A Unified Approach for Multi-Object Triangulation, Tracking and Camera Calibration

Jeremie Houssineau, Daniel E. Clark, Spela Ivekovic, Chee Sing Lee, and Jose Franco

***Abstract*—Object triangulation, 3-D object tracking, feature correspondence, and camera calibration are key problems for estimation from camera networks. This paper addresses these problems within a unified Bayesian framework for joint multi-object tracking and camera calibration, based on the finite set statistics methodology. In contrast to the mainstream approaches, an alternative parametrization is investigated for triangulation, called disparity space. The approach for feature correspondence is based on the probability hypothesis density (PHD) filter, and hence inherits the ability to handle the initialization of new tracks as well as the discrimination between targets and clutter within a Bayesian paradigm. The PHD filtering approach then forms the basis of a camera calibration method from static or moving objects. Results are shown on simulated and real data.**

***Index Terms*—Camera calibration, disparity space, finite set statistics.**

INTRODUCTION

DETECTION, localization and tracking of an object's state from active sensors, such as, e.g., radar, range-finding laser and sonar, are usually determined from the sensor measurements using a stochastic filter, such as the Kalman filter [26], to provide statistically optimal estimates. When the use of active sensors is not possible, passive sensors, such as cameras, are the alternative.

Calculating the distance of objects from cameras requires triangulation. The traditional means of triangulation from a pair of image observations are well known if the observations of the object are perfect, in which case the triangulated position can be calculated using knowledge of the sensor geometry [15], also known as the camera projection matrix, obtained through the process of camera calibration.

The objective of this paper is to describe a statistical framework for joint 3-D object state estimation and camera calibration, which considers both the geometry and the observation characteristics of the cameras. The framework presented makes use of a proxy state space, called disparity space, which allows for parts of the estimation process to be expressed in linear Gaussian form, thereby enabling the use of the Kalman filter methodology.

The proposed framework encompasses a logical hierarchy of algorithms for estimation from noisy image measurements and addresses the following research problems: single-object triangulation, single-object tracking, multi-object triangulation, multi-object tracking, and camera calibration. It builds on two existing approaches: localization from non-rectified cameras [19] and sensor calibration based on the Probability Hypothesis Density (PHD) filter [40]. The novel contribution is the generalization of [19] to the estimation of moving objects from non-rectified cameras and the use of this approach within the existing sensor calibration technique [40], which has not been applied to the case of camera networks, to obtain a unified Bayesian framework for joint multi-object tracking and camera calibration, applicable to an arbitrary camera setup and an arbitrary number of objects.

The statistical framework is presented in a series of steps, as follows. First, the problem of triangulation from cameras and the concept of disparity space are described in Section I, followed by a discussion on the representation of object-state and object-measurement uncertainty in Section II. The simplest and most constrained case of a single-object state estimation from calibrated cameras is then considered in Section III, followed by the case of multi-object state estimation from calibrated cameras in Section IV, and finally joint multi-object state estimation and camera calibration in Section V. Experimental results on simulated and real data are shown in Section VI.

I. TRIANGULATION FROM CAMERAS

Triangulation is of importance in various engineering applications, e.g., surveying, navigation, metrology, astrometry, binocular vision and target tracking, and is the fundamental estimation problem underlying all of the algorithms presented in this paper. The principle of triangulation from a pair of cameras is shown in Fig. 1, where a point $\mathbf{x}$ in the real world $\mathbb{X} = \mathbb{R}^3$ is projected onto the left and right camera image planes, $\mathbb{P}_\ell$ and $\mathbb{P}_\mathrm{r}$, and its respective projections are denoted $\mathbf{z}_\ell$ and $\mathbf{z}_\mathrm{r}$. Triangulation can then be formulated as the process of recovering the point $\mathbf{x}$ from its projections $\mathbf{z}_\ell$ and $\mathbf{z}_\mathrm{r}$. For this

Manuscript received April 27, 2015; revised November 11, 2015; accepted January 14, 2016. Date of publication February 03, 2016; date of current version April 18, 2016. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Ana Perez-Neira. J. Houssineau has a Ph.D. scholarship sponsored by DCNS and a tuition fees scholarship by Heriot-Watt University. This work was supported by a Royal Society Research Grant. This work was supported by a Royal Academy of Engineering/EPSRC Research Fellowship and the Engineering and Physical Sciences Research Council grant EP/J012432/1.

J. Houssineau, D. E. Clark, and J. Franco are with Heriot-Watt University, Edinburgh, EH14 4AS, U.K. (e-mail: declark@gmail.com).

S. Ivekovic is with the Sophrodyne Ltd., Glasgow G2 4JR, U.K.

C. S. Lee is with BigML Inc., Corvallis, OR 97330 USA.

This paper has supplementary downloadable multimedia material available at http://ieeexplore.ieee.org provided by the authors. This a supplementary video that demonstrates the approach for jointly triangulating and tracking moving objects and simultaneously calibrating the cameras. This material is 3.4 MB in size.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSP.2016.2523454

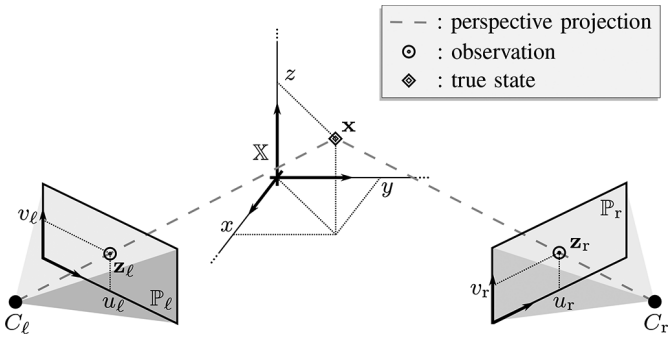


Fig. 1. Two Cameras C_ℓ and C_r observing the same point $\mathbf{x} \in \mathbb{X}$.

purpose, the relation between $\mathbb{X}$ and the image planes must be formulated. Such a formulation can be made easier by using the concepts of projective geometry [11], as described next.

A. Triangulation and Projective Geometry

A point $\mathbf{x} = (x, y)^T$ in $\mathbb{R}^2$ is represented by any triple $\bar{\mathbf{x}} = (\alpha x, \alpha y, \alpha)^T$ with $\alpha \in \mathbb{R} \setminus \{0\}$, and any such triple is referred to as the *homogeneous coordinates* of the point $\mathbf{x}$. A general perspective projection is a linear transformation in homogeneous coordinates, represented by an $(n-1) \times n$ matrix, where n is the dimension of the original projective space. Henceforth, projective equivalents of spaces and points will be denoted with a bar. A perspective projection matrix relates the homogeneous point $\bar{\mathbf{x}} = (x, y, z, 1)^T$ in $\bar{\mathbb{X}}$ with a homogeneous point $\bar{\mathbf{z}} = (\bar{u}, \bar{v}, \bar{w})^T$ in any of the image planes $\bar{\mathbb{P}}_\ell$ and $\bar{\mathbb{P}}_\mathrm{r}$ through a matrix-vector product:

$$\bar{\mathbf{z}} \propto P\bar{\mathbf{x}}, \tag{1}$$

where P is a 3×4 matrix and where "$\propto$" refers to equality up to a scaling factor. Homogeneous coordinates simplify the notation needed to describe perspective projections and allow for projective-geometric concepts such as points and lines at infinity [15]. For the purposes of Bayesian estimation, however, the perspective projection must be expressed in Euclidean coordinates, in order to allow for a meaningful definition of a distance between points, namely the Euclidean distance. The point $\bar{\mathbf{z}}$ is then expressed in Euclidean coordinates as $\mathbf{z} = (u, v)^T = (\bar{u}/\bar{w}, \bar{v}/\bar{w})^T$, which is thus a nonlinear function of the coordinates of the real-world point $\mathbf{x}$. If P is the projection onto the left (resp. right) image plane, then $\mathbf{z}$ will be the point $\mathbf{z}_\ell$ (resp. $\mathbf{z}_\mathrm{r}$).

Triangulation is typically performed in 3-D directly, since we are generally interested in the object's state expressed with respect to the world coordinate system. However, Bayesian estimation requires the modelling of uncertainties, as described in Section II, and in 3-D, due to the nonlinear nature of the perspective projection, uncertainties will tend to be highly range-dependent [42] and the possible distance of the object from the cameras might become unbounded. These aspects make the integration of uncertainties difficult in the world coordinate system, and a re-parametrization is required. One of the most well-known methods for tackling this problem is referred to as the *inverse depth approach* [34], [7], and has been successfully used for Simultaneous Localization and Mapping (SLAM) problems. It relies on a parametrization in which the uncertainty is more easily quantified as a Gaussian distribution. Although it is applicable to most reasonable SLAM configurations, the performance of the inverse depth approach degrades as the baseline becomes larger [19]. We thus investigate an alternative parametrization, called disparity space, and assess performance against inverse depth in Section VI-A for triangulation from cameras. The principle of disparity space is described in the next section.

B. Disparity Space

The notion of binocular disparity, defined by the difference in the location of an object in two images, arose from research into mammalian visual systems to reflect the horizontal separation of the left and right eyes [24]. Perception of depth is obtained in stereopsis as a consequence of this binocular disparity. The same concept is applied to problems in computer vision for extracting depth information from stereo cameras and researchers have designed algorithms for 3-D estimation from cameras by considering the disparity space as a state space [2], [13], [14], [21], [22].

The concept of disparity space is closely linked to the idea of a rectified camera setup, as shown in Fig. 2 (*cf.* Fig. 1, showing a more general, non-rectified camera setup). Formally, assuming that the projection matrix P_ℓ of the left camera C_ℓ is of the form $P_\ell = K[I\ \mathbf{0}]$, then the pair (C_ℓ, C_r) is called horizontally (resp. vertically) rectified if the projection matrix P_r of the right camera C_r is of the form $P_\mathrm{r} = K[I\ \mathbf{t}]$ where $\mathbf{t} = (b, 0, 0)^T$ (resp. $\mathbf{t} = (0, b, 0)^T$); the parameter b is called the *baseline*. Henceforth, we will consider rectified cameras to be horizontally rectified, as in Fig. 2. Let (C_ℓ, C_r) be the rectified camera pair, let $\mathbb{P}_\ell$ and $\mathbb{P}_\mathrm{r}$ be the respective camera image planes, and let the projections of a real-world point $\mathbf{x} \in \mathbb{X}$ be denoted with $\mathbf{z}_\ell = (u_\ell, v_\ell)^T$ in $\mathbb{P}_\ell$ and $\mathbf{z}_\mathrm{r} = (u_\mathrm{r}, v_\mathrm{r})^T$ in $\mathbb{P}_\mathrm{r}$. The point $\mathbf{x}$ is represented in the disparity space $\mathbb{D} = \mathbb{R}^3$ associated with the rectified camera pair (C_ℓ, C_r) by a point $\mathbf{y}$ of the form

$$\mathbf{y} = (u_\ell, v_\ell, d)^T,$$

where $d = u_\mathrm{r} - u_\ell$ is referred to as the *disparity*, as it measures the difference in the camera views of the point $\mathbf{x}$. The point $\mathbf{y}$ characterizes both the left and right projections, $\mathbf{z}_\ell$ and $\mathbf{z}_\mathrm{r}$, as depicted in Fig. 2.

In the context of projective geometry, it is possible to relate the points $\mathbf{x}$ and $\mathbf{y}$ through a linear transformation P_d as

$$\bar{\mathbf{y}} \propto P_\mathrm{d}\bar{\mathbf{x}}, \tag{2}$$

where $\bar{\mathbf{y}}$ and $\bar{\mathbf{x}}$ denote, as in the previous section, the projective equivalents of the points $\mathbf{y}$ and $\mathbf{x}$.

It is useful to express the transformation P_d in terms of the elements of the camera projection matrices P_r and P_ℓ. As a consequence of the fact that the camera pair is horizontally rectified, it holds that

$$(P_\ell)_{i\cdot} = (P_\mathrm{r})_{i\cdot},$$

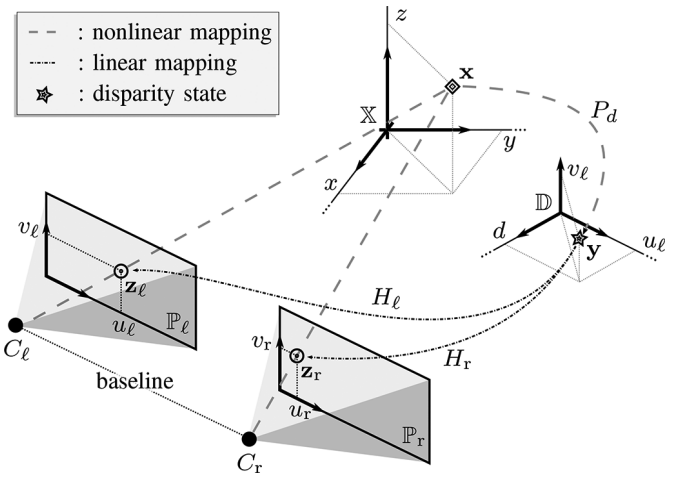


Fig. 2. A rectified camera pair (C_ℓ, C_r) and the related disparity space $\mathbb{D}$.

for $i = 2, 3$, where $(P)_{i\cdot}$ is the ith row of the matrix P. The matrix P_d can then be expressed as

$$P_\mathrm{d} = \begin{bmatrix} (P_\ell)_{1\cdot} \\ (P_\ell)_{2\cdot} \\ (P_\mathrm{r})_{1\cdot} - (P_\ell)_{1\cdot} \\ (P_\ell)_{3\cdot} \end{bmatrix}. \tag{3}$$

The existence of transformation P_d means that the disparity space $\mathbb{D}$ can be used as a proxy space for triangulation from cameras and any point in $\mathbb{D}$ can be converted to its equivalent in $\mathbb{X}$ via the inverse transform of P_d.

To allow for triangulation, a link between the disparity space and the image planes must also be established. With the rectified camera setup, the point $\mathbf{y}$ is projected onto the left- and right-camera image plane, $\mathbb{P}_\ell$ and $\mathbb{P}_\mathrm{r}$, by applying the respective orthographic projections, H_ℓ and H_r, defined as

$$H_\ell = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad \text{and} \quad H_\mathrm{r} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}. \tag{4}$$

In summary, the disparity space associated with a pair of rectified cameras allows for expressing the process of observation (1) as a linear mapping (4), while maintaining a one-to-one correspondence with the real world $\mathbb{X}$, as shown in (2). As the concept of disparity is related to the concept of inverse depth, the disparity space $\mathbb{D}$ inherits from the advantages of the inverse-depth parametrization [7], [34], but the fact that it also enables a linear projection onto the image planes $\mathbb{P}_\ell$ and $\mathbb{P}_\mathrm{r}$ makes it particularly suitable for Bayesian tracking.

II. Representing Uncertainty

The purpose of this section is to describe the sources of uncertainty in an object's 3-D state when observed from cameras, for a static object in Section II-A and for a moving object in Section II-B.

A. Static Object

The most common approach in Bayesian tracking is to assume that objects are point-like. This is justified in radar applications by the relatively small extent of the objects in the scene, when compared to the radar resolution. In such a case, if an object of interest is static, its state can be described by its position, represented by a point state in $\mathbb{X}$ [39].

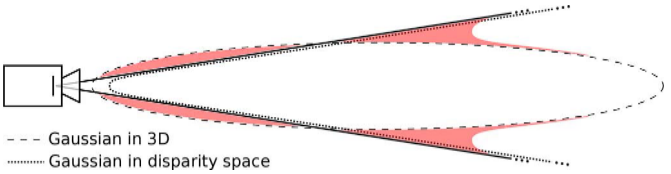


Fig. 3. Modeling of uncertainty in triangulated camera observations. The solid lines show the actual uncertainty in 3-D, defined by the camera's field of view. When the Gaussian uncertainty in camera observations is mapped from the image space, via disparity space $\mathbb{D}$ (4), into the 3-D space $\mathbb{X}$ (3), it takes on a distinctly non-Gaussian nature, as shown by the dotted curve. In contrast, the corresponding Gaussian uncertainty in 3-D is shown by the dashed ellipse and the difference between the Gaussian and non-Gaussian representation is highlighted in red.

When the sensor is a camera, the extent of the objects is often observable, so that the shape of the objects can be estimated [27]. Yet, in the context of camera calibration, the estimation of the extent, or of the shape, of the objects is not always desirable, as it significantly increases the difficulty of the problem without directly contributing to the convergence of the estimation of calibration parameters. We approach this problem by modelling the extended observation of an object on the camera image planes as an uncertainty on the point-like object state, which affects the estimation in a similar way to a point spread function.

Even if an object is actually point-like, an important source of uncertainty stems from the observation process itself, namely the camera observations. The fact that image measurements are inherently noisy, and hence estimation from them requires statistical methodology, has been recognized by many researchers. The observation errors in $\mathbb{P}_\ell$ and $\mathbb{P}_\mathrm{r}$ are modelled as Gaussian, which is generally a reasonable assumption [42]. It follows that the corresponding uncertainty in the space $\mathbb{X}$ is non-Gaussian, as illustrated in Fig. 3. This raises the question of how to characterise the distribution p describing the state of the object in $\mathbb{X}$. Although the Gaussian distribution is very popular, it is clearly not appropriate in this context. The choice of a good model for p is further complicated by the fact that the uncertainty in the triangulated object state is range-dependent, i.e., *heteroscedastic*. In this situation, one typically resorts to particle representations [3] to approximate p.

However, the particle representation of p also has its limitations. One of the most serious limitations is its inability to represent objects that are infinitely far away from the camera. In this case, the support of the distribution p is not bounded and infinitely many particles are required to represent it fairly.

The inapplicability of the usual representations to modelling the distribution p in $\mathbb{X}$ motivates the use of *another state space*, the *disparity space*, in which this can be achieved more easily.

As shown in the previous section, the disparity space $\mathbb{D}$ is related to the camera image planes $\mathbb{P}_\ell$ and $\mathbb{P}_\mathrm{r}$ via a linear transformation (4). It follows directly that a Gaussian uncertainty in these image planes back-transforms into a Gaussian distribution on $\mathbb{D}$. The fact that $\mathbb{D}$ is also in one-to-one relation with $\mathbb{X}$ through (2) makes the disparity space a suitable space for the representation of the uncertainty for purposes of 3-D estimation. Estimating the position of the object of interest can then

be achieved via a Kalman filter update, as demonstrated in [8] and [21].

B. Dynamic Object

If the object of interest is dynamic, its state will include its position in $\mathbb{X}$, as well as parameters modelling its dynamics. The dimensionality of the state space depends on the type of dynamics required to model the motion of the object. For instance, if the object is assumed to have a constant, but unknown, velocity, then the state of the object is a vector in $\mathbb{R}^6$, where the 2×3 coordinates represent the object's position and velocity in $\mathbb{X}$. Therefore, let $\hat{\mathbb{X}} = \mathbb{R}^6$ be the space of vectors of the form $\mathbf{x} = (x, y, z, \dot{x}, \dot{y}, \dot{z})^T$, where $\dot{x} = dx/dt, \dot{y} = dy/dt$ and $\dot{z} = dz/dt$. The disparity space $\mathbb{D}$ also has to be extended to model velocity, and we define $\hat{\mathbb{D}} = \mathbb{R}^6$ as the space of vectors of the form $\mathbf{y} = (u_\ell, v_\ell, d, \dot{u}_\ell, \dot{v}_\ell, \dot{d})$, with $\dot{u}_\ell, \dot{v}_\ell$ and $\dot{d}$ defined as time derivatives of u_ℓ, v_ℓ and d.

Let $\mathbb{T} = \mathbb{N}$ be the set of time steps. The dynamics of the object of interest are usually uncertain and are modelled by a Markov transition $M_{t+1\,|\,t}$ from $\hat{\mathbb{X}}$ to $\hat{\mathbb{X}}$, such that if the object is at point $\mathbf{x} \in \hat{\mathbb{X}}$ at time $t \in \mathbb{T}$, then the probability for it to be at point $\mathbf{x}'$ at time $t+1$ is $M_{t+1\,|\,t}(\mathbf{x}' \,|\, \mathbf{x})$.

The uncertainty associated with the object's dynamic transition from t to $t+1$ is often assumed to be Gaussian in $\hat{\mathbb{X}}$. As discussed in Section II-A, however, the uncertainty in the position of the object is more naturally represented as a Gaussian in $\mathbb{D}$. This raises the following question: how to relate these two types of uncertainty in the estimation process?

Denoting with p_t the Gaussian distribution on $\hat{\mathbb{D}}$ representing the state of the object at time t, the proposed solution to this question can be divided into 6 steps:

a) Sample a particle representation of p_t in $\hat{\mathbb{D}}$;
b) Map this representation into $\hat{\mathbb{X}}$;
c) Apply the Markov transition $M_{t+1\,|\,t}$ in $\hat{\mathbb{X}}$;
d) Map the resulting particle representation back into $\hat{\mathbb{D}}$;
e) Recover the Gaussian distribution $p_{t+1\,|\,t}$ by computing the statistics of the resulting representation in $\hat{\mathbb{D}}$;
f) Compute p_{t+1} by applying the Kalman update in $\hat{\mathbb{D}}$.

This approximation will be consistent with the proposed update method as long as propagating the particles through the Markov transition kernel in 3-D preserves the shape of the distribution in disparity space, which is required for the basic Kalman update of step *f*). This can be shown experimentally by testing statistically whether the particle cloud in disparity space is Gaussian after prediction. A robust test to verify whether an empirical distribution is multivariate normal is the BHEP test [16], which compares the empirical characteristic function of the sample residuals with the theoretical characteristic function of the multivariate normal distribution. To perform this test, a particle cloud was initialized in $\hat{\mathbb{D}}$ from a simulated measurement, with velocity initialized in 3-D with a Gaussian distribution. The Gaussianity of the sample was tested using the BHEP test to evaluate multivariate normality for increasing lengths of prediction time. The resulting p-values averaged over 100 Monte Carlo (MC) runs can be seen in Fig. 4(a). As it can be seen, the distribution can still be considered Gaussian after over a second (at 24 fps), which suggests that the approximation is suitable for the estimation process as long as observations are relatively frequent. The same test was carried out on the point clouds in 3-D, but these yielded p-values of near zero every time and are not displayed.

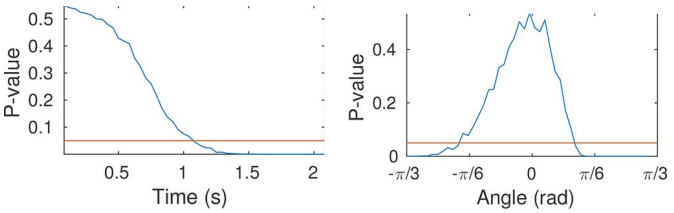
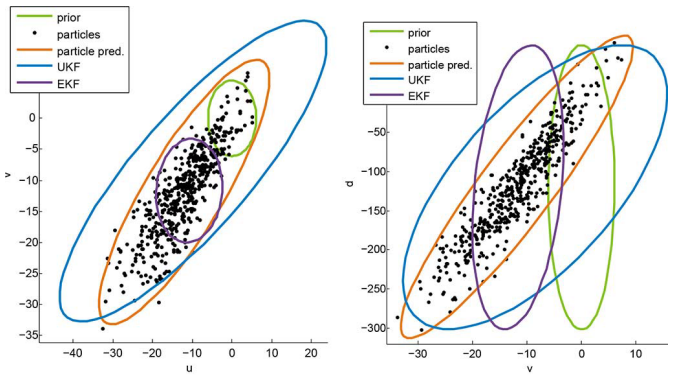


Fig. 4. Averaged p-values of the BHEP test over 100 MC runs for (a) different lengths of prediction time and (b) different relative angles between the camera pair ($-\pi/3, \pi/3$ radians). The red line is the 5% confidence threshold, under which the test fails to assert Gaussianity.

Although the approach described above (see also Algorithm 1) bears some similarity with the Unscented Kalman Filter (UKF) [25], in the sense that we are approximating a Gaussian distribution with particles, the important difference is that samples are drawn randomly from the posterior, which enables us to maintain the nonlinearity when re-parametrizing. The reason for not using the UKF itself, or even the Extended Kalman Filter (EKF) [23], is that the non-linearity in the observation model is too pronounced to be fairly represented by a point (EKF) or by a set of σ-points (UKF). The approximation that is applied in step *e*) above, by recovering a Gaussian distribution from the particle representation, might be very optimistic, yet the objective is to be explicitly aware of the uncertainty, which might not be the case with the UKF and EKF. This aspect is exemplified in Fig. 5, where the distribution before and after prediction in the u-v and v-d planes is displayed for several prediction methods. In the case depicted, the EKF manages to capture the overall motion, as the mean of the associated Gaussian distribution and the mean of the set of particles seem to match, yet, it fails to understand the evolution of the uncertainty and clearly underestimates it in the u-v plane. In the case of UKF, it appears that even though the shift of the mean and the general evolution of the uncertainty is better captured than with the EKF, there is still a non-negligible error in the estimation of the covariance. This is mainly due to the noise on the motion model in $\mathbb{X}$, which becomes non-linear in the disparity space $\mathbb{D}$. The particle prediction, which relies on 500 particles, manages to capture both the non-linearity of the motion and of the associated noise.

III. Single-Object Estimation

3-D object tracking refers to the problem of estimating the position *and* dynamics of the object at each point in time, based on a sequence of noisy measurements which originates from one or several sensors. This definition coincides with the mathematical theory of stochastic or Bayes filtering and these terms have become synonymous in the sensor fusion community due to the widespread deployment of filtering techniques in practical applications. The importance of fusing the estimates in the case where the considered sensors are cameras is recognized as an instrumental way of reducing the uncertainty in triangulated estimates [1], [12].



(a) (b)

Fig. 5. Transformation of a Gaussian prior (green) for the prediction of a dynamical object. The model is constant velocity with $\dot{x} = \dot{y} = 2$ cm·s^{-1} and $\dot{z} = 0.5$ cm·s^{-1} and a noise with variance 0.08 cm^2·s^{-2}. (a) u-v plane in $\mathbb{D}$, (b) v-d plane in $\mathbb{D}$.

Statistical methods for estimating the uncertainty in 3-D, such as finding the Cramer-Rao Lower Bound (CRLB), have previously been investigated [5], [6], [51], though researchers often transform the measurement or linearize the system before estimating the uncertainty, thereby losing the underlying statistical sensor characteristics in the process. Indeed, like in the case of triangulation, it is usual to express the tracking problem in the 3-D Euclidean space, since the operator is ultimately interested in knowing the state of the object, such as position and velocity, in the world coordinate system. There is a long history of using Kalman filters and their extensions to non-linear systems, such as EKF, for solving 3-D motion estimation from images ([31], p. 437). Unfortunately, in the presence of the nonlinear observation model and range-dependent uncertainty in 3-D estimates, the usual assumption of Gaussianity in the Kalman filter leads to a poor characterisation of the posterior distribution in 3-D, particularly in the depth estimate, and hence the use of the Kalman filter and its non-linear variants will almost inevitably lead to filter divergence and poor tracking performance. This is particularly acute for targets in long-range stereo applications [42].

Because of the aforementioned disadvantages of dynamic estimation in 3-D space, we instead turn to dynamic estimation in disparity space. Estimation in disparity space has three key advantages over estimation in 3-D Euclidean space: (i) the projections into the observation space (the two image planes) are linear, (ii) the noise in the state estimate is range-independent, and (iii) the range of the estimated variable is bounded by the image size. Consequently, in disparity space, the position of an object can be estimated with the linear-Gaussian assumptions required for the Kalman filter update, and hence optimally and in closed form. This allows for a straightforward error analysis, for example, the computation of the Fisher information and CRLB [10]. A solution for the estimation of moving objects from a rectified camera pair is introduced in Section III-A, and an extension to non-rectified cameras is proposed in Section III-B.

A. Rectified Camera Pair

As mentioned in the previous section, the estimation of a single static object can be handled via a Kalman filter update in $\mathbb{D}$. A *particle prediction* between two time steps is used when the object is dynamic (i.e., when its state lives in $\hat{\mathbb{D}}$). Let $(\mathbf{z}_i, R_i)$ be the mean and covariance of an observation at time $t \in \mathbb{T}$ from the camera C_i, with $i \in \{l, \mathrm{r}\}$. The likelihood $L_t^i(\mathbf{z}_i \mid \mathbf{y})$ can be expressed as

$$L_t^i(\mathbf{z}_i \mid \mathbf{y}) = \mathcal{N}(\mathbf{z}_i; H_i\mathbf{y}, R_i),$$

where $\mathcal{N}(\mathbf{x}; \mathbf{m}, P)$ is a normal distribution with mean $\mathbf{m}$ and covariance matrix P, evaluated at point $\mathbf{x}$. If the object is dynamic, the observation matrices (4) have to be suitably augmented. For example,

$$H_\ell = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix},$$

is the observation matrix from $\hat{\mathbb{D}}$ to the left camera image plane $\mathbb{P}_\ell$. Note that the velocity is assumed to be unobserved.

Algorithm 1: Single-object estimation with a rectified camera pair at time $t \in \mathbb{T}$.

Data:
- Gaussian distribution $(\mathbf{y}_{t-1}, Q_{t-1})$,
- Observation $\mathbf{z}_i \in \mathbb{P}_i$ with uncertainty R_i, where i is ℓ or r.

Result: Initialised/updated Gaussian distribution at time t

```
if t == 0 then
    [(y_0, Q_0)] = initialisation[(z_i, R_i)]
else
    if object is static then
        (ŷ_t, Q̂_t) = (y_{t-1}, Q_{t-1});
    else
        [(ŷ_t, Q̂_t)] = particle_prediction[(y_{t-1}, Q_{t-1})]
    end
    [(y_t, Q_t)] = Kalman_update[(ŷ_t, Q̂_t), (z_i, R_i)]
end
```

In order to completely specify an estimation algorithm for the object of interest, the initialization has to be described as well. As we do not assume that the camera pair (C_ℓ, C_r) is synchronised, initialization has to be dealt with using a single camera, say the left camera C_ℓ. Consider that we receive the first observation $\mathbf{z}_\ell = (u_\ell, v_\ell)$ with covariance R_ℓ at time $t = 0$. This observation can be used directly to initialise the first two components of the Gaussian distribution p_0 in $\mathbb{D}$. However, the disparity, and possibly the velocity, are not known a priori and have to be initialized in some other way. The mean of the disparity can be computed by considering the expected distance between the left camera C_ℓ and the object. The variance has to be taken sufficiently large for the disparity 0 to be likely enough, whenever the object is possibly infinitely far away from the camera. As a consequence, negative disparity, which represents objects behind the camera pair, must be included. This is necessary in order to maintain a Gaussian distribution in $\mathbb{D}$ and does not represent an issue in general. The mean and covariance $\mathbf{y}_0$ and Q_0 of the Gaussian distribution p_0 can now be determined, and the estimation carried out, as described in Algorithm 1. In this algorithm, a hat is used to refer to the predicted mean and covariance in order to underline that prediction is necessary in the space $\hat{\mathbb{X}}$ only, where moving objects are modelled.

B. Non-Rectified Camera Pair

Estimating the state of an object from a non-rectified camera pair (C_ℓ, C_r) is a challenging problem, as the linear observa-

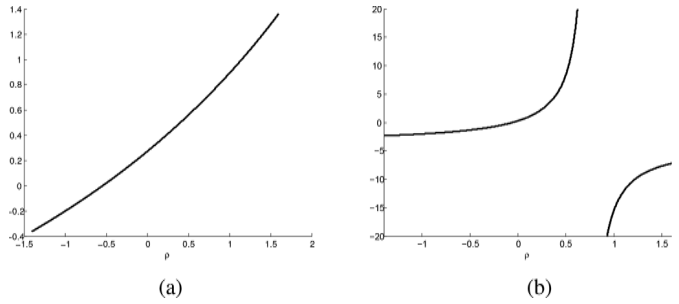


Fig. 6. Illustration of the non-linearity of one of the components of the observation function with respect to inverse-depth ρ, as the camera pair becomes non-rectified. Starting from a rectified configuration with baseline b, the right camera C_{r} is rotated by an angle of $\pi/12$ radians. (a) $b = 0.5$ m. (b) $b = 5$ m.

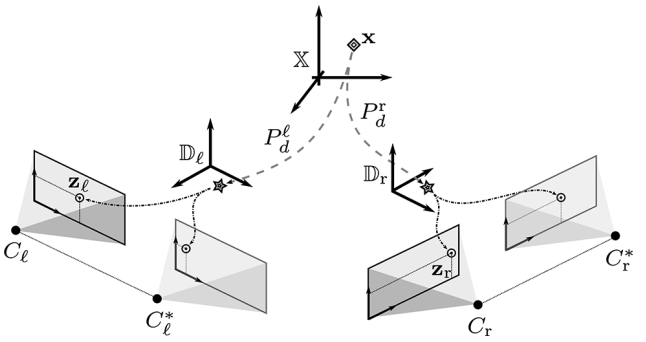


Fig. 7. Projection of a point $\mathbf{x}$ in $\mathbb{X}$ onto the disparity spaces $\mathbb{D}_\ell$ and $\mathbb{D}_{\mathrm{r}}$ and onto the image planes of the two rectified camera pairs (C_ℓ, C_ℓ^*) and $(C_{\mathrm{r}}, C_{\mathrm{r}}^*)$.

tion model obtained from the rectified camera geometry is not available anymore. This aspect is illustrated in Fig. 6, where the nonlinearity of the observation function is shown for two different non-rectified camera pairs. Yet, taking advantage of the approach which applies in the rectified case, and has been detailed in the previous sections, is still beneficial. This idea is described in detail and assessed against the standard inverse-depth parametrization in [19], so that only the underlying principles are restated here.

In the previous section, a particle-based prediction has been used in order to handle the possible motion of the object of interest. In the case of a non-rectified camera pair, a similar idea can be used to map the distribution from a disparity space specifically constructed for the left camera C_ℓ to another disparity space, constructed for the right camera C_{r}.

The properties of disparity spaces are still strong assets, even when considering a single camera. Yet, a disparity space requires two cameras in order to be defined. The idea is then to introduce two *abstract* cameras C_ℓ^* and C_{r}^* that are rectified with respect to the left and right cameras, respectively. These cameras are said to be abstract as they do not exist physically, and hence never produce observations. Two disparity spaces $\mathbb{D}_\ell$ and $\mathbb{D}_{\mathrm{r}}$ are thus defined based on the rectified camera pairs (C_ℓ, C_ℓ^*) and $(C_{\mathrm{r}}, C_{\mathrm{r}}^*)$ and are related to $\mathbb{X}$ via the projective transformations P_{d}^{ℓ} and $P_{\mathrm{d}}^{\mathrm{r}}$, as shown in Fig. 7. The process of predicting a probability distribution while starting from the disparity space $\mathbb{D}_\ell$ (resp. $\mathbb{D}_{\mathrm{r}}$) and arriving into the disparity space $\mathbb{D}_{\mathrm{r}}$ (resp. $\mathbb{D}_\ell$) will be called a *particle move*. Indeed, the principle of this approach is to use particle representations in order to perform the mapping of the Gaussian distribution representing the object of interest from one disparity space to another. Following the same conventions as in Algorithm 1, the principle of the proposed single-object estimation from non-rectified cameras is detailed in Algorithm 2, where the prediction and update of a distribution in the space $\mathbb{D}_i$ is performed for an observation provided by camera C_j, for any indices $i, j \in \{\ell, \mathrm{r}\}$.

When the pair of cameras are rectified with respect to one another, the mapping between $\mathbb{D}_\ell$ and $\mathbb{D}_{\mathrm{r}}$ is linear, and so the particle move preserves the Gaussian shape of distributions in each space. If the cameras are not rectified, the transformation is nonlinear, and so Gaussianity is not guaranteed after mapping from one space to the other. However, it is reasonable to assume that

Algorithm 2: Single-object estimation with a non-rectified camera pair at time $t \in \mathbb{T}$.

Data:
- Gaussian distribution $(\mathbf{y}_{t-1}^j, Q_{t-1}^j)$ in $\mathbb{D}_j$, $j \in \{\ell, \mathrm{r}\}$,
- Observation $\mathbf{z}_i \in \mathbb{P}_i$ with uncertainty R_i, $i \in \{\ell, \mathrm{r}\}$.

Result: Initialised/updated Gaussian distribution at time t

```
if t == 0 then
    [(y_0^i, Q_0^i)] = initialisation[(z_i, R_i)]
else
    if i == j then
        [(ŷ_t^i, Q̂_t^i)] = particle_prediction[(y_{t-1}^j, Q_{t-1}^j)]
    else
        [(ŷ_t^i, Q̂_t^i)] = particle_move[(y_{t-1}^j, Q_{t-1}^j)]
    end
    [(y_t^i, Q_t^i)] = Kalman_update[(ŷ_t^i, Q̂_t^i), (z_i, R_i)]
end
```

the resulting distribution is Gaussian if the extent of non-rectification is not too extreme. To show this, a simulated pair of cameras at a distance of 30 centimeters was used to evaluate the Gaussianity of a cloud of particles after transforming from $\mathbb{D}_\ell$ to $\mathbb{D}_{\mathrm{r}}$. Fig. 4(b) shows the resulting p-values of the BHEP test [16] for the transformed particle cloud as the relative angle between the cameras increases from $-\pi/3$ to $\pi/3$ radians, averaged over 100 MC runs. It can be seen that the approximation preserves Gaussianity for non-rectified cameras for a range of angles from $-\pi/6$ to $\pi/8$, which is a useful working range in everyday applications. Although the distribution is not Gaussian outside of this interval, a Gaussian approximation might still be sufficiently accurate to allow for localizing and tracking the target. For instance, in Section VI-D, calibration is successfully performed with an initial uncertainty on the rotation ranging from $-3\pi/12$ to $\pi/12$.

As mentioned before, this approach for a static object has been assessed in [19]. However, its use for a moving object, as in Algorithm 2, is novel. The performance of this extension will be evaluated, together with other generalizations, in Section VI.

Fig. 8 illustrates the form of the distributions obtained when handling the mapping from $\mathbb{D}_\ell$ to $\mathbb{D}_{\mathrm{r}}$ with different methods. 500 particles have been used to represent the actual distribution in $\mathbb{D}_{\mathrm{r}}$. Once again, the UKF shows inaccuracies in its representation of the objective distribution, even though the noise on the motion is lower than for the example shown in Fig. 5. This can be explained by the non-linearity of the mapping between $\mathbb{D}_\ell$ and $\mathbb{D}_{\mathrm{r}}$, which makes the representation of the uncertainty even

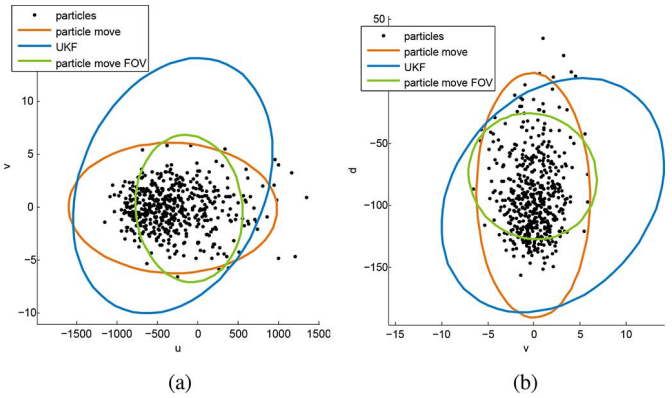


Fig. 8. Transformation of a Gaussian prior in the disparity space $\mathbb{D}_\ell$ to the disparity space $\mathbb{D}_\mathrm{r}$ for the prediction of a dynamical object. The two cameras have the same intrinsic parameters, with a resolution of 800×600 and a focal length of 8 mm. The distance between the two cameras is 200 cm and the yaw is $-\pi/8$ for the left camera and $\pi/8$ for the right camera. The velocity has mean 0 and variance 0.03 $\mathrm{cm}^2 \cdot \mathrm{s}^{-2}$. (a) u-v plane in $\mathbb{D}_\mathrm{r}$, (b) v-d plane in $\mathbb{D}_\mathrm{r}$.

more difficult. Note that, in Fig. 8(a), the range on the axis u is much larger than on the axis v, and actually extends outside of the field of view of the right camera.

When estimating a single object, we can assume that the object is detected and hence the particles outside of the field of view can be discarded before fitting a Gaussian distribution. The result of this operation is represented as an indication by the green ellipse in Fig. 8. This can be justified by the use of a function describing the probability of detection together with the likelihood function, as described in Section IV-C in a multi-object context.

Although we have described the procedure for two cameras, the approach can be straightforwardly extended to more cameras by introducing a disparity space for each camera.

IV. Multi-Object Estimation

Multi-camera multi-object estimation, in computer vision also referred to as the *feature correspondence* [15] and *feature tracking* problem, is a fundamental problem in estimation from images, the solution of which has a wide range of applications from object recognition, camera calibration and 3-D reconstruction to mosaicing, motion segmentation, and image morphing. It is related to the *data association* [4] problem in the sensor fusion literature: both relate to the problem of finding the measurements which correspond to the same object that have come from different sensors, or in dynamical systems, from the same sensor at different times.

In a multi-object environment, this is a challenging task, since we may not know how many objects are in the scene, there may be many false alarms from the sensor, and there may not always be a measurement at each time-step or in each sensor. Methods for reducing the complexity of the problem in the sensor community usually rely on *gating* [4] around the object or measurement to identify possible matches, or using the *epipolar constraint* [15], in the computer vision literature.

Recent developments in the sensor fusion community have enabled practitioners to overcome the computational limitations of combinatorial data association approaches by modelling the system as an integrated multi-object Bayesian estimation problem. A Bayesian solution to the multi-object filtering and estimation problem can be found with Finite Set Statistics (FISST) [33], a set of mathematical tools developed from point process theory, random finite sets, and stochastic geometry.

There are a number of advantages in developing an integrated mathematical framework for multi-object detection and tracking: (i) the number of objects and their locations can both be optimally estimated from multiple sensors; (ii) false alarms/outliers do not need to be explicitly discarded since they will not be confirmed by the model; (iii) the sensors are not required to provide measurements of the objects in each image and the sensor characteristics and frame rates are not required to be the same; and (iv) advance matching of the measurements from each object is not necessary. The FISST approach to multi-sensor multi-object tracking has attracted significant international attention in the sensor fusion community due to the success of practical implementations of first-moment multi-object approximation filters, known as Probability Hypothesis Density (PHD) filters [33], [37], [9].

The advantage of viewing this problem as a multi-object statistical estimation problem and using the PHD filter means that, in addition to providing a rigorous mathematical foundation for multi-object estimation, (i) there is no explicit data association for assigning measurements to targets, (ii) the PHD filter has a linear complexity in the number of targets and the number of measurements. Furthermore, given a video sequence of a static scene, we can recursively apply the multi-object Bayes update on image measurements, using disparity space, and re-parametrize the state estimates into 3-D, which makes the proposed approach directly extendible to the stochastic triangulation of multiple objects in cluttered environments.

The choice of the PHD filter as the underlying multi-object estimation algorithm is motivated by a) its simple formulation which enables an accessible proof of concept for joint multi-object and camera calibration, and b) its principled foundations which allow for the derivation of the corresponding inference for the camera parameters. Moreover, the tracking problems considered in this article are sufficiently simple in terms of detectability to be well handled by the PHD filter. Other multi-object filters could be used in a similar fashion and would allow for handling more sophisticated scenarios. Filters that can be considered include the cardinalized PHD filter [33], the generalized labeled multi-Bernoulli filter [50] or the hypothesized filter for independent stochastic populations [17].

The method presented in this section will underpin the method for camera calibration in the next section, since it provides the likelihood to update the probability density on the sensor parameters. Throughout this section, objects will be considered to be moving according to a constant velocity model, so that the spaces $\mathbb{D}_\ell$ and $\mathbb{D}_\mathrm{r}$ are augmented with velocity and respectively denoted with $\hat{\mathbb{D}}_\ell$ and $\hat{\mathbb{D}}_\mathrm{r}$.

A. General Solution

We consider a population $\mathcal{X}_t$, defined as the set of objects of interest in the scene, at time $t \in \mathbb{T}$. Most often, the size of the population $\mathcal{X}_t$ is not known and might vary in time. Additionally, the correspondences between the estimated population and

the received observations are not generally known. As a consequence, a sufficiently general model has to be constructed in order to allow for the estimation of the population $\mathcal{X}_t$ for any time $t \in \mathbb{T}$.

The most popular estimation framework applicable in this context is the FISST framework [33]. In the following, we provide a brief summary of this framework, necessary to motivate the remainder of the paper, and refer the interested reader to [18], [33] for a more exhaustive description.

Denoting with $p_S(\mathbf{y})$ and $p_D^i(\mathbf{y})$ the probability of survival and the probability of detection from camera $i \in \{\ell, \mathrm{r}\}$ at state $\mathbf{y}$, the FISST framework allows for modelling that:

1) a new set of objects $\mathcal{X}_t^{\mathrm{b}}$ might appear a each time $t \in \mathbb{T}$, so that $\mathcal{X}_t = \mathcal{X}_{t-1} \cup \mathcal{X}_t^{\mathrm{b}}$,
2) every object's motion is independent of the other objects,
3) an object in $\mathcal{X}_{t-1}$ with state $\mathbf{y}' \in \hat{\mathbb{D}}_i$ might disappear from the scene with probability $1 - p_S(\mathbf{y}')$,
4) an object in $\mathcal{X}_t$ with state $\mathbf{y} \in \hat{\mathbb{D}}_i$ can be either non detected with probability $1 - p_D^i(\mathbf{y})$ or detected through the observation $\mathbf{z} \in \mathbb{P}_i$ with probability $p_D^i(\mathbf{y}) L_t^i(\mathbf{z} \,|\, \mathbf{y})$, with $i \in \{\ell, \mathrm{r}\}$, and
5) the set Z_t^i of observations in $\mathbb{P}_i$ at time t contains independent object-originated observations, as well as independent spurious observations, spatially distributed according to the probability density c_i on $\mathbb{P}_i$, and the number of which is driven by a Poisson distribution with parameter λ_i.

We assume that p_D^i only depends on the coordinates (u_i, v_i) in the image plane $\mathbb{P}_i$, so that the choice of state space has no consequence on the probability of detection.

As the correspondences between objects and observations are not assumed to be known, we introduce association functions of the form $\theta : Y \to \phi \cup Z_t^i$, where ϕ is the empty observation. Denoting with Y_z the inverse image of Z_t^i through θ, we assume that the restriction $\theta|_{Y_z} : Y_z \to Z_t^i$ of the function θ is a bijection. The set of such association functions is denoted with Θ.

With these models and assumptions, and following [33], we can proceed to the estimation of the population via the following prediction and update steps:

$$\hat{P}_t^i(Y) = \int \mathbf{M}_{t\,|\,t-1}^{i\,|\,j}(Y \,|\, Y') P_{t-1}^j(Y') \delta Y',$$

$$P_t^i(Y) = \frac{\mathbf{L}_t^i(Z_t \,|\, Y) \hat{P}_t^i(Y)}{\int \mathbf{L}_t^i(Z_t \,|\, Y') \hat{P}_t^i(Y') \delta Y'},$$

where $\int \cdot \delta Y$ refers to the set integral [33], $\hat{P}_t^i(Y)$ and $P_t^i(Y)$ are the predicted and updated multi-object densities describing the probability for the objects in $\mathcal{X}_t$ to be at given points in the set Y of points in $\hat{\mathbb{D}}_i$, and $\mathbf{M}_{t\,|\,t-1}^{i\,|\,j}$ and $\mathbf{L}_t^i$ are the conditional multi-object densities describing prediction and update, with $\mathbf{L}_t^i$ expressed as

$$\mathbf{L}_t^i\left(Z_t^i \middle| Y\right) = e^{-\lambda_i} \left[\prod_{\mathbf{z} \in Z_t^i} \lambda_i c_i(\mathbf{z})\right] \times \left[\prod_{\mathbf{y} \in Y} \left(1 - p_D^i(\mathbf{y})\right)\right] \times \sum_{\theta \in \Theta} \left[\prod_{\mathbf{y} \in Y_z} \frac{p_D^i(\mathbf{y}) L_t^i(\theta(\mathbf{y}) \,|\, \mathbf{y})}{\left(1 - p_D^i(\mathbf{y})\right) \lambda_i c_i(\theta(\mathbf{y}))}\right].$$

The evaluation of every possible association in Θ is extremely costly in practice and the complexity becomes exponential in time. It is therefore useful to avoid resorting explicitly to Θ. This is made possible by reducing the multi-object densities $\hat{P}_t^i$ and P_t^i to their first moment densities $\hat{\mu}_t^i$ and μ_t^i. With additional assumptions, the estimation can be performed using only these first moment densities, and the resulting filter is the PHD filter [32].

B. *The* PHD *Filter*

As stated in the previous section, it is possible, with some assumptions, to propagate only the first moment of the multi-object densities of interest. These assumptions are as follows.

A.1 At any time $t \in \mathbb{T}$, all the objects in $\mathcal{X}_t$ have the same probability density p_t^i on $\hat{\mathbb{D}}_i, i \in \{\ell, \mathrm{r}\}$.

A.2 The cardinality distribution of the set $\mathcal{X}_t$ follows a Poisson distribution.

Under these two assumptions, and following [32], the first-moment density describing the population of interest can be propagated as follows

$$\hat{\mu}_t^i(\mathbf{y}) = \mu_t^{\mathrm{b}}(\mathbf{y}) + \int p_S(\mathbf{y}') M_{t\,|\,t-1}^{i\,|\,j}(\mathbf{y} \,|\, \mathbf{y}') \mu_t^j(\mathbf{y}') \mathrm{d}\mathbf{y}',$$

$$\mu_t^i(\mathbf{y}) = \left(1 - p_D^i(\mathbf{y})\right) \hat{\mu}_t^i(\mathbf{y}) + \sum_{\mathbf{z} \in Z_t^i} \int \frac{p_D^i(\mathbf{y}) L_t^i(\mathbf{z} \,|\, \mathbf{y}) \hat{\mu}_t^i(\mathbf{y})}{\lambda^i c^i(\mathbf{z}) + \int p_D^i(\mathbf{y}') L_t^i(\mathbf{z} \,|\, \mathbf{y}') \hat{\mu}_t^i(\mathbf{y}') \mathbf{y}'},$$

where μ_t^{b} is the first-moment density representing the appearing set of individuals $\mathcal{X}_t^{\mathrm{b}}$.

Two implementations of the PHD filter are available, the Gaussian Mixture PHD filter [48], or GM-PHD filter, and the sequential Monte Carlo PHD filter [49].

As the objective is to incorporate the single-object filter, designed in the previous sections, into a multi-object framework, the choice of a Gaussian mixture implementation of the PHD filter is the most appropriate. The transition $M_{t\,|\,t-1}^{i\,|\,j}$ is then the particle move between the disparity spaces $\hat{\mathbb{D}}_j$ and $\hat{\mathbb{D}}_i$, from time $t-1$ to time t. Note that the use of the Gaussian mixture implementation requires additional assumptions:

A.3 The probability of survival p_S is state-independent.

A.4 The probability of detection p_D is state-independent.

With these assumptions it can be demonstrated [48] that the equations of the PHD filter propagate in closed form a Gaussian mixture of the form:

$$\mu_t^i(\mathbf{y}) = \sum_{k=1}^{N_t} w_k \mathcal{N}\left(\mathbf{y}; \mathbf{y}_k^i, Q_k^i\right).$$

Note that the weight w_k of the kth term in the mixture does not depend on the space in which the Gaussian distribution is expressed.

However, Assumption **A.4** is too strong when considering a pair of cameras, as their field of view might, and will, significantly differ. It is then necessary to relax such an assumption and we discuss this in detail in the next section.

Following the choice of initializing the probability density with the first observation available, we adopt the observation-driven birth, detailed in [20]. Note that previous attempts to use

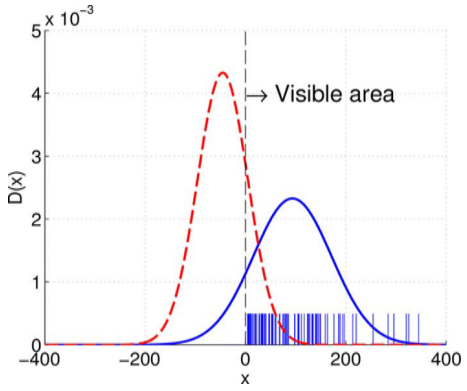


Fig. 9. Gaussian fitting for a state-dependent probability of detection in the 1-D case. $p_D(\mathbf{x}) = 0.9$ for $\mathbf{x} \geq 0$ and $p_D(\mathbf{x}) = 0$ for $\mathbf{x} < 0$. Solid line: detection term—Dashed line: missed-detection term.

the PHD filter with cameras, e.g., [38] or [28], required the scene to be bounded and/or the use of at least 3 cameras. These restrictions limit the impact of the error made when representing the uncertainty by a Gaussian distribution in $\hat{\mathbb{D}}$ and increase the observability of the objects as the problem of triangulation from 3 points of view is better constrained than from only 2.

C. State-Dependent Probability of Detection

As opposed to radar applications, the estimation of multiple objects from a camera pair requires the fields of view to be properly modelled. For this reason, Assumption **A.4** must be relaxed. Once again, we can resort to a solution similar to the particle move, introduced in the previous sections, in order to consider a state-dependent probability of detection.

Formally, the following two Gaussian distributions can be computed for each original Gaussian term in the mixture $\hat{\mu}_t^i$:

- one corresponding to the missed detection term:

$$w_{\circ,k}\mathcal{N}\left(\mathbf{y};\hat{\mathbf{y}}_{\circ,k}^i,\hat{Q}_{\circ,k}^i\right) \approx \left(1-p_D^i(\mathbf{y})\right) w_k\mathcal{N}\left(\mathbf{y};\hat{\mathbf{y}}_k^i,\hat{Q}_k^i\right),$$

- one corresponding to the detection term:

$$w_{\bullet,k}\mathcal{N}\left(\mathbf{y};\hat{\mathbf{y}}_{\bullet,k}^i,\hat{Q}_{\bullet,k}^i\right) \approx p_D^i(\mathbf{y}) w_k\mathcal{N}(\mathbf{y};\hat{\mathbf{y}}_k^i,\hat{Q}_k^i),$$

where the subscripts "$\circ$" and "$\bullet$" indicate missed detection and detection respectively. This is achieved by sampling particles according to the predicted law and then applying the state-dependent probability of detection before computing the mean and covariance of the obtained weighted set of particles. An example of such an approach is depicted in Fig. 9. Denoting by $\hat{\mu}_{\circ,t}^i$ and $\hat{\mu}_{\bullet,t}^i$ the modified missed detection and detection first-moment densities, the PHD update can be expressed as

$$\mu_t^i(\mathbf{y}) = \hat{\mu}_{\circ,t}^i(\mathbf{y}) + \sum_{\mathbf{z}\in Z_t^i} \int \frac{L_t^i(\mathbf{z}\,|\,\mathbf{y})\hat{\mu}_{\bullet,t}^i(\mathbf{y})}{\lambda^i c^i(\mathbf{z}) + \int L_t^i(\mathbf{z}\,|\,\mathbf{y}')\hat{\mu}_{\bullet,t}^i(\mathbf{y}')\mathrm{d}\mathbf{y}'}. \quad (5)$$

Note that the Gaussian distributions which depict objects that are almost surely inside or outside of the field of view can be kept as they are, so that only the weight w_k changes. For instance, if the object is almost surely inside the field of view, $w_{\circ,k} = (1 - p_D^{i,\text{in}})w_k$ and $w_{\bullet,k} = p_D^{i,\text{in}}w_k$, where $p_D^{i,\text{in}}$ is the constant probability of detection within the field of view of the camera C_i.

Equipped with a suitable way of estimating multiple objects from a non-rectified camera pair, we now proceed to describe a solution for the problem of camera calibration in the next section.

V. Camera Calibration From Moving Objects

Camera calibration refers to the estimation of the parameters of the imaging process, such that when two or more views of the same scene are available, the original 3-D scene and its dimensions can be reconstructed by solving an inverse problem. How accurately the original scene can be reconstructed depends on the number of parameters that can be estimated and consequently different calibration methods exist. If some ground-truth knowledge about the scene is provided, e.g., a calibration object with known Euclidean 3-D coordinates, the Euclidean calibration can be performed directly [47]. Alternatively, the so called *stratified approach* is used [43], which gradually refines the calibration from projective to Euclidean.

In practice, a calibration object is not always available and hence the stratified approach, which relies only on the information extracted from the images, is more appropriate. Projective calibration is usually achieved by structure-from-motion techniques [46] which unrealistically assume perfect knowledge of measurement correspondences as an input to the calibration process. This in turn means that such projective calibration implicitly assumes that the estimated correspondences were updated with the correct measurements and the corresponding points are known in at least a certain number of images. The possibility of incorrect data association or correspondence is not considered as such cases are pruned from the input data and similarly, the possibility of incorrect estimation of the number of correspondences is also not considered. As a consequence, useful information is removed from the input data before the calibration process even begins.

To remove the dependency of the calibration method on perfect input data, the calibration can instead be formulated as an extension of the multi-object stochastic estimation problem, discussed in the previous section. In fact, given that the projective camera calibration relies on information obtained from the multi-object state estimation, estimating the multi-object state of an *uncalibrated* dynamic system is inherently suboptimal if the camera parameters are not estimated as a part of the same process.

We propose to address this problem as a doubly-stochastic inference problem [45], where the measurements are conditioned on the multiple object locations, that are in turn conditioned on the relative camera orientation. A similar method using random finite sets has been developed for the related problem of simultaneous localization and mapping for autonomous robot navigation [36], [29], [30], where each object measurement contributes both to a feature in the world and self-localization of the vehicle.

Reliable estimation requires reliable knowledge of the sensor parameters, and thus sensor calibration has been a central problem in multi-object multi-sensor tracking. In the context of FISST, solutions to this problem have been derived recently

[29], [35], [40]. However, these solutions have not been used for calibrating cameras. The objective of this section is to extend the multi-object estimation framework, described in the previous section, and present a method for calibrating a non-rectified camera pair by formulating a joint multi-object tracking and camera calibration algorithm.

A. Model Parameters

The origin of the coordinate system is assumed to be aligned with the left camera position and orientation, so that only the right camera has to be calibrated in order to define the camera pair (C_ℓ, C_r). Let $\mathbb{S}_\mathrm{r} = \mathbb{R}^d, d > 0$, be the space in which the state of the right camera is described. In general, the components of a given state vector $\mathbf{s}$ in $\mathbb{S}_\mathrm{r}$ can be

- the camera's position and orientation in $\mathbb{X}$ (6-D),
- the velocity and rotation rates (6-D),
- the focal length (1-D),
- the coordinates of the principal point (2-D), and
- the image distortion (1-D) for a non-pinhole camera,

so that the dimension d of the right-camera's state space can be as high as 16.

The objective is to jointly estimate the state of the multiple objects in the scene, as well as the state of the right camera, C_r, relative to the left camera, C_ℓ. We thus introduce the joint probability distribution $\mathbf{P}_t^i$ which encompasses the right camera state $\mathbf{s} \in \mathbb{S}_\mathrm{r}$, as well as the multi-object state Y:

$$\mathbf{P}_t^i(Y, \mathbf{s}) = P_t^i(Y \mid \mathbf{s}) p_t(\mathbf{s}),$$

where p_t is a probability distribution over $\mathbb{S}_\mathrm{r}$.

For the same reasons as the ones discussed in Section IV-A, it continues to be impractical to work with multi-object densities directly, and the first-moment density

$$\boldsymbol{\mu}_t^i(\mathbf{y}, \mathbf{s}) = \mu_t^i(\mathbf{y} \mid \mathbf{s}) p_t(\mathbf{s}) \tag{6}$$

is preferred. This relation holds as the first-moment density corresponding to a single-variate distribution is the distribution itself. Equation (6) indicates that the use of the PHD filter for propagating the first-moment density $\boldsymbol{\mu}_t^i$ can be considered. We describe this approach in the next section.

B. Conditional PHD Filtering

Due to the conditional nature of (6), the derivation of the PHD filter results in an expression that is different to the usual PHD filter equations. The result of this derivation, detailed in [40], can be expressed as

$$\boldsymbol{\mu}_t^i(\mathbf{y}, \mathbf{s}) = \mu_t^i(\mathbf{y} \mid \mathbf{s}) \alpha_t(\mathbf{s}) \hat{p}_t(\mathbf{s}),$$

where $\mu_t^i(\cdot \mid \mathbf{s})$ is found via the PHD update (5), where λ^i, c^i, L_t^i and p_D^i might be dependent on $\mathbf{s}$, and where $\alpha_t(\mathbf{s}) \in [0, 1]$ relates to the probability for the sensor state $\mathbf{s}$ to generate a successful multi-object update, expressed as

$$\alpha_t(\mathbf{s}) = \frac{\mathbf{L}_t^\mathrm{c}\left(Z_t^i \mid \mathbf{s}\right)}{\int \mathbf{L}_t^\mathrm{c}\left(Z_t^i \mid \mathbf{s}'\right) \hat{p}_t(\mathbf{s}') \mathrm{d}\mathbf{s}'},$$

where $\mathbf{L}_t^\mathrm{c}(Z_t^i \mid \mathbf{s})$, with "c" standing from "calibration", is interpreted as the likelihood of the observation set Z_t^i, given the camera state $\mathbf{s}$, defined as

$$\mathbf{L}_t^\mathrm{c}\left(Z_t^i \mid \mathbf{s}\right) = \exp\left(-\lambda(\mathbf{s}) - \int \mu_{\bullet,t}^i(\mathbf{y} \mid \mathbf{s}) \dot{\mathbf{y}}\right) \times \prod_{z \in Z_t^i} \left[\lambda(\mathbf{s}) c(z \mid \mathbf{s}) + \int L_t^i(z \mid \mathbf{y}, \mathbf{s}) \mu_{\bullet,t}^i(\mathbf{y} \mid \mathbf{s}) \mathrm{d}\mathbf{y}\right].$$

The expression of $\mathbf{L}_t^\mathrm{c}$ contains a product over the observations, assessing the probability for each of these to be either a spurious observation or to come from an object in $\mathcal{X}_t$. This form confirms the status of a multi-object likelihood for $\mathbf{L}_t^\mathrm{c}$.

Interestingly, the structure of the joint multi-object tracking and camera calibration is similar to the one derived for group tracking, see, e.g., [44] and [45]. This similarity can be explained by the hierarchical structure shared by the two estimation problems.

As the single-object likelihood L_t^i exhibits the same kind of non-linearity as the mapping from $\hat{\mathbb{X}}$ to $\hat{\mathbb{D}}_i$ or $\mathbb{P}_i$, we can readily conclude that the distribution p_t is likely to be non-Gaussian in $\mathbb{S}_\mathrm{r}$. However, we do not wish to model the possibility for the right camera to be infinitely far from the left camera, and thus a particle representation is now suitable.

For these reasons, we select a particle representation of the camera distribution p_t, composed of M_t particles $\{\mathbf{s}_k\}_{k=1}^{M_t}$, expressed as

$$p_t(\mathbf{s}) \approx \sum_{k=1}^{M_t} \omega_k \delta_{\mathbf{s}_k}(\mathbf{s}),$$

where $\delta_{\mathbf{s}_k}$ is the Dirac function at point $\mathbf{s}_k$. The updated joint first-moment density $\boldsymbol{\mu}_t^i$ can then be rewritten as

$$\boldsymbol{\mu}_t^i(\mathbf{y}, \mathbf{s}) \approx \sum_{k=1}^{\hat{M}_t} \mu_t^i(\mathbf{y} \mid \hat{\mathbf{s}}_k) \alpha_t(\hat{\mathbf{s}}_k) \hat{\omega}_k \delta_{\hat{\mathbf{s}}_k}(\mathbf{s}),$$

so that each possible camera predicted state $\hat{\mathbf{s}}_k$ is associated with a specific conditional first-moment density $\mu_t^i(\cdot \mid \hat{\mathbf{s}}_k)$, propagated with a GM-PHD filter.

In practice, particle implementations are known to be sensitive to the curse of dimensionality. The number of particles needed to maintain a certain approximation error grows exponentially with the number of state dimensions. Therefore, every effort should be made to decrease the number of calibration parameters being estimated. Rather than estimate all of the above mentioned parameters in one pass in a 16-dimensional state space, we suggest the following approach:

1) Assume that the camera pair is in a static configuration, in order to temporarily ignore the 6 dimensions required for the motion estimation. The intrinsic parameters can then be estimated within a 10-dimensional state space.
2) Once the intrinsic parameters are known, the estimation of the position and velocity can then take place within a 12-dimensional state space.

VI. Results on Simulated and Real Data

The proposed approach was validated with several simulated scenarios and one real dataset, depicting interesting examples

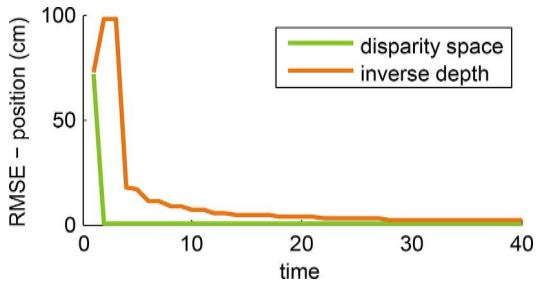


Fig. 10. Performance of the estimation in disparity space compared with inverse depth for the localization of a static object with a non-rectified pair of cameras.

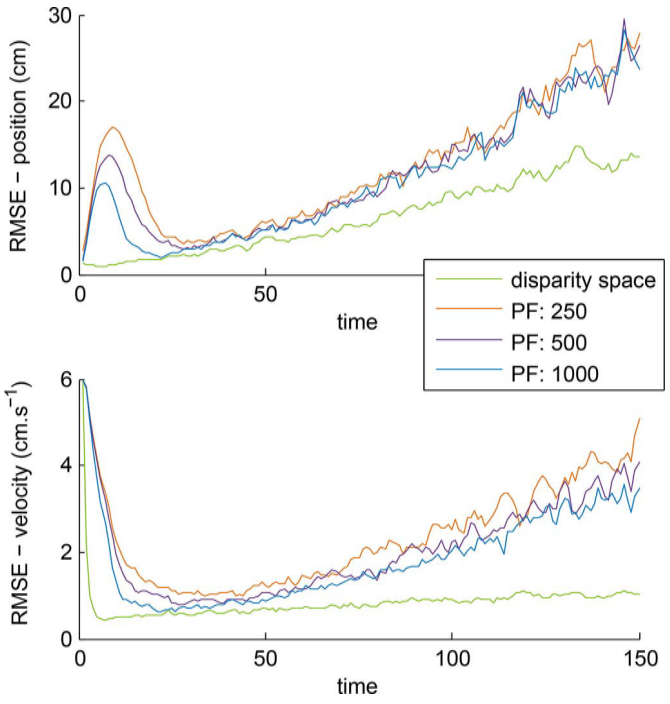


Fig. 11. Performance of the proposed single-object tracking algorithm (disparity space) compared against a particle filter with 250, 500 and 1000 particles (PF:250, PF:500 and PF:1000).

of use. The basic experimental configuration is a pair of non-rectified cameras that observe a scene with objects that behave in different ways:

A. *Single-object localization*: the disparity space approach is compared with inverse-depth (Fig. 10)
B. *Single-object tracking*: the proposed method for single-object tracking, detailed in Algorithm 1, is compared with a particle filter (Fig. 11)
C. *Multi-object tracking*: the approach of joint calibration and tracking is assessed for random camera configurations (Fig. 12)
D. *Camera calibration*: the approach is assessed in simulations in order to show the convergence of the extrinsic camera-parameter estimates (Fig. 13) and the tracking performance of the underlying multi-object estimation (Fig. 12)
E. *Real Data*: the proposed framework is tested on real data and results are presented in Figs. 14 to 17 and in the supplementary material

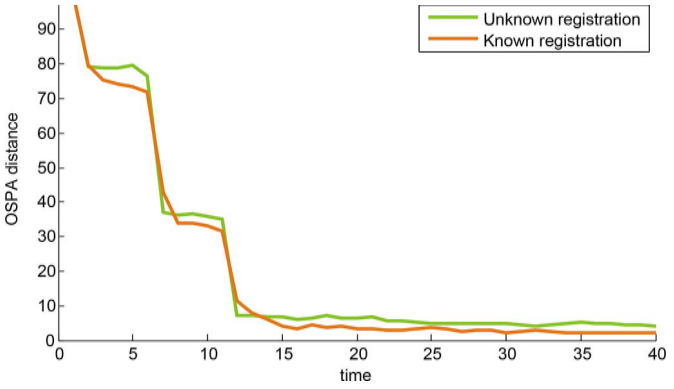


Fig. 12. Average performance over 50 MC runs for the multi-object tracking algorithm with and without calibration. Parameters of the OSPA distance with cutoff $c = 100$ and 1-norm.

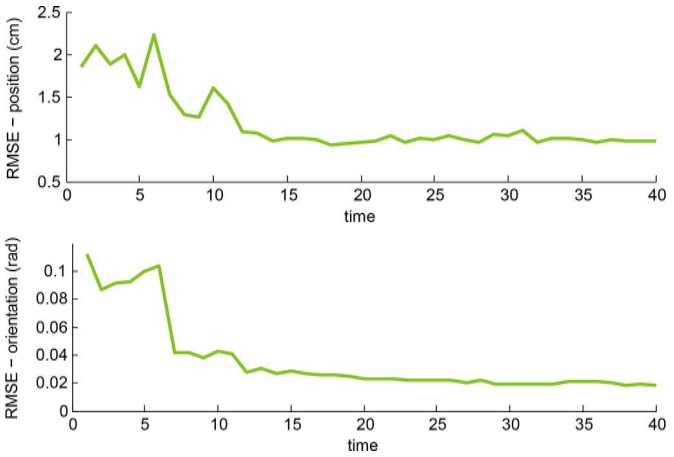
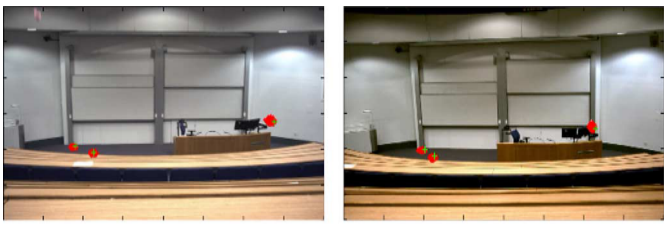


Fig. 13. Average performance over 50 MC runs for the estimation of the extrinsic parameters of the right camera for the proposed joint multi-object tracking and camera calibration algorithm.



(a) (b)

Fig. 14. Measurements (green crosses) and tracks (red dots) projected on the left and right camera planes. (a) Paper planes, left view, (b) Paper planes, right view.

These examples are presented in the following sections.

A. Single-Object Localization

One of the strengths of the disparity space representation is that it allows for the definition of prior distributions, where a large range of distance values are taken into account, using a single Gaussian representation. This is advantageous for triangulation, since it limits the amount of resources that are necessary to define a prior distribution for a newly observed object, and then localise it using a Bayes update.

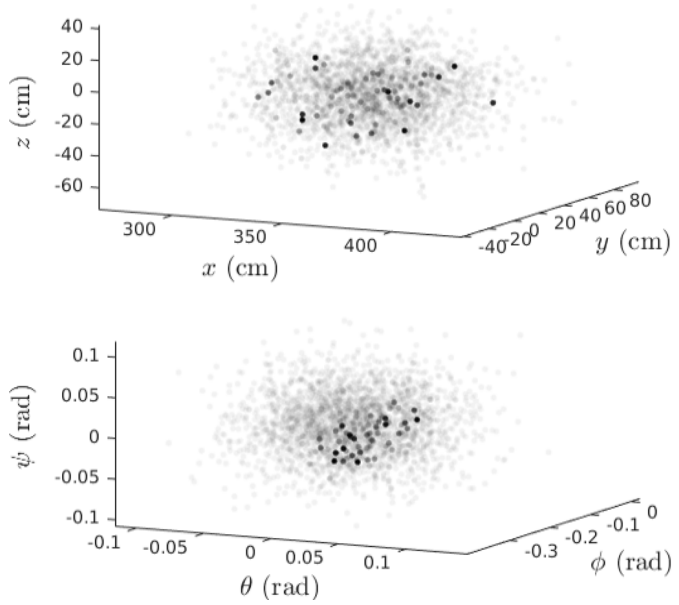


Fig. 15. 3-D views of the estimation of the intrinsic parameters view 2500 particles. Particles are displayed with a color ranging from white to black, depending on their weight - the higher, the darker.

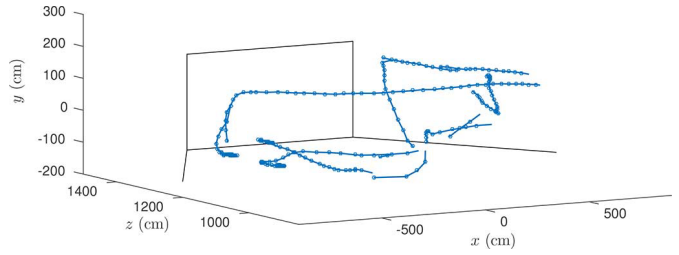


Fig. 16. Estimated 3-D trajectories of the paper planes (blue lines) with the back wall and the ground of the auditorium indicated in black.

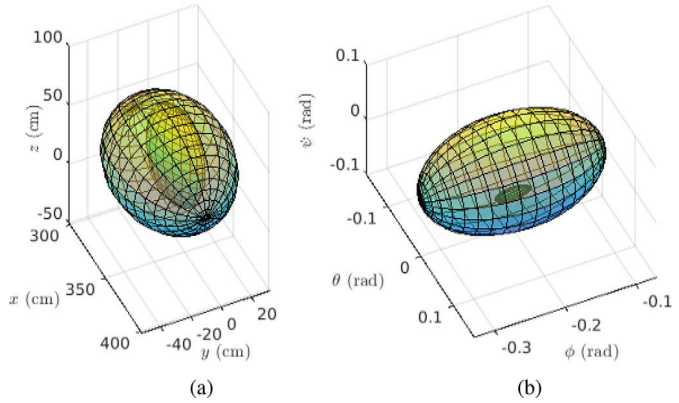


Fig. 17. Mean and covariance of the Maximum A Posteriori (MAP) for the extrinsic parameters over 15 MC runs (small ellipsoid) compared to the initial uncertainty (large ellipsoid). (a) Position, (b) Orientation.

In this scenario, the localization performance of the disparity space-based solution is compared against inverse depth, as in [19]. Two cameras are set up as follows: the first camera is at the centre of the coordinate system and the second camera is translated by 80 cm along the x axis with respect to the first, and rotated $-\pi/4$ radians around the y axis, i.e., the setup is non-rectified. The object is located along the z axis, 150 cm away from the left camera, and is observed by the modelled pinhole cameras. The camera parameters are given in Table I. The initialization of the inverse depth component is made equivalent to the one used for disparity, with mean $\mu_d = 12.5$ and standard deviation $\sigma_d = 4.2$.

TABLE I
SIMULATED CAMERA PARAMETERS

Parameter	Value	Parameter	Value
pixel size d_u	8.9μm	principal point u_0	400
d_v	9.0μm	v_0	300
observation noise σ_u^2	2	baseline b	80cm
σ_v^2	2	focal length f	-8mm

The average performance over 100 MC runs for the two localization algorithms is shown in Fig. 10. It appears that the inverse depth approach does not cope well with the non-linearity of the observation function and makes a significant error at the second time step, whereas the disparity space approach manages to localise the target almost instantly. This result can be explained by the limitations of the EKF, which is used for the inverse-depth approach, as originally published in [34], when dealing with non-linear functions such as the one depicted in Fig. 6. This example corroborates the fact that the proposed solution does manage to propagate the uncertainty between the left and right disparity spaces, $\mathbb{D}_\ell$ and $\mathbb{D}_\mathrm{r}$, as already suggested in Fig. 8.

B. Single-Object Tracking

The suitability of the disparity space parametrization for tracking was evaluated through an experiment where an object moves away from the left camera with a nearly-constant velocity. This was done to analyze how capable the parametrization is to deal with smooth changes in distance. The configuration of the camera pair is as follows: the left camera C_ℓ is at $(-20, 0, 0)$ and is rotated by an angle of $\pi/12$ about the y axis, while the right camera C_r is at (20, 0, 0) and is rotated by an angle of $-\pi/12$ about the y axis. As before, the filter was initialized with a prior distribution in disparity space, and then it was successively updated with measurements that were acquired synchronously from both cameras. Even though synchronicity of the camera pair is not a necessary assumption in our algorithm, it is shown here that this special case can be treated equally well. The experiments consisted of tracking an object with an initial velocity of 6 cm·s^{-1} along the z axis.

The performance of the proposed solution is compared against a particle filter. The objective is to demonstrate that the approximation made when fitting a Gaussian distribution after the particle move is compensated for by the gain in accuracy obtained during the observation update. The average performance over 100 MC runs for the disparity space approach is shown in Fig. 11, together with the performance of a particle filter for different numbers of particles. It appears that the particle filter struggles at the initialization, both in the estimation of position and of velocity of the target. This can be explained by the degeneracy of the set of particles when the prior distribution is updated by the observation from the right camera. Towards the end of the experiment, the target is up to 10 m away from the camera pair, and the estimation is once again made difficult

for the particle filter, as resampling is needed more and more frequently to cope with the noise in the observations. The disparity space approach only uses 250 particles for the particle move and does not require resampling to be applied. As a consequence, the computational time is equivalent to a particle filter with 250 particles, while the performance is better than that of a particle filter with 1000 particles.

C. Multi-Object Tracking

Having assessed the performance of the disparity space representation for tracking a single target, an experiment was done to evaluate the performance of a PHD filter equipped with the disparity space representation for simultaneously tracking multiple objects. To evaluate the performance of the multiple target tracker, the OSPA metric [41] was utilized. This metric is commonly employed to measure the performance of multi-object tracking filters. It gives the distance between two sets of points by first solving the optimal assignment problem and returning a weighted combination of the average distance between the matched points and the difference in cardinality between the two sets.

For this experiment, the basic camera configuration is similar to the one considered in Section VI-B, i.e., the cameras were located on the $y = z = 0$ plane at -20 cm and 20 cm along the x axis, and rotated $\pi/12$ and $-\pi/12$ radians around the y axis, respectively. In order to test the robustness of the proposed method, the right camera position and orientation are changed randomly for each Monte Carlo run, with the following respective standard deviations: 0.2 cm, 0.5 cm and 0.2 cm on x, y and z as well as $\pi/180$ rad, $\pi/90$ rad and $\pi/180$ rad for the rotations about x, y and z. The model parameters used for the GM-PHD filter are as follows: the merging distance is equal to 7, the pruning threshold is set to 10^{-6}, the false alarm Poisson parameter is $\lambda^i = 10$ and the probability of detection p_D^i is equal to 0.95. Seven objects moving according to a constant velocity model were observed by the two cameras. In Fig. 12, the evolution of the OSPA metric is displayed with the label "Known registration", showing good agreement between the ground truth and the obtained estimates.

D. Camera Calibration

Following the assessment of the proposed method for tracking one to many targets in Sections VI-A to VI-C, the objective in this section is to demonstrate that the extrinsic parameters of the right camera can additionally be estimated by tracking 7 non-cooperative moving targets.

For this experiment, the cameras were set up in a configuration similar to the one considered in Section VI-C. Fig. 13 shows the convergence of the estimation in position and orientation for a 6-D calibration problem with 2500 particles for the calibration and the following values for the uncertainty:

- Prior position: $\sigma_x = \sigma_y = \sigma_z = 2$ cm,
- Prior orientation: $\sigma_\theta = \sigma_\psi = \pi/48$ and $\sigma_\phi = \pi/24$.

Fig. 13 demonstrates that the proposed solution enables the calibration of non-rectified cameras from multiple, non-cooperative, moving objects, when the data association is not known. The considered scenario is the same as in Section VI-C, so that the two OSPA distances can be compared. The average OSPA distance can also be found in Fig. 12 with the label "Unknown registration". It first appears that the difference between the two is larger before the initialization of the second set of tracks around the 7th time step, and then stabilizes for the rest of the scenario. This shows that the first appearing targets bring more information than the ones appearing later on in the scenario. A surprising result is that the OSPA distance for unknown registration becomes smaller for one or two time steps when new objects appear. This phenomenon can be explained by the fact that before convergence, the objects tend to be seen at a closer range than their true position, which, because of the greater localization accuracy at shorter ranges, causes the early confirmation of the tracks.

Note that the values of the standard deviation in position are large enough to set up the initial value by the naked eye, as it covers a 12 cm error in each direction, whereas the actual distance between the two cameras is 40 cm. The uncertainty for the orientation ϕ around the y-axis is also relatively large, as it covers up to $45°$. The orientations θ and ψ around the x and z axis, respectively, are assumed to be better known, with only $22.5°$ of coverage for these components. A higher number of particles would be required to allow for a larger uncertainty interval for extrinsic parameters.

E. Results on Real Data

In this section, the solution detailed in Section VI-D for non-rectified camera calibration is tested on real data. This section focuses on testing the proposed methodology on practical examples, rather than evaluating its performance with respect to a known ground truth, which was the topic of Section VI-D. The devices used for the test are two Point Grey Flea®3 cameras, equipped with 8 mm lenses. As these lenses have a very low distortion, this parameter is considered negligible and will not be estimated. The two cameras are plugged in through FireWire 800 cables to a MAGMA® PCI Express box, which is connected to a laptop computer.

In the considered dataset, the objects of interest are paper planes of different colors and shapes. The paper planes are thrown through an auditorium and sustain their flight for several time steps, thus exhibiting sufficient observability for their state to be estimated. They are observed by two cameras located at the back of the auditorium, with views as shown in Figs. 14(a) and (b). Most of the time, the motion of the paper planes is well represented by a simple constant velocity model with small random accelerations modelled by additive noise. An example of estimation of the intrinsic parameters is given in Fig. 15, where it appears that the convergence of orientation parameters happens faster than the convergence of the position parameters. The estimation of the state of the planes has also been achieved, and the resulting 3-D trajectories of all the paper planes are displayed in Fig. 16. Finally, in Fig. 17, the variability of the MAP estimate for the extrinsic parameters is displayed as an ellipsoid and is compared to the initial uncertainty on these parameters, displayed as the larger ellipsoid. The variability of the position parameters can be explained by their limited observability when compared to the orientation parameters.

This data set demonstrates the applicability of the proposed solution to a realistic case, where the prior knowledge on the position and orientation of the cameras is highly uncertain (up to 1 m uncertainty on the x axis), and where the observations come from moving, non-cooperative targets. A video of the results is available as supplementary material.

VII. Conclusion

A parametrization based on the concept of disparity space has been presented for non-rectified camera networks, extended to moving objects, and integrated into a Bayesian multi-object tracking and sensor calibration technique. The proposed single-object filter relies on approximations that have been shown to be valid for a suitably large spectrum of camera setups and object motions. The performance of the obtained framework has been demonstrated, not only for camera calibration on simulated and real data, but also for the underlying problems of single-object localization and tracking, as well as for multi-object tracking. This framework can therefore be described as a unified Bayesian multi-object tracking and camera calibration method that only requires the presence of non-cooperative moving objects.

Future work will cover the extension of the proposed method to other multi-object filters and a comparative study of these approaches for camera-based tracking as well as for camera calibration.

References

[1] G. Adiv, "Inherent ambiguities in recovering 3-D motion and structure from a noisy flow field," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, no. 5, pp. 477–489, 1989.

[2] M. Agrawal and K. Konolige, "Real-time localization in outdoor environments using stereo vision and inexpensive GPS," in *Proc. Int. Conf. Pattern Recognit. (ICPR)*, Hong Kong, 2006, vol. 3, pp. 1063–1068.

[3] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174–188, 2002.

[4] Y. Bar-Shalom and T. Fortmann, *Tracking and Data Association*. New York, NY, USA: Academic, 1988.

[5] T. J. Broida and R. Chellappa, "Estimating the kinematics and structure of a rigid object from a sequence of monocular images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 6, pp. 497–513, 1991.

[6] A. R. Chowdhury and R. Chellappa, "Statistical bias in 3-D reconstruction from a monocular video," *IEEE Trans. Image Process.*, vol. 14, no. 8, pp. 1057–1062, 2005.

[7] J. Civera, A. Davison, and J. M. M. Montiel, "Inverse depth parametrization for monocular SLAM," *IEEE Trans. Robot.*, vol. 24, no. 5, pp. 932–945, 2008.

[8] D. Clark and S. Ivekovic, "The Cramer-Rao lower bound for 3-D state estimation from rectified stereo cameras," in *Proc. IEEE FUSION Conf.*, 2010.

[9] D. E. Clark and J. Bell, "Convergence results for the particle PHD filter," *IEEE Trans. Signal Process.*, vol. 54, no. 7, pp. 2652–2661, 2006.

[10] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York, NY, USA: Wiley, 1991.

[11] H. S. M. Coxeter, *Projective Geometry*. New York, NY, USA: Springer, 2003.

[12] K. Daniilidis and H. H. Nagel, "The coupling of rotation and translation in motion estimation of planar surfaces," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 1993.

[13] D. Demirdjian and T. Darrell, "Using multiple-hypothesis disparity maps and image velocity for 3-D motion estimation," *Int. J. Comput. Vis.*, vol. 47, no. 1–3, pp. 219–228, 2002.

[14] K. Derpanis and P. Chang, "Closed-form linear solution to motion estimation in disparity space," in *Proc. IEEE Intell. Veh.*, 2006.

[15] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge, U.K.: Cambridge Univ. Press, 2003.

[16] N. Henze and B. Zirkler, "A class of invariant consistent tests for multivariate normality," *Commun. Statist.—Theory and Methods*, vol. 19, no. 10, pp. 3595–3617, 1990.

[17] J. Houssineau, "Representation and estimation of stochastic populations," Ph.D. dissertation, School of Engineering and Physical Sciences, Heriot-Watt Univ., Edinburgh, U.K., 2015.

[18] J. Houssineau, E. Delande, and D. Clark, "Notes of the summer school on finite set statistics," 2013, arXiv preprint arXiv:1308.2586.

[19] J. Houssineau, S. Ivekovic, and D. Clark, "Disparity space: A parameterisation for Bayesian triangulation from multiple cameras," in *Proc. 15th IEEE FUSION Conf.*, 2012.

[20] J. Houssineau and D. Laneuville, "PHD filter with diffuse spatial prior on the birth process with applications to GM-PHD filter," in *Proc. 13th IEEE FUSION Conf.*, 2010.

[21] S. Ivekovic and D. Clark, "Multi-object stereo filtering in disparity space," in *Proc. COGnitive Syst. With Interact. Sensors (COGIS)*, 2009.

[22] S. Ivekovic and E. Trucco, "Articulated 3-D modelling in a wide-baseline disparity space," in *Proc. Eur. Conf. Vis. Media Product.*, 2007.

[23] A. Jazwinski, *Stochastic Processes and Filtering Theory*. New York, NY, USA: Academic, 1970.

[24] B. Julesz, *Foundations of Cyclopean Perception*. Chicago, IL, USA: Univ. of Chicago, 1971.

[25] S. J. Julier and J. K. Uhlmann, "A general method for approximating nonlinear transformations of probability distributions," RRG, Eng. Sci. Dept., Univ. of Oxford, Oxford, U.K., Tech. Rep., 1996.

[26] R. E. Kalman, "A new approach to linear filtering and prediction problems," *J. Basic Eng. (ASME)*, vol. 82, no. D, pp. 35–45, 1960.

[27] J. W. Koch, "Bayesian approach to extended object and cluster tracking using random matrices," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 44, no. 3, pp. 1042–1059, 2008.

[28] D. Laneuville and J. Houssineau, "Passive multi target tracking with GM-PHD filter," in *Proc. 13th IEEE FUSION Conf.*, 2010.

[29] C.-S. Lee, D. Clark, and J. Salvi, "SLAM with dynamic targets via single-cluster PHD filtering," *IEEE J. Sel. Top. Signal Process. (Special Issue on Multi-Target Tracking)*, vol. 7, no. 3, pp. 543–552, 2013.

[30] C. S. Lee, S. Nagappa, N. Palomeras, D. Clark, and J. Salvi, "SLAM with SC-PHD filters: An underwater vehicle application," *IEEE Robot. Autom. Mag.*, vol. 21, no. 2, pp. 38–45, 2014.

[31] Y. Ma, S. Soatto, J. Kosecka, and S. Sastry, *An Invitation to 3-D Vision, From Images to Geometric Models*. New York, NY, USA: Springer Science, 2006.

[32] R. P. S. Mahler, "Multitarget Bayes filtering via first-order multitarget moments," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 39, no. 4, pp. 1152–1178, 2003.

[33] R. P. S. Mahler, "PHD filters of higher order in target number," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 43, no. 4, pp. 1523–1543, 2007.

[34] J. M. M. Montiel, J. Civera, and A. Davison, "Unified inverse depth parametrization for monocular SLAM," in *Proc. Robot.: Sci. Syst.*, 2006.

[35] J. Mullane, B.-N. Vo, M. Adams, and B.-T. Vo, "A random-finite-set approach to Bayesian SLAM," *IEEE Trans. Robot.*, vol. 27, no. 2, pp. 268–282, 2011.

[36] J. Mullane, B.-N. Vo, M. D. Adams, and W. Wijesoma, "A random set formulation for Bayesian SLAM," in *Proc. IEEE Int. Conf. Intell. Robot. Syst. (IROS)*, 2008.

[37] K. Panta, D. E. Clark, and B.-N. Vo, "Data association and track management for the Gaussian mixture probability hypothesis density filter," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 45, no. 3, pp. 1003–1016, 2009.

[38] N. T. Pham, W. Huang, and S. Ong, "Probability hypothesis density approach for multi-camera multi-object tracking," in *Computer Vision-ACCV*. New York, NY, USA: Springer, 2007, pp. 875–884.

[39] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter, Particle Filters for Tracking Applications*. Norwood, MA, USA: Artech House, 2004.

[40] B. Ristic, D. Clark, and N. Gordon, "Calibration of multi-target tracking algorithms using non-cooperative targets," *IEEE J. Sel. Topics Signal Process. (Special Issue on Multi-Target Tracking)*, vol. 7, no. 3, pp. 390–398, 2013.

[41] D. Schuhmacher, B.-T. Vo, and B.-N. Vo, "A consistent metric for performance evaluation of multi-object filters," *IEEE Trans. Signal Process.*, vol. 56, no. 8, pp. 3447–3457, 2008.

[42] G. Sibley, L. Matthies, and G. Sukhatme, *Bias Reduction and Filter Convergence for Long Range Stereo*. New York, NY, USA: Springer, 2007.
[43] T. Svoboda, D. Martinec, and T. Pajdla, "A convenient multi-camera self-calibration for virtual environments," *PRESENCE: Teleoperators Virtual Environ.*, vol. 14, no. 4, pp. 407–422, 2005.
[44] A. Swain and D. Clark, "Extended object filtering using spatial independent cluster processes," in *Proc. 13th FUSION Conf.*, 2010.
[45] A. Swain and D. Clark, "First-moment filters for spatial independent cluster processes," in *Proc. SPIE Defense, Secur., Sens.*, 2010.
[46] C. Tomasi and T. Kanade, "Shape and motion from image streams under orthography: A factorization method," *Int. J. Comput. Vis.*, vol. 9, no. 2, pp. 137–154, 1992.
[47] R. Tsai, "A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf cameras and lenses," *J. Robot. Autom.*, vol. 3, no. 4, pp. 323–344, 1987.
[48] B.-N. Vo and W.-K. Ma, "The Gaussian mixture probability hypothesis density filter," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4091–4104, 2006.
[49] B.-N. Vo, S. Singh, and A. Doucet, "Sequential Monte Carlo methods for multitarget filtering with random finite sets," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 41, no. 4, pp. 1224–1245, 2005.
[50] B.-T. Vo and B.-N. Vo, "Labeled random finite sets and multi-object conjugate priors," *IEEE Trans. Signal Process.*, vol. 61, no. 13, pp. 3460–3475, 2013.
[51] G.-S. J. Young and R. Chellappa, "Statistical analysis of inherent ambiguities in recovering 3-D motion from a noisy flow field," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 10, pp. 995–1013, 1992.

Jeremie Houssineau received an Eng. degree in mathematical and mechanical modelling from MATMECA, Bordeaux, and a M.Sc. degree in mathematical modelling and statistics from the University of Bordeaux, both in 2009. From 2009 to 2011, he was a Research Engineer with DCNS, Toulon, and then with INRIA Bordeaux. He received his Ph.D. degree in statistical signal processing from Heriot-Watt University, Edinburgh, in 2015. His research interests include applied probability, point process theory and multi-object estimation.

Daniel E. Clark is an Associate Professor in the School of Engineering and Physical Sciences at Heriot-Watt University. His research interests are in the development of the theory and applications of multi-object estimation algorithms for sensor fusion problems. He has led a range of projects spanning theoretical algorithm development to practical deployment. He was awarded his Ph.D. in 2006 from Heriot-Watt University.

Spela Ivekovic received her Ph.D. in computer vision from Heriot-Watt University in 2008. The focus of her dissertation was on articulated human body pose estimation in disparity space as a mechanism for novel-view synthesis in an immersive videoconferencing environment. As a postdoctoral researcher, she worked in the area of computer vision, computational intelligence, molecular dynamics and parallel computation. She currently directs Sophrodyne Ltd., a scientific research consultancy working to address cutting-edge research challenges in science and engineering from an interdisciplinary, systems-theory perspective.

Chee Sing Lee is a Machine Learning Engineer at BigML Inc. His doctoral research focused on simultaneous localization and mapping using probability hypothesis density filters, and autonomous underwater vehicles. He also has prior experience working in computer vision, machine learning, FPGA design, and electric vehicles. He graduated from Oregon State University with an H.B.S. in electrical and electronic engineering in 2008, earned an Erasmus Mundus Masters in computer science and robotics (VIBOT) in 2010, and obtained a Ph.D. from the University of Girona in 2015.

Jose Franco received a B.Sc. degree in mathematical engineering from EAFIT University in 2010 and a M.Sc. in computer vision and robotics jointly from the University of Burgundy, the University of Girona and Heriot-Watt University in 2013. He is currently a doctoral student in Heriot Watt University under the supervision of Dr. Daniel Clark, where he studies multiple object estimation techniques for a variety of applications.