Optical Flow Estimation Using the Fisher-Rao Metric

Abstract The optical flow in an event camera is estimated using measurements in the Address Event Representation (AER). Each measurement consists of a pixel address and the time at which a change in the pixel value equalled a given fixed threshold. The measurements in a small region of the pixel array and within a given window in time are approximated by a probability distribution defined on a finite set. The distributions obtained in this way form a three dimensional family parameterized by the pixel addresses and by time. Each parameter value has an associated Fisher-Rao matrix obtained from the Fisher-Rao metric for the parameterized family of distributions. The optical flow vector at a given pixel and at a given time is obtained from the eigenvector of the associated Fisher-Rao matrix with the least eigenvalue. The Fisher-Rao algorithm for estimating optical flow is tested on eight datasets, of which six have ground truth optical flow. It is shown that the Fisher-Rao algorithm performs well in comparison with two state of the art algorithms for estimating optical flow from AER measurements.

Keywords address event representation · AER · asynchronous image sensor · event camera · Fisher-Rao metric · Kullback-Leibler divergence · optical flow · OptiTrack motion capture

1 Introduction

The Address Event Representation (AER) [9, 27, 29] is a new paradigm in computer vision. Each pixel in an AER camera emits a signal when the change in the pixel value equals a fixed threshold. If the change in value is less than the threshold, then no signal is emitted. The pixels emit their signals asynchronously, i.e. without coordination. There is no concept of an image frame of pixel values all of which are obtained at the same instant in time [4]. Cameras which supply data in the AER format are referred to as event cameras or as silicon retinas. The latter term arises from an analogy between the AER and the human retina. The advantages of event cameras over conventional cameras are a low power consumption, a very rapid response to changes in the image and a high dynamic range [6]. Event based vision is surveyed in [17].

1.1 Notation for the AER

Each pixel emits a measurement when the value recorded by the pixel changes by an amount $\pm d$, where d is a fixed positive threshold. If the change in value is less than d in absolute value, then the pixel in question does not emit any measurement. The measurements emitted by a pixel q form a list

$$(q, s(i), \Delta(i)), \quad i = 1, 2, 3, \ldots \tag{1}$$

where $s(1), s(2), \ldots$ is an increasing sequence of times and $\Delta(i) \in \{0, 1\}$ is the polarity of event i. The component $\Delta(i)$ of the measurement specifies the sign of the change in the pixel value. If $\Delta(i) = 0$, then the value decreases by d and if $\Delta(i) = 1$, then the value increases by d.

The full list of measurements obtained from an event camera consists of the union of the lists of measurements (1) over all pixels of the sensor array. Event cameras have lower data rates and lower power consumption than conventional cameras because there is no wasteful output from pixels for which there is only a small or zero change in value.

The times $s(i)$ at which the measurements are emitted are known with a high accuracy. For example, in the event camera described by Benosman et al. [6], the errors in the times $s(i)$ are of the order of $350\mu s$ in regions where there is low to moderate texture variation. In regions with dense texture the errors in the times are larger because many pixels emit measurements simultaneously.

1.2 Optical flow

The changes in an image sequence over time are often modeled by a vector field, which is usually referred to as the optical flow [15, 36, 37]. The optical flow is represented by velocity vectors based at individual pixels in an image. The literature on optical flow in conventional image sequences is vast. Fortun et al. [15] provide a survey with 287 references. They list many applications including action recognition, video indexing and retrieval, video compression, video restoration, automated visual surveillance, estimation of crowd motions, pedestrian behaviour analysis, gesture recognition, facial expression recognition, automatic robot guidance, obstacle detection and avoidance, medical image registration, cell tracking, blood flow estimation, the measurement of organ deformations and fluid mechanics.

1.3 Method for estimating optical flow

A new method for estimating optical flow from event camera measurements is investigated. The patterns in the measurements obtained over a short interval of time are modelled by probability distributions which are obtained by normalizing local histograms. The result is a family of probability distributions parameterised by points (q, t) in $\mathbb{R}^3$, where q is a point in the pixel array and t is a time. If brightness constancy holds to a good approximation, as described in Section 3.3, then the distributions associated with a moving object are translates of one another in the parameter space $\mathbb{R}^3$. This translation is estimated by comparing probability distributions. The comparison is based on the Fisher-Rao metric [2, 14, 25] on the parameter space $\mathbb{R}^3$. The optical flow vector is obtained from the translation.

Experiments with the Fisher-Rao method were carried out on eight datasets. Three of these datasets include the ground truth optical flow obtained by the OptiTrack motion capture system. An additional dataset is obtained from the MVSEC dateset [38,39,40] which includes the ground truth optical flow

1.4 Overview

Related work is described in Section 2. Optical flow estimation using the Fisher-Rao metric is described in Section 3. The implementation of the Fisher-Rao method for estimating optical flow is described in Section 4. Experiments with the Fisher-Rao method are reported in Sections 5, 6 and 7. In particular, an experimental comparison of the Fisher-Rao algorithm with two state of the art algorithms is described in Section 7. Section 8 is a conclusion. The OptiTrack system for obtaining ground truth optical flow is described in Appendix A. The Fisher-Rao method for estimating optical flow is summarised in Appendix B by three algorithms in an informal notation.

2 Related Work

Research on processing techniques suitable for AER data has been prolific in the past few years. AER datasets with ground truth are described by Barranco et al. [5] and by Zhu et al. [38, 39, 40]. The latter very extensive dataset is used in the experiments described in Section 7. Direct conversions of state of the art computer vision algorithms to AER based algorithms are usually achieved by using the intensity information estimated by the local integration of the events (1). This approach is adopted for event correlation applied to stereo matching [24], for photoconsistency based estimation of optical flow [6] and for machine learning using convolution networks [31]. However, local integration of events does not preserve the event camera's temporal accuracy. Akolkar et al. [1] show that the high temporal accuracy of event camera measurements yields up to 70% more information, compared with conventional frame based methods. This is a motivation for focusing on truly event-based techniques. For example, Benosman et al. [7] reformulate optical flow estimation as a robust local plane fitting problem. The fitted planes are updated as new events arrive. The same technique is generalized by Ieng et al. [22] to 3D scenes by fitting planes to ruled surfaces generated by point clouds. The point clouds can be reconstructed either synchronously or asynchronously. The technique can handle different forms of data such as 3D events [13, 33], Lidar measurements and 3D points obtained from image frames by classical triangulation. The plane fitting algorithm in [7] for estimating optical flow is compared with the Fisher-Rao method in Section 7.2 below.

Rueckauer and Delbruck [34] evaluate nine algorithms for estimating the components of the optical flow normal to moving edges. The first algorithm searches for moving edges using the time differences between nearby events. The motions of the edges are measured. The next four algorithms are based on the Lucas-Kanade approach in which the optical flow is assumed to be locally constant and linear constraints on the optical flow are obtained using the motion constraint equation together with estimates of the intensity gradient. The remaining four algorithms are based on planes fitted to the measurements. The nine algorithms are evaluated on computer generated data obtained firstly from a translating square and secondly from a rotating bar. The algorithms are then evaluated on three experimental datasets obtained using a rotating camera. In contrast with [34], our Fisher-Rao method is a new way of estimating optical flow that does not assume that the optical flow is locally constant and that does not require estimates of the intensity gradient. In Section 6, the Fisher-Rao method is applied to two of the datasets in [34].

Bardow et al. [3] estimate the optical flow and the full intensity image from event camera data by

minimizing a complicated objective function that penalizes high optical flow gradients, high intensity gradients and large deviations from the motion constraint equation for optical flow. The objective function also includes terms to take into account the fact that only differences in image intensities can be measured. The objective function is minimized using iteration, to yield estimates of optical flow and of the full intensity image. In contrast, our Fisher-Rao method for estimating optical flow does not require the full intensity image. In any case, the task of obtaining the full intensity image is ill-posed in that regularisation is required in order to obtain a unique solution. Our Fisher-Rao matrix is obtained by a standard least squares calculation which has a unique solution without the necessity for regularisation.

Brosch et al. [11] and Brosch et al. [12] construct filters for event camera data by analogy with the filters found in biological vision. The filtered data are used to estimate the components of the optical flow normal to moving edges. In contrast, our Fisher-Rao method does not assume that the event measurements originate from moving edges and it does not require any filtering, except for a single Gaussian smoothing of a spike count array.

Barranco et al. [4] estimate normal velocities along a moving contour using event camera measurements. Motion boundaries are located by finding connected groups of pixels such that each pixel emits at least one event during a specified time interval. The components of the normal velocity are estimated separately by considering first the horizontal motion and then the vertical motion. Experiments are carried out using data from a dynamic and active vision based sensor, DAVIS, which is referred to as the ApsDVS sensor by Berner et al. [8]. The DAVIS sensor provides both AER measurements and complete frames of intensity values. The data from the frames is used to improve the accuracy of the estimates of contour motion. The estimates of the optical flow and the moving contours are checked using AER measurements synthesized from conventional image sequences. In contrast, the Fisher-Rao method does not require frames of intensity values and it does not attempt to identify any contours in the image. The Fisher-Rao method estimates full optical flow vectors, rather than particular components of the flow vectors.

Zhu et al. [40] discretize event camera measurements in the time domain. The discrete measurements are input to a neural network to estimate optical flow. Discrete measurements from a stereo pair are input to a second neural network to estimate ego motion and scene depths. Gallego et al. [16] specify a constant value for the optical flow in a small 3D neighbourhood. Each measurement in the neighbourhood defines a trajectory parameterised by time. The point on the trajectory at a fixed reference time is obtained. An objective function is defined using the resulting set of points. The objective function is maximised iteratively over the space of possible values for the optical flow. In contrast, our Fisher-Rao method does not assume that the optical flow is locally constant and it does not require the iterative maximisation of an objective function.

Gherig et al. [18] describe a general framework for obtaining a grid based representation for event camera measurements. The measurements are initially represented by a weighted sum of Dirac functions. The Dirac functions are convolved with a kernel and the convolved measurements are sampled in space and time to produce a fourth order array, taking the event polarities in (1) into account. A range of different arrays can be produced by varying the weighting of the Dirac functions, varying the kernel or by projection from the fourth order array. Applications to object recognition and optical flow estimation are described. The optical flow is estimated using EV-FlowNet [39].

Liu and Delbruck [28] record events in three time slice memories. The first memory simply accumulates events. The optical flow associated with an incoming event is estimated by matching blocks of data in the second time slice memory with blocks of data in the third time slice memory. The three memories are updated periodically: the previous first time slice becomes the new second time slice, and the previous second time slice becomes the new third time slice. Three different methods for choosing the times to make the updates are evaluated experimentally. In contrast, our Fisher-Rao method does not rely on block matching. Instead, it estimates optical flow by matching probability distributions using the Fisher-Rao metric. The results of the matching are invariant under the choice of parameterisation of the image.

Ghosh et al. [19] use slow feature analysis to extract features from event camera measurements. The features remain stable when events are missed. A convolutional neural network is used to classify actions given the filter responses. In [20] Ghosh et al. summarise the information in sets of events using neighbourhood spike count arrays. Features are extracted from the spike count arrays using princi-

pal components analysis and slow feature analysis. The features are applied to the tracking of cars in a traffic dataset. Algorithms for high speed tracking using event camera measurements are described by Lagorce et al. [26].

The use of local histograms for matching in conventional images is well established [21, 23, 35]. In this paper the local histograms are normalised to produce probability distributions. Once these distributions are obtained, the optical flow is estimated using powerful methods taken from probability theory, in particular, methods based on the Fisher-Rao metric. The Fisher-Rao metric is described by Amari [2] and by Cover and Thomas [14]. As far as the authors are aware, there is no previous application of the Fisher-Rao metric to the estimation of optical flow using AER measurements.

3 Optical Flow Estimation Using the Fisher-Rao Metric

The relevant properties of the Fisher-Rao metric are described in Section 3.1. The metric is applied to a family of discrete probability distributions obtained in Section 3.2 by dividing the AER measurement space into rectangular cuboids and counting the number of events in each cuboid. Brightness constancy is discussed in Section 3.3. The details of the Fisher-Rao method for estimating optical flow are given in Section 3.4.

3.1 Overview

The event camera measurements in a small spatiotemporal volume are summarized by a probability distribution defined on a three dimensional grid centred at the mid point (q, t) of the spatiotemporal volume, where q is a pixel and t is a time. In this way, a three parameter family of probability distributions is obtained. These distributions have the role of spatiotemporal features. The Fisher-Rao metric is a Riemannian metric defined on the parameter space for the probability distributions. In this case the parameter space is a subset of $\mathbb{R}^3$. The metric is specified at each point (q, t) of the parameter space by a 3×3 symmetric non-negative matrix $J(q, t)$. Further information is given by Amari [2], Cover and Thomas [14] and Kullback [25]. The squared distance between the probability distribution with parameters (q, t) and the probability distribution with parameters $(q + \Delta q, t + \Delta t)$ is given to leading order by

$$(\Delta q, \Delta t) J(q, t) (\Delta q, \Delta t)^\top. \tag{2}$$

The squared distance (2) is estimated directly from the two probability distributions using the Kullback-Leibler divergence [14, 25]. The matrix $J(q, t)$ is estimated in turn using the squared distances obtained for a range of different values of $(\Delta q, \Delta t)$. It is assumed that brightness constancy holds to a good approximation over a short time interval. With this assumption a moving object gives rise to a sequence of distributions that are close together in the Fisher-Rao metric. If (q, t) and $(q + \Delta q, t + \Delta t)$ are the parameter values for two distributions in this sequence, then the squared distance (2) is small, and $(\Delta q, \Delta t)$ is an estimate of the eigenvector $(\Delta q_e, \Delta t_e)$ of $J(q, t)$ with the least eigenvalue. The optical flow vector at (q, t) is given by $\Delta q_e / \Delta t_e$. If two of the eigenvalues of $J(q, t)$ are small, then the full optical flow cannot be estimated. Instead, one component only of the optical flow can be estimated. This is the well known aperture problem. Further details are included at the end of Section 3.4, below.

The advantages of the Fisher-Rao algorithm for estimating optical flow are as follows.

- The Fisher-Rao algorithm has a small number of parameters. There is no learning stage and no requirement for application specific features. It is not necessary to estimate the pixel grey levels.
- The use of the Fisher-Rao metric ensures that the squared distances in (2), on which the Fisher-Rao algorithm depends, are fundamental quantities, in that they are unaffected by the choice of the parameterisation of the family of probability distributions.
- The aperture problem can be described cleanly, using the eigenvalues of the Fisher-Rao matrix.

3.2 Implementation details

Each point (q, t) of the parameter space has a box neighbourhood, as noted in [20]. To be specific, let m, n be odd positive integers and let $\tau > 0$ be a time interval. An event (r, s), $r \equiv (r_1, r_2)$, is in the box neighbourhood of (q, t) if

$$|r_i - q_i| \leq (m-1)/2, \quad i = 1, 2,$$

and

$$|\tau^{-1}(s - t)| \leq (n-1)/2,$$

where $|.|$ is the absolute value.

The events in the box neighbourhood of (q,t) are used to define a neighbourhood spike array, $L(q,t)$, as in [20]. The array $L(q,t)$ has dimensions $m \times m \times n$. Each element of $L(q,t)$ corresponds to a voxel in $\mathbb{R}^3$. Let $\lfloor . \rfloor$ be the floor function. A point (r,s) in the box neighbourhood of (q,t) is in the voxel corresponding to the array indices i, j, k defined by

$$i = \lfloor r_1 - q_1 \rfloor + (m+1)/2,$$
$$j = \lfloor r_2 - q_2 \rfloor + (m+1)/2,$$
$$k = \lfloor \tau^{-1}(s-t) \rfloor + (n+1)/2.$$

The array element $L_{ijk}(q,t)$, $1 \leq i,j \leq m$, $1 \leq k \leq n$, is equal to the number of events in the voxel corresponding to (i,j,k). The array $L(q,t)$ is scaled to produce a discrete probability distribution, $g(q,t)$, defined on the set

$$\{(i,j,k), 1 \leq i,j \leq m, 1 \leq k \leq n\}.$$

The sum of the elements $g_{ijk}(q,t)$ over i, j and k is equal to one.

It is convenient to choose coordinates in $\mathbb{R}^3$ such that $q = (0,0)$ and $t = 0$. With this choice, the probability distribution $g(q,t)$ is denoted by g_0. Let $a = (a_1, a_2, a_3)$ be a vector in $\mathbb{R}^3$. Then g_a is defined to be the probability distribution obtained from the neighbourhood spike array $L(r,s)$ centred at the point $r = (a_1, a_2)$, $s = a_3\tau$. The probability distributions g_a for a in $\{-1,0,1\}^3$ are used in Section 3.4 to estimate the 3×3 matrix $J(0)$ that specifies the Fisher-Rao metric at $q = 0$, $t = 0$.

3.3 Optical flow

Suppose that a moving object is observed by a camera for a short period of time. It is assumed that brightness constancy holds, in that the appearance of a point on the object does not change significantly as the point moves through a short distance in the field of view. If a point is observed at the pixel (i_0, j_0) at time t_0 and if the same point is observed at the pixel (i_1, j_1) at a later time t_1 near to t_0, then the value of the pixel (i_0, j_0) at time t_0 is approximately equal to the value of the pixel (i_1, j_1) at time t_1. This brightness constancy is the basis of many methods for estimating optical flow [15]. The optical flow (u,v) at (i_0, j_0) at time t_0 is estimated by

$$(u,v) \approx$$
$$(t_1 - t_0)^{-1}(i_1 - i_0, j_1 - j_0) \ \text{ pixels } s^{-1}. \tag{3}$$

Let $\tau = t_1 - t_0$. It follows from (3) that

$$(i_1, j_1, t_1) \approx (i_0, j_0, t_0) + (u\tau, v\tau, \tau). \tag{4}$$

The optical flow (u,v) and the time interval τ together define a translation $(u\tau, v\tau, \tau)$ in the measurement space $\mathbb{R}^3$. The magnitude of this translation is proportional to τ.

In some cases it is not possible to establish a unique match between points (i_0, j_0, t_0) and (i_1, j_1, t_1). For example, if the optical flow is due to a moving straight edge and if (i_0, j_0, t_0) matches (i_1, j_1, t_1), then (i_0, j_0, t_0) also matches any point (i_2, j_2, t_1) for which $(i_2 - i_1, j_2 - j_1)$ is parallel to the edge. The component of the optical flow parallel to the edge cannot be measured. This ambiguity is known as the aperture problem.

3.4 Estimation of the optical flow

The optical flow is estimated at a point (q,t), where q is a pixel and t is a time. The estimate is obtained using the Fisher-Rao matrix $J(q,t)$. As noted at the end of Section 3.2, it is convenient to choose coordinates in $\mathbb{R}^3$ such that $q = (0,0)$ and $t = 0$. In this context, the Fisher-Rao matrix is written as $J(0)$, in place of the notation $J(q,t)$ used in Section 3.1. Let a be a vector in $\mathbb{R}^3$ and let g_a be the associated probability distribution, as defined in Section 3.2.

A particular value for $\Delta(i)$ in (1) is chosen, for example $\Delta(i) = 1$. The Fisher-Rao matrix is obtained using the fact that a scaled version of the Fisher-Rao matrix is a leading order approximation to the Kullback-Leibler divergence [2, 25] as shown in (6) below. The Kullback-Leibler divergence $D(0||a)$ of g_a from g_0 is defined by

$$D(0||a) = \sum_{i,j=1}^{m} \sum_{k=1}^{n} g_{0,ijk} \ln(g_{0,ijk}/g_{a,ijk}). \tag{5}$$

If $D(0\|a)$ is sufficiently smooth as a function of a, then the leading order term in a Taylor expansion of $D(0\|a)$ at $a = 0$ is quadratic in a [25], in that

$$D(0\|a) = \frac{1}{2} a J'(0) a^\top + O(\|a\|^3),$$

where $J'(0)$ is a symmetric 3×3 non-negative matrix. The matrix $J'(0)$ is estimated using the 26 values of $D(0\|a)$ for a in $\{-1,0,1\}^3$, $a \neq 0$, together with the approximation

$$D(0\|a) \approx \frac{1}{2} a J'(0) a^\top. \tag{6}$$

In fact it is only necessary to estimate accurately the eigenvector of $J'(0)$ associated with the least

eigenvalue. The relevant values of $D(0\|a)$ are those near to zero.

Let $a = (u\tau, v\tau, \tau)$, where $(u\tau, v\tau, \tau)$ is as defined in (4). The square of the distance between the distributions g_0 and g_a is estimated by

$$(u\tau, v\tau, \tau)J'(0)(u\tau, v\tau, \tau)^\top. \tag{7}$$

It follows from brightness constancy, as described in Section 3.3, that the measurements used to estimate g_a are translates in $\mathbb{R}^3$ of the measurements used to estimate g_0. It follows that g_a is equal to g_0, thus the Fisher-Rao distance between g_a and g_0 is zero and $(u\tau, v\tau, \tau)^\top$ is an eigenvector of $J'(0)$ with eigenvalue 0.

In the above calculations the terms $\Delta(i)$ in (1) have the value 1. The measurements for which $\Delta(i) = 0$ are also used to obtain a set of probability distributions and an associated Fisher-Rao matrix $J''(0)$. Let $J(0)$ be defined by

$$J(0) = J'(0) + J''(0). \tag{8}$$

The optical flow at the point (q, t) corresponding to the point $0 \equiv \{0, 0, 0\}$ is estimated using the eigenvector of $J(0)$ with the least eigenvalue.

The eigenvalues of $J(0)$ can be used to detect the aperture problem described in Section 3.3. If two of the eigenvalues of $J(0)$ are near to zero and the third eigenvalue is significantly different from zero, then only one component of the optical flow can be measured accurately, i.e. the aperture problem appears. In detail, let (u, v) be the optical flow. If $J(0)$ has two eigenvalues equal to zero, then it has the form $J(0) = e^\top e$, where e is a row vector with coordinates $e = (e_1, e_2, e_3)$. It follows from the definition of (u, v) that for a short time interval τ,

$$\begin{aligned} 0 &= (u\tau, v\tau, \tau)J(0)(u\tau, v\tau, \tau)^\top \\ &= (ue_1 + ve_2 + e_3)^2\tau^2, \end{aligned}$$

thus

$$(u, v)(e_1, e_2)^\top = -e_3.$$

The magnitude of the component of the optical flow parallel to (e_1, e_2) is obtained by taking the scalar product of (u, v) with the unit vector in the direction (e_1, e_2), namely

$$(e_1^2 + e_2^2)^{-1/2}(u, v)(e_1, e_2)^\top,$$

which is equal to

$$-(e_1^2 + e_2^2)^{-1/2}e_3. \tag{9}$$

The component of the optical flow normal to (e_1, e_2) cannot be measured.

4 Implementation

The algorithm described in Section 3.4 for estimating the optical flow requires some modifications and choices of parameters in order to obtain accurate results in practice. In the following description it is assumed that the quantity $\Delta(i)$ in (1) is fixed, for example $\Delta(i) = 1$. Let the pixel array in the event camera have dimensions $x_{max} \times y_{max}$, and let m, n be odd positive integers.

Let t be a time. It is convenient to define a single large spike count array A_t with dimensions $x_{max} \times y_{max} \times (n + 2)$. Let E be the list of events and let $\tilde{E}$ be the sub-list of E defined by

$$\tilde{E} = \{(r, s), (r, s) \in E, |s - t| \leq \tau(n + 1)/2\}.$$

The spike count array A_t is defined by

$$A_t(i, j, k) =$$

$$\#\{(r, s), (r, s) \in \tilde{E}, k = \lfloor \tau^{-1}(s - t) \rfloor + (n + 3)/2\},$$

for $1 \leq i \leq x_{max}$, $1 \leq j \leq y_{max}$ and $1 \leq k \leq n+2$, where $\#\{.\}$ is the number of elements in the set $\{.\}$.

The array A_t contains all the spike count arrays

$$L(r + (a_1, a_2), t + a_3\tau), \tag{10}$$

such that r is a pixel and $(a_1, a_2, a_3) \equiv a$ is in $\{-1, 0, 1\}^3$. Let $\tilde{a}_t(q)$ be the $(m + 2) \times (m + 2) \times (n + 2)$ sub-array of A_t defined by

$$\begin{aligned} \tilde{a}_t(q) = A_t(&q_1 - (m + 3)/2 : q_1 + (m + 3)/2, \\ &q_2 - (m + 3)/2 : q_2 + (m + 3)/2), \end{aligned} \tag{11}$$

where : is the MATLAB notation for a range of array entries. The sub-array $\tilde{a}_t(q)$ is referred to as a block centred at (q, t). It contains the 27 sub-arrays obtained by setting $r = q$ in (10). Let n_q be the number of non-zero entries in $\tilde{a}_t(q)$. A list C_t is made of the pixels q for which

$$n_q \geq (m + 2)^2(n + 2)f \tag{12}$$

where f is a fixed parameter taking a value in $[0, 1]$. The optical flow is estimated only for the pixels contained in C_t.

A small strictly positive quantity ϵ is added to each element of A_t to ensure that the elements are all strictly larger than zero. This is to avoid numerical instabilities in the calculation (5) of the Kullback-Leibler divergence. The resulting array is smoothed with a mask that approximates to a Gaussian function with covariance $\sigma^2 I$ where I is the 3×3 identity matrix. Let B_t be the smoothed array.

Let $\tilde{b}_t(q)$ be the block obtained by replacing A_t in (11) with B_t. For each pixel q in C_t, let g_a for a in $\{-1,0,1\}^3$ be the set of 27 probability distributions obtained from $\tilde{b}_t(q)$. At this point, it is convenient to choose coordinates such that $q=(0,0)$ and $t=0$. The Fisher-Rao matrix $J'(0)$ is estimated using (6). There are six parameters to be estimated, namely $J'_{11}(0)$, $J'_{12}(0)$, $J'_{13}(0)$, $J'_{22}(0)$, $J'_{23}(0)$ and $J'_{33}(0)$. There are 26 equations of the form

$$D(0\|a)=\frac{1}{2}aJ'(0)a^\top,\quad a\in\{-1,0,1\}^3, a\neq 0. \tag{13}$$

A solution $J'(0)$ to (13) is estimated using least squares.

Similar calculations are carried out using the measurements with $\Delta(i)=0$, to obtain a Fisher-Rao matrix $J''(0)$. If $J'(0)$ and $J''(0)$ are both defined, in that the corresponding sub-arrays $\tilde{a}_t(0)$ and $\tilde{a}_t(0)$ each have a sufficient number of non-zero entries, then they are added, as in (8), to yield a matrix $J(0)$. If $J'(0)$ or $J''(0)$ is not defined, then the calculation is abandoned.

Let $\lambda_1\geq\lambda_2\geq\lambda_3$ be the eigenvalues of the matrix $J(0)$. Thresholds β_1, β_2 are chosen and $J(0)$ is accepted only if λ_3 is sufficiently small, in that

$$\lambda_1\geq\beta_1\lambda_3 \quad\text{and}\quad \lambda_2\geq\beta_2\lambda_3. \tag{14}$$

If λ_3 is comparable in magnitude to λ_1 and λ_2, then there is no match between nearby probability distributions and the optical flow is not defined. Let $w\equiv(w_1,w_2,w_3)^\top$ be the eigenvector corresponding to the least eigenvalue of an accepted matrix $J(0)$. The optical flow at the corresponding pixel (x,y) is estimated by

$$(u,v)=(w_1/w_3, w_2/w_3).$$

The units for the components u, v of the optical flow are pixels τ^{-1}. The estimate (u,v) of the optical flow is accepted only if $\|(u,v)\|\leq maxFlow$, where $maxFlow$ is a physically plausible threshold and $\|.\|$ is the Euclidean norm.

The time complexity for computing the array A_t is linear in the number of events. The time complexity for smoothing the array A_t and obtaining the list C_t of pixels is $O(x_{max}y_{max}(n+2))$. The time complexity for estimating each Fisher-Rao matrix is the sum of the $O(m^2n)$ cost of calculating the Kullback-Leibler divergences (5) and the $O(1)$ cost of the least squares estimate of $J(0)$. The time complexity also depends on the parameter f in (12). If f is large then few flow vectors are obtained.

A summary of the algorithm for estimating optical flow is included in Appendix B.

5 Experiments with Five Datasets

This section describes experiments to test the Fisher-Rao method for estimating optical flow using five new datasets, namely Data 1, Data 2, Data 3, Data 4 and Data 5. The datasets Data 1, Data 2, Data 3 and Data 4 were obtained using the Asynchronous Time-based Image Sensor (ATIS) [32] made by Prophesee. Data 5 was obtained using the next generation sensor, H-VGA. Ground truth optical flow was obtained for Data 3, Data 4 and Data 5 using the OptiTrack motion capture system [30]. Further information about OptiTrack is given in Appendix A.

The relevant properties of the proposed datasets are summarized in Table 2. The sixth dataset, MV/SEC, is described in Section 7.1 below. The parameter values used to test the Fisher-Rao algorithm on these datasets are summarised in Table 3. Changes in the parameter values from one dataset to the next were avoided, as far as possible. For example, five of the six datasets in Table 3 use the same size $11\times11\times11$ arrays to construct the box neighbourhoods, as described in Section 3.2. This indicates that the tuning of the parameter values is stable.

5.1 Data 1

The data were obtained in a laboratory using an ATIS event camera which was rotated about a fixed axis while viewing two flat pages. The camera was initially at rest, then it was rotated about the axis and finally brought to rest at the end of the motion. The first measurement was obtained at time $t_{min}=8513875\mu s$ and the last measurement was obtained at time $t_{max}=11791554\mu s$. The estimated optical flow is shown in Fig. 1 for four consecutive time intervals, each one of length $400000\mu s$, with the parameter τ in Sections 3.3 and 3.4 given by $\tau=400000/(n+2)\mu s$. The size of the pixel array was 240×304.

A pixel in Fig. 1 is blue if the events with $\Delta(i)=0$ predominate. A pixel is yellow if the events with $\Delta(i)=1$ predominate. If the number of events with $\Delta(i)=0$ is equal to the number of events with $\Delta(i)=1$, then the pixel is white. The estimated optical flow vectors are shown superposed in red. Each flow vector is scaled up by a factor of 10 in order to make it more visible. If the number of flow vectors is large, then some vectors are removed in order to improve the visibility of the remaining vectors. The parameters used to obtain the results in Fig. 1 are $m=11$ pixels, $n=11$ pixels, $f=1/20$, $\sigma=2$ pixels, $\beta_1=10$, $\beta_2=4$,

Dataset	Sensor	Camera Motion	Object Motion	Object	GT Flow
Data 1	ATIS	rotation	static	two planes	no
Data 2	ATIS	static	translation	car	no
Data 3	ATIS	rotation	static	plane	yes
Data 4	ATIS	static	random	plane	yes
Data 5	H-VGA	static	random	plane	yes
MVSEC	DAVISm346B	hexacopter	static	indoors	yes

Table 1 Summary of the properties of six of the datasets. The rightmost column is ground truth optical flow.

Dataset	$m \times m \times n$ (pixels^3)	$\tau(\mu s)$	f	β_1, β_2	σ (pixels)	*maxFlow* (pixels τ^{-1})	ϵ
Data 1	$11 \times 11 \times 11$	$4 \times 10^5/(n+2)$	1/20	10, 4	2	5	0.01
Data 2	$11 \times 11 \times 11$	$10^5/(n+2)$	1/20	10, 4	2	5	0.01
Data 3	$11 \times 11 \times 11$	$4 \times 10^5/(n+2)$	1/20	10, 4	2	5	0.025
Data 4	$11 \times 11 \times 11$	$10^5/(n+2)$	1/30	10, 4	4	5	0.025
Data 5	$5 \times 5 \times 5$	$62500/(n+2)$	1/160	10, 4	2	2	0.025
MVSEC	$11 \times 11 \times 11$	$10^5/(n+2)$	1/240	5, 2	2	1.5	10^{-5}

Table 2 Parameter values for the Fisher-Rao algorithm. For m, n and τ, see Section 3.2. For f, *maxFlow*, ϵ, β_1, β_2, σ: see Section 4.

$maxFlow = 5$ pixels τ^{-1} and $\epsilon = 0.01$, using the notation in Section 4. The parameter values are chosen empirically. The parameters m, n and the time interval $400000\mu s$ are chosen large enough to ensure that the probability distributions are stable. The threshold $maxFlow$ is necessary in order to remove outliers from the optical flow vectors.

5.2 Data 2

The data consist of measurements obtained by an ATIS event camera placed in a city street. The camera was mounted on a tripod and directed towards the road traffic. The pixel array is of size 240×304. The first measurement was obtained at $t_{min} = 13\mu$s and the last was obtained at $t_{max} = 5.06 \times 10^7 \mu$s. The four images shown in Fig. 2 were obtained from consecutive time intervals, each one of length $10^5 \mu$s. The parameter τ was given by $\tau = 10^5/(n+2)\mu$s. As in Fig. 1, each optical flow vector is scaled up by a factor of 10 and some flow vectors are not shown in order to improve the visibility of the remaining vectors. The parameters m, n, f, β_1, β_2, σ, *maxFlow* and ϵ have the same values as in Section 5.1 for Data 1.

Grey level image frames are available for Data 2. See for example Fig. 3. However, these frames were not used in the estimation of the optical flow.

5.3 Comparison with ground truth for Data 3 and Data 4

The ATIS datasets Data 3 and Data 4 are used to test the Fisher-Rao method by comparing the estimated optical flow with a ground truth optical flow provided by a motion capture system. The position and orientation of a moving camera relative to a planar set of points are measured over time. The ground truth optical flow is obtained by projecting the points into the camera. The motions of the camera and the motions of the planar set of points are measured using the motion capture system OptiTrack [30], which consists of 8 Flex13 cameras. The motion capture has a sub-millimeter spatial resolution and an acquisition frequency of 120Hz. All technical specifications, including information about accuracy, can be found on the web page [30]. Further information about the calculations used in OptiTrack is given in Appendix A.

In order to track the event-based camera, markers are put on its casing so that when the camera moves the trajectory of the casing can be updated in real time. The camera observes a planar printed pattern which contains a set of points that provide the data for the Fisher-Rao algorithm. The planar pattern also contains markers from which the ground truth optical flow is calculated.

The Fisher-Rao algorithm estimates the optical flow using time slices of $400ms$ $(2.5Hz)$ in Data 3 and slices of $100ms$ $(10Hz)$ in Data 4. The ground truth image velocities are computed from the sequences of positions of the tracked points obtained at times between two consecutive Fisher-Rao estimates of the optical flow.

The camera motion to obtain Data 3 is similar to that used for Data 1: the camera is placed on a tripod and rotated about a vertical axis, first clockwise and then counterclockwise, while the pla-

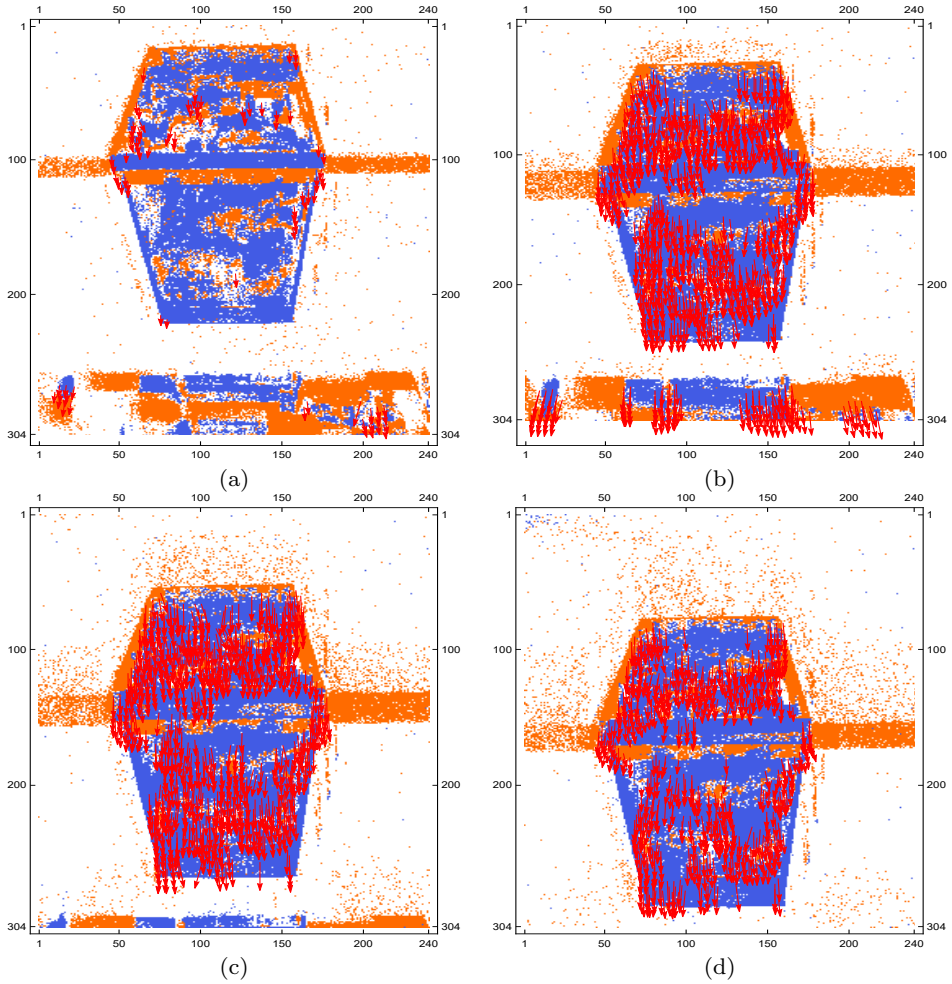


Fig. 1 Optical flow for Data 1 obtained from two flat pages viewed by a rotating camera. Blue: $\Delta = 0$ predominates. Yellow: $\Delta = 1$ predominates. White: number of events with $\Delta = 0$ equals the number of events with $\Delta = 1$.

nar pattern is held stationary. An example of the optical flow obtained by the Fisher-Rao algorithm is shown in Fig. 4. The flow vectors are scaled up by a factor of 15 and some flow vectors are not shown in order to improve the visibility of the flow field. Histograms of the directions errors, i.e. the differences between the orientations of the ground truth optical flow vectors and the orientations of the estimated optical flow vectors, are shown in the upper part of Fig. 5 for the two sweeps of the camera. The mean amplitude error shown for the two sweeps in Fig. 5 is the mean of the Euclidean norms of the differences between the empirical flow vectors and the corresponding ground truth flow vectors. The parameters are $m = 11$ pixels, $n = 11$ pixels, $f = 1/20$, $\sigma = 2$ pixels, $\beta_1 = 10$, $\beta_2 = 4$, $maxFlow = 5$ pixels τ^{-1} and $\epsilon = 0.025$. The value of τ is $\tau = 400000/(n+2)\mu$s.

A normal distribution is fitted to the scaled histogram of the directions errors. The mean value of the distribution is -1.5×10^{-3} rad and the standard deviation is 5×10^{-3} rad. The estimated amplitudes are close, with a mean Euclidean error of 7 pixels s^{-1}. The last sample in sweep 1 and the first sample in sweep 2 produce large errors because the pattern leaves the field of view and there are only a few pixels for which the optical flow can be estimated.

Data 4 differs from Data 3 in that the camera is static while the pattern moves in the world reference frame. An example of the optical flow obtained by the Fisher-Rao algorithm is shown in Fig. 4b. The flow vectors are scaled up by a factor of 10 and some flow vectors are not shown in order to improve the visibility of the flow field. The parameters are $m = 11$ pixels, $n = 11$ pixels, $f = 1/30$, $\sigma = 4$ pixels, $\beta_1 = 10$, $\beta_2 = 4$, $maxFlow = 5$ pixels τ^{-1} and $\epsilon = 0.025$. The value of τ is $\tau = 100000/(n+2)\mu$s. Fig. 6 shows the distribution of the directions errors and the mean

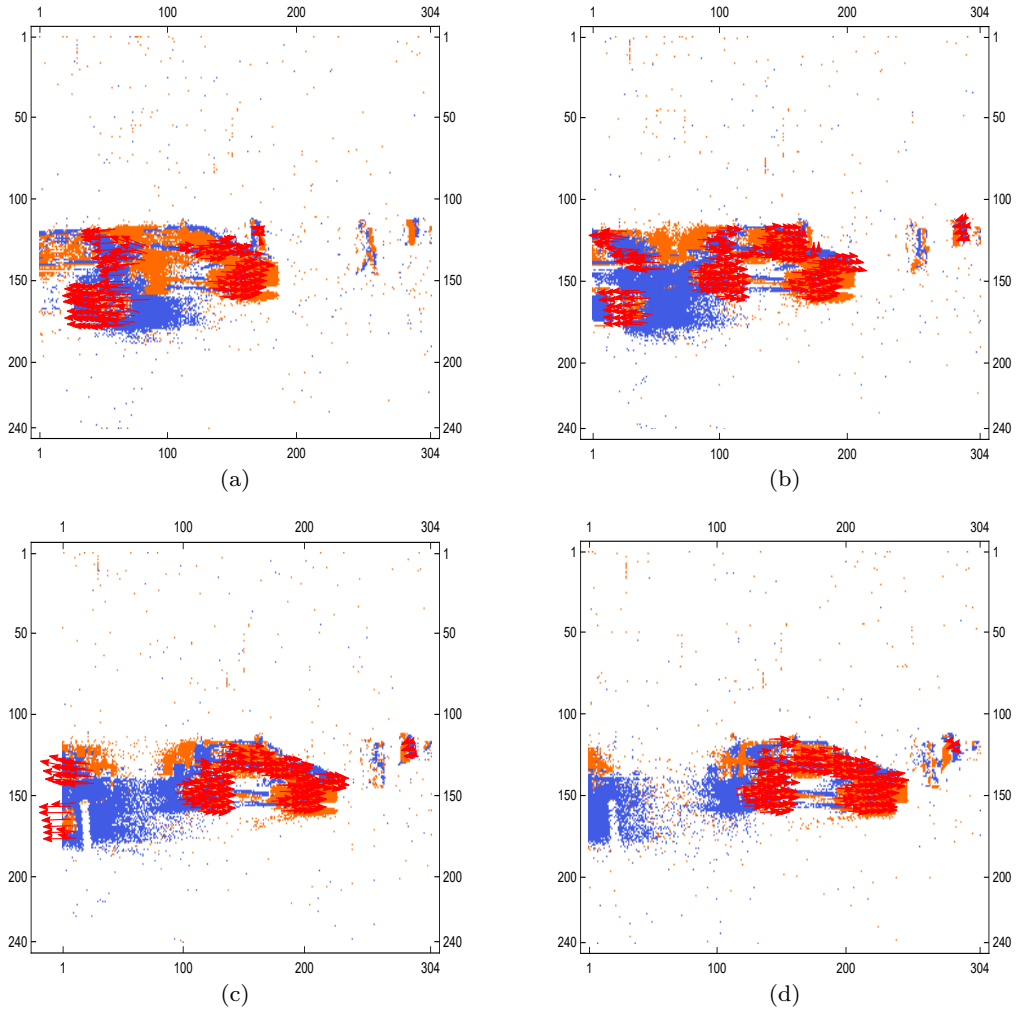


Fig. 2 Optical flow for Data 2 obtained from a street scene viewed by a fixed camera.

Fig. 3 Example of a grey level image frame for Data 2.

amplitude errors, calculated as for Data 3. The errors are higher than for Data 3 because the pattern was moved manually with a velocity varying in amplitude and direction, as required to retain the pattern within the field of view. The estimated mean and standard deviation of the directions errors are respectively -7.5×10^{-2} rad and 4×10^{-2} rad. The lower performance of the Fisher-Rao flow in this experiment is likely to be due to the varying velocity of the pattern. The matrices for the Fisher-Rao metric are estimated over spatiotemporal volumes whose dimensions have to fit the motion. If the volume is too small, then not enough events are available to estimate the flow and if the volume is too big, then the velocity estimate is an averaged value.

5.4 Data 5

Data 5 was obtained using a new generation H-VGA of the ATIS sensor. H-VGA and ATIS are compared in Table 1. The H-VGA sensor has an increased spatial resolution, a higher signal to noise ratio, sharper images and better performance in low light, as compared with the ATIS sensor. As a

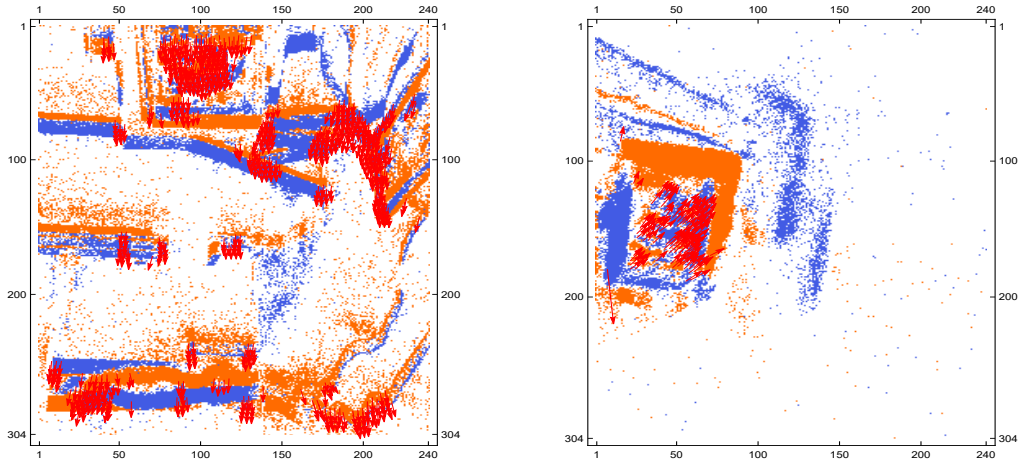


Fig. 4 a) Optical flow for Data 3; b) Optical flow for Data 4.

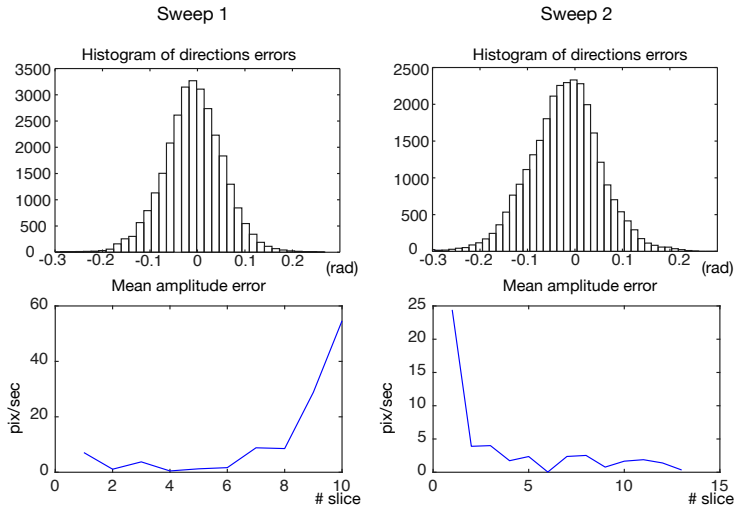


Fig. 5 Comparison of the Fisher-Rao optical flow with ground truth for Data 3, obtained from a static pattern viewed by a camera rotating about a fixed axis.

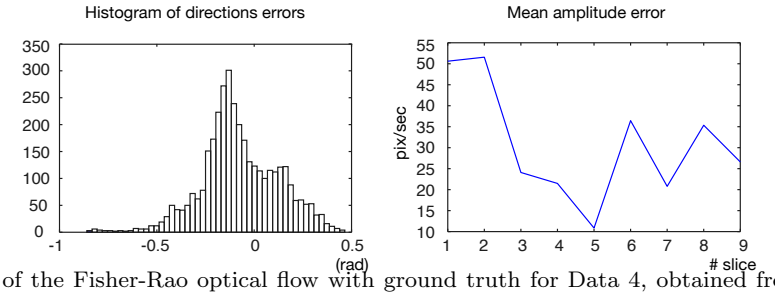


Fig. 6 Comparison of the Fisher-Rao optical flow with ground truth for Data 4, obtained from a moving pattern viewed by a static camera.

	ATIS	H-VGA
Resolution	240× 304	480×360
Dynamic range (dB)	143	120
Min. contrast sensitivity (%)	13	12
Pixel size (μm^2)	30×30	15×15
Fill factor (%)	20	25

Table 3 Comparison of the ATIS sensor and the H-VGA sensor.

result, more accurate estimates of the optical flow can be obtained from H-VGA measurements.

An example of the optical flow obtained from Data 5 is shown in Fig. 8. Some flow vectors are omitted in order to make the flow field clearer. The flow vectors are also scaled up by a factor of 10. The ground truth optical flow is obtained from OptiTrack using markers attached to a plane surface held by the experimenter, as shown in Fig. 8. The results of the Fisher-Rao algorithm are compared with ground truth only for the optical flow arising from the moving plane.

The increase in quality of the estimated optical flow can be seen by comparing the error curves for Data 4, given in Fig. 6, with those of Data 5, given in Fig. 7. In both cases the flow arises from a plane moved with an unconstrained velocity while keeping the camera static. The directions errors for Data 5 have a more uniform and narrower distribution than the direction errors for Data 4. The mean amplitude error is also lower, going from few pixels s^{-1} to a maximum of 20 pixels s^{-1}. The parameters for Data 5 are $m = 5$ pixels, $n = 5$ pixels, $f = 1/160$, $\sigma = 2$ pixels, $\beta_1 = 10$, $\beta_2 = 4$, $maxFlow = 2$ pixels τ^{-1} and $\epsilon = 0.025$. The value of τ is $\tau = 62500/(n+2)\mu$s. The errors in estimating optical flow with the H-VGA sensor are less than the errors obtained using the ATIS sensor, even though the parameters m, n, which control the size of the window for each probability distribution, are reduced from $m = n = 11$ pixels to $m = n = 5$ pixels.

6 Experiments with the Rueckauer-Delbruck Data

The Fisher-Rao algorithm was applied to two of the datasets in [34], namely the translating square (translSquare) and the rotating disk (rotDisk). These datasets were chosen because the ground truth is known. The results for translSquare and rotDisk are discussed in Sections 6.1 and 6.2 respectively.

6.1 Translating square

The data set translSquare is computer generated. It shows a textureless square of size 40×40 pixels^2 translating with a constant velocity of (20, 20) pixels s^{-1}. The first measurement is obtained at time $t_{min} = 50,000\mu$s and the last measurement is obtained at time $t_{max} = 5,000,000\mu$s. Nineteen consecutive subintervals of width $\Delta s = 247500\mu$s are chosen from $[t_{min}, t_{max}]$, such that the central (i.e. 10th) subinterval is

$$[(t_{min} + t_{max})/2 - \Delta s/2, (t_{min} + t_{max})/2 + \Delta s/2].$$

It is not possible to obtain the full optical flow vectors along the sides of the square because of the aperture problem. The normal components of the optical flow vectors are obtained using (9). The parameters used by the Fisher-Rao algorithm are $m = 11$ pixels, $n = 11$ pixels, $f = 1/100$, $\beta_1 = 5$, $\sigma = 2$ pixels, $maxFlow = 2$ pixels Δs^{-1}. The condition $\lambda_2 \geq \beta_2 \lambda_3$ in (14) is discarded because the eigenvalues λ_2, λ_3 of the Fisher-Rao matrix both tend to be small compared with λ_1.

It was found that very few spatiotemporal volumes contain enough measurements with $\Delta(i) = 0$ and with $\Delta(i) = 1$ to enable the calculation of both of the matrices $J'(0)$, $J''(0)$ in (8). If only one of $J'(0)$, $J''(0)$ is available, then $J(0)$ is set equal to that matrix. The optical flow estimated by the Fisher-Rao algorithm for the 10th subinterval is shown in Fig. 9a. The flow vectors are scaled up by a factor of 10 and some flow vectors are removed to improve the visibility of the remaining vectors. The estimated optical flows are similar for the other 18 subintervals.

Under ideal conditions, each normal flow vector is parallel or anti-parallel to a coordinate axis and has a norm of 20 pixels s^{-1}. The errors in the directions of the normal flow vectors over all 19 subintervals have a mean value of -0.00014 radians and a standard deviation of 0.046 radians. The scale factor to convert the Fisher-Rao flow to a flow in pixels s^{-1} is $(n + 2)/(10^{-6}\Delta s)$. The errors in the magnitudes of the normal flow vectors over all 19 subintervals have a mean value of -0.80 pixels s^{-1} and a standard deviation of 2.86 pixels s^{-1}.

6.2 Rotating disk

The data set rotDisk [34] is obtained from a 240 × 180 pixel Dynamic and Active-pixel Vision Sensor (DAVIS). The camera observes a disk divided

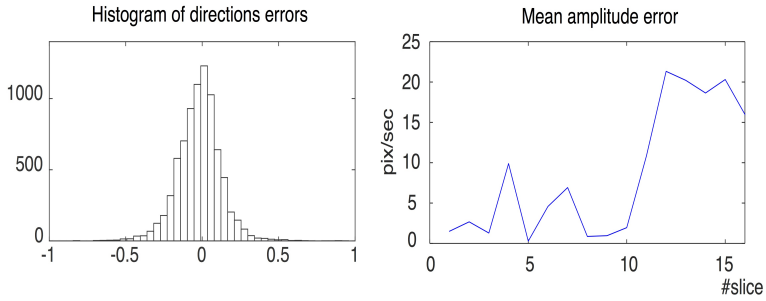


Fig. 7 Comparison of the Fisher-Rao optical flow with ground truth for Data 5, obtained from a moving pattern viewed by a static H-VGA camera.

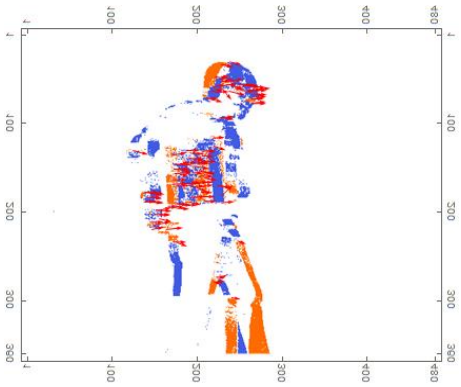


Fig. 8 Example of optical flow obtained using Data 5.

into eight sectors with varying grey levels. The disk is kept stationary while the camera rotates about a fixed axis. The disk appears to rotate in a clockwise direction about a centre $c = (115, 86)$. The first measurement is obtained at time $t_{min} = 341678\mu$s and the last measurement is obtained at time $t_{max} = 3508437\mu$s. Nineteen consecutive subintervals of width $\Delta s = 158338\mu$s are defined following the example in Section 6.1.

It is not possible to obtain the full optical flow vectors because of the aperture problem. The boundaries between the different sectors of the disc are straight lines with uniform grey levels on either side. As in Section 6.1, the normal component of the optical flow is obtained using (9). In this particular case the normal optical flow coincides with the full optical flow. The parameters used by the Fisher-Rao algorithm are the same as for translSquare. The condition $\lambda_2 \geq \beta_2 \lambda_3$ in (14) is discarded.

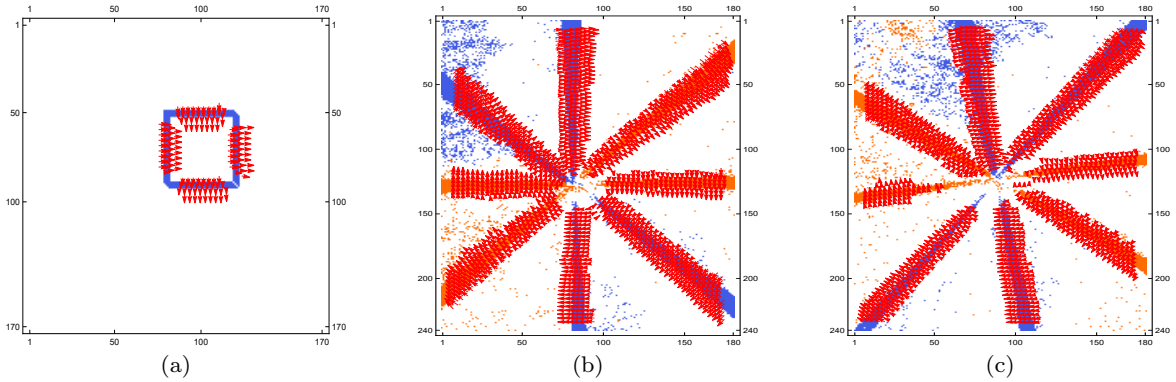
As in the case of the translating square, if only one of $J'(0)$, $J''(0)$ in (8) is available, then $J(0)$ is set equal to that matrix. The estimated optical flows for the second subinterval and the eleventh subinterval are shown in Figs 9b and 9c respectively. The flow vectors are scaled up by a factor of 10 and some flow vectors are removed in order to improve the visibility of the remaining vectors. The flow vectors in the remaining subintervals are similar.

The flow vector data have the form $(q, (u, v))$, such that (u, v) is the normal component of the flow vector and q is the base point. Flow vectors with $\|q - c\| \leq 20$ are discarded. There remain 251,179 flow vectors from all nineteen subintervals. The angular velocity of each flow vector is given by $\|q - c\|^{-1}\|(u, v)\|$. The average angular velocity is 0.48 radians s^{-1} with a standard deviation of 0.13 radians s^{-1}. Ideally, the angle between $(x, y) - c$ and (u, v) should be $\pi/2$ radians. The deviations of this angle from $\pi/2$ have a mean value of 0.0065 radians and a standard deviation of 0.12 radians.

It is apparent by visual inspection that the upper vertical bar in Fig. 9b moves clockwise to the corner of the image in Fig. 9c. The change in orientation of the bar is estimated to be 0.69 radians. The time interval is $9\Delta s$, thus the angular velocity is estimated to be 0.48 radians s^{-1}, in agreement with the value obtained by the Fisher-Rao algorithm.

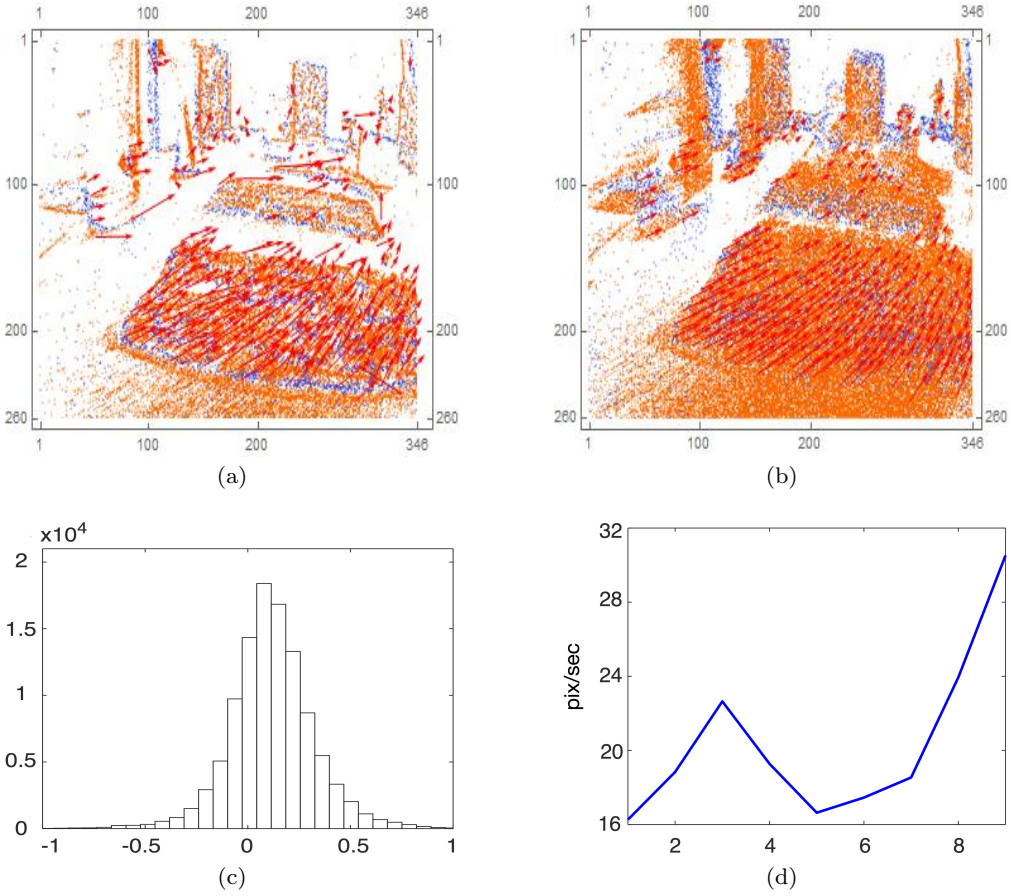
7 Comparison with the State of the Art

In this section the Fisher-Rao algorithm is compared with two state of the art algorithms for estimating optical flow from event camera measurements. The first algorithm has two forms, namely EV-FlowNet$_{2R}$ and EV-FlowNet$_{4R}$ (Zhu et al. [39]). The results are described in Section 7.1. The second algorithm fits planes to sets of events [6, 7]. The results are described in Section 7.2.



(a) (b) (c)

Fig. 9 Examples of optical flow obtained from (a) translSquare and (b), (c) from rotDisk.



(a) (b) (c) (d)

Fig. 10 Comparison of the Fisher-Rao optical flow with ground truth for the MVSEC data:(a) Fisher-Rao flow; (b) ground truth flow; (c) histogram of directions errors; d) graph of mean end point errors.

7.1 Comparison with EV-FlowNet$_{2R}$ and EV-FlowNet$_{4R}$

The data for the comparison between the Fisher-Rao algorithm and EV-FlowNet$_{2R}$ were obtained from the Multi Vehicle Stereo Event Camera (MVSEC) dataset [38, 39, 40]. The dataset contains stereo event camera measurements from a car, motorcycle, hexacopter and hand held camera. The data were obtained from both indoor and outdoor environments. The Fisher-Rao algorithm was applied to the indoor hexacopter measurements. Zhu et al. [39] obtained the ground truth optical flow using the Vicon motion capture system with 20 cameras to observe the hexacopter. The camera was

Algorithm	$t = 0.05$(s)	$t = 0.1$(s)	$t = 0.2(s)$
EV-FlowNet$_{2R}$	1.03	-	2.25
EV-FlowNet$_{4R}$	1.14	-	2.75
Fisher-Rao	1.88	2.05	2.95

Table 4 Mean amplitude errors for three algorithms applied to the MVSEC data indoor flying1.

a DAVIS m346B with a resolution of 346×260 pixels^2.

The list of events was obtained from the file indoor_flying1_data.hdf5. The ground truth optical flow and a list of time stamps were obtained from the MVSEC file indoor_flying1_gt_flow_dist.npz. Let ts be the list of time stamps and let t be a time interval. Nine time slices were chosen, namely

$$[ts(i) - t/2, ts(i) + t/2], \tag{15}$$

for $681 \leq i \leq 689$. The time step τ is given by $\tau = t/(n+2) = t/13$. The remaining parameter values are $m = 11$ pixels, $n = 11$ pixels, $f = 1/240$, $\sigma = 2$ pixels, $\beta_1 = 5$, $\beta_2 = 2$, $maxFlow = 3/2$ pixels τ^{-1} and $\epsilon = 10^{-5}$.

Fig. 10a shows the optical flow obtained by the Fisher-Rao algorithm for the first time slice, with $t = 10^5 \mu$s. In order to make Fig. 10 clearer, the original flow vectors are scaled by 130/4. The units for this scaled flow are 1/4 pixels s^{-1}. The corresponding ground truth flow is shown in Fig. 10b for time stamp 681. The original ground truth flow vectors are scaled by 5 to ensure that the units are the same as for Fig. 10a. A histogram of directions errors is shown in Fig. 10c and a graph of mean amplitude error as a function of the time slice is shown in Fig. 10d. The histogram of directions errors is accumulated over all nine time slices.

Mean amplitude errors for the algorithms EV-FlowNet$_{2R}$, EV-FlowNet$_{4R}$ and Fisher-Rao are shown in Table 2 for the data indoor flying1. The entries for EV-FlowNet$_{2R}$ and EV-FlowNet$_{4R}$ in Table 2 are taken from Table 1 in [39], where they are referred to as average end point errors. The mean amplitude errors for the Fisher-Rao algorithm, for example as shown in Fig. 10d, are scaled to give the values obtained after one second of the flow. In order to make the comparison with EV-FlowNet$_{2R}$ and EV-FlowNet$_{4R}$ in Table 2 it is necessary to scale the errors to give the values obtained after t seconds of the flow. It is apparent from Table 2 that the errors for the Fisher-Rao algorithm are near to the errors for EV-FlowNet$_{2R}$ and EV-FlowNet$_{4R}$. The advantage of the Fisher-Rao algorithm is that it is much simpler to initialize. Grey level images and ground truth optical flow are not required. It is not necessary to train a complicated deep neural network. It is only necessary to tune the values of the nine parameters listed in Table 3. The experimental results show that the Fisher-Rao algorithm is stable, in that only minor changes in the values of the parameters are required if the dataset is changed.

7.2 Comparison with a Plane Fitting Algorithm

The Fisher-Rao algorithm is compared with a state of the art event based algorithm for estimating optical flow. The algorithm fits planes to sets of measurements in $\mathbb{R}^3$ [6, 7]. The algorithm is applied to Data 3 and its results compared with those obtained from the Fisher-Rao algorithm. The parameters of the algorithms are made as similar as possible: the size of the spatial neighborhood is set to 11×11 pixels^2 and only the latest events within this spatial neighborhood are used for estimating the flow. A local plane fitting is carried out for each incoming event and the optical flow is estimated using the orientation of the plane. The plane based algorithm only estimates the components of the flow normal to the moving edges that generate the events, as shown in the image in Fig. 11.

As the results of the Fisher-Rao algorithm have already been compared with the ground truth in Section 5.3, only the directions of the estimated flow vectors are compared. The histograms of directions computed by the two algorithms are shown in Fig. 12. The most interesting result that can be obtained from the histograms is a comparison of the robustness of each algorithm to the aperture problem. The estimated flow directions for the Fisher-Rao algorithm agree with ground truth. The estimated flow directions for the plane based algorithm have a larger spread around the ground truth direction.

8 Conclusion

A new algorithm for estimating optical flow from event camera measurements has been described. The algorithm is based on the Fisher-Rao metric, which is defined on the parameter spaces for families of probability distributions. In this application, the distributions are obtained from the measurements in small spatiotemporal volumes referred to as box neighbourhoods. The time component of the measurement is quantized, to ensure that each distribution is defined on a three dimensional neigh-

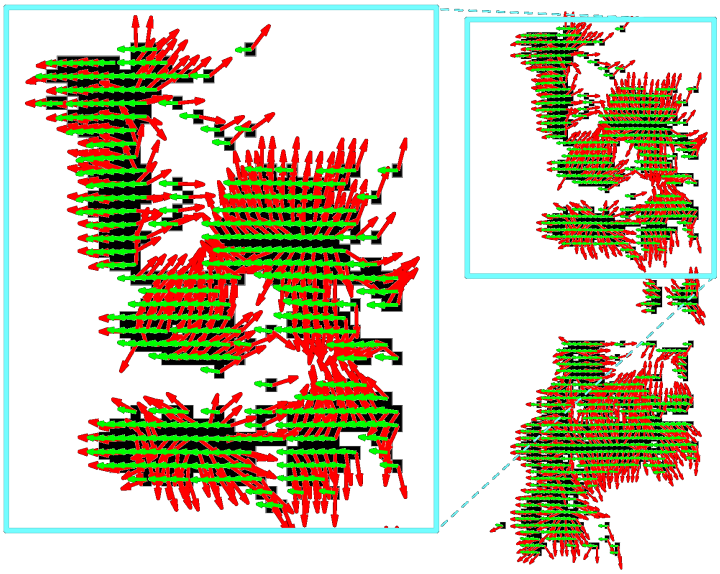
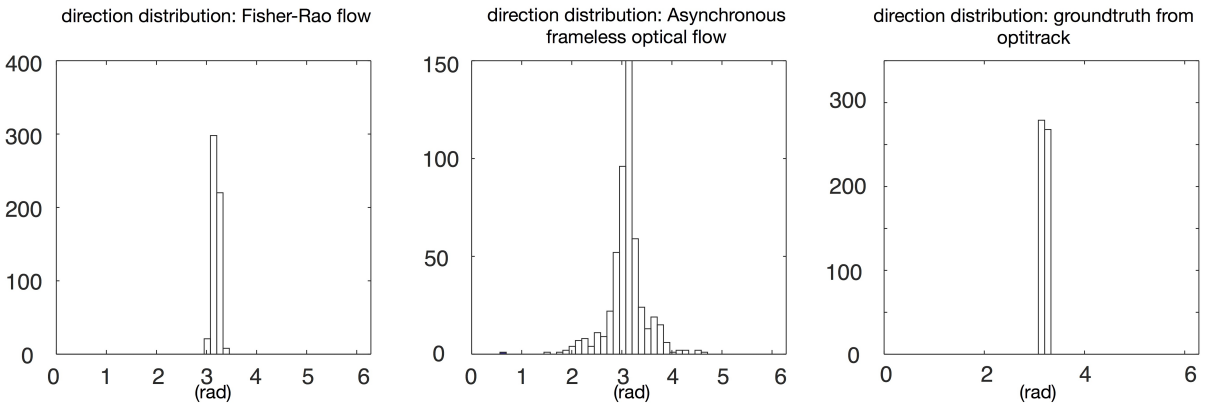


Fig. 11 Comparison of the Fisher-Rao algorithm with the plane based algorithm in [7] using zooms of the flows. The red arrows show the normal flow obtained in [7]. The green arrows show the Fisher-Rao flow.



(a) (b) (c)

Fig. 12 Comparison of the Fisher-Rao algorithm with the plane based algorithm; a) Fisher-Rao based histogram for the directions of the flow vectors; b) plane based histogram for the directions of the flow vectors; c) ground truth histogram.

bourhood spike array. The parameter space for the family of distributions is also three dimensional. The parameter value corresponding to a given distribution is, by definition, the centre of the spatiotemporal volume from which the distribution is obtained. The Fisher-Rao metric is used to find distributions that are near translates of each other in space time. The corresponding translations provide estimates of the optical flow vectors. Experiments with event camera measurements show that excellent estimates of the optical flow can be obtained, provided the flow does not fluctuate too rapidly.

The Fisher-Rao algorithm requires a sufficient number of measurements to establish the probability distributions from which the optical flow is obtained. In the experiments reported in Sections 5, 6 and 7, time slices of several hundred milliseconds are required to accumulate the measurements. The event cameras native time precision is the order of a microsecond. An optical flow algorithm to meet this time precision might be obtained using a sliding window technique, but this is a topic for future research.

Acknowledgements This work received support from LABEX LIFESENSES [ANR-10-LABX-65], managed by the French state funds (ANR) within the Investissements d'Avenir program [ANR-11-IDEX-0004-02]

References

1. Akolkar, H., Meyer, C., Clady, Z., Marre, O., Bartolozzi, C., Panzeri, S., Benosman, R. What can neuromorphic event-driven precise timing add to spike-based pattern recognition? *Neural Computation*, vol. 27, pp. 1-33 (2015).
2. Amari, S-I. *Differential-geometrical Methods in Statistics*. Lecture Notes in Statistics, Springer-Verlag, Berlin (1985).
3. Bardow, P., Davison, A. J., Leutenegger, S. Simultaneous optical flow and intensity estimation from an event camera. *Computer Vision and Pattern Recognition CVPR* (2016).
4. Barranco, F., Fermüller, C., Aloimonos, Y. Contour motion estimation for asynchronous event-driven cameras. *Proceedings of the IEEE*, vol. 102, no. 10, pp. 1537-1556 (2014).
5. Barranco, F., Fermuller, C., Aloimonos, Y., Delbruck, T. A dataset for visual navigation with neuromorphic methods. *Frontiers in Neuroscience*, vol. 10, article 49, DOI: 10.3389/fnins. 2016.00049 (2016)
6. Benosman, R., Ieng, S.-H., Clerq, C., Bartolozzi, C., Srinivasan, M. Asynchronous frameless event-based optical flow. *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, pp. 32-37 (2012).
7. Benosman, R., Clercq, C., Lagorce, X., Ieng, S.-H., Bartolozzi, C. Event-based visual flow. *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 2, pp. 407-417 (2014).
8. Berner, R., Brandli, C., Yang, M., Liu, S.-C., Delbruck, T. A 240 $\times$ 180 10 mW 12 us latency sparse-output vision sensor for mobile applications. *Proc. Symposium on VLSI Circuits*, C186-C187 (2013).
9. Boahen, K.A. Point-to-point connectivity between neuromorphic chips using address-events. *IEEE Transactions on Circuits and Systems*, vol. 74, no. 5, pp. 416-434 (2000).
10. Bouguet, J.Y. Camera Calibration Toolbox for Matlab (2015). http://www.vision.caltech.edu/bouguetj/. Accessed 14 August 2019.
11. Brosch, T., Tschechne, S., Neumann, H. On event-based optical flow detection. *Frontiers in Neuroscience*, vol. 9 (2015).
12. Brosch, T., Tschechne, S., Sailer, R., von Egloffstein, N., Abdul-Kreem, L. I., Neumann, H. On event-based motion detection and integration. *Proc. 8th International Conf. on Bio-inspired Information and Communications Technology, BICT 2014* (2015).
13. Carneiro, J., Ieng, S.-H., Posch, C., Benosman, R. Asynchronous event-based 3D reconstruction from neuromorphic retinas. *Neural Networks*, vol. 45, pp. 27-38 (2013).
14. Cover, T.M., Thomas, J.A. *Elements of Information Theory*. 2nd Edition, Wiley-Interscience (2006).
15. Fortun, D., Bouthemy, P., Kervrann, C. Optical flow modeling and computation: a survey. *Computer Vision and Image Understanding*, Issue C, vol. 134, pp. 1-21 (2015).
16. Gallego, G., Rebecq, H., Scaramuzza, D. A unifying contrast maximization framework for event cameras, with applications to motion, depth, and optical flow estimation. *Proceedings IEEE Conference on Computer Vision and Pattern Recognition CVPR* (2018).
17. Gallego, G., Delbruck, T., Orchard, G., Bartolozzi, C., Taba, B., Censi, A., Leutenegger, S., Davison, A., Conradt, J., Daniilidis, K., Scaramuzza, D. Event-based vision: a survey. *arXiv: 1904.08405v1* (2019).
18. Gehrig, D., Loquercio, A., Derpanis, K. G., Scaramuzza, D. End-to-end learning of representations for asynchronous event-based data. *IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea (South)*, pp. 5632-5642 (2019).
19. Ghosh, R., Gupta, A., Nakagawa, A., Soares, A. B., Thakor, N. V. Spatiotemporal filtering for event-based action recognition. *arXiv: 1903.07067v1* (2019a).
20. Ghosh, R., Gupta, A. K., Tang, S., Soares, A. B., Thakor, N. V. Spatiotemporal feature learning for event-based vision. *arXiv:1903.06923v1* (2019b).
21. Hu, W., Xie, N., Hu, R., Ling, H., Chen, Q., Yan, S., Maybank, S.J. Bin ratio-based histogram distances and their application to image classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, number 12, pp. 2338-2352 (2014).
22. Ieng, S.-H., Carneiro, J., Benosman, R. Event-based 3D motion flow estimation using 4D spatio temporal subspaces properties. *Frontiers in Neuroscience*, vol. 10 (2017).
23. Jia, W., Zhang, H., He, X., Wu, Q. A comparison on histogram based image matching methods. *IEEE Int. Conf. on Video and Signal Based Surveillance, AVSS'06.* (2006)

24. Kogler, J., Humenberger, M., Sulzbachner, C. Event-based stereo matching approaches for frameless address event stereo data. *Proceedings of the 7th International Conference on Advances in Visual Computing*, Lecture Notes in Computer Science, vol. 6938, pp. 674-685 (2011).
25. Kullback, S. *Information Theory and Statistics*, John Wiley and Sons (1959).
26. Lagorce, X., Meyer C., Ieng, S.H., Filliat, D., Benosman, R. Asynchronous event-based multikernel algorithm for high-speed visual features tracking. *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, pp.1-12. doi:10.1109/TNNLS 2014.2352401 (2015)
27. Lazzaro, J., Wawrzynek, J. A multi-sender asynchronous extension to the AER protocol. *Proceedings Sixteenth Conference on Advanced Research in VLSI* (1995).
28. Liu, M., Delbruck, T. Adaptive time-slice block-matching optical flow algorithm for dynamic vision sensors. *British Machine Vision Conference (BMVC)* (2018).
29. Mahowald, M. A. VLSI Analogs of Neuronal Visual Processing: A Synthesis of Form and Function. PhD Thesis, California Institute of Technology (1992).
30. OptiTrack Documentation Wiki (2017) http://v110.wiki.optitrack.com/index.php?title=OptiTrack_Documentation_Wiki. Accessed 14 August 2019.
31. Perez-Carrasco, J.A., Zhao, B., Serrano, C., Acha, B., Serrano-Gotarredona, T., Chen, S., Linares-Barranco, B. Mapping from frame-driven to frame-free event-driven vision systems by low-rate rate coding and coincidence processing–application to feed forward ConvNets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2706-2719 (2013).
32. Posch, C., Matolin, D., Wohlgenannt, R., Hofstätter, M., Schön, P., Litzenberger, M., Bauer, D., Garn, H. Live demonstration: asynchronous tie-based image sensor (ATIS) camera with full-custom AE processor. *Proceedings IEEE International Symposium on Circuits and Systems* (2010).
33. Rogister, P., Benosman, R., Ieng, S.H., Lichtsteiner, P., Delbruck, T. Asynchronous event-based binocular stereo matching. *IEEE Transactions on Neural Networks*, vol. 23, pp. 347-353 (2011).
34. Rueckauer, B., Delbruck, T. Evaluation of event-based algorithms for optical flow with ground-truth from inertial measurement sensor. *Frontiers in Neuroscience*, vol. 10 (2016).
35. Shen, D. Image registration by local histogram matching. *Pattern Recognition*, vol. 40, pp. 1161-1172 (2007).
36 Szeliski, R. *Computer Vision: algorithms and applications*. Springer-Verlag London Limited (2011).
37. Wedel, A., Cremers, D. *Stereo Scene Flow for 3D Motion Analysis*. Springer Verlag London Ltd (2011).
38. Zhu, A. Z., Thakur, D., Özaslan, T., Pfrommer, B., Kumar, V., Daniilidis, K. The multivehicle stereo event camera dataset: an event camera dataset for 3D perception. *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2032-2039, (2018a).
39. Zhu, A. Z., Yuan, L., Chaney, K., Daniilidis, K. EV-flownet: self-supervised optical flow estimation for event based cameras. *Robotics: Science and Systems (RSS)*, (2018b).
40. Zhu, A. Z., Yuan, L., Chaney, K., Daniilidis, K. Unsupervised event-based learning of optical flow, depth, and egomotion. *Proceedings IEEE Conference on Computer vision and Pattern Recognition, CVPR* (2019)

A Ground Truth Flow Using OptiTrack

The world reference frame is by default the OptiTrack's frame. Poses relative to the world reference frame are defined by a rotation matrix and a translation vector. For example, the pose of the camera casing is specified by the pair $(R_{ca}, \mathbf{T}_{ca})$, in which R_{ca} is the rotation matrix and $\mathbf{T}_{ca}$ is the translation vector. If a point has coordinates $\mathbf{X}$ in the world reference frame then the coordinates of the point in the camera casing frame are given by

$$R_{ca}\mathbf{X} + \mathbf{T}_{ca}.$$

The ground truth measurement of the optical flow is based on the following information.

- the camera casing pose, $(R_{ca}, \mathbf{T}_{ca})$ provided by Optitrack,
- the planar pattern pose, $(R_p, \mathbf{T}_p)$ provided by Optitrack,
- the camera pose, $(R_c, \mathbf{T}_c)$, which is not provided by Optitrack; it has to be updated as the camera moves.

The camera and the planar pattern move independently. In order to calculate the projections of the planar pattern points into the camera, it is necessary to update the camera projection matrix $P = K[R, \mathbf{T}]$, where $(R, \mathbf{T})$ is the pose of the camera relative to the pattern and K is a 3×3 matrix of intrinsic parameters.

Standard calibration techniques provided by Bouguet (2015) are used at time $t = 0$, i.e. before the acquisition starts, to estimate K and the initial pose $(R, \mathbf{T})|_{t=0}$ of the camera relative to the pattern. A point with coordinates $\mathbf{X}$ in the world reference frame has coordinates $R_p\mathbf{X} + \mathbf{T}_p$ in the pattern frame, and coordinates

$$R(R_p\mathbf{X} + \mathbf{T}_p) + \mathbf{T} \tag{16}$$

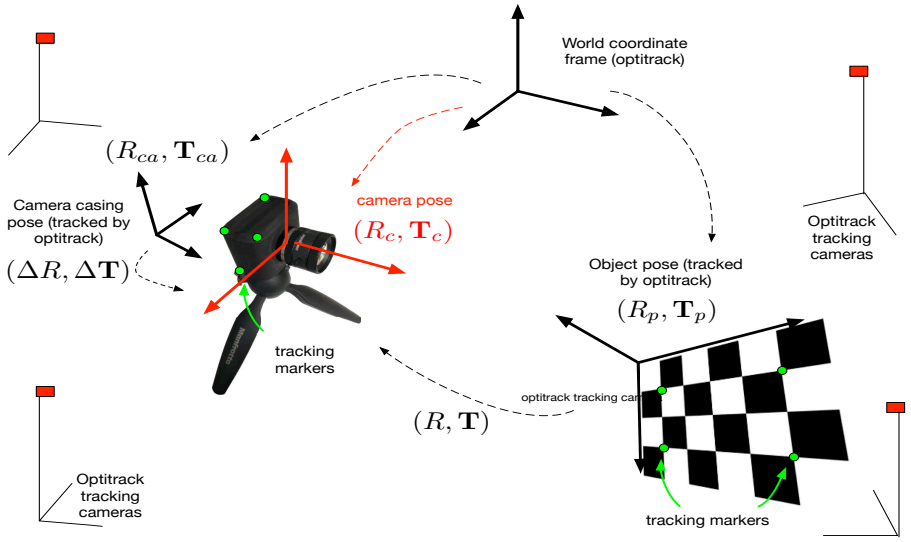


Fig. 13 Relative poses of the components in the tracking setup for obtaining the ground truth optical flow.

in the frame defined by the initial pose of the camera. It follows that

$$(R_c, \mathbf{T}_c)|_{t=0} = (RR_p, R\mathbf{T}_p + \mathbf{T})|_{t=0}.$$

Let $(\Delta R, \Delta \mathbf{T})$ be the pose of the camera with respect to the casing. A point with coordinates $\mathbf{X}$ in the world coordinate frame has coordinates $R_c\mathbf{X} + \mathbf{T}_c$ in the camera frame. It follows from the definition of $(\Delta R, \Delta \mathbf{T})$ that $\mathbf{X}$ also has coordinates

$$\Delta R(R_{ca}\mathbf{X} + \mathbf{T}_{ca}) + \Delta \mathbf{T}$$

in the camera frame, thus

$$(\Delta R, \Delta \mathbf{T}) = (R_c R_{ca}^\top, \mathbf{T}_c - R_c R_{ca}^\top \mathbf{T}_{ca})|_{t=0}. \quad (17)$$

The pose $(\Delta R, \Delta \mathbf{T})$ is constant when the camera moves. It is used to estimate the camera pose when the casing pose changes. A rearrangement of (17) yields

$$(R_c, \mathbf{T}_c) = (\Delta R\, R_{ca}, \Delta \mathbf{T} + \Delta R\, \mathbf{T}_{ca}).$$

Finally, when the camera and the pattern both move, the coordinates of $\mathbf{X}$ in the camera frame are given by $R_c\mathbf{X} + \mathbf{T}_c$ and also by (16). It follows that

$$(R, \mathbf{T}) =$$

$$(\Delta R\, R_{ca} R_p^\top, \Delta \mathbf{T} + \Delta R\, \mathbf{T}_{ca} - \Delta R\, R_{ca} R_p^\top \mathbf{T}_p).$$

Fig. 13 shows all the coordinate frames used for the pose estimation. The camera pose and the camera projection matrix $P = K[R, \mathbf{T}]$ are updated continuously. With these requirements, the projections of known 3D points to the camera's image can be computed at the acquisition frequency of the OptiTrack ($120Hz$).

B Algorithms

The Fisher-Rao algorithm for estimating optical flow is summarised by the following three algorithms. Algorithm 1 takes a set of events as input and returns a list of smoothed blocks. Algorithm 2 takes a smoothed block as input and returns a Fisher-Rao matrix. Algorithm 3 takes a Fisher-Rao matrix as input and returns an optical flow vector.

B.1 Algorithm 1

Input: list E of events, odd positive integers m, n, time t, time interval τ, threshold f, $\epsilon > 0$, standard deviation σ, dimensions x_{max}, y_{max} of the pixel array.
Output: list of smoothed blocks

1. $\tilde{E} = \{((i,j),s) \in E, |s-t| \le \tau(n+1)/2\}$.
2. Define the $x_{max} \times y_{max} \times (n+2)$ array A_t by
   $A_t(i,j,k) =$
   $\#\{((i,j),s) \in \tilde{E}, k = \lfloor \tau^{-1}(s-t) \rfloor + (n+3)/2\}$
3. Define the blocks $\tilde{a}_t(q)$ for pixels $q = (q_1, q_2)$ by
   $\tilde{a}_t(q) = A_t(q_1 - (m+3)/2 : q_1 + (m+3)/2,$
   $q_2 - (m+3)/2 : q_2 + (m+3)/2)$
4. n_q = number of non-zero entries in $\tilde{a}_t(q)$.
5. $C_t = \{q, n_q \ge (m+2)^2(n+2)f\}$.
6. Add ϵ to all entries of A_t.
7. M = Gaussian mask with covariance
   $\sigma^2\, IdentityMatrix$.
8. B_t = convolve A_t with M
9. Define the smoothed blocks $\tilde{b}_t(q)$ for pixels $q = (q_1, q_2)$ by
   $\tilde{b}_t(q) = B_t(q_1 - (m+3)/2 : q_1 + (m+3)/2,$
   $q_2 - (m+3)/2 : q_2 + (m+3)/2).$
10. **Return** $\{\tilde{b}_t(q), q \text{ in } C_t\}$.

B.2 Algorithm 2

Input: A smoothed block $\tilde{b}_t(q)$.
Output: Fisher-Rao matrix at q.

1. Extract from $\tilde{b}_t(q)$ the 27 probability distributions g_a for a in $\{-1,0,1\}^3$.
2. Calculate the Kullback-Leibler divergences $D(0\|a)$ for a in $\{-1,0,1\}^3$.
3. Find the symmetric matrix J that minimises
   $$\sum_{a \in \{-1,0,1\}^3} \|D(0\|a) - 2^{-1} a J a^\top\|^2.$$
4. **Return** J

B.3 Algorithm 3

Input: Fisher-Rao matrix J, parameters $\beta_1, \beta_2, maxFlow$.
Output: Optical flow vector (u, v).

1. Obtain the eigenvalues $\lambda_1 \geq \lambda_2 \geq \lambda_3$ of J.
2. If $\lambda_1 < \beta_1 \lambda_3$ or $\lambda_2 < \beta_2 \lambda_3$ then stop the calculation.
3. Find the eigenvector $w = (w_1, w_2, w_3)$ of J with eigenvalue λ_3.
4. $(u, v) = (w_1/w_3, w_2/w_3)$.
5. If $\|(u, v)\| > maxFlow$, then stop the calculation.
6. **Return** (u, v)