Silhouette Measurements for Bayesian Object Tracking in Noisy Point Clouds

Florian Faion, **Marcus Baum**, and **Uwe D. Hanebeck**
Intelligent Sensor-Actuator-Systems Laboratory (ISAS)
Institute for Anthropomatics
Karlsruhe Institute of Technology (KIT), Germany
florian.faion@kit.edu, marcus.baum@kit.edu, uwe.hanebeck@ieee.org

Abstract—**In this paper, we consider the problem of jointly tracking the pose and shape of objects based on noisy data from cameras and depth sensors. Our proposed approach formalizes object silhouettes from image data as measurements within a Bayesian estimation framework. Projecting object silhouettes from images back into space yields a visual hull that constrains the object. In this work, we focus on the 2D case. We derive a general equation for the silhouette measurement update that explicitly considers segmentation uncertainty of each pixel. By assuming a bounded error for the silhouettes, we can reduce the complexity of the general solution to only consider uncertain edges and derive an approximate measurement update. In simulations, we show that the proposed approach dramatically improves point-cloud-based estimators, especially in the presence of high noise.**

Index Terms—**Silhouettes, Point Clouds, Extended Object Tracking, Shape and Pose Estimation**

I. Introduction

Object tracking based on data from cameras and depth sensors has several applications in health care, automotive safety, telepresence, entertainment, and surveillance. Especially since the launch of the Microsoft Kinect in 2010, tracking objects with RGBD-cameras has become very popular. These sensors use structured light as their measurement principle. Unfortunately, interference between multiple depth sensors with overlapping views is an inherent problem of the structured light. As a consequence, the sensors are often scheduled by sequentially turning on and off the depth stream [1], [2]. However, while the depth becomes occasionally unavailable, the color stream remains online, though often unused.

In this work, we incorporate this image information from a camera into a Bayesian object tracking algorithm. In doing so, we focus on the 2D case and simply connected objects. The resulting algorithm can be applied to the described scenario as well as to arbitrary heterogeneous sensor networks with depth sensors and cameras. We start from a general formulation for silhouette measurements in a Bayesian sense that models a pixel-wise segmentation uncertainty for each pixel. However, these formulas turn out to be rather inconvenient as each pixel has to be considered explicitly and, in addition, the evaluation carries numeric instabilities. We propose to simplify the problem by making a few assumptions on the segmentation. In more detail, we assume that the object can be extracted from the image very accurately up to a certain bounded error. This allows approximating the silhouette by its edge and transforming the pixel-wise uncertainty into a spatial uncertainty. This reduces the dimension of the measurement from all image pixels to only two edge pixels, as we consider the 2D case.

A. Related Work

The concept of shape estimation from silhouettes by means of the visual hull was introduced in [3] and formalizes geometric constraints on an object. These constraints were applied to image based shape estimation [4] in a Bayesian framework. Incorporation geometric constraints into shape estimation was also proposed by applying *projection onto convex sets* (POCS) [5]. Another approach [6] presented a recursive Bayesian pose estimation based on feature projections. Exploiting edges and discontinuities in images in order to infer the object pose was also proposed in [7]. Fusing depth and color images for segmentation and tracking based on mean-shift was discussed in [8]. A grid based surface reconstruction was presented in [9] by fusing time-of-flight data with multiple images. In [10], a similar strategy of fusing depth with image data is pursued that aims at estimating the pose of a human. However this approach does not explicitly model the noise, and thus fusion of multiple sensors is not straight forward. The point-cloud-based estimator for ellipses presented in our previous work [11] is used for comparison in the simulation. The measurement update for point clouds of the estimator proposed in this work is also quite similar, with exception of the employed distance measure. In [12], they estimate the shape of extended objects based on radar measurements and infer different target types. Also related is the work of [13], where the object shape is inferred from support and diameter function measurements. This work differs from related work in the sense that

1) measurement uncertainty of silhouettes is explicitly modeled,
2) and thus, fusion with point clouds can be performed in a natural and mathematically sound way within the Bayesian estimation framework.

B. Overview

The remainder of this paper is structured as follows. In Sec. II, we outline given and desired parameters and introduce silhouette measurements. Based on the problem formulation,

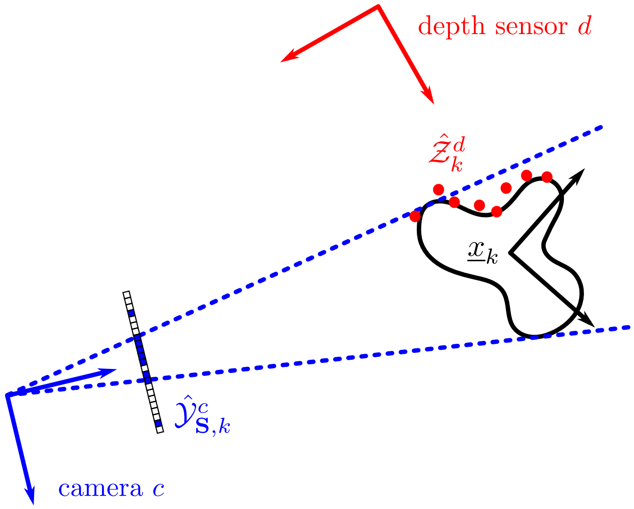


Figure 1: Sketch of the problem formulation. **Given**: Noisy silhouette $\hat{\mathcal{Y}}^c_{\mathbf{S},k}$ and point cloud $\hat{\mathcal{Z}}_k^d$ measurements of an object, measured at time k by camera c and depth sensor d, respectively. **Desired**: Shape and pose parameters $\underline{x}_k$ of the object.

theoretic concepts for a point-cloud-based and silhouette-based Bayesian estimator are explained in Sec. III. Starting from these theoretic concepts, the novel silhouette-based estimator is derived step-by-step in Sec. IV. Sec. V considers a specific instance of the proposed estimator for ellipse objects objects. In Sec. VI, an extensive comparison to a state-of-the-art approach is performed. Finally, Sec. VII concludes with a short summary and an outlook to future work.

II. Problem Formulation

We consider the problem of estimating the pose and shape of an object based on silhouette data from cameras and point clouds from depth sensors. An illustration is given in Fig. 1.

The state $\underline{x}_k$ of the object is to be determined as a set of pose (position, orientation) and shape parameters at each time k with

$$\underline{x}_k = \begin{bmatrix} \underline{x}_k^{\text{pose}} \\ \underline{x}_k^{\text{shape}} \end{bmatrix} \quad .$$

Shape parameters could be, e.g., the radius of a circle, or the axes lengths of an ellipse. As mentioned, two measurement modalities, point clouds and silhouettes, are considered.

Point Cloud Data

A measurement of depth sensor d at time k is a set of surface points

$$\hat{\mathcal{Z}}_k^d = \{\underline{\hat{z}}_{i,k}^d | i = 1, \dots, n_k^d\}$$

of the object[1]. Each point is assumed to be generated by a source on the object surface $\underline{z}_{src}$ by

$$\underline{\hat{z}} = \underline{z}_{src} + \underline{\boldsymbol{w}}_{\underline{z}_{src}} \tag{1}$$

that is subject to an additive, zero-mean Gaussian noise term $\underline{\boldsymbol{w}}_{\underline{z}_{src}} \sim \mathcal{N}(\underline{0}, \mathbf{C}_{\underline{z}_{src}})$.

Silhouette Data

The second modality, i.e., a silhouette measurement of camera c at time k is a set of image coordinates

$$\hat{\mathcal{Y}}^c_{\mathbf{S},k} = \{\underline{\hat{y}}^c_{i,k} | i = 1, \dots, n_k^c\}$$

of pixels that lie within the projected object. All other pixels are aggregated to a set of background pixel coordinates $\hat{\mathcal{Y}}^c_{\neg\mathbf{S}}$, so that $\hat{\mathcal{Y}}_{\mathbf{S}} \cap \hat{\mathcal{Y}}_{\neg\mathbf{S}} = \emptyset$. All silhouette pixels have a certain probability to be classified correctly $p(\text{TP})$, denoted as *true positive*. Analogously, all background pixels have a probability $p(\text{TN})$ to be *true negative*. An example of this classification for each pixel of an image is shown in Fig. 2.

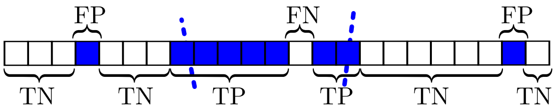


Figure 2: Close-up of the silhouette measurement that is shown in Fig. 1. Each pixel is marked as *true positive* (TP), *true negative* (TN), *false positive* (FP), or *false negative* (FN). The dashed line reflects the real object edge.

In addition, the extrinsic parameters

$$\mathbf{H} = \begin{bmatrix} \mathbf{R} & \underline{t} \\ \underline{0}^{\mathrm{T}} & 1 \end{bmatrix}$$

with rotation matrix $\mathbf{R}$ and translation vector $\underline{t}$, are assumed to be known for all sensors, as well as the intrinsic parameters $\mathbf{K}$ for each camera.

III. Theoretical Concept

In this section, we look at the theoretical concept of estimating $\underline{x}$ from a Bayesian point of view. Based on a prior estimate of the object parameters $p(\underline{x})$ a measurement update is derived for both types of measurements.

A. Point-Cloud-Based Estimator

Fitting a parametric shape $\underline{x}$ to noisy points $\hat{\mathcal{Z}}$ is a well-studied problem [11], [14]. The general solution by means of a Bayes update can be written as

$$\begin{aligned} p(\underline{x}|\hat{\mathcal{Z}}) &= \frac{p(\hat{\mathcal{Z}}|\underline{x})}{\int p(\hat{\mathcal{Z}}|\underline{x}) \cdot p(\underline{x}) d\underline{x}} \cdot p(\underline{x}) \\ &\propto p(\hat{\mathcal{Z}}|\underline{x}) \cdot p(\underline{x}) \quad . \end{aligned}$$

In accordance to related work we assume all measured points $\underline{\hat{z}} \in \hat{\mathcal{Z}}$ to be independent. This leads to

$$p(\underline{x}|\hat{\mathcal{Z}}) \propto p(\underline{x}) \cdot \prod_i p(\underline{\hat{z}}_i|\underline{x}) \quad .$$

In order to evaluate the likelihood $p(\underline{\hat{z}}_i|\underline{x})$, the relationship between points and the object $\underline{x}$ has to be considered. This can be done by a *signed distance function* (SDF), such that

$$\text{SDF}(\underline{x}, \underline{z}) = \begin{cases} + \parallel \underline{z}_{\underline{x}} - \underline{z} \parallel, & \underline{z} \text{ inside or on contour,} \\ - \parallel \underline{z}_{\underline{x}} - \underline{z} \parallel, & \underline{z} \text{ outside contour,} \end{cases}$$

[1] Note that when possible, time indices k, sensor indices and measurement indices i have been omitted for clarity.

where $\underline{z}_{\underline{x}}$ is the closest point to $\underline{z}$ on the object contour, according to the specific distance $\|\ \|$. Especially, for all points $\underline{z}_{src}$ that lie on the object contour

$$\text{SDF}(\underline{x}, \underline{z}_{src}) = 0$$

holds. Together with (1), this leads to a nonlinear measurement equation with multiplicative noise

$$\text{SDF}(\underline{x}, \hat{\underline{z}} - \underline{\boldsymbol{w}}_{\underline{z}_{src}}) = 0 \ .$$

Statistical linearization [15] can be applied to approximately calculate the update by means of a sample-based filter, e.g., the *Unscented Kalman Filter* (UKF) [16].

B. Silhouette-Based Estimator

Let us consider the estimation of $\underline{x}$ based on silhouette measurements. Given the measurement $\hat{\mathcal{Y}}_{\mathbf{S}}$, $\hat{\mathcal{Y}}_{\neg\mathbf{S}}$ for a certain camera c, the prior estimate $p(\underline{x})$ can be updated according to

$$\begin{aligned} p(\underline{x}|\hat{\mathcal{Y}}_{\mathbf{S}}, \hat{\mathcal{Y}}_{\neg\mathbf{S}}) =& \frac{p(\hat{\mathcal{Y}}_{\mathbf{S}}, \hat{\mathcal{Y}}_{\neg\mathbf{S}}|\underline{x})}{\int p(\hat{\mathcal{Y}}_{\mathbf{S}}, \hat{\mathcal{Y}}_{\neg\mathbf{S}}|\underline{x})\cdot p(\underline{x}) d\underline{x}} \cdot p(\underline{x}) \qquad (2) \\ &\propto p(\hat{\mathcal{Y}}_{\mathbf{S}}, \hat{\mathcal{Y}}_{\neg\mathbf{S}}|\underline{x}) \cdot p(\underline{x}) \ . \end{aligned}$$

Under the assumption that all pixels are measured independently, the likelihood function $p(\hat{\mathcal{Y}}_{\mathbf{S}}, \hat{\mathcal{Y}}_{\neg\mathbf{S}}|\underline{x})$ can be written as

$$\begin{aligned} p(\hat{\mathcal{Y}}_{\mathbf{S}}, \hat{\mathcal{Y}}_{\neg\mathbf{S}}|\underline{x}) &= p(\hat{\mathcal{Y}}_{\mathbf{S}}|\underline{x}) \cdot p(\hat{\mathcal{Y}}_{\neg\mathbf{S}}|\underline{x}) \qquad (3) \\ &= \prod_i p(\hat{y}_{\mathbf{S},i}|\underline{x}) \cdot \prod_j p(\hat{y}_{\neg\mathbf{S},j}|\underline{x}) \ . \end{aligned}$$

For evaluating (3), the relationship between an object and its silhouette has to be defined.

Definition 1 (Silhouette)
For a given object represented by the parameters $\underline{x}$ and a given camera, the mapping $\mathbf{S}(\underline{x})$ determines a set of pixel coordinates in the camera image that belong to the object with

$$\begin{aligned} \mathbf{S} : \mathbb{R}^{n_{\underline{x}}} &\to \mathcal{P}(\mathbb{N}^n) \\ \underline{x} &\mapsto \{y_i \mid y_i \in \textit{projection of } \underline{x}\} \ . \end{aligned}$$

All pixels with $y_i \in \mathbf{S}(\underline{x})$ form the silhouette of object $\underline{x}$ in the camera image.

Employing this silhouette mapping leads to

$$p(\hat{y}_{\mathbf{S},i}|\underline{x}) = \begin{cases} p(\text{TP}), & \hat{y}_{\mathbf{S},i} \in \mathbf{S}(\underline{x}) \\ 1 - p(\text{TN}), & \hat{y}_{\mathbf{S},i} \in \neg\mathbf{S}(\underline{x}) \end{cases} , \qquad (4)$$

and

$$p(\hat{y}_{\neg\mathbf{S},j}|\underline{x}) = \begin{cases} p(\text{TN}), & \hat{y}_{\neg\mathbf{S},j} \in \neg\mathbf{S}(\underline{x}) \\ 1 - p(\text{TP}), & \hat{y}_{\neg\mathbf{S},j} \in \mathbf{S}(\underline{x}) \end{cases} . \qquad (5)$$

Due to the nonlinearity of (4) and (5), the Bayes update cannot be derived analytically. Instead, as an approximation, a sample-based update can be performed. However, applying (3) directly causes numeric instabilities as a consequence of multiplying a huge number of probability values between 0 and 1. In the following section, we show that the problem can be dramatically reduced by stating a few simplifying assumptions.

IV. Proposed Silhouette-Based Estimator

The key idea of this work is to simplify the presented silhouette-based estimator from Sec. III-B by approximating the pixel-wise segmentation uncertainty with a spatial uncertainty in the image domain. In a first preliminary step, the relationship between the edge and measured pixels is defined.

Definition 2 (Edge)
For a given object that is represented by the parameters $\underline{x}$ that is visible to a given camera (so that $\mathbf{S}(\underline{x}) \neq \emptyset$), the mapping $\boldsymbol{E}_*(\underline{x})$ determines the pixel coordinate of the left silhouette edge

$$\begin{aligned} \boldsymbol{E}_* : \mathbb{R}^{n_{\underline{x}}} &\to \mathbb{N} \\ \underline{x} &\mapsto \min(\mathbf{S}(\underline{x})) \ , \end{aligned}$$

and $\boldsymbol{E}^*(\underline{x})$ the coordinate of the right edge

$$\begin{aligned} \boldsymbol{E}^* : \mathbb{R}^{n_{\underline{x}}} &\to \mathbb{N} \\ \underline{x} &\mapsto \max(\mathbf{S}(\underline{x})) \ . \end{aligned}$$

See Fig. 3 for a visualization.

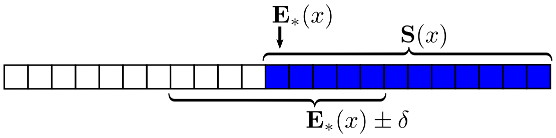


Figure 3: Sketch of a silhouette $\mathbf{S}(\underline{x})$ that is generated by an object $\underline{x}$. The left edge pixel $\mathbf{E}_*(\underline{x})$ is marked, as well as an area $\mathbf{E}_*(\underline{x}) \pm \delta$ around this edge.

The segmentation is assumed to produce no clutter measurements (*false positive*) outside of a sensor- and segmentation-specific distance δ around the object [17], i.e., for $\hat{y}_{\neg\mathbf{S}} \notin \mathbf{E}_*(\underline{x}) \pm \delta$ and $\hat{y}_{\neg\mathbf{S}} \notin \mathbf{E}^*(\underline{x}) \pm \delta$, $p(\text{TN}) = 1$ holds. This can be achieved by employing prior knowledge by means of, e.g., gating. In addition, we assume that the object is sufficiently large so that $\mathbf{E}_*(\underline{x}) \pm \delta \cap \mathbf{E}^*(\underline{x}) \pm \delta = \emptyset$ holds. The edge pixels of $\hat{\mathcal{Y}}_{\mathbf{S}}$ are determined according to

$$\hat{y}_{\mathbf{E}_*} = \min(\hat{\mathcal{Y}}_{\mathbf{S}}), \ \ \hat{y}_{\mathbf{E}^*} = \max(\hat{\mathcal{Y}}_{\mathbf{S}}) \ .$$

Note that these edges only can be evaluated when the object is visible to the camera. An example is shown in Fig. 4.

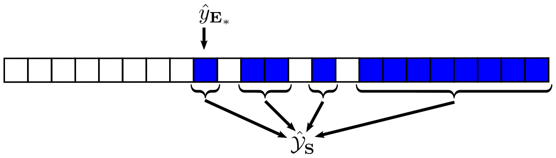


Figure 4: Measurement $\hat{y}_{\mathbf{E}_*}$ of the left edge, derived as the minimum of all silhouette pixels $\hat{\mathcal{Y}}_{\mathbf{S}}$.

Mathematically, the key idea is to translate the problem of estimating (2), i.e., $p(\underline{x}|\hat{\mathcal{Y}}_{\mathbf{S}}, \hat{\mathcal{Y}}_{\neg\mathbf{S}})$, to estimating $p(\underline{x}|\hat{y}_{\mathbf{E}_*}, \hat{y}_{\mathbf{E}^*})$ by solving

$$p(\underline{x}|\hat{y}_{\mathbf{E}_*}, \hat{y}_{\mathbf{E}^*}) \propto p(\hat{y}_{\mathbf{E}_*}, \hat{y}_{\mathbf{E}^*}|\underline{x}) \cdot p(\underline{x}) \ . \qquad (6)$$

For this purpose, we have to derive the likelihood $p(\hat{y}_{\mathbf{E}_*}, \hat{y}_{\mathbf{E}^*}|\underline{x})$ starting from $p(\hat{\mathcal{Y}}_{\mathbf{S}}, \hat{\mathcal{Y}}_{\neg\mathbf{S}}|\underline{x})$. Due to the fact that all pixels are segmented independently, we can write

$$p(\hat{y}_{\mathbf{E}_*}, \hat{y}_{\mathbf{E}^*}|\underline{x}) = p(\hat{y}_{\mathbf{E}_*}|\underline{x}) \cdot p(\hat{y}_{\mathbf{E}^*}|\underline{x}) \ .$$

These likelihoods can be derived by marginalizing

$$p(\hat{y}_{\mathbf{E}_*}|\underline{x}) = \sum_{\hat{\mathcal{Y}}_{\mathbf{S}}, \hat{\mathcal{Y}}_{\neg\mathbf{S}}} p(\hat{\mathcal{Y}}_{\mathbf{S}}, \hat{\mathcal{Y}}_{\neg\mathbf{S}}, \hat{y}_{\mathbf{E}_*}|\underline{x}) \ , \tag{7}$$

and

$$p(\hat{y}_{\mathbf{E}^*}|\underline{x}) = \sum_{\hat{\mathcal{Y}}_{\mathbf{S}}, \hat{\mathcal{Y}}_{\neg\mathbf{S}}} p(\hat{\mathcal{Y}}_{\mathbf{S}}, \hat{\mathcal{Y}}_{\neg\mathbf{S}}, \hat{y}_{\mathbf{E}^*}|\underline{x}) \ .$$

An analytic derivation for arbitrary δ is given in the appendix. Essentially, the resulting likelihood gives for each pixel in the boundary around the predicted silhouette edge the corresponding probability to be the measured edge. Thus, the likelihood function can be seen as a Dirac-mixture density in the image domain around the edge $\mathbf{E}_*(\underline{x})$. A Gaussian density

$$p(\hat{y}_{\mathbf{E}_*}|\underline{x}) = \mathcal{N}(\mathbf{E}_*(\underline{x}) - \hat{y}_{\mathbf{E}_*}; 0, \mathrm{Var}_\delta) \tag{8}$$

is used as an approximation and can be obtained from the Dirac-mixture by means of moment matching. Due to symmetry, this likelihood can easily be adapted to the right edge, i.e.,

$$p(\hat{y}_{\mathbf{E}^*}|\underline{x}) = \mathcal{N}(\mathbf{E}^*(\underline{x}) - \hat{y}_{\mathbf{E}^*}; 0, \mathrm{Var}_\delta) \ . \tag{9}$$

Fig. 5 gives a visualization. Deriving a Bayes update for (6) can be performed by a sample-based estimator, e.g., a UKF.

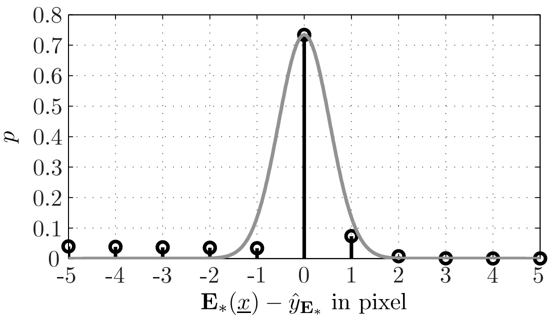


Figure 5: Gaussian approximation (*gray*) of the analytic likelihood (*black*) for a bounded error of $\delta = 5$ pixel. This analytic likelihood can only be evaluated at discrete pixels.

Remark 1 (Interpretation)

From a geometric point of view, the derived likelihoods (8), (9) *force the edge of the estimated object silhouette to coincide with the measured edges. This is equivalent to constrain the estimated object parameters according to the visual hull. The derived spatial uncertainty in the image domain directly affects the hull. In Fig. 6, an example of a visual hull, generated by two cameras is shown.*

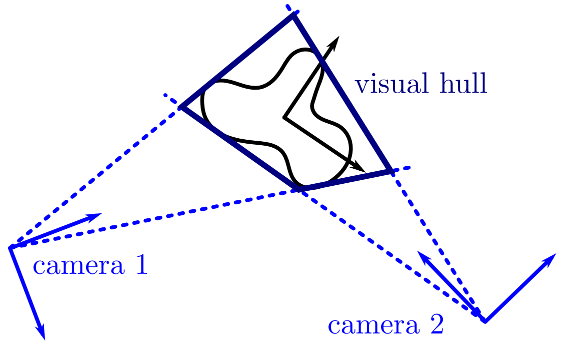


Figure 6: Geometric interpretation of the proposed approach. The object contour is forced to touch its projected silhouette edges, i.e., the visual hull.

V. Example: Ellipse Estimation

In order to demonstrate the proposed approach, we consider the specific problem of estimating the shape and pose parameters of an ellipse. For this purpose, we first define a proper state representation and a time update. Subsequently we show, how to apply the presented concepts to perform the measurement update with silhouette and point cloud measurements.

A. State Representation

An ellipse in 2D space can be represented by five parameters, where the current pose is given by an angle ϕ_k and the translation of the ellipse center $\underline{t}_k$. The shape is determined by the length of the two axes a_k and b_k. According to Sec. II, this yields a system state with

$$\underline{x}_k^{\text{pose}} = \begin{bmatrix} \phi_k \\ \underline{t}_k \end{bmatrix} \ , \ \underline{x}_k^{\text{shape}} = \begin{bmatrix} a_k \\ b_k \end{bmatrix} \ .$$

The uncertain knowledge of $\underline{x}_k$ is modeled by a Gaussian random vector $\underline{\boldsymbol{x}}_k \sim \mathcal{N}(\hat{\underline{x}}_k, \mathbf{C}_{\underline{x}_k})$.

B. Time Update

We make no assumptions on the ellipse behavior, and thus use the random walk model

$$\underline{x}_{k+1} = \underline{x}_k + \underline{\boldsymbol{v}} \ ,$$

with zero-mean Gaussian system noise $\underline{\boldsymbol{v}} \sim \mathcal{N}(\underline{0}, \mathbf{C}_{\underline{v}})$.

C. Measurement Update for Silhouettes

In the sample-based estimator, evaluating the likelihoods (8), (9) requires the explicit calculation of the edge functions $\mathbf{E}_*(\underline{x})$, $\mathbf{E}^*(\underline{x})$ for given ellipse instances $\underline{x}$ and cameras. The ellipse has to be projected onto the camera and the minimum and maximum pixel has to be found. This can be done by approximating the ellipse as a polygon $\mathrm{Poly}(\underline{x}) = \{\underline{p}_i | i = 1, \ldots, n_{\underline{p}}\}$. Based on this polygon, the edges can be easily be computed

$$\mathbf{E}_*(\underline{x}) = \min_i(\mathbf{K} \cdot (\mathbf{R}^{\mathrm{T}} \cdot \underline{p}_i - \mathbf{R}^{\mathrm{T}} \cdot \underline{t})) \ ,$$

and

$$\mathbf{E}^*(\underline{x}) = \max_i(\mathbf{K} \cdot (\mathbf{R}^{\mathrm{T}} \cdot \underline{p}_i - \mathbf{R}^{\mathrm{T}} \cdot \underline{t})) \ ,$$

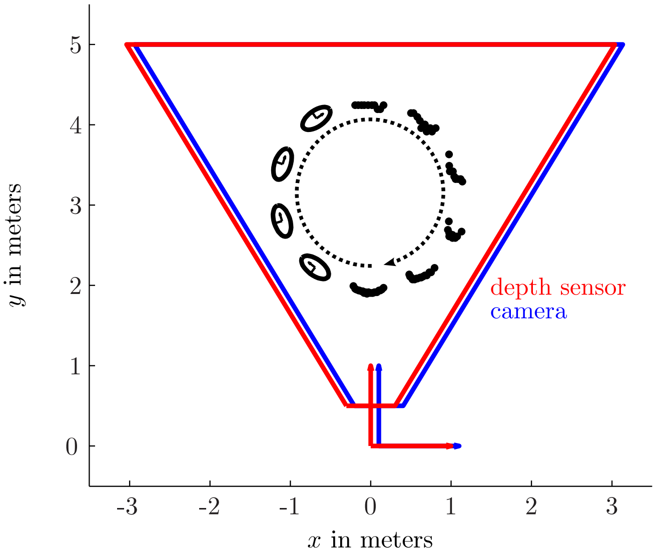


Figure 7: **Experimental setup**: An ellipse (*black*) moves clockwise along a circular path and is observed by a depth sensor (*red*) and a camera (*blue*). Note the viewing frustums are depicted in the respective colors. For some selected locations, the ellipse and point cloud measurements are drawn.

respectively. The variance term Var_δ can be computed, using the formulas in the appendix.

D. *Measurement Update for Point Clouds*

According to Sec. III-A, an SDF has to be found for relating point measurements to a given ellipse. This SDF can be calculated analytically by means of the *Mahalanobis distance* [11]. However, in simulations we observed a better convergence by using the *Euclidian distance*, even though it can only be computed approximately. Approximations could be done either iteratively or by discretization of the ellipse [18]. In order to determine a proper covariance matrix $\mathbf{C}_{\underline{z}_{src}}$ for measured points, a sensor model similar to [1] can be employed.

VI. Evaluation

In this section, we compare the proposed approach to the approach [11] that only uses point cloud information. The experimental setup consists of a depth sensor with an aligned camera that observes a moving ellipse within its fields of view. The ellipse starts at $[0\,\mathrm{m}, 2\,\mathrm{m}]$ and moves clock-wise 360° along a circular path. The axes of the ellipse are $[0.2\,\mathrm{m}, 0.1\,\mathrm{m}]$. An illustration of the setup is shown in Fig. 7. Simulated point cloud measurements are affected by stochastic noise and quantization errors according to a realistic sensor model [1]. Especially, this model reflects how measurement quality decreases with distance (see Fig. 7). For the edge measurements, a camera with a resolution of 640 pixels and 60° field of view was simulated. The resulting measurements were distorted with an additive, zero-mean Gaussian noise with a variance of 2 pixel^2. This corresponds to a $\delta = 5$ pixel and $p(\mathrm{TP}) = 0.95$, $p(\mathrm{TN}) = 0.95$.

Error	Point cloud only approach	Proposed approach
Position Avg.	6.28 cm	2.9 cm
Position Std.	4.68 cm	2.06 cm
Orientation Avg.	9.90 °	5.0 °
Orientation Std.	5.90 °	3.42 °
Shape Avg.	5.33 cm	2.25 cm
Shape Std.	3.27 cm	1.48 cm

Table I: Average estimation error over all 100 runs and all angles.

A. *Results*

The following results are based on 100 Monte-Carlo runs of the described simulation. Fig. 8 visualizes the median result of one lap for both approaches. An interesting result is that the point-cloud-based approach is not able to follow the change in orientation (Fig. 8a). Instead, it flips the axes of the ellipse, assuming a shape change (Fig. 8c). This behavior is also reflected in the error diagram in Fig. 9, where the average error of each parameter is drawn over a full lap. Looking at the errors, two main improvements by incorporating silhouette measurements can be highlighted:

1) Better initial convergence, and
2) better estimation in the presence of high noise.

Overall, the average estimation error in this scenario can be reduced by a factor of two (see Table I for details).

VII. Conclusion

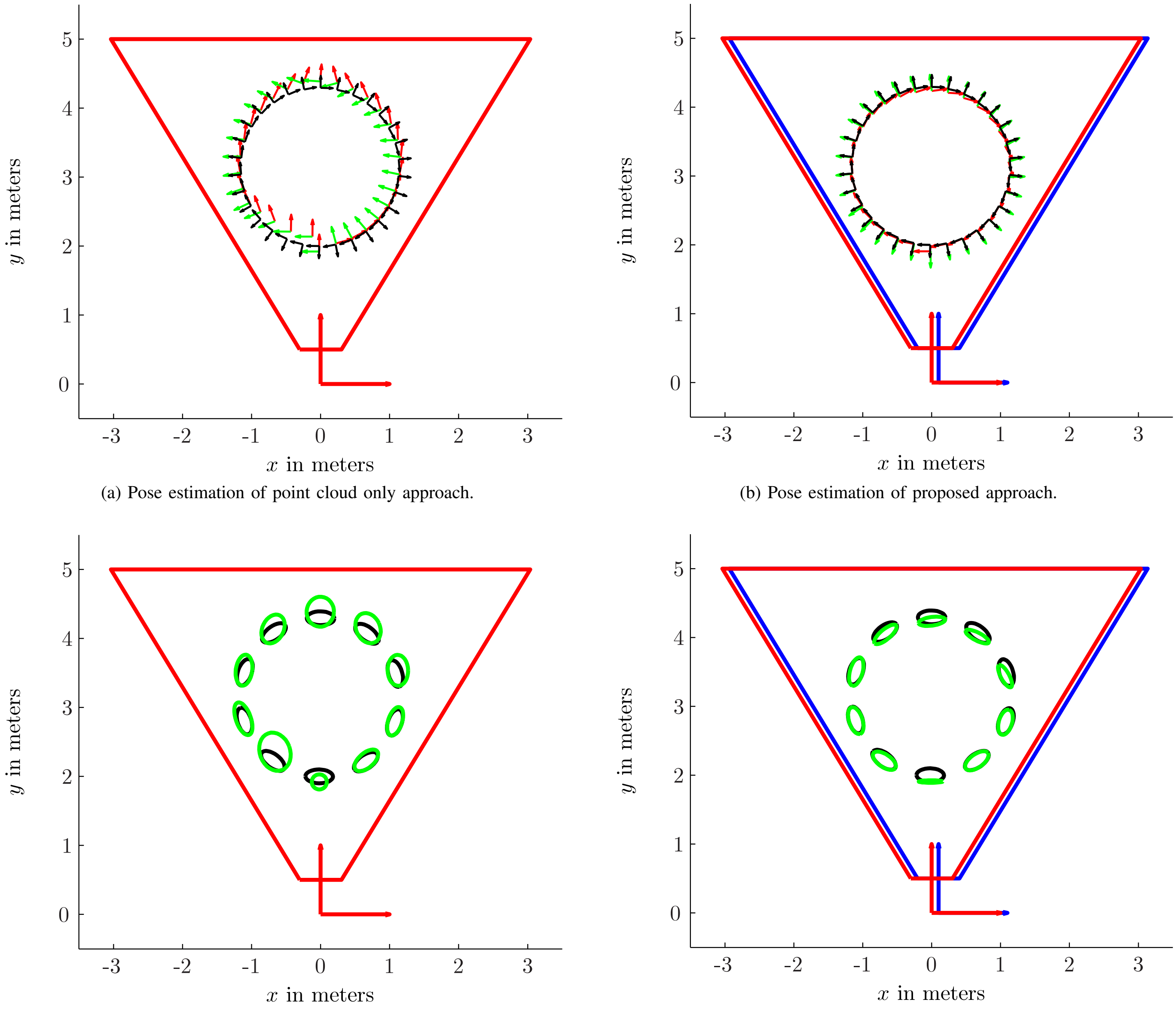
In this work, we presented a novel approach[2] for incorporating silhouette information in Bayesian shape and pose estimation of extended objects. Starting from a general expression of the silhouette likelihood function that assumes segmentation uncertainty for each pixel, we derived an approximative likelihood function that models spatial uncertainty for the silhouette edge in the image domain. In doing so, only the edge pixels of a silhouette have to be considered for the update instead of the whole image. This reduces the complexity as well as the numerical instabilities of the general update.

Our approach comes with two major highlights. First, the uncertainty for silhouette measurements is explicitly modeled. This allows for consideration of sensor and segmentation properties. Second, this explicit uncertainty model allows fusing other measurement modalities, i.e., point clouds, in a mathematically sound way, as both types of information are individually weighted by its corresponding uncertainty. In simulations, we demonstrated the improvements of incorporating silhouettes in a point-cloud-based Bayesian pose and shape estimator. In summary, the proposed approach yields better initial convergence and an overall reduced estimation error, especially in the presence of high noise.

A. *Future Work*

The next step would be extending the algorithm for estimating 3D objects. For this purpose, a suitable representation for

[2] Code is available online: http://www.cloudrunner.eu/algorithm/120/silhouette-measurements-for-bayesian-object-tracking-in-noisy-point-clouds/



(a) Pose estimation of point cloud only approach.

(b) Pose estimation of proposed approach.

(c) Ellipse estimation of point cloud only approach.

(d) Ellipse estimation of proposed approach.

Figure 8: Illustration of the simulated experiment. Median of the estimated ellipse parameters over 100 *Monte-Carlo* runs is shown for the point-cloud only (a,c) and the proposed (b,d) approach. For comparison, the ground truth is also drawn in *black*. Note that the point cloud only approach fails in capturing the orientation (a) by adjusting the shape instead (b).

2D silhouette edges has to be chosen. A second interesting aspect would be to apply silhouette measurements directly to the depth sensor. As many depth sensors measure organized point clouds, i.e., depth images, information about object silhouettes is available. However, this would require an analysis of statistical correlations between point cloud measurements and silhouettes from the same sensor.

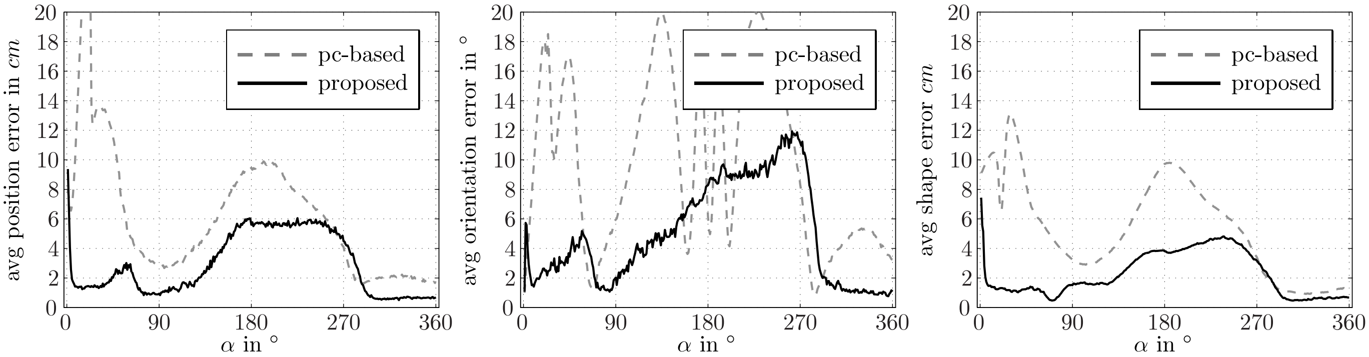
Appendix

This section considers the analytic derivation of the edge likelihood $p(\hat{y}_{\mathbf{E}_*}|\underline{x})$ for a given segmentation quality, i.e., bounded error δ and probabilities $p(\text{TP})$, $p(\text{TN})$. According to (7), the variance Var_δ has to be determined. Due to symmetry, the calculation for the right edge $p(\hat{y}_{\mathbf{E}^*}|\underline{x})$ is analogous. The abbreviation e_* denotes the signed distance between the calculated edge $\mathbf{E}_*(\underline{x})$, and the measured edge $\hat{y}_{\mathbf{E}_*}$ with

$$e_* = \mathbf{E}_*(\underline{x}) - \hat{y}_{\mathbf{E}_*} \ .$$

The bounded segmentation error ensures $e_* \leq \delta$. Three cases can be distinguished: if $e_* = 0$, the measured and calculated edge match

$$\begin{aligned} p(e_* = 0) &= p(\text{TN})^\delta \cdot p(\text{TP}) \cdot \underbrace{(p(\text{FN}) + p(\text{TP}))^\delta}_{=1} \\ &= p(\text{TN})^\delta \cdot p(\text{TP}) \ , \end{aligned}$$



(a) Position error.

(b) Orientation error.

(c) Shape error.

Figure 9: Average error over 100 runs of a full clock-wise circle, by starting at the closest point to the sensors. The proposed approach features a fast initial convergence in all parameters and an overall improved estimation. The point-cloud-based approach fails in estimating the orientaion between 90° and 270°.

i.e., all δ pixels left of $\mathbf{E}_*(\underline{x})$ have been measured correctly as *true negative*. If $e_* < 0$ the measured edge is a *false positive* as it lies left of the calculated

$$\begin{aligned} p(e_* < 0) &= p(\mathrm{TN})^{\delta + e_*} \cdot p(\mathrm{FP}) \\ &\quad \cdot \underbrace{(p(\mathrm{FN}) + p(\mathrm{TP}))^{\delta+1}}_{=1} \cdot \underbrace{(p(\mathrm{FP}) + p(\mathrm{TN}))^{|e_*|-1}}_{=1} \\ &= p(\mathrm{TN})^{\delta + e_*} \cdot p(\mathrm{FP}) \ . \end{aligned}$$

Analogously, if $e_* > 0$, the measured edge lies right of the calculated

$$\begin{aligned} p(e_* > 0) &= p(\mathrm{TN})^{\delta} \cdot p(\mathrm{FN})^{e_*} \cdot p(\mathrm{TP}) \\ &\quad \cdot \underbrace{(p(\mathrm{FN}) + p(\mathrm{TP}))^{\delta - e_*}}_{=1} \\ &= p(\mathrm{TN})^{\delta} \cdot p(\mathrm{FN})^{e_*} \cdot p(\mathrm{TP}) \ . \end{aligned}$$

In Fig. 5, these probabilities are evaluated for $\delta = 5$ and drawn in *black*. Then, the variance Var_δ of the approximated Gaussian (*gray*) can be computed by means of moment matching.

REFERENCES

[1] F. Faion, S. Friedberger, A. Zea, and U. D. Hanebeck, “Intelligent Sensor-Scheduling for Multi-Kinect-Tracking,” in *Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2012)*, Vilamoura, Algarve, Portugal, Oct. 2012, pp. 3993–3999.

[2] K. Berger, K. Ruhl, and C. Brümmer, “Markerless motion capture using multiple color-depth sensors,” in *Vision, Modeling, and Visualization Workshop (VMV)*, Berlin, Germany, 2011, pp. 317–324.

[3] A. Laurentini, “The visual hull concept for silhouette-based image understanding,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 2, pp. 150–162, 1994.

[4] K. Grauman, G. Shakhnarovich, and T. Darrell, “A bayesian approach to image-based visual hull reconstruction,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Madison, WI, 2003, pp. 1–8.

[5] A. Murat Tekalp, N. Navab, and V. Ramesh, “Interactive optimization of 3D shape and 2D correspondence using multiple geometric constraints via POCS,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 4, pp. 562–569, Apr. 2002.

[6] S. Soatto, R. Frezza, and P. Perona, “Motion estimation via dynamic vision,” *IEEE Transactions on Automatic Control*, vol. 41, no. 3, pp. 393–413, Mar. 1996.

[7] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, “Real-time human pose recognition in parts from single depth images,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, Colorado Springs, USA, 2011.

[8] A. Bleiweiss and M. Werman, “Fusing time-of-flight depth and color for real-time segmentation and tracking,” *Dynamic 3D Imaging*, pp. 58–69, 2009.

[9] Y. M. Kim, C. Theobalt, J. Diebel, J. Kosecka, B. Miscusik, and S. Thrun, “Multi-view image and ToF sensor fusion for dense 3D reconstruction,” in *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*, Kyoto, Japan, Sep. 2009, pp. 1542–1549.

[10] D. Grest, V. Krüger, and R. Koch, “Single view motion tracking by depth and silhouette information,” *Image Analysis*, 2007.

[11] M. Baum and U. D. Hanebeck, “Fitting conics to noisy data using stochastic linearization,” in *Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2011)*, San Francisco, California, USA, Sep. 2011.

[12] D. Angelova and L. Mihaylova, “Extended object tracking using Monte Carlo methods,” *IEEE Transactions on Signal Processing*, vol. 56, no. 2, pp. 825–832, 2008.

[13] A. Poonawala, P. Milanfar, and R. J. Gardner, “Shape estimation from support and diameter functions,” *Journal of Mathematical Imaging and Vision*, vol. 24, no. 2, pp. 229–244, Jan. 2006.

[14] M. Werman and D. Keren, “A Bayesian method for fitting parametric and nonparametric models to noisy data,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 5, pp. 528–534, 2001.

[15] T. Lefebvre, H. Bruyninckx, and J. D. Schutter, *Nonlinear Kalman filtering for force-controlled robot tasks*. Leuven (Heverlee), Belgium: Springer-Verlag, 2005.

[16] S. Julier and J. Uhlmann, “Unscented filtering and nonlinear estimation,” *Proceedings of the IEEE*, vol. 92, no. 3, 2004.

[17] S. C. Zhu and A. Yuille, “Region competition: Unifying snakes, region growing, and Bayes/MDL for multiband image segmentation,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 18, no. 9, pp. 884–900, 1996.

[18] P. Schneider and D. Eberly, *Geometric tools for computer graphics*. San Francisco, California, USA: Morgan Kaufmann Publishers, 2003.