18th International Conference on Information Fusion
Washington, DC - July 6-9, 2015

Tracking of Dolphins in a Basin Using a Constrained Motion Model

Clas Veibäck*, Gustaf Hendeby*†, Fredrik Gustafsson*
*Dept. Electrical Engineering, Linköping University, SE-581 83 Linköping, Sweden.
Email: firstname.lastname@liu.se
†Dept. of Sensor & EW Systems, Swedish Defence Research Agency (FOI), SE-581 11, Linköping, Sweden.
Email: gustaf.hendeby@foi.se

***Abstract*—Visual animal tracking is a challenging problem generally requiring extended target models, group tracking and handling of clutter and missed detections. Furthermore, the dolphin tracking problem we consider includes basin constraints, shadows, limited field of view and rapidly changing light conditions. We describe the whole pipeline of a solution based on a ceiling-mounted fisheye camera that includes foreground segmentation and observation extraction in each image, followed by a target tracking framework. A novel contribution is a potential field model of the basin edges as a part of the motion model, that provides a robust prediction of the dolphin trajectories in phases with long segments of missed detections. The overall performance on real data is quite promising.**

I. Introduction

Tracking animal movement is a multi-faceted emerging application area of target tracking. On the one hand, we have recent legislations of visual surveillance of the catch in legal fishing by inspecting the fishing net, and the upcoming legislations of having real-time positions of cattle. On the other hand, the research of animal movement is in great need for automatic tracking, where today tedious manual work is needed. Research directions include better understanding of genetic control programs of migration, what 'sensor information' that is used for animal navigation, and the evolution of migration. Cross-disciplinary research between the target tracking community and biologists has the potential to generate large amounts of animal data to the biologists, at the same time as posing challenging problems for the target tracking algorithms. One example of such a collaboration is [1], where data from a $4\,\mathrm{g}$ light logger mounted on a common swift was used to track the bird from the summer residence in Sweden through its migration to Africa and back again. The development involved an astronomic sensor model defining the sun angle as a function of position.

In this work, we describe another challenging application, to track dolphins swimming around in a basin using a fisheye camera mounted in the ceiling. The biological purpose is to understand how the behavioural pattern is affected by underwater sonar transponders. In this way, a better understanding can be obtained for how the dolphins' internal navigation system works. Today, tracking is done manually from the video. There are many similarities with classic target tracking problems with individuals forming tight groups, the need for extended target models, and clutter from the pre-processing. New challenges include shadows at the bottom of the basin, sun light through the ceiling windows that gives large local changes in light conditions. The special scene also includes hard constraints, occlusion from a platform and missed detections caused by a limited field of view from the fisheye camera. Another challenge is the difficulty to obtain sufficient data to calibrate the camera.

For visual tracking, the computer vision community has made a lot of progress to solve this problem in video. The methods used rely on several different principles, often used in combination. Sophisticated foreground extraction methods (*e.g.*, [2–5]) can be used to bring out moving objects from stationary backgrounds. The usage of these methods requires stationary cameras and is complicated by rapid changes in light conditions and irrelevant motions in the scene. Machine learning can be used to train a detector that finds previously known objects in an image (*e.g.*, [6–8]). To work properly, these methods require considerable amounts of training data that cover all possible object appearances and backgrounds. Yet other types of methods detect possible objects and try to locate the same patch in consecutive frames (*e.g.*, [9]). These methods tend to be sensitive to appearance changes, which limit their applicability in practice.

In the target tracking community standard computer vision algorithms are used as input to more sophisticated tracking algorithms (*e.g.*, [10, 11]). The approach we describe aims at using state of the art algorithms for foreground segmentation described in [2, 3] and estimation of an extended target model in each distorted image frame. The position and shape of each detected object is then undistorted and used as input to a target tracking algorithm, where false detections are compensated for and occlusions and missed detections are handled gracefully using a suitable motion model. A novel contribution is to describe the physical constraints of the basin in terms of potential fields as part of the motion model, inspired by the potential fields used for collision avoidance in the robotics community (*e.g.*, [12]).

II. Proposed Tracking Solution and Paper Outline

Fig. 1 depicts the data processing pipeline suggested to solve the dolphin tracking problem described above. The solution is divided into two principal parts: measurement pre-processing and target tracking.

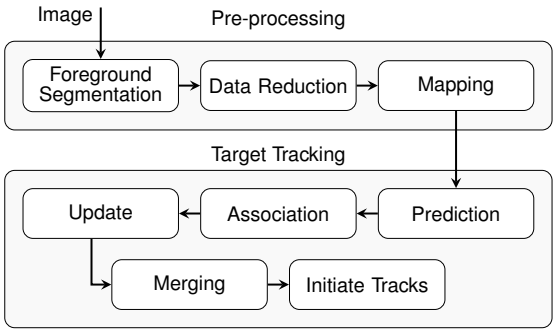

Fig. 1. The pipeline of processing a frame from the sensor.

In the measurement pre-processing block, the raw images provided by the fisheye camera in the ceiling are processed and observations are fed to the target tracking block. The purpose is to obtain as high quality dolphin observations as possible, while introducing as few false observations as possible. This is done in three steps: foreground segmentation, data reduction and mapping described in Sec. IV. The segmentation is obtained by estimating a background model and extracting non-matching pixels. The result, which can be quite noisy, is then further refined in the data reduction step where connected regions are clustered and extracted. These observations are then compensated for by the camera parameters in the mapping step. Sec. III describes how the camera parameters are derived. The result, that need not be perfect, is then passed on to the target tracking block.

The target tracking block is designed around a standard target tracking loop comprising: track prediction, track-observation association, measurement update, track merging, and track initiation. Important components in the target tracking solution are the novel motion model that constrains the tracked dolphins to the basin, as derived in Sec. V, and the PDA inspired method used to incorporate the observed regions in the measurement update step. Combined, these result in a tracking solution that is able to produce useful tracks based on the input from the measurement pre-processing block.

Finally, the solution is evaluated on experimental data in Sec. VII and relevant aspects of the suggested tracking solution are highlighted. Conclusions and further work are discussed in Sec. VIII.

III. Calibration

The fisheye camera is used as a solution to local regulations concerning audience integrity, but exhibits severe radial distortion and must be calibrated before being used. Usually, the camera calibration (intrinsic and extrinsic parameters and the lens distortion) can be obtained using standard software [13] based on images containing a checkerboard in different angles and positions. In this case the camera is mounted in a fixed position in the ceiling, hence the calibration must be estimated from available images and a map of the monitored region by identifying corresponding points as described below.

Let $\mathbf{x}^d$, $\mathbf{x}^u$ and $\mathbf{x}^m$ denote the coordinates of a point in the distorted image, undistorted image and on the map respectively. Furthermore, denote with $\tilde{\mathbf{x}} = \begin{pmatrix}\mathbf{x}^T & 1\end{pmatrix}^T$ the homogenous vector corresponding to $\mathbf{x}$. Then the intrinsic and extrinsic parameters can be combined into a homography

$$\mathbf{H} = \begin{pmatrix} H_{11} & H_{12} & H_{13} \\ H_{21} & H_{22} & H_{23} \\ H_{31} & H_{32} & H_{33} \end{pmatrix} = \begin{pmatrix}\bar{\mathbf{h}}_1 & \bar{\mathbf{h}}_2 & \bar{\mathbf{h}}_3\end{pmatrix}^T \tag{1a}$$

such that

$$\mathbf{x}^m = \frac{1}{\bar{\mathbf{h}}_3^T \tilde{\mathbf{x}}^u} \begin{pmatrix}\bar{\mathbf{h}}_1^T \\ \bar{\mathbf{h}}_2^T\end{pmatrix} \tilde{\mathbf{x}}^u \tag{1b}$$

gives a one-to-one mapping between undistorted and mapped coordinates.

Commonly a polynomial distortion model is used, but [14] suggests the following model for fisheye lenses

$$r_d = R(r_u) = \frac{1}{\omega} \arctan\left(2 r_u \tan\frac{\omega}{2}\right) \tag{2a}$$

$$r_u = R^{-1}(r_d) = \frac{\tan(r_d \omega)}{2\tan\frac{\omega}{2}}, \tag{2b}$$

where $r_d = |\mathbf{x}_d - \mathbf{x}_c|$ is the radial distance from the center of distortion $\mathbf{x}_c$ in the distorted image, $r_u = |\mathbf{x}_u - \mathbf{x}_c|$ is the radial distance in the undistorted image, and ω is a parameter determining the amount of distortion. The mapping is computed as

$$\mathbf{x}_d = \mathbf{x}_c + \frac{R(r_u)}{r_u}(\mathbf{x}_u - \mathbf{x}_c) \tag{3a}$$

$$\mathbf{x}_u = \mathbf{x}_c + \frac{R^{-1}(r_d)}{r_d}(\mathbf{x}_d - \mathbf{x}_c). \tag{3b}$$

The method described in [15] is used to estimate the homography in (1b) by finding the linear least-squares solution as an initial guess and refining it with the Levenberg-Marquardt algorithm. The parameters ω and $\mathbf{x}_c$ are estimated using the Levenberg-Marquardt algorithm. These solutions are computed in an alternated manner until convergence is achieved to find the complete mapping, as suggested by [15].

Having estimated the model, (1b) and (3b) can be used to derive a measurement function $\mathbf{h}(\mathbf{x})$ relating a point on the map with a point in the image. However, since the mapping is static and bijective, each measurement is transformed to the map as a final step of the measurement pre-processing block to reduce the dependence between the target tracking filter and the measurement model.

IV. Measurement Pre-processing

A. Foreground Segmentation

To bring out the objects the video is segmented into background and foreground. For this purpose, a Gaussian mixture background model [2, 3] is estimated with some modifications. The basic idea is to estimate mixtures of Gaussians to represent the pixel intensities using *expectation maximization* (EM) [16], but considering the number of pixels, several approximations of this algorithm are applied to make the computations

tractable. The intensities of a new image are gated and associated to the Gaussian mixture components of the model. If an association is found, the model is updated, otherwise a new Gaussian is initialized with low weight. Gaussian components with large weights are considered background whereas those with small weights are considered foreground.

The segmentation is based on a one channel image, which is obtained as a function of the red, green and blue channels. The function is chosen to achieve a reduction in the variance of the background pixels. Furthermore, the mean scene intensity is subtracted to make the model less sensitive to the light conditions.

The following applies to each pixel, currently measuring the intensity I, with a Gaussian mixture background model consisting of components $j = 1, \ldots, K_B, \ldots, K$ with mean μ_j and variance σ_j^2 and where the first K_B components are considered background. The parameter γ^2 determines the maximum squared Mahalanobis distance d_j considered a match through the criteria

$$d^j(I) = \frac{(I - \mu_j)^2}{\sigma_j^2} \leq \gamma^2. \tag{4}$$

If no $j \leq K_B$ exists such that $d^j(I) \leq \gamma^2$, the pixel is considered to be part of the foreground.

Selecting γ^2 is a trade-off between tolerating variations in the background and detecting foreground. According to [2], it can be advantageous to let γ^2 vary over time and different regions in the scene. The following heuristics are used for selecting γ^2

$$\gamma_t^2 = \gamma_0^2 + \gamma_g^2 \max_{s \in [t-\tau, t]} \sqrt{|\bar{\mathbf{I}}_s - \bar{\mathbf{I}}_{s-1}|}, \tag{5}$$

where τ, γ_0^2 and γ_g^2 are design parameters and $\bar{\mathbf{I}}_t$ is the mean intensity in the image at time t. The second term in (5) increases the tolerance for all pixels when the light conditions in the scene change drastically for some time determined by τ, allowing the current background components to adapt to the new conditions rather than to estimate new background components, which otherwise would result in many false detections.

An additional extension to the method is to compute

$$d = \min_{j \leq K_B} d^j(I) \tag{6}$$

for each foreground pixel, providing the Mahalanobis distance to the nearest background component. This value provides information about the confidence in the detection, which could allow for more sophisticated methods in the post-processing to globally segment the foreground or improve the tracking as in [17], but is here only used as described in Sec. IV-B. Additional extensions can be made to improve the performance, *e.g.*, as suggested by [4].

The foreground segmentation is generally noisy and is filtered using morphological operations [18], after which the output is M_f observations consisting of the coordinates $\breve{\mathbf{y}}_i$ of the foreground pixels and their values d_i obtained from (6). The set is denoted

$$\breve{\mathbf{Z}} = \{\breve{\mathbf{y}}_i,\, d_i\}_{i=1}^{M_f}. \tag{7}$$

B. Data Reduction

In general there are many measurements per target and their abundancy is intractable for a target tracking filter to handle, so the following method to reduce the amount of data is proposed. A first step is to obtain the indices i of connected components $\tilde{\mathcal{C}}_j$ for $j = 1, \ldots, M_c$ from the measurements $\breve{\mathbf{y}}_i$ using the flood fill algorithm [18, ch. 9].

Then use the k-means clustering algorithm [19] on the measurements $\{\breve{\mathbf{y}}_i | i \in \tilde{\mathcal{C}}_j\}$ for each connected component to obtain the clusters $\mathcal{C}_m$, for $m = 1, \ldots, M$, of measurements in $\breve{\mathbf{Z}}$. To obtain clusters of approximate size m_r the number of clusters for each component is chosen as $\lceil |\tilde{\mathcal{C}}_j| / m_r \rceil$.

To reduce the number of measurements, the means

$$\bar{\mathbf{y}}_j = \frac{1}{|\mathcal{C}_j|} \sum_{i \in \mathcal{C}_j} \breve{\mathbf{y}}_i \quad \text{and} \quad \bar{d}_j = \frac{1}{|\mathcal{C}_j|} \sum_{i \in \mathcal{C}_j} d_i \tag{8a}$$

are computed, where $|\cdot|$ denotes the set cardinality, and to keep some information regarding the extent of the connected component the covariance of the measurements

$$\bar{\mathbf{Y}}_j = \frac{1}{|\mathcal{C}_j|} \sum_{i \in \mathcal{C}_j} (\breve{\mathbf{y}}_i - \bar{\mathbf{y}}_j)(\breve{\mathbf{y}}_i - \bar{\mathbf{y}}_j)^T, \tag{8b}$$

is computed and a reduced measurement set is obtained as

$$\bar{\mathbf{Z}} = \{\bar{\mathbf{y}}_j, \bar{d}_j, \bar{\mathbf{Y}}_j\}_{j=1}^{M_c}. \tag{8c}$$

To exactly map the ellipsoid represented by the covariance in the reduced measurement set (8b) using the nonlinear measurement functions (1b) and (3b) is not trivial. Since approximations have already been introduced, the extent is approximated using the unscented transform [20] of (8), and the sigma-points are mapped using (1b) and (3b). The mapped centroids $\mathbf{y}_j$ and covariances $\mathbf{Y}_j$ are recomputed and a mapped, reduced measurement set is obtained,

$$\mathbf{Z} = \{\mathbf{y}_j,\, \bar{d}_j, \mathbf{Y}_j\}_{j=1}^{M}. \tag{9}$$

V. Constrained Motion Model

To accurately track targets suitable motion and measurement models are important, and a nonlinear discrete state-space model is chosen on the form

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k) + \mathbf{w}_k \tag{10a}$$

$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k \tag{10b}$$

where $\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q})$ is the process noise, $\mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$ is the measurement noise, $\mathbf{x}_k$ is the state and $\mathbf{y}_k$ is the measurement, all at time k and using sampling time T.

Since the measurements are undistorted and mapped as described in Sec. III, the measurement model $\mathbf{h}(\mathbf{x}) = \begin{pmatrix} x & y \end{pmatrix}^T$ is used where x and y represent the target position.

A conventional motion model in target tracking applications is the constant velocity model [21], where the target state vector is

$$\mathbf{x}_k = \begin{pmatrix} x_k & y_k & \dot{x}_k & \dot{y}_k \end{pmatrix}^T \tag{11}$$

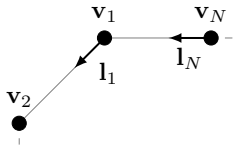

Fig. 2. Polygon representation.

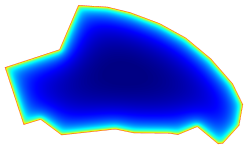

Fig. 3. Potential field illustration.

and the linear motion model is given by

$$\mathbf{f}(\mathbf{x}) = \begin{pmatrix} \mathbf{I}_2 & T\mathbf{I}_2 \\ \mathbf{0}_2 & \mathbf{I}_2 \end{pmatrix} \mathbf{x}. \tag{12}$$

Another conventional motion model is the coordinated turn model [21], where the target state vector is $\mathbf{x}_k = \begin{pmatrix} x_k & y_k & \dot{x}_k & \dot{y}_k & \omega_k \end{pmatrix}^T$ and the model is given by

$$\mathbf{f}(\mathbf{x}) = \begin{pmatrix} x + \frac{\dot{x}}{\omega}\sin(\omega T) - \frac{\dot{y}}{\omega}\big(1 - \cos(\omega T)\big) \\ y + \frac{\dot{x}}{\omega}\big(1 - \cos(\omega T)\big) + \frac{\dot{y}}{\omega}\sin(\omega T) \\ \dot{x}\cos(\omega T) - \dot{y}\sin(\omega T) \\ \dot{x}\sin(\omega T) + \dot{y}\cos(\omega T) \\ \omega \end{pmatrix}. \tag{13}$$

A. Constraint Model

When a target is constrained to a region, adapting the motion model to reflect this can improve the tracking performance. In the following, this is achieved by making a few assumptions about target behaviour close to the boundary of the region. The inspiration comes from research on potential fields [12] and collision avoidance for autonomous robots.

It is reasonable for a target moving towards a boundary to avoid it by turning when it gets close. A target moving along a nearby boundary is also assumed to follow it by turning to align its velocity. In general a target is assumed to move either in a clockwise or counter clockwise direction along the boundary of the region, determining the turning direction. The strength of the influence from each point $\mathbf{n}$ along the boundary is assumed to be a function $w(\mathbf{x}, \mathbf{n})$ of the state of the target and the position of the point.

Combining the effect of each point on the angular velocity by integrating along the boundary $\mathbf{N}$ of the region gives

$$\omega(\mathbf{x}) = d_r(\mathbf{x}) \int_{\mathbf{N}} \big(\beta_d + \beta_a(\dot{\mathbf{p}}_\perp \cdot \mathbf{l}(\mathbf{n}))\big) w(\mathbf{x}, \mathbf{n})\, d\mathbf{n}, \tag{14}$$

where $d_r(\mathbf{x}) \in \{-1, 1\}$ gives the rotational direction of the target, β_d and β_a are design parameters giving the strengths of avoidance and alignment respectively, $\dot{\mathbf{p}} = \dot{\mathbf{p}}(\mathbf{x}) = \begin{pmatrix} \dot{x} & \dot{y} \end{pmatrix}^T$, $\mathbf{l}(\mathbf{n})$ is the tangent of the boundary and the notation $\begin{pmatrix} a & b \end{pmatrix}_\perp^T = \begin{pmatrix} b & -a \end{pmatrix}^T$ is used.

B. Constraint Region Model

The boundary of the constraint region is modeled as a simple two-dimensional polygon and to avoid unexpected behaviour the polygon is assumed to be nearly convex. The polygon is defined by N vertices $\mathbf{v}_i$ for $i = 1, \ldots, N$, given in counter clockwise order. Points on each segment of the polygon are obtained from

$$\mathbf{n}_i(s) = \mathbf{v}_i + s\mathbf{l}_i, \quad s \in [0,\, m_i] \tag{15}$$

where $m_i = \|\mathbf{v}_{i+1} - \mathbf{v}_i\|$ and $\mathbf{l}_i = (\mathbf{v}_{i+1} - \mathbf{v}_i)/m_i$, as shown in Fig. 2, with obvious adjustments for m_N and $\mathbf{l}_N$.

The strength of the influence $w(\mathbf{x}, \mathbf{n})$ in (14) for a point $\mathbf{n}_i(s)$ on the boundary is modeled to diminish as

$$w_i(\mathbf{x}, s) = \frac{1}{\|\mathbf{p} - \mathbf{n}_i(s)\|^2} = \frac{1}{\|\mathbf{e}_i - s\mathbf{l}_i\|^2}, \tag{16}$$

where $\mathbf{p} = \mathbf{p}(\mathbf{x}) = \begin{pmatrix} x & y \end{pmatrix}^T$ and $\mathbf{e}_i = \mathbf{p} - \mathbf{v}_i$. Inserting (16) into (14) and using the region model in (15) gives the angular velocity

$$\omega(\mathbf{x}) = d_r(\mathbf{x}) \sum_{i=1}^{N} \big(\beta_d + \beta_a(\dot{\mathbf{p}}_\perp \cdot \mathbf{l}_i)\big) w_i(\mathbf{x}), \tag{17}$$

where, using $\|\mathbf{l}_i\| = 1$,

$$\begin{aligned} w_i(\mathbf{x}) &= \int_0^{m_i} w_i(\mathbf{x}, s)\, ds = \int_0^{m_i} \frac{1}{\|\mathbf{e}_i - s\mathbf{l}_i\|^2}\, ds \\ &= \frac{1}{\mathbf{l}_i^T \mathbf{e}_{i\perp}} \arctan\left[\frac{m_i \mathbf{l}_i^T \mathbf{e}_{i\perp}}{\|\mathbf{e}_i\|^2 - m_i \mathbf{l}_i^T \mathbf{e}_i}\right]. \end{aligned} \tag{18}$$

Fig. 3 illustrates (17) with $\beta_a = 0$ for a constraint region.

The Jacobians of the weights are given by

$$\frac{\partial w_i}{\partial \mathbf{x}}(\mathbf{x}) = \begin{pmatrix} w_{ix}(\mathbf{x}) & w_{iy}(\mathbf{x}) & 0 & 0 \end{pmatrix} \tag{19}$$

where, using the notation $\mathbf{a} = \begin{pmatrix} a_x & a_y \end{pmatrix}^T$,

$$\begin{aligned} w_{ix}(\mathbf{x}) = \frac{1}{\mathbf{l}_i^T \mathbf{e}_{i\perp}} \Bigg(& l_{iy} w_i(\mathbf{x}) + \frac{e_{iy}}{\|\mathbf{e}_i\|^2} \\ & - \frac{e_{iy} - l_{iy} m_i}{\|\mathbf{e}_i\|^2 - 2\mathbf{l}_i^T \mathbf{e}_i m_i + m_i^2} \Bigg) \end{aligned} \tag{20a}$$

and

$$\begin{aligned} w_{iy}(\mathbf{x}) = \frac{1}{\mathbf{l}_i^T \mathbf{e}_{i\perp}} \Bigg(& -l_{ix} w_i(\mathbf{x}) - \frac{e_{ix}}{\|\mathbf{a}_i\|^2} \\ & + \frac{e_{ix} - l_{ix} m_i}{\|\mathbf{e}_i\|^2 - 2\mathbf{l}_i^T \mathbf{e}_i m_i + m_i^2} \Bigg). \end{aligned} \tag{20b}$$

The direction of the rotation $d_r(\mathbf{x})$ can either be chosen using prior information or be estimated by comparing the target velocity $\dot{\mathbf{p}}$ to the boundary directions $\mathbf{l}_i$, *e.g.* using

$$d_r(\mathbf{x}) = \operatorname{sign}\left(\sum_{i=1}^{N} (\dot{\mathbf{p}} \cdot \mathbf{l}_i) w_i(\mathbf{x}) \right). \tag{21}$$

C. Constrained Motion Model

The motion model chosen is a coordinated turn model with known angular velocity [21]. The continuous state vector is $\mathbf{x} = \begin{pmatrix} x & y & \dot{x} & \dot{y} \end{pmatrix}^T$ and the motion model is

$$\dot{\mathbf{x}}(t) = \mathbf{f}_c\big(\mathbf{x}(t), t\big) + \mathbf{w}(t) \tag{22a}$$

where $\mathbf{w}(t)$ is the process noise and

$$\mathbf{f}_c\big(\mathbf{x}(t),t\big) = \begin{pmatrix} \dot{x}(t) \\ \dot{y}(t) \\ -\omega\big(\mathbf{x}(t)\big)\dot{y}(t) \\ \omega\big(\mathbf{x}(t)\big)\dot{x}(t) \end{pmatrix}. \tag{22b}$$

With a temporary zero-order hold assumption on $\omega(\mathbf{x}) = \omega$ and the state vector in (11), (22b) is discretized exactly as

$$\mathbf{f}(\mathbf{x},\omega) = \begin{pmatrix} x + \frac{\dot{x}}{\omega}\sin(\omega T) - \frac{\dot{y}}{\omega}\big(1-\cos(\omega T)\big) \\ y + \frac{\dot{x}}{\omega}\big(1-\cos\omega T\big) + \frac{\dot{y}}{\omega}\sin(\omega T) \\ \dot{x}\cos(\omega T) - \dot{y}\sin(\omega T) \\ \dot{x}\sin(\omega T) + \dot{y}\cos(\omega T) \end{pmatrix}. \tag{23}$$

Reintroducing $\omega = \omega(\mathbf{x})$, the Jacobian with regards to $\mathbf{x}$ is

$$\mathbf{F}\big(\mathbf{x},\omega(\mathbf{x})\big) = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}\big(\mathbf{x},\omega(\mathbf{x})\big) + \frac{\partial \mathbf{f}}{\partial \omega}\big(\mathbf{x},\omega(\mathbf{x})\big)\frac{\partial \omega(\mathbf{x})}{\partial \mathbf{x}} \tag{24a}$$

using the chain rule, where

$$\frac{\partial \mathbf{f}}{\partial \mathbf{x}} = \begin{pmatrix} 1 & 0 & \frac{\sin\omega T}{\omega} & -\frac{1-\cos\omega T}{\omega} \\ 0 & 1 & \frac{1-\cos\omega T}{\omega} & \frac{\sin\omega T}{\omega} \\ 0 & 0 & \cos\omega T & -\sin\omega T \\ 0 & 0 & \sin\omega T & \cos\omega T \end{pmatrix}, \tag{24b}$$

$$\frac{\partial \mathbf{f}}{\partial \omega} = \begin{pmatrix} \frac{(\omega T\dot{x}-\dot{y})\cos(\omega T)-(\dot{x}+\omega T\dot{y})\sin(\omega T)+\dot{y}}{\omega^2} \\ \frac{(\dot{x}-\omega T\dot{y})\cos(\omega T)+(\dot{y}-\omega T\dot{x})\sin(\omega T)-\dot{x}}{\omega^2} \\ -T\dot{y}\cos(\omega T) - T\dot{x}\sin(\omega T) \\ T\dot{x}\cos(\omega T) - T\dot{y}\sin(\omega T) \end{pmatrix} \tag{24c}$$

and using (20)

$$\frac{\partial \omega}{\partial \mathbf{x}} = \sum_{i=1}^{N} \begin{pmatrix} (\beta_d + \beta_a \dot{\mathbf{p}}_\perp^T \mathbf{l}_i) w_{ix}(\mathbf{x}) \\ (\beta_d + \beta_a \dot{\mathbf{p}}_\perp^T \mathbf{l}_i) w_{iy}(\mathbf{x}) \\ -\beta_a l_{iy} w_i(\mathbf{x}) \\ \beta_a l_{ix} w_i(\mathbf{x}) \end{pmatrix}^T. \tag{24d}$$

Care is needed in implementations when $\omega \to 0$ for (24b) and (24c), where, *e.g.*, (24c) reduces to

$$\lim_{\omega\to 0} \frac{\partial \mathbf{f}}{\partial \omega} = \begin{pmatrix} -\frac{T^2\dot{y}}{2} & \frac{T^2\dot{x}}{2} & -T\dot{y} & T\dot{x} \end{pmatrix}^T. \tag{25}$$

VI. Target Tracking

To associate related measurements generated by the measurements pre-processing over time and estimate target trajectories, a target tracking filter is needed. The *probabilistic data association* (PDA) filter [22, Ch. 6] with some modifications is used for association.

The *extended Kalman filter* (EKF) [23] is chosen for estimating the target states $\mathbf{x}_k$ from measurements $\mathbf{y}_k$ using the models described in Sec. V. The EKF is separated into a prediction update, using the motion model $\mathbf{f}(\mathbf{x})$, its Jacobian $\mathbf{F}(\mathbf{x})$ and the noise covariance $\mathbf{Q}$, and a measurement update, using the measurement $\mathbf{y}_k$, the measurement model $\mathbf{h}(\mathbf{x})$, and the noise covariance $\mathbf{R}$. The noise covariances are considered design parameters and are selected to achieve good performance. The output is the state estimate $\hat{\mathbf{x}}_k$ and its covariance $\mathbf{P}_k$

A. Probabilistic Data Association Filter

A common filter used for point targets is the PDA filter, which constructs a hypothesis for each gated measurement that it is generated by the target, and then proceeds to merge all hypotheses weighted by the probability that the measurement was generated by the target. Although the assumption in Sec. IV-B is that each target generates several measurements, the filter assumes that there is at most one measurement generated by the target, which gives the side-effect that the state covariance grows, and the innovation covariance can be seen as an approximate measure of the extent.

All measurements not associated with a track are considered clutter, which is modeled with a Poisson-Uniform distribution with intensity density β, resulting in the probability

$$\Pr(\theta_j|\mathbf{Z}^k) \propto \beta^{N-1}\mathcal{N}(\mathbf{y}_j;\hat{\mathbf{y}}_{k|k-1},\mathbf{S}_{k|k-1})P_D \tag{26a}$$

of hypothesis θ_j that measurement j was generated by the target, where $\hat{\mathbf{y}}_{k|k-1}$ and $\mathbf{S}_{k|k-1}$ are the predicted EKF measurement and innovation covariance respectively and P_D is a design parameter defining the probability of detection. The probability of hypothesis θ_0 that all measurements are clutter is

$$\Pr(\theta_0|\mathbf{Z}^k) \propto \beta^N(1 - P_D P_G), \tag{26b}$$

where P_G is the gate probability. The weights for the hypotheses are computed as

$$\mu_j = \frac{\Pr(\theta_j|\mathbf{Z}^k)}{\sum_{i=0}^{N}\Pr(\theta_i|\mathbf{Z}^k)} \tag{27}$$

and the hypotheses are merged using

$$\hat{\mathbf{x}}_{k|k} = \sum_{j=0}^{M_r} \mu_j \hat{\mathbf{x}}_{k|k}^j \tag{28a}$$

$$\mathbf{P}_{k|k} = \mu_0\mathbf{P}_{k|k-1} + (1-\mu_0)\,\mathbf{P}_{k|k} + \sum_{j=0}^{M_r} \mu_j(\hat{\mathbf{x}}_{k|k}^j - \hat{\mathbf{x}}_{k|k})(\hat{\mathbf{x}}_{k|k}^j - \hat{\mathbf{x}}_{k|k})^T \tag{28b}$$

where $\hat{\mathbf{x}}_{k|k}^0 = \hat{\mathbf{x}}_{k|k-1}$ is the EKF predicted state estimate, $\hat{\mathbf{x}}_{k|k}^j$ is the EKF updated state estimate using measurement $\mathbf{y}_j$ and $\mathbf{P}_{k|k-1}$ is the EKF predicted state covariance.

B. Modified Probabilistic Data Association Filter

The filter used in the proposed solution is inspired by the PDA filter equations in Sec. VI-A. The PDA has been modified so that the probability of a measurement anywhere in the gate is the same, that is measurements are uniformly distributed in the gate. Furthermore, each measurement represents a number of actual measurements (foreground pixels), hence multiplicity is approximated by the size of the observation

$$n_j = |\mathbf{Y}_j| \tag{29a}$$

or by the size and confidence, interpreted as the density of measurements, as

$$n_j = \bar{d}_j|\mathbf{Y}_j|. \tag{29b}$$

The probability for θ_j becomes

$$\Pr(\theta_j|\mathbf{Z}^k) \propto \frac{\beta^{N-1}P_D}{V_k} = \frac{\beta^{N-1}P_D}{\pi\sqrt{|\mathbf{S}_{k|k-1}|\gamma}}, \tag{30a}$$

where $|\cdot|$ is the determinant and γ is a design parameter determining the area V_k of the gate and for θ_0

$$\Pr(\theta_0|\mathbf{Z}^k) \propto \beta^N(1-P_D). \tag{30b}$$

The weights are computed, where $n_0 = 1$, using

$$\mu_j = \frac{n_j \Pr(\theta_j|\mathbf{Z}^k)}{\sum_{i=0}^{N} n_i \Pr(\theta_i|\mathbf{Z}^k)}. \tag{31}$$

C. Track Management

Track management is performed in three steps:

1) Measurements are associated to and used to update only confirmed tracks.
2) Unassociated measurements are associated and used to update tentative tracks.
3) All remaining measurements are used to initiate new tracks.

Furthermore, M/N-logic is used to determine whether to confirm or delete tracks. If a track has N_1 gated measurements in consecutive frames and subsequently M gated measurements in the next N_2 frames, the track is confirmed, otherwise deleted. If a confirmed track has D missed consecutive measurements while in the detection region, shown in Fig. 4 and 9, it is deleted.

On top of this, similar tracks are merged based on the Bhattacharyya distance [24], $d_B(i,j)$,

$$d_B(i,j) = \frac{1}{4}(\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_j)^T(\mathbf{P}_i + \mathbf{P}_j)^{-1}(\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_j) + \frac{1}{2}\ln\left(\frac{|(\mathbf{P}_i + \mathbf{P}_j)/2|}{\sqrt{|\mathbf{P}_i||\mathbf{P}_j|}}\right). \tag{32}$$

If a set of tracks $\mathcal{M}$ satisfies $d_B(i,j) \leq \gamma_m$ for all $i,j \in \mathcal{M}$, where γ_m is a design parameter, the tracks are merged. Tracks are merged into one using [25]

$$\hat{\mathbf{x}}_n = \sum_{i\in\mathcal{M}} w_i\hat{\mathbf{x}}_i, \tag{33a}$$

$$\mathbf{P}_n = \sum_{i\in\mathcal{M}} w_i\big(\mathbf{P}_i + (\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_n)(\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_n)^T\big), \tag{33b}$$

where the weight $w_i = |\mathbf{P}_i|/\sum_{i\in\mathcal{M}}|\mathbf{P}_i|$ is chosen to prioritize tracks with a large extent.

VII. Examples and Results

In this section, the proposed tracking solution is evaluated using actual video footage from the dolphinarium at Kolmården Wildlife Park. See Fig. 4 for an example. Preferably the solution should be able to extract a trajectory for each individual dolphin, however, due to resolution and occlusion this is very difficult and in many situations impossible. Additionally, that level of detail is not required for the intended behavioural study. The aim is therefore to instead track groups of dolphins with the goal to maintain a track for a group and maintain the track in occluded regions.

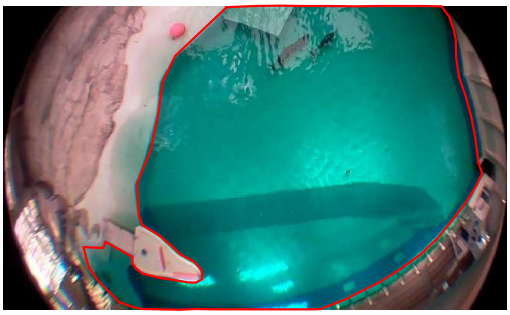

Fig. 4. A frame from the video with the chosen detection region marked in red. The reflections at the top have a high variance, while reflections at the bottom are more stable. There is one group of dolphins that is slightly difficult to segment due to the reflections, but they are easily visible to the eye and there is one hard-to-see dolphin down at the bottom left.

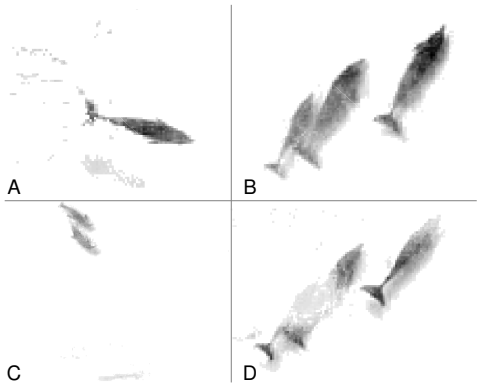

Fig. 5. Shows the Mahalanobis distance output for segmented pixels in the foreground segmentation for four situations. A properly segmented target together with noise and a prominant shadow (A). Three separate targets where the two on the left would be combined by thresholding (B). A faint target at the bottom for which the track is maintained (C). Targets partly disappearing in a reflective region (D).

The performances of the solutions are evaluated qualitatively by comparing the performances for the various filters and models in difficult situations. The main setup used is the modified PDA filter described in Sec. VI-B using multiplicity (29b) and the constrained motion model based on (23).

A. Foreground Segmentation

The tracking relies on the output from the measurement preprocessing and some examples of segmented targets are shown in Fig. 5. The quality of the output varies over the region and over time depending on *e.g.* the stability of the background, separation of targets, camera resolution and distortion and light conditions. Although more information could probably be extracted using tailored computer vision techniques, thresholding is good enough for the intended group target tracking and to use general methods is beneficial when applying the same solution to other similar problems.

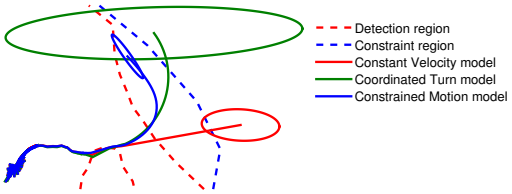

Fig. 6. Compares the predictive capabilities and the innovation covariance of different models in a non-detection region when a track stops receiving measurements. The coordinated turn and constant velocity models do not take the constraint region into account resulting in infeasible predictions. The constrained motion model keeps predictions within the constraint region.

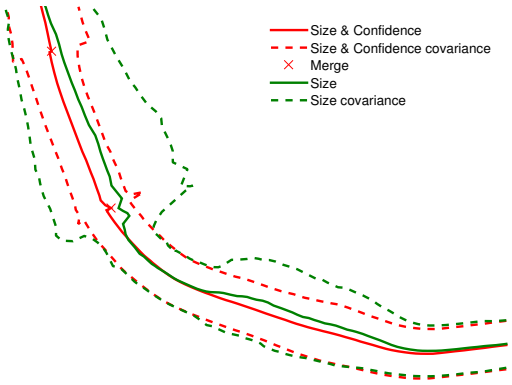

Fig. 7. Compares the estimated trajectory and the innovation covariance size in the presence of a shadow using size and confidence as well as only size as multiplicity of the measurements.

B. Model Comparison

Conventional models do not take the physical constraints into account, this is why the constrained motion model was proposed. To show the differences in behaviour between the models, the prediction of each model with the resulting innovation covariance when no measurements are received is shown in Fig. 6. The conventional models produce infeasible predictions and if the target is rediscovered, due to a large gate or the prediction returning to the constraint region, the estimated trajectory is infeasible. The constrained motion model prediction, however, follows the boundary of the constraint region with an innovation covariance better adapted to the actual uncertainty of the position. The uncertainty in the velocities is propagated to the uncertainty in the position, causing rapidly increasing innovation covariance for the conventional models, while the constrained motion model starts by increasing the uncertainty in position and then decreases it as the boundary is approached, although eccentricity still increases.

To improve the constrained motion model even further, the predictions from it should cover motions in both directions along a boundary until measurements have been acquired to distinguish which direction the target went. However, in most situations this seems not to be a major problem.

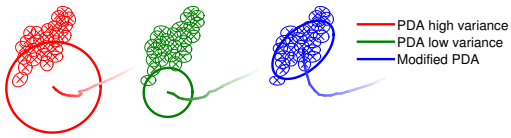

Fig. 8. Compares the track innovation covariance for the standard PDA filter, using high and low Gaussian measurement noise covariance, with the modified PDA filter during a sharp turn.

C. Multiplicity Comparison

Targets cast shadows, as seen in Fig. 5, which often has a smaller confidence than the targets. Using only size as in (29a) to determine the measurement cluster multiplicity, all foreground measurements will have similar weights in the target tracking filter, while including the confidence as in (29b) puts higher weights on true targets than shadows. To compare the two options the results in the presence of the shadow in Fig. 5A is shown in Fig. 7. Using only size the trajectory is seen to be sensitive to the shadow, since the same weight is put on all measurements, but when including the confidence the shadow measurements are seen to have less impact, giving a smoother trajectory estimate and an innovation covariance that mainly covers the target.

The side effect is that a new track is initiated on the shadow, which, however, is quickly merged into the original track with little effect to its trajectory.

D. Filter Comparison

Using a standard PDA filter the measurements are assumed to be Gaussian distributed around the target position, effectively giving more weight to measurements near the centroid resulting in poor estimation of the target extent. To handle this, a variation of the PDA filter was proposed. It assumes uniformly distributed measurements in the gate, (30a), and utilizes the multiplicity of the measurements (31). To compare the two options the filter performances were evaluated in a sharp turn. The result is given in Fig. 8. The standard PDA filters struggle to track the target through the turn for various choices of process and measurement noise covariances, while the modified PDA filter not only finds the centroid, but also adapts its innovation covariance to match the extent of the target, improving the performance.

E. Trajectory Extraction

Using the main setup, the trajectory for one group of dolphins is shown in Fig. 9. The red line shows the mapped detection region and it can be seen that the mapping is inaccurate in some areas. Several tracks, initiated at the blue circles, are merged into the track along the way and, although not showing in the figure, several individuals leave the group along the trajectory, initiating new tracks. The advantage of the constrained motion model is displayed at the bottom left where the target disappears for over 100 frames while the track is maintained.

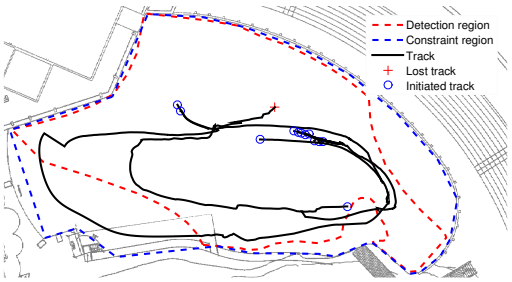

Fig. 9. A track of a group of targets, with individuals joining and splitting from the group. The filter manages to keep the track while the target passes straight through the non-detection region at the bottom right and when the target disappears for a long time in the non-detection region at the bottom left.

VIII. Conclusion

This paper has proposed a method to track dolphins using a ceiling-mounted fisheye camera intended to help biologists obtain trajectories for further studies of their behaviour. The whole pipeline from foreground segmentation to target tracking is described, where the target tracking techniques are used to handle pre-processing imperfections. To achieve this a novel motion model that affects the heading to avoid collisions with the basin edges at the same time as it favours trajectories along the edges is suggested, as well as adaptations to the standard PDA filter to handle extended targets.

The solution performs very well on recorded video data, and will provide a tool for biologists to avoid a lot of tedious manual work. The results show that the foreground segmentation is able to extract the dolphins from the video with sufficient accuracy, despite complicating factors, such as reflections, shadows, distortions and changing light conditions. The target tracking framework is shown to be able to handle false detections, limited field of view and occlusions. It is also shown that the proposed constrained motion model can maintain tracks during long periods without detections when conventional constant velocity and coordinated turn models fail. The feedback from the involved biologists regarding the results has also been very positive.

Each individual step of the pipeline can be improved. The most interesting possibility is to introduce feedback from the target tracking block to improve the measurement preprocessing. This could be beneficial for the foreground segmentation, especially if combined with explicit handling of extended targets and groups of targets.

Acknowledgment

The authors greatly acknowledge funding from Vinnova Industry Excellence Center LINK-SIC and the Swedish strategic research center Security Link. The authors would like to thank Prof. Mats Amundin and Laura van Zonneveld at Kolmården Wildlife Park for inspiring discussions and providing invaluable data. The authors would also like to thank Dr. Emre Özkan and Dr. Martin Skoglund for contributing valuable input and feedback.

References

[1] N. Wahlström, F. Gustafsson, and S. Åkesson, "A voyage to africa by mr Swift," in *International Conference on Information Fusion (FUSION)*, Jul. 2012, pp. 808–815.

[2] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. Computer Vision and Pattern Recognition*, Jun. 1999, pp. 2246–2253.

[3] P. W. Power and J. A. Schoonees, "Understanding background mixture models for foreground segmentation," in *Proc. Image and Vision Computing*, New Zealand, 2002.

[4] L. Taycher, J. W. Fischer, and T. Darrel, "Incorporating object tracking feedback into background maintenance framework," in *Proc. IEEE Workshop on Motion and Video Computing*, 2005.

[5] S. Brutzer, B. Hoferlin, and G. Heidemann, "Evaluation of background subtraction techniques for video surveillance," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Providence, RI, USA, Jun. 2011.

[6] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 743–761, Apr. 2012.

[7] C. Szegedy, A. Toshev, and D. Erhan, "Deep neural networks for object detection," in *Advances in Neural Information Processing Systems 26*, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2013, pp. 2553–2561.

[8] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, vol. 1, San Diego, CA, USA, Jun. 2005.

[9] R. T. Collins, "Mean-shift blob tracking through scale space," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, vol. 2, Jun. 2003.

[10] D. Reid, "An algorithm for tracking multiple targets," *IEEE Trans. Autom. Control*, vol. 24, no. 6, pp. 843–854, Dec. 1979.

[11] K.-C. Chang and Y. Bar-Shalom, "Joint probabilistic data association for multitarget tracking with possibly unresolved measurements and maneuvers," *IEEE Trans. Autom. Control*, vol. 29, no. 7, pp. 585–594, Jul. 1984.

[12] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in *Proc. IEEE International Conference onRobotics and Automation*, vol. 2, Mar. 1985, pp. 500–505.

[13] J. Y. Bouguet, "Camera calibration toolbox for Matlab," http://www.vision.caltech.edu/bouguetj/calib_doc/, 2010.

[14] F. Devernay and O. Faugeras, "Straight lines have to be straight: Automatic calibration and removal of distortion from scenes of structured enviroments," *Mach. Vision Appl.*, vol. 13, no. 1, pp. 14–24, Aug. 2001.

[15] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 11, pp. 1330–1334, Nov. 2000.

[16] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society, Series B*, vol. 39, no. 1, pp. 1–38, 1977.

[17] M. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. Van Gool, "Online multiperson tracking-by-detection from a single, uncalibrated camera," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 9, pp. 1820–1833, Sep. 2011.

[18] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 3rd ed. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 2008.

[19] J. B. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability*, L. M. L. Cam and J. Neyman, Eds., vol. 1. University of California Press, 1967, pp. 281–297.

[20] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte, "A new method for the nonlinear transformation of means and covariances in filters and estimators," *IEEE Trans. Autom. Control*, vol. 45, no. 3, Mar. 2000.

[21] R. X. Li and V. P. Jilkov, "Survey of maneuvering target tracking. Part I: Dynamic models," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 39, no. 4, pp. 1333–1364, Oct. 2003.

[22] S. S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*. Artech House, 1999.

[23] A. H. Jazwinski, *Stochastic Processes and Filtering Theory*, ser. Mathematics in Science and Engineering. Academic Press, Inc, 1970, vol. 64.

[24] A. Bhattacharyya, "On a measure of divergence between two statistical populations defined by their probability distributions," *Bulletin of Cal. Math. Soc.*, vol. 35, no. 1, pp. 99–109, 1943.

[25] A. R. Runnalls, "A Kullback-Leibler approach to Gaussian mixture reduction," *IEEE Trans. Aerosp. Electron. Syst.*, pp. 989–999, 2007.