formulation with the $SE(3)$ exponential and logarithm maps (more computationally expensive) could be used.

Once the corresponding points of the event coordinates $(\mathbf{u}, t_k)$ and $(\mathbf{u}, t_k - \Delta t)$ have been computed, we use their intensity values on the reference image I^r to approximate the contrast:

$$\Delta \ln I \approx \ln I^r(\mathbf{u}'(t_k)) - \ln I^r(\mathbf{u}'(t_k - \Delta t)), \qquad (13)$$

where t_k is the time of the current event and Δt is the time since the last event at the same pixel. This approach is more accurate than linearizing $\Delta \ln I$. We assume that for a small pose change there is a relatively large number of events from different pixels. In this case, the information contribution of a new event to an old pose will be negligible, and the new event will mostly contribute to the most recent pose.

Next, we linearize the measurement function in (11) around the expected state $\bar{s}_k = E_{p(s_k|o_{1:k-1})}[s_k]$, prior to incorporating the measurement correction:

$$M(o_k, \tilde{s}_k) \approx M(o_k, \bar{\tilde{s}}_k) + \nabla_{\tilde{s}} M(o_k, \bar{\tilde{s}}_k) \cdot (\tilde{s}_k - \bar{\tilde{s}}_k)$$
$$= \bar{M}_k + J_k \cdot \Delta \tilde{s}_k, \qquad (14)$$

where $\bar{M}_k$ and J_k are the predicted measurement and Jacobian at $\bar{s}_k$, respectively. Substituting (14) in (11) we get:

$$p(o_k|s_k) = \pi_m \mathcal{N}(\bar{M}_k + J_k \cdot \Delta \tilde{s}_k; 0, \sigma_m^2) + (1 - \pi_m)\mathcal{U}. \quad (15)$$

We assume that the linearization is a good approximation to the original measurement function.

Finally, we may re-write the likelihood (15) in a more general and convenient form for deriving the filter equations, as a sum of exponential families for the state parameters s_k (see the Appendix):

$$p(o_k|s_k) = \sum_j h(s_k) \exp(\eta_{o,j} \cdot T(s_k) - A_{o,j}). \qquad (16)$$

4.4 Posterior Approximation and Filter Equations

Our third contribution pertains to the approximation of the posterior distribution using a tractable distribution. For this, we consider variational inference theory [27], and choose a distribution in the exponential family as well as conjugate priors, minimizing the relative entropy error in representing the true posterior distribution with our approximate distribution, as we explain next.

Exponential families of distributions are useful in Bayesian estimation because they have *conjugate priors* [27]: if a given distribution is multiplied by a suitable prior, the resulting posterior has the same form as the prior. Such a prior is called a conjugate prior for the given distribution. The prior of a distribution in the exponential family is also in the exponential family, which clearly simplifies recursion. A mixture distribution like (16) does not, however, have a conjugate prior: the product of the likelihood and a prior from the exponential family is not in the family. Instead, the number of terms of the posterior doubles for each new measurement, making it unmanageable. Nevertheless, for tractability and flexibility, we choose as conjugate prior a distribution in the exponential family and approximate the product, in the sense of the Kullback-Leibler (KL) divergence [28], by a distribution of the same form, as expressed by (6). This choice of prior is optimal if either the uniform or the normal terms of the likelihood dominates the mixture; we expect that small deviations from this still gives good approximations.

Letting the KL divergence (or relative entropy) from a distribution f to a distribution g be

$$D_{\mathrm{KL}}(f\|g) = \int f(x) \ln \frac{f(x)}{g(x)} dx, \qquad (17)$$

which measures the information loss in representing distribution f by means of g, the posterior parameters η_k are calculated by minimization of the KL divergence from the distribution on the right hand side of (6) to the approximating posterior (left hand side of (6)):

$$\eta_k = \arg \min_\eta D_{\mathrm{KL}}\big(C\, p(o_k|s_k) q(s_k; \eta_{k-1}) \| q(s_k; \eta)\big).$$

It can be shown [27, p.505] that for g in the exponential family, the necessary optimality condition $\nabla_\eta D_{\mathrm{KL}}(f\|g) = 0$ gives the system of equations (in η)

$$E_{f(s)}[T(s)] = E_{g(s)}[T(s)], \qquad (18)$$

Algorithm 1 Event-based pose tracking

Initialize state variables (event camera pose, contrast threshold, inlier ratio). Then, for each incoming event:
- propagate state covariance (zero-mean random diffusion)
- transfer the event to the map, compute the depth and evaluate the measurement function M function (14).
- compute K_k in (20), the inlier probability π_m, the weight w_k in (21), and the gain $w_k K_k$.
- update filter variables and covariance (e.g., (22)-(23)).

i.e., the expected sufficient statistics must match. Additionally, the right hand side of (18) is $\nabla A \equiv \nabla_\eta A = E_{g(s)}[T(s)]$ since g is in the exponential family. In our case, $g \equiv q(s_k; \eta)$, $f \propto p(o_k|s_k)q(s_k; \eta_{k-1})$ and (18) can also be written in terms of the parameters of (16) [(3)-(6) in the Appendix], the log-normalizer A and its gradient:

$$0 = \sum_j \exp\big(A(\eta_{o,j} + \eta_{k-1}) - A(\eta_{k-1}) - A_{o,j})\big)$$
$$\times \big(\nabla A(\eta_{o,j} + \eta_{k-1}) - \nabla A(\eta)\big). \quad (19)$$

Equation (19) describes a system of equations that can be solved for η, yielding the update formula for η_k in terms of η_{k-1} and the current event o_k. For a multivariate Gaussian distribution over the event camera poses, explicit calculation of all update rules has the simple form of an Extended Kalman Filter (EKF) [25], [29] weighted by the inlier probability of that event:

$$K_k = P_k J_k^\top \big(J_k P_k J_k^\top + \sigma_m^2\big)^{-1} \quad (20)$$

$$w_k = \frac{\pi_m \mathcal{N}(\bar{M}_k; 0, \sigma_m^2)}{\pi_m \mathcal{N}(\bar{M}_k; 0, \sigma_m^2) + (1 - \pi_m)\mathcal{U}} \quad (21)$$

$$\boldsymbol{\xi}_{k+1} = \boldsymbol{\xi}_k + w_k K_k \bar{M}_k \quad (22)$$

$$P_{k+1} = (\mathbb{1} - w_k K_k J_k) P_k, \quad (23)$$

where $\mathbb{1}$ is the identity, $\bar{M}_k$ and J_k are given in (14), $\boldsymbol{\xi}$ are the 6-DOF coordinates (3 for translation and 3 for rotation) of the event camera pose, P is the pose covariance matrix, and $w_k K_k$ acts as the Kalman gain. A pseudocode of the approach is outlined in Algorithm 1.

The posterior approximation described in this section allows us to fuse the measurements and update the state-vector efficiently, without keeping multiple hypothesis in the style of particle filters, which would quickly become intractable due to the dimension of the state-vector.

5 EXPERIMENTAL RESULTS

Our event-based pose estimation algorithm requires an existing photometric depth map of the scene. As mentioned at the beginning of Section 4, without loss of generality we describe the map in terms of depth maps with associated reference frames. These can be obtained from a previous mapping stage by means of an RGB-D camera or by classical dense reconstruction approaches using standard cameras (e.g., DTAM [30] or REMODE [31]), RGB-D sensors [32], or even using an event camera (future research). In this work we use an Intel Realsense R200 RGB-D camera. We show experiments with both nearly planar scenes and scenes with large depth variations.

We evaluated the performance of our algorithm on several indoor and outdoor sequences. The datasets also contain fast motion with excitations in all six degrees of freedom (DOF). For the interested reader, we would like to point out that sequences similar to the ones used in these experiments can be found in the publicly available Event Camera Dataset [33].

Indoor Experiments

First, we assessed the accuracy of our method against ground truth obtained by a motion-capture system. We placed the event camera in front of a scene consisting of rocks (Fig. 5) at a mean scene depth of 60 cm and recorded eight sequences. Fig. 5 shows the position and orientation errors (i.e., difference between the estimated ones and ground truth)[4] for one of the sequences, while Fig. 10 shows the actual values of the estimated trajectory and ground truth over time. Fig. 6 summarizes the errors of the estimated trajectories for all sequences. The mean RMS errors in position and orientation are 1.63 cm and $2.21°$, respectively, while the mean and standard deviations of the position and orientation errors are $\mu = 1.38$ cm, $\sigma = 0.84$ cm, and $\mu = 1.89°$, $\sigma = 1.15°$, respectively. Notice that the RMS position error corresponds to 2.71% of the average scene depth, which is very good despite the poor spatial resolution of the DVS.

Outdoor Experiments

For the outdoor experiments, we used structure from motion from a standard camera as ground truth, more specifically we used SVO [35]. To this end, we rigidly mounted the DVS and a standard camera on a rig (see Fig. 7), and the same lens model was mounted on both sensors. The DVS has a spatial resolution of 128×128 pixels and operates asynchronously, in the microsecond scale. The standard camera is a global shutter MatrixVision Bluefox camera with a resolution of 752×480 pixels and a frame rate of up to 90 Hz. Both camera and DVS were calibrated intrinsically and extrinsically. For reference, we measured the accuracy of the frame-based method against the motion-capture system, in the same sequences previously mentioned (`rocks`, as in Fig. 10). The average RMS errors in position and orientation are 1.08 cm (i.e., 1.8% of the mean scene depth) and $1.04°$, respectively. Comparing these values to those of the event-based method, we note that, in spite of the limited resolution of the DVS, the accuracy of the results provided by our event-based algorithm is only slightly worse (2.71% vs. 1.8% in position, and $2.21°$ vs. $1.04°$ in orientation) than that obtained by a standard camera processing $20\times$ higher resolution images. This is made possible by the DVS temporal resolution being ten thousand times larger than the standard camera.

The three outdoor sequences (`ivy`, `graffiti`, and `building`) were recorded with the DVS-plus-camera rig viewing an ivy, a graffiti covered by some plants, and a building with people moving in front of it, respectively (see Fig. 8, 1st column and accompanying video submission). The rig was moved by hand with increasing speed. All sequences exhibit significant translational and rotational motion. The error plots in position and orientation of all 6-DOFs are given in Fig. 8. The reported error peaks in the `graffiti` and `building` sequences are due to a decrease of overlap between the event camera frustum and the reference map, thus making pose estimation ambiguous for some motions (e.g., Y-translation vs. X-rotation).

Table 1 summarizes the statistics of the pose tracking error for the three outdoor sequences. For the `ivy` dataset, the mean and

---

4. The rotation error is measured using the angle of their relative rotation (i.e., geodesic distance in $SO(3)$ [34]).
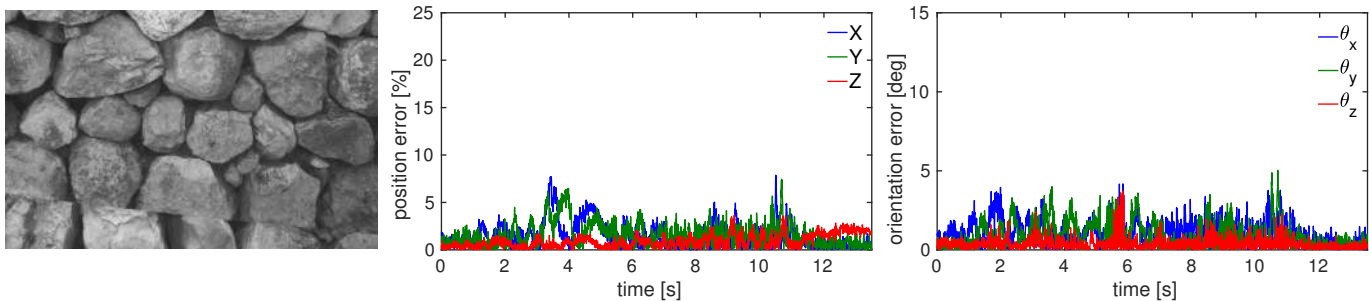
Fig. 5: Error plots in position (relative to a mean scene depth of $60\,\mathrm{cm}$) and in orientation (in degrees) for one of the test sequences with ground truth provided by a motion capture system with sub-millimeter accuracy.
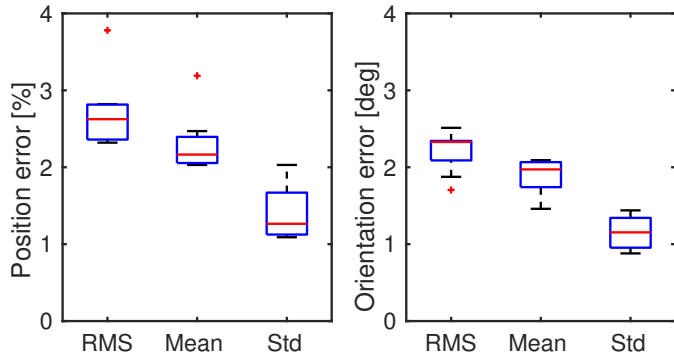

Fig. 6: Error in position (relative to a mean scene depth of $60\,\mathrm{cm}$) and orientation (in degrees) of the trajectories recovered by our method for *all* `rocks` sequences (ground truth is given by a motion capture system). We provide box plots of the root-mean-square (RMS) errors, the mean errors and the standard deviation (Std) of the errors.

Fig. 7: An event camera (DVS) and a standard camera mounted on a rig. The standard camera was only used for comparison.

standard deviation of the position error are $9.93\,\mathrm{cm}$ and $4.60\,\mathrm{cm}$,

TABLE 1: Error measurements of three outdoor sequences. Translation errors are relative (i.e., scaled by the mean scene depth).

	Position error [%]			Orientation error [°]		
	RMS	μ	σ	RMS	μ	σ
ivy	4.37	3.97	1.84	2.21	2.00	0.94
graffiti	5.88	5.23	2.70	3.58	3.09	1.80
building	7.40	6.47	3.60	3.99	3.43	2.05

which correspond to $3.97\,\%$ and $1.84\,\%$ of the average scene depth ($2.5\,\mathrm{m}$), respectively. The mean and standard deviation of the orientation error are $2.0°$ and $0.94°$, respectively. For the `building` dataset, which presents the largest errors, the mean and standard deviation of the orientation error are $3.43°$ and $2.05°$, respectively, while, in position error, the corresponding figures are $1.94\,\mathrm{m}$ and $1.08\,\mathrm{m}$, that correspond to $6.47\,\%$ and $3.60\,\%$ of the average scene depth ($30\,\mathrm{m}$), respectively.

As reported by the small errors in Table 1, overall our event-based algorithm is able to accurately track the pose of the event camera also outdoors. We expect that the results provided by our approach would be even more accurate with the next generation of event-based sensors currently being developed [22], [23], which will have higher spatial resolution (640×480 pixels). Finally, observe that in the `building` sequence (Fig. 8, bottom row), our method gracefully tracks the pose in spite of the considerable amount of events generated by moving objects (e.g., people) in the scene (see Fig. 9).

5.1 Tracking during High-Speed Motions

In addition to the error plots in Fig. 5, we show in Fig. 10 the actual values of the trajectories (position and orientation) acquired by the motion capture system (dashed line) and estimated by the event-based method (solid line) and the frame-based method (dash-dot). Notice that they are all are almost indistinguishable relative to the amplitude of the motion excitation, which gives a better appreciation of the small errors reported in Figs. 5 and 6.

Figure 11 shows a magnified version of the estimated trajectories during high-speed motions (occurring at $t \geq 7\,\mathrm{s}$ in Fig. 10). The frame-based method is able to track in the shaded region, up to $t \approx 8.66\,\mathrm{s}$ (indicated by a vertical dashed line), at which point it loses tracking due to motion blur, while our event-based method continues to accurately estimate the pose.

5.2 Experiments with Large Depth Variation

In the following set of experiments, we also assessed the accuracy of our method on scenes with large depth variation and, therefore larger parallax than in previous experiments. We recorded seven sequences with ground truth from a motion-capture system of a scene consisting of a set of textured `boxes` (Fig. 12, top row). We also recorded two outdoor sequences: `pipe` and `bicycles` (middle and bottom rows of Fig. 12). The `pipe` sequence depicts a creek going through a pipe, surrounded by rocks and grass; the `bicycle` sequence depicts some parked bicycles next to a building; both outdoor scenes present some occlusions. All sequences exhibit significant translational and rotational motion.
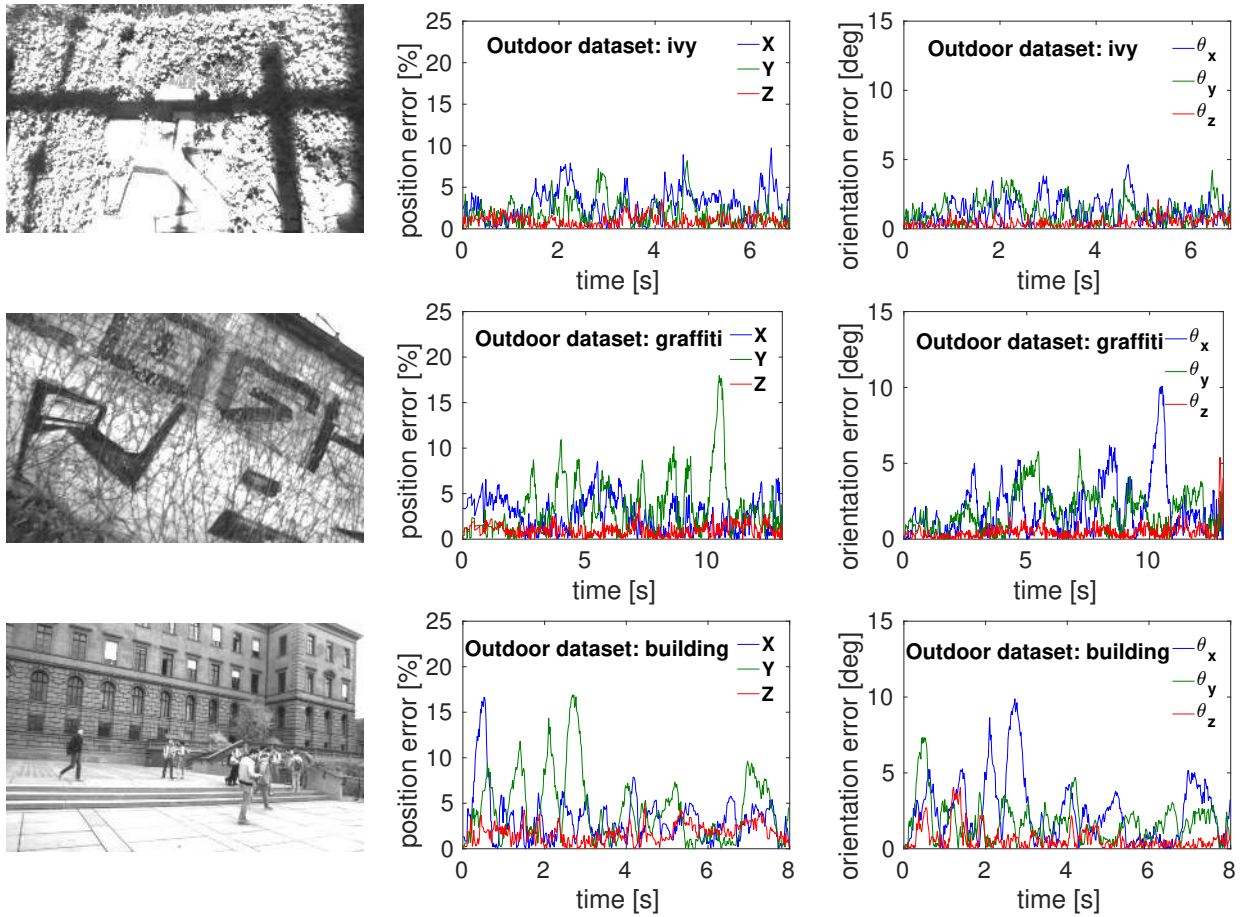
Fig. 8: Error plots in position (2nd column, relative to the mean scene depth) and in orientation (3rd column, in degrees) for three outdoor test sequences (1st column): `ivy`, `graffiti`, and `building`. The mean scene depths are $2.5\,\mathrm{m}$, $3\,\mathrm{m}$, and $30\,\mathrm{m}$, respectively.
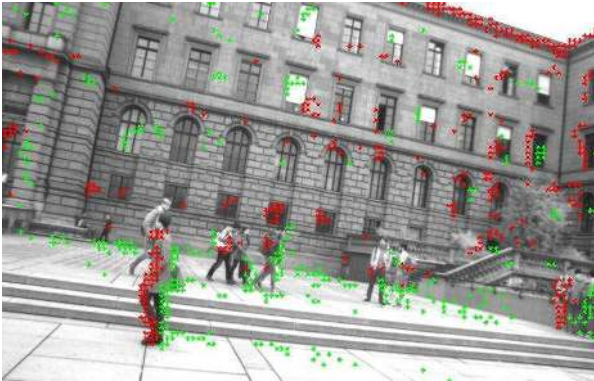

Fig. 9: The algorithm is able to track the pose of the event camera in spite of the considerable amount of events generated by moving objects (e.g., people) in the scene.

Fig. 13 summarizes the position and orientation error statistics of our algorithm on the `boxes` sequences (compared with ground truth from the motion-capture system). The position error is given relative to the mean scene depth, which is $1.9\,\mathrm{m}$. As it is observed, the errors are very similar to those in Fig. 6, meaning that our pose tracking method can handle arbitrary 3D scenes, i.e., not necessarily nearly planar.

Table 2 reports the numerical values of the trajectory errors

TABLE 2: Error measurements of the sequences in Fig. 12. Translation errors are relative (i.e., scaled by the mean scene depth).

	Position error [%]			Orientation error [°]		
	RMS	μ	σ	RMS	μ	σ
boxes	2.50	2.23	1.17	1.88	1.65	1.02
pipe	4.04	3.04	2.66	2.90	2.37	1.67
bicycles	2.14	1.724	1.27	1.46	1.19	0.84

in both indoors and outdoor sequences. The row corresponding to the `boxes` sequences is the average of the errors in the seven indoor sequences (Fig. 13). For the position error, the mean scene depths of the `pipe` and `bicycles` sequences are $2.7\,\mathrm{m}$ and $2.2\,\mathrm{m}$, respectively. The mean RMS errors in position and orientation are in the range $2.5\,\%$ to $4.0\,\%$ of the mean scene depth and $1.4°$ to $2.9°$, respectively, which are in agreement with the values in Table 1 for the scenes with mean depths smaller than $3\,\mathrm{m}$. It is remarkable that the method is able to track despite some lack of texture (as in the `pipe` sequence, where there are only few strong edges), and in the presence of occlusions, which are more evident in the `bicycles` sequence.

5.3 Computational Effort

We measured the computational cost of our algorithm on a single core of an Intel(R) i7 processor at $2.60\,\mathrm{GHz}$. The processing time
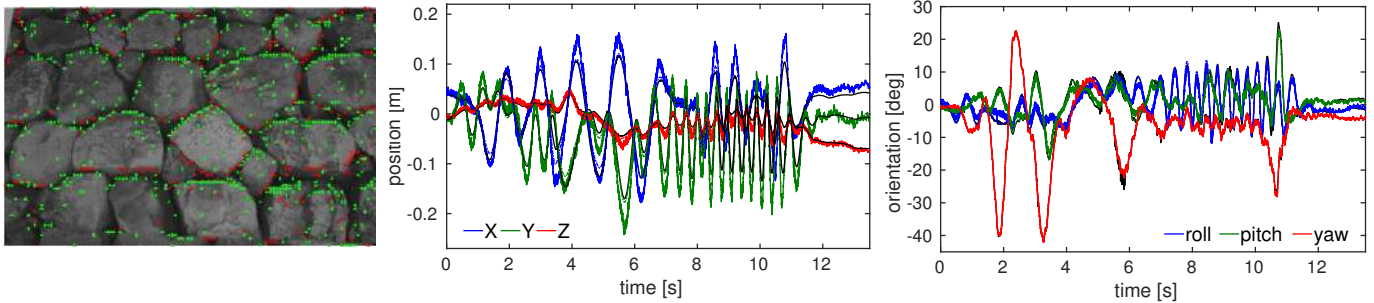
Fig. 10: Indoor experiment with 6-DOF motion. Left: Image of the standard camera overlaid with events (during mild motion). Events are displayed in red and green, according to polarity. Estimated position (center) and orientation (right) from our event-based algorithm (solid line), a frame-based method (dash-dot line) and ground truth (black line) from a motion capture system.
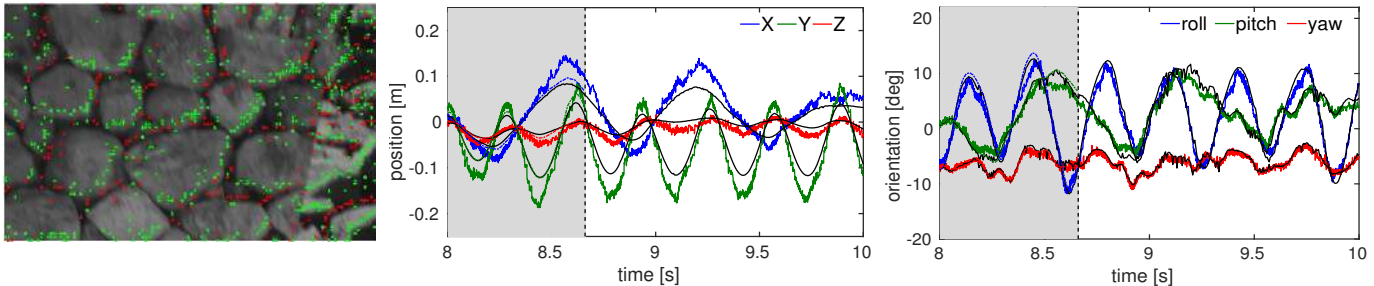

Fig. 11: Zoom of Fig. 10. Left: Image of the standard camera overlaid with events (red and green points, according to polarity) during high-speed motion. Center and right: estimated trajectories. Due to the very high temporal resolution, our algorithm can still track the motion even when the images of the standard camera are sufficiently blurred so that the frame-based method (FB) failed. The event-based method (EB) provides pose updates even in high-speed motions, whereas the frame-based method loses track (it only provides pose updates in the region marked with the shaded area, then it fails).

per event is $32\,\mu s$, resulting in a processing event rate of 31.000 events per second. Depending on the texture of the scene and the speed of motion, the data rate produced by an event camera ranges from tens of thousands (moderate motion) to over a million events per second (high-speed motion). However, notice that our implementation is not optimal; many computations can indeed still be optimized, cached, and parallelized to increase the runtime performance of the algorithm.

6 CONCLUSION

We have presented an approach to track the 6-DOF pose of an arbitrarily moving event camera from an existing photometric depth map in natural scenes. Our approach follows a Bayesian filtering methodology: the sensor model is given by a mixture-model likelihood that takes into account both the event-generation process and the presence of noise and outliers; the posterior distribution of the system state is approximated according to the relative-entropy criterion using distributions in the exponential family and conjugate priors. This yields a robust EKF-like filter that provides pose updates for every incoming event, at microsecond time resolution.

We have compared our method against ground truth from a motion capture system and a state-of-the-art frame-based pose-tracking pipeline. The experiments revealed that the proposed method accurately tracks the pose of the event-based camera, both in indoor and outdoor experiments in scenes with significant depth variations, and under motions with excitations in all 6-DOFs.

REFERENCES

[1] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM: a versatile and accurate monocular SLAM system," *IEEE Trans. Robot.*, vol. 31, no. 5, pp. 1147–1163, 2015. 1

[2] J. Engel, J. Schöps, and D. Cremers, "LSD-SLAM: Large-scale direct monocular SLAM," in *Eur. Conf. Comput. Vis. (ECCV)*, 2014. 1

[3] C. Forster, Z. Zhang, M. Gassner, M. Werlberger, and D. Scaramuzza, "SVO: Semidirect visual odometry for monocular and multicamera systems," *IEEE Trans. Robot.*, vol. 33, no. 2, pp. 249–265, 2017. 1

[4] A. N. Belbachir, *Smart Cameras.* Springer US, 2009. 1

[5] P. Lichtsteiner, C. Posch, and T. Delbruck, "A 128x128 120dB 30mW asynchronous vision sensor that responds to relative intensity change," in *IEEE Intl. Solid-State Circuits Conf. (ISSCC)*, Feb. 2006, pp. 2060–2069. 1

[6] M. Cook, L. Gugelmann, F. Jug, C. Krautz, and A. Steger, "Interacting maps for fast visual interpretation," in *Int. Joint Conf. Neural Netw. (IJCNN)*, 2011, pp. 770–776. 1, 2

[7] P. Bardow, A. J. Davison, and S. Leutenegger, "Simultaneous optical flow and intensity estimation from an event camera," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 2016. 1

[8] C. Reinbacher, G. Graber, and T. Pock, "Real-time intensity-image reconstruction for event cameras using manifold regularisation," in *British Machine Vis. Conf. (BMVC)*, 2016. 1

[9] H. Rebecq, T. Horstschäfer, G. Gallego, and D. Scaramuzza, "EVO: A geometric approach to event-based 6-DOF parallel tracking and mapping in real-time," *IEEE Robot. Autom. Lett.*, vol. 2, pp. 593–600, 2017. 1, 2

[10] H. Kim, S. Leutenegger, and A. J. Davison, "Real-time 3D reconstruction and 6-DoF tracking with an event camera," in *Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 349–364. 1, 2

[11] P. Lichtsteiner, C. Posch, and T. Delbruck, "A 128×128 120 dB 15 μs latency asynchronous temporal contrast vision sensor," *IEEE J. Solid-State Circuits*, vol. 43, no. 2, pp. 566–576, 2008. 1, 3, 4

[12] D. Weikersdorfer and J. Conradt, "Event-based particle filtering for robot self-localization," in *IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, 2012, pp. 866–870. 2, 4
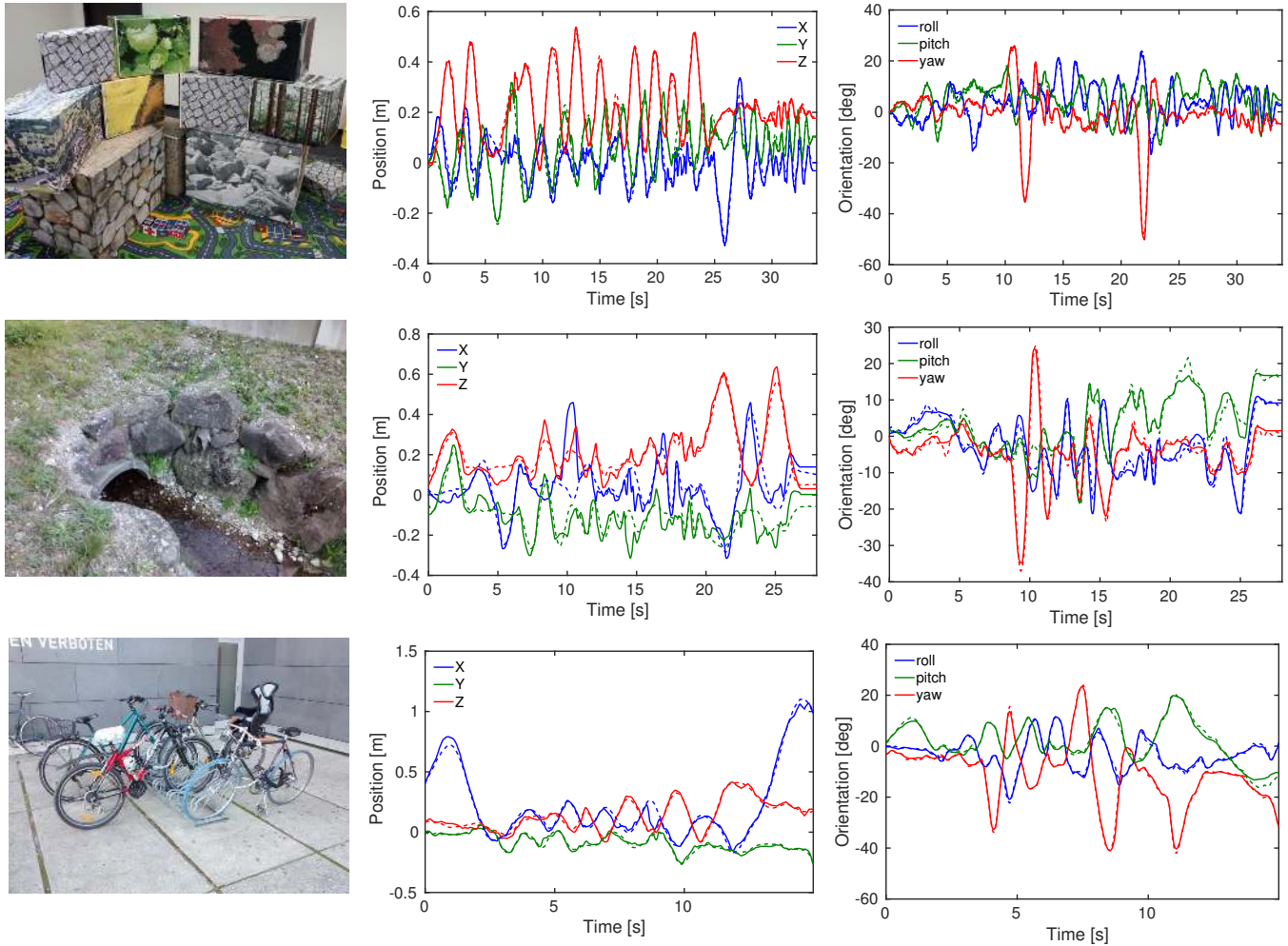
Fig. 12: Experiments on scenes with significant depth variation and occlusions. Scene impressions (1st column): `boxes`, `pipe`, and `bicycles`. Estimated position (2nd column, in meters) and orientation (3rd column, in degrees) from our event-based algorithm (solid line) compared with ground truth (dashed line). The mean scene depths are $1.8\,\text{m}$, $2.7\,\text{m}$, and $2.3\,\text{m}$, respectively.
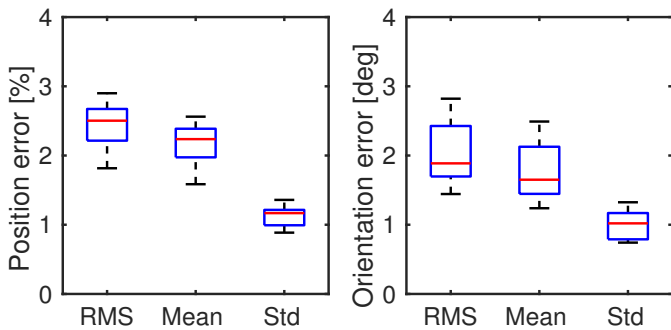

Fig. 13: Error in position (relative to a mean scene depth of $1.9\,\text{m}$) and orientation (in degrees) of the trajectories recovered by our method for *all* `boxes` sequences (ground truth is given by a motion-capture system). We provide box plots of the root-mean-square (RMS) errors, the mean errors and the standard deviation (Std) of the errors.

[13] D. Weikersdorfer, D. B. Adrian, D. Cremers, and J. Conradt, "Event-based 3D SLAM with a depth-augmented dynamic vision sensor," in *IEEE Int. Conf. Robot. Autom. (ICRA)*, Jun. 2014, pp. 359–364. 2

[14] A. Censi and D. Scaramuzza, "Low-latency event-based visual odometry," in *IEEE Int. Conf. Robot. Autom. (ICRA)*, 2014. 2

[15] E. Mueggler, B. Huber, and D. Scaramuzza, "Event-based, 6-DOF pose tracking for high-speed maneuvers," in *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, 2014, pp. 2761–2768. 2, 4

[16] H. Kim, A. Handa, R. Benosman, S.-H. Ieng, and A. J. Davison, "Simultaneous mosaicing and tracking with an event camera," in *British Machine Vis. Conf. (BMVC)*, 2014. 2, 4

[17] G. Gallego and D. Scaramuzza, "Accurate angular velocity estimation with an event camera," *IEEE Robot. Autom. Lett.*, vol. 2, pp. 632–639, 2017. 2

[18] C. Reinbacher, G. Munda, and T. Pock, "Real-time panoramic tracking for event cameras," in *IEEE Int. Conf. Comput. Photography (ICCP)*, 2017. 2

[19] E. Mueggler, G. Gallego, H. Rebecq, and D. Scaramuzza, "Continuous-time visual-inertial trajectory estimation with event cameras," 2017, `arXiv:1702.07389`. 3

[20] A. Zhu, N. Atanasov, and K. Daniilidis, "Event-based visual inertial odometry," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 2017. 3

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

[22] C. Brandli, R. Berner, M. Yang, S.-C. Liu, and T. Delbruck, "A 240x180 130dB 3us latency global shutter spatiotemporal vision sensor," *IEEE J. Solid-State Circuits*, vol. 49, no. 10, pp. 2333–2341, 2014. 3, 7

[23] C. Li, C. Brandli, R. Berner, H. Liu, M. Yang, S.-C. Liu, and T. Delbruck, "An RGBW color VGA rolling and global shutter dynamic and active-pixel vision sensor," in *International Image Sensor Workshop (IISW)*, Vaals, Netherlands, Jun. 2015. 3, 7

[24] B. Son, Y. Suh, S. Kim, H. Jung, J.-S. Kim, C. Shin, K. Park, K. Lee,

J. Park, J. Woo, Y. Roh, H. Lee, Y. Wang, I. Ovsiannikov, and H. Ryu, "A 640x480 dynamic vision sensor with a 9um pixel and 300Meps address-event representation," in *IEEE Intl. Solid-State Circuits Conf. (ISSCC)*, 2017. 3

[25] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. The MIT Press, Cambridge, MA, 2005. 3, 6

[26] G. Vogiatzis and C. Hernández, "Video-based, real-time multi view stereo," *Image Vis. Comput.*, vol. 29, no. 7, pp. 434–441, 2011. 5

[27] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer-Verlag New York, Inc., 2006. 5

[28] S. Kullback and R. A. Leibler, "On information and sufficiency," *Ann. Math. Statist.*, vol. 22, no. 1, pp. 79–86, 1951. 5

[29] R. Kalman, "A new approach to linear filtering and prediction problems," *J. Basic Eng.*, vol. 82, pp. 35–45, 1960. 6

[30] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, "DTAM: Dense tracking and mapping in real-time," in *Int. Conf. Comput. Vis. (ICCV)*, Nov. 2011, pp. 2320–2327. 6

[31] M. Pizzoli, C. Forster, and D. Scaramuzza, "REMODE: Probabilistic, monocular dense reconstruction in real time," in *IEEE Int. Conf. Robot. Autom. (ICRA)*, 2014, pp. 2609–2616. 6

[32] R. A. Newcombe, A. J. Davison, S. Izadi, P. Kohli, O. Hilliges, J. Shotton, D. Molyneaux, S. Hodges, D. Kim, and A. Fitzgibbon, "KinectFusion: Real-time dense surface mapping and tracking," in *IEEE ACM Int. Sym. Mixed and Augmented Reality (ISMAR)*, Basel, Switzerland, Oct. 2011, pp. 127–136. 6

[33] 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," *Int. J. Robot. Research*, vol. 36, pp. 142–149, 2017. 6

[34] D. Q. Huynh, "Metrics for 3D rotations: Comparison and analysis," *J. Math. Imaging Vis.*, vol. 35, no. 2, pp. 155–164, 2009. 6

[35] C. Forster, M. Pizzoli, and D. Scaramuzza, "SVO: Fast semi-direct monocular visual odometry," in *IEEE Int. Conf. Robot. Autom. (ICRA)*, 2014, pp. 15–22. 6

Elias Mueggler received his Ph.D. degree from the University of Zurich, Switzerland in 2017, where he was working at the Robotics and Perception Group, lead by Prof. Davide Scaramuzza, in the topics of event-based vision for high-speed robotics and air-ground robot collaboration. He received B.Sc. (2010) and M.Sc. (2012) degrees in Mechanical Engineering from ETH Zurich, Switzerland. He is a recipient of the KUKA Innovation Award (2014), the Qualcomm Innovation Fellowship (2016) and the Misha Mahowald Prize for Neuromorphic Engineering (2017). He has been a visiting researcher with Prof. John Leonard (Massachusetts Institute of Technology) and Dr. Chiara Bartolozzi (Istituto Italiano di Tecnologia). His research interests include computer vision and robotics.

Henri Rebecq is a Ph.D. student in the Robotics and Perception Group at the University of Zurich, where he is working on on event-based vision for robotics. In 2014, he received an M.Sc.Eng. degree from Télécom ParisTech, and an M.Sc. degree from Ecole Normale Supérieure de Cachan, both located in Paris, France. Prior to pursuing graduate studies, he worked as a research and software engineer at VideoStitch, Paris, France. His research interests include omnidirectional vision, visual odometry, 3D reconstruction and SLAM. He is a recipient of the Misha Mahowald Prize for Neuromorphic Engineering (2017).

Tobi Delbruck (M'99–SM'06–F'13) received the B.Sc. degree in physics and applied mathematics from the University of California, San Diego, CA, USA, and the Ph.D. degree from the California Institute of Technology, Pasadena, CA, USA, in 1986 and 1993, respectively. He has been a Professor of Physics with the Institute of Neuroinformatics, University of Zurich and ETH Zurich, Switzerland, since 1998. His group focuses on neuromorphic sensory processing. He worked on electronic imaging at Arithmos, Synaptics, National Semiconductor, and Foveon. Dr. Delbruck has co-organized the Telluride Neuromorphic Cognition Engineering summer workshop and the live demonstration sessions at International Symposium on Circuits and Systems. He is also co-founder of iniLabs and Insightness. He was the Chair of the IEEE CAS Sensory Systems Technical Committee, is current Secretary of the IEEE Swiss CAS/ED Society, and an Associate Editor of the IEEE Transactions on Biomedical Circuits and Systems. He has received 9 IEEE awards.

Guillermo Gallego received the Ph.D. degree in electrical and computer engineering from the Georgia Institute of Technology, Atlanta, GA, USA, in 2011. He received the Ingeniero de Telecomunicación degree (five-year engineering program) from the Universidad Politécnica de Madrid (UPM), Madrid, Spain, in 2004, the M.S. degree in mathematical engineering (Magíster en Ingeniería Matemática) from the Universidad Complutense de Madrid, Madrid, in 2005, and the M.S. degrees in electrical and computer engineering and mathematics from the Georgia Institute of Technology, Atlanta, GA, USA, in 2007 and 2009, respectively. From 2011 to 2014, he was a Marie Curie Post-Doctoral Researcher with the UPM. Since 2014, he has been with the Robotics and Perception Group, University of Zurich, Zurich, Switzerland. His current research interests include computer vision, signal processing, robotics, geometry, optimization, and ocean engineering. Dr. Gallego was a recipient of the Fulbright Scholarship to pursue graduate studies at the Georgia Institute of Technology in 2005. He is a recipient of the Misha Mahowald Prize for Neuromorphic Engineering (2017).

Davide Scaramuzza Davide Scaramuzza (born in 1980, Italian) is Associate Professor of Robotics and Perception at both departments of Informatics (University of Zurich) and Neuroinformatics (University of Zurich and ETH Zurich), where he does research at the intersection of robotics, computer vision, and neuroscience. He did his PhD in robotics and computer vision at ETH Zurich and a postdoc at the University of Pennsylvania. From 2009 to 2012, he led the European project sFly, which introduced the PX4 autopilot and pioneered visual-SLAM–based autonomous navigation of micro drones. For his research contributions, he was awarded the Misha Mahowald Neuromorphic Engineering Award, the IEEE Robotics and Automation Society Early Career Award, the SNSF-ERC Starting Grant, a Google Research Award, the European Young Research Award, and several conference paper awards. He coauthored the book Introduction to Autonomous Mobile Robots (published by MIT Press) and more than 100 papers on robotics and perception.

Jon E.A. Lund is a flight control and video engineer at Prox Dynamics/FLIR UAS in Oslo, Norway. He received his M.Sc. (2015) in Neural Systems and Computation from ETH and University of Zurich, Switzerland. Before that, he earned a B.Sc. (2013) in Physics at the University of Oslo, Norway.

Event-based, 6-DOF Camera Tracking from Photometric Depth Maps

Guillermo Gallego, Jon E.A. Lund, Elias Mueggler, Henri Rebecq, Tobi Delbruck, Davide Scaramuzza

◆

SUPPLEMENTARY MATERIAL

Video of the experiments: https://youtu.be/iZZ77F-hwzs.

APPENDIX A
REWRITING THE LIKELIHOOD FUNCTION

A distribution in the exponential family can be written as

$$p(x; \eta) = h(x) \exp(\eta \cdot T(x) - A(\eta)), \qquad (1)$$

where η are the natural parameters, $T(x)$ are the sufficient statistics of x, $A(\eta)$ is the log-normalizer, and $h(x)$ is the base measure.

The likelihood (15) can be rewritten as:

$$
\begin{aligned}
p(o_k|s_k) = & \frac{1}{\sqrt{2\pi}} \exp(\ln(\pi_m) - \ln(\sigma_m) \qquad (2) \\
& - \frac{1}{2}\left[J_k^i J_k^i \frac{\tilde{s}_k^i \tilde{s}_k^j}{\sigma_m^2} + 2\bar{M}_k J_k^i \frac{\tilde{s}_k^i}{\sigma_m^2} + \frac{\bar{M}_k^2}{\sigma_m^2} \right] \\
& + \exp(\ln((1-\pi_m)/(M_{\max} - M_{\min}))),
\end{aligned}
$$

where we use the Einstein summation convention for the indices of $J_k = (J_k^i)$ and $\tilde{s}_k = (\tilde{s}_k^i)$. Collecting the sufficient statistics into

$$T(s_k) = \left[\frac{\tilde{s}_k^i \tilde{s}_k^j}{\sigma_m^2}, \frac{\tilde{s}_k^i}{\sigma_m^2}, \frac{1}{\sigma_m^2}, \ln(\sigma_m), \ln(\pi_m), \ln(1-\pi_m) \right],$$

the likelihood can be conveniently rewritten as a sum of two exponential families (16), $j = 1, 2$, with $h(s) = 1$,

$$\eta_{o,1} = \left[-\frac{1}{2} J_k^i J_k^j, -\bar{M}_k J_k^i, -\frac{1}{2}\bar{M}_k^2, -1, 1, 0 \right] \qquad (3)$$

$$\eta_{o,2} = [0_{ij}, 0_i, 0, 0, 1] \qquad (4)$$

$$A_{o,1} = \ln\sqrt{2\pi} \qquad (5)$$

$$A_{o,2} = -\ln(M_{\max} - M_{\min}). \qquad (6)$$