2013 IEEE International Conference on Robotics and Automation (ICRA)
Karlsruhe, Germany, May 6-10, 2013

GMM-based 3D Object Representation and Robust Tracking in Unconstructed Dynamic Environments

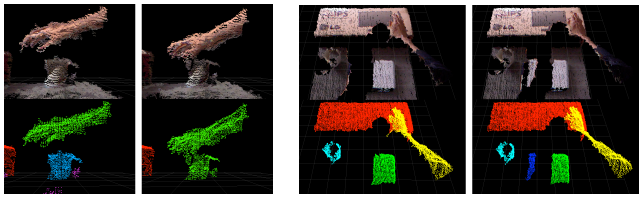
Seongyong Koo, Dongheui Lee and Dong-Soo Kwon

Abstract— **Operating in unstructured dynamic human environments, it is desirable for a robot to identify dynamic objects and robustly track them without prior knowledge. This paper proposes a novel model-free approach for probabilistic representation and tracking of moving objects from 3D point set data based on Gaussian Mixture Model (GMM). GMM is inherently flexible such that represents any shape of objects as 3D probability distribution of the true positions. In order to achieve the robustness of the model, the proposed tracking method consists of GMM-based 3D registration, Gaussian Sum Filtering, and GMM simplification processes. The tracking performance of the proposed method was evaluated in the moving two human hands with one object, and it performed over 87% tracking accuracy together with processing 5 frames per second.**

I. Introduction

With the advent of 3D RGB-D cameras and improvements to 3D data processing technologies [25], service robots operating in unstructured human environments can capture the environments as 3D point set data and its interpretation technology have been explored in recent years. In particular, identifying multiple objects from the data is an important and challenging task for understanding the unstructured environment such as reconstructing 3D indoor environments [19] and representing the semantic information of a human environment [5], and operating a robot manipulator for manipulating multiple objects [6], [16].

In most cases in previous works, a robot has pre-knowledge or uses a learned model of objects to recognize and track them. In reality, however, the modeling and learning of all objects in advance is not always possible, and unforeseen objects might be present while performing a tracking task. One approach to solve this problem is to construct general models for arbitrary objects from the appearances of shape and/or color information. [21], [26] represented each object based on shape primitive models such as a sphere, a plane, a cylinder, and a cone from a point data set, and [24] obtained a more precise object model by combining the primitives and triangular meshes for the remaining point parts outside of the model. In more recent robotics research, [16] suggested a graphical model to represent the appropriate features of multiple objects, such as supporting contacts, caging, and object geometry for placing the objects into the another space. Another approach is to construct a specific model for the unknown object by robot itself in on-line manner. [18] modeled a new object in a robot hand as a set of surfels that is robust to noise and occlusions by using both the shape and appearance information, and [13] proposed a method to construct complete 3D models of articulated objects by interacting with objets.



(a) Two contacted objects (b) An occluded object by a human hand

Fig. 1. Two cases of falsely detected point data set in the dynamic situations. In each case, point set data captured by two Kinect cameras at time $t-1$ and t are displayed in the left and right figures, respectively. The upper figures show the RGB data of each point, and the bottom figures show the segmented points of each object by using euclidian clustering methods in [25].

When it comes to interacting with human in dynamic environments such as learning actions from human demonstration and cooperating with human [1], [8], [20], [22], there are many challenging issues of tracking moving objects in the human environments. First, many 3D entities, including human body, can be considered as articulated objects that have components connected by joints and move with respect to each other. In this case, the tracking problem involves estimating position and orientation of the object and those of all the components constructing the object [23]. In addition, when moving objects become adjacent (contacted) to another or when some parts of the object are undetected due to occlusion or detection error, the observed point set data of the object becomes distorted as shown in Fig. 1.

This paper aims to achieve both flexibility and robustness for modeling and tracking multiple objects without prior-knowledge of them. At first, we propose a novel object representation approach from the point set data based on Gaussian Mixture Model (GMM). The basic idea is that the estimated position of an object is probabilistically distributed around the true position and the distribution function represents the shape of the object. GMM is not only good flexible stochastic model to represent any 3D shapes of an object, but also useful for manipulating the object models analytically owing to its functional expression such as distance measure and probabilistic multiplications. Second, in order to compensate the weak robustness of the GMM representation due to

S. Koo is with Mechanical Engineering and HRI Research Center, KAIST, Deajeon, Republic of Korea koosy@robot.kaist.ac.kr

D. Lee is with Faculty of Department of Electrical Engineering and Information Technology, Technical University of Munich, 80290 Munich, Germany dhlee@tum.de

D. Kwon is with Faculty of Mechanical Engineering and HRI Research Center, KAIST, Deajeon, Republic of Korea kwonds@kaist.ac.kr

the model's adaptability to the falsely detected point set as depicted in Fig. 1, we propose GMM-based robust tracking method by using 3D registration method [14] and Gaussian Sum Filtering [17]. The proposed tracking method represents each object as a function of probability distribution and performs tracking not only for the positions of the objects but also for the shapes of them, which results in identifying each detected point to be involved in which object. This point-wise object shape tracking helps to segment objects correctly even when they are in contact as shown in Fig. 1(a). The robustness of the proposed tracking method have been evaluated by calculating the point tracking accuracy in the dynamic situations of moving two hands with an object. Finally, in order to investigate the relation of the exactness and the computational efficiency according to the expressiveness of the model, the tracking accuracy and the computation time were measured according to the down-sampling distance and the simplification ratio. As a result, the optimal values of the two control parameters have been found experimentally to achieve 87% accuracy and 5 frames per second of computation time.

The remainder of this paper is structured as follows. GMM-based object representation method is described in chapter II, and the overview of the proposed tracking method and the detailed explanations of the processes are delineated in chapter III and chapter IV, respectively. In chapter V, the results of the proposed methods are discussed with several experiments involving various movements of human hands with an object. Finally, a conclusion is given in chapter VI.

II. GMM-BASED OBJECT REPRESENTATION

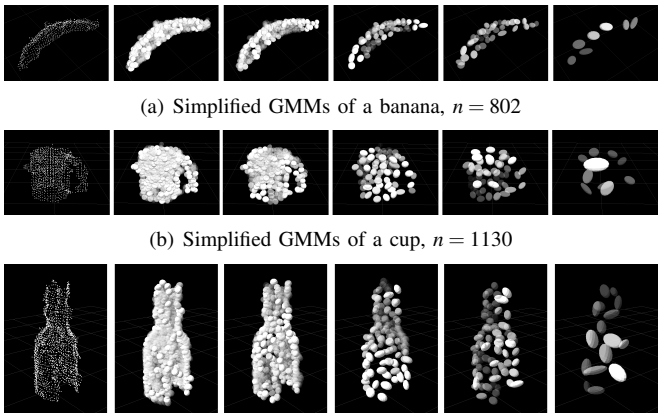
With a 3D RGB-D camera, an object can be detected as a point set data $O = \{p_1, ..., p_n\}$, in each of which contains the 3D position and RGB color information of a point. In order to describe an object only with shape information in this research, the 3D position data of a point $p_i \in \mathbb{R}^3$ is used for constructing probability distribution function of an object. The most simple case is to design one multivariate Gaussian distribution consisting of a mean ($\mu \in \mathbb{R}^3$) and covariance matrix ($\Sigma \in \mathbb{R}^{3\times3}$). The probability density of a point ($\mathbf{x} \in \mathbb{R}^3$) belonging to the object can be represented as (1).

$$\phi(\mathbf{x}|\mu,\Sigma) = \frac{1}{\sqrt{(2\pi)^3\,|\Sigma|}} \exp\left(-\frac{1}{2}(\mathbf{x}-\mu)^T\Sigma^{-1}(\mathbf{x}-\mu)\right) \tag{1}$$

In particular, the Gaussian Mixture Model, which is defined as (2), can represent any arbitrary shape of functions when the number of Gaussians, k, goes to infinity.

$$p(\mathbf{x}) = \sum_{i=1}^{k} w_i \phi(\mathbf{x}|\mu_i,\Sigma_i), \;\; \sum_{i=1}^{k} w_i = 1 \tag{2}$$

This probability distribution function of GMM is defined by a set of parameters $\mathscr{G} = \{k, \mathbf{w}, \boldsymbol{\mu}, \boldsymbol{\Sigma}\}$, where each of $\mathbf{w}$, $\boldsymbol{\mu}$, and $\boldsymbol{\Sigma}$ has k components that represent k Gaussians. Learning the parameters of $\mathscr{G}$ from the given set of points has been investigated in many ways. One of the typical methods involves using the Expectation-Maximization (EM) algorithm [7], [4] given the number of Gaussians, k. In recent years, hierarchical GMM has been proposed to determine the number of Gaussians efficiently through a hierarchical clustering method [9]. On the other hand, if the assumption that the point set of an object O is obtained using the same sampling distance, the corresponding GMM can be represented by evenly weighted n Gaussians centered at each point with the same spherical covariance matrix [14]. Although a parameter learning process is not needed in such a case, the model includes such a number of Gaussians as much as points that related algorithms are inefficient due to the expensive computational time.



(a) Simplified GMMs of a banana, $n = 802$

(b) Simplified GMMs of a cup, $n = 1130$

(c) Simplified GMMs of a bottle, $n = 1919$

Fig. 2. The first column denotes the n source points. The simplified GMMs are displayed as a set of 3D ellipsoids with reduction ratios of 0.5, 0.3, 0.1, 0.05 and 0.01, respectively, from the second to the sixth column.

For this reason, we construct an initial GMM directly from the down-sampled point set with a constant sampling distance using a VoxelGrid filter in [25], and simplify the GMM with the given number of Gaussians. There have been proposed several GMM approximation methods. Hierarchical clustering (HC) method constructed local point groups to approximate the original GMM by minimizing the KL-divergence using EM-algorithm in [11], [10]. Later, functional approximation (FA) method [27] which used the measured L2 distance to minimize the upper bound of the approximation error, showed better performance than the HC method in terms of model approximation, but the computation time is nearly three times greater than that of HC. In this research, HC method [11] is used with L2-distance measure because of the fast computational time compared with other methods, and its reasonable approximation performance with L2-distance.

The simplified GMM consists of k 3D Gaussians with different weight values. The GMM examples of three objects that have different number of points are illustrated in Fig. 2 with different reduction ratio which determines the number of Gaussians of the simplified GMM from the number of original points. The 3D ellipsoid shows each Gaussian and its transparency represent the corresponding weight value.

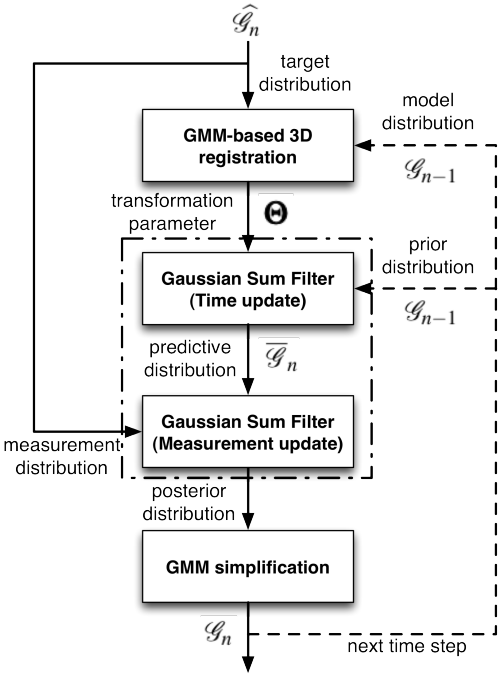


Fig. 3. Overview of the proposed tracking processes

III. Framework of the Proposed Tracking Method

The aim of this research is to estimate the probability distribution of the current position of object $\mathbf{x}_n$ (filtering distribution) from observations of the point set $O_{0:n}$. As in the Hidden Markov Model (HMM) formulation, the true position of an object $\mathbf{x_n}$ can not be detected directly, but the detected point set O_n can be thought as the probability distribution of the position of the observed object $\mathbf{y_n}$ originated from the hidden $\mathbf{x_n}$. With the assumption that the object position probability is distributed on the arbitrary shape of the object, the probability distribution of $\mathbf{x_n}$ given the sequence of observed object position $\mathbf{y}_{0:n}$ can be expressed as the form of GMM with the parameters of $\mathscr{G}_n = \{k_n, \mathbf{w}_n, \boldsymbol{\mu}_n, \boldsymbol{\Sigma}_n\}$.

$$p(\mathbf{x}_n|\mathbf{y}_{0:n}) \triangleq \sum_{i=1}^{k_n} w_{ni}\phi(\mathbf{x}_n|\mu_{ni}, \Sigma_{ni}). \tag{3}$$

The distribution of the measured point set (measurement distribution) is also defined as the form of GMM with the parameters of $\widehat{\mathscr{G}}_n = \{\widehat{k}_n, \widehat{\mathbf{w}}_n, \widehat{\boldsymbol{\mu}}_n, \widehat{\boldsymbol{\Sigma}}_n\}$.

$$p(\mathbf{y}_n|\mathbf{x}_n) \triangleq \sum_{i=1}^{\widehat{k}_n} \widehat{w}_{ni}\phi(\mathbf{x}_n|\widehat{\mu}_{ni}, \widehat{\Sigma}_{ni}) \tag{4}$$

Fig. 3 demonstrates the whole tracking process of estimating the filtering distribution, $\mathscr{G}_n$, from the measurement distribution, $\widehat{\mathscr{G}}_n$, at every time n in iterative way.

A dynamic model of a moving object can be expressed as the dynamic state-space (DSS) model,

$$\begin{aligned} \mathbf{x}_n &= \mathbf{f}(\mathbf{x}_{n-1}) + \mathbf{u}_{n-1} \\ \mathbf{y}_n &= \mathbf{h}(\mathbf{x}_n) + \mathbf{v}_n \end{aligned} \tag{5}$$

where $\mathbf{f}(\cdot)$ and $\mathbf{h}(\cdot)$ are possibly nonlinear functions, and $\mathbf{u}_{n-1}$ and $\mathbf{v}_n$ are independent and identically distributed random noise sequences. Once the dynamic model is obtained, the target distribution can be estimated by Gaussian Sum Filtering (GSF). In case of an unknown dynamic model, Gaussian Sum Particle Filtering (GSPF) can be applied [17], but, in this research, the unknown transition function $\mathbf{f}(\cdot)$ can be approximated as piece-wise linear between time n and $n-1$ with the assumption that the time step is small enough in the real-time tracking task. The piece-wise linear function $\tilde{\mathbf{f}}(\cdot)$ between time n and $n-1$ is estimated by GMM-based robust 3D registration method [14]. This method uses the target distribution, $\widehat{\mathscr{G}}_n$, and the model distribution, $\mathscr{G}_{n-1}$ which is the filtering distribution at the previous time step,

$$p(\mathbf{x}_{n-1}|\mathbf{y}_{0:n-1}) \triangleq \sum_{i=1}^{k_{n-1}} w_{(n-1)i}\phi(\mathbf{x}_{n-1}|\mu_{(n-1)i}, \Sigma_{(n-1)i}) \tag{6}$$

The estimated transformation parameter $\boldsymbol{\Theta}$ is used for the time update step in GSF, which produces the predictive distribution of the current position of the object from the prior distribution (6). It is also expressed as the GMM form with the parameters of $\overline{\mathscr{G}}_n = \{\overline{k}_n, \overline{\mathbf{w}}_n, \overline{\boldsymbol{\mu}}_n, \overline{\boldsymbol{\Sigma}}_n\}$.

$$p(\mathbf{x}_n|\mathbf{y}_{0:n-1}) \triangleq \sum_{i=1}^{\overline{k}_n} \overline{w}_{ni}\phi(\mathbf{x}_n|\overline{\mu}_{ni}, \overline{\Sigma}_{ni}) \tag{7}$$

In the measurement update step in GSF, the target distribution can be obtained from the measurement distribution and the predictive distribution by Bayes' theorem and Markov property.

$$p(\mathbf{x}_n|\mathbf{y}_{0:n}) = C_n p(\mathbf{y}_n|\mathbf{x}_n)p(\mathbf{x}_n|\mathbf{y}_{0:n-1}) \tag{8}$$

The next two chapters explain the details of the each step of GMM-based registration and Gaussian Sum Filtering based on GMM representation.

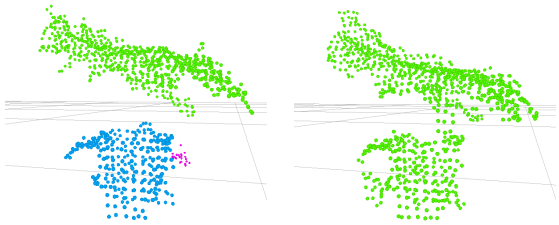
IV. GMM-based 3D Registration

The 3D registration is a process that finds the transformation parameter $\boldsymbol{\Theta}$ to minimize the distance or maximize the similarity between the point set of the transformed model, $T(O_m, \boldsymbol{\Theta})$ and the point set of the scene O_s. With the piece-wise linear assumption, the unknown parameter $\boldsymbol{\Theta}$ consists of rotation and translation matrices, $\mathbf{R}$ and $\mathbf{t}$, and each point at $n-1$ can be transformed as following.

$$\mathbf{x}_n = \tilde{\mathbf{f}}(\mathbf{x}_{n-1}) = \mathbf{R}\mathbf{x}_{\mathbf{n-1}} + \mathbf{t} \tag{9}$$

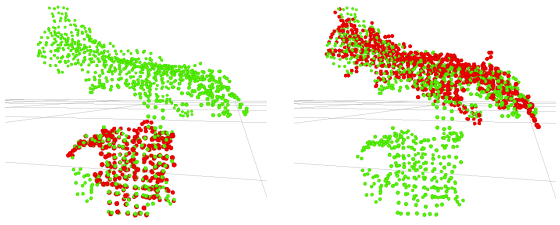
In dynamic situations, point set of an object at each time step is easily distorted with many outliers as in the cases of Fig. 1. Fig. 4(a) shows the case that the two objects (a human hand and a cup) are detected correctly at time $n-1$ but are merged at time n, as shown in Fig. 4(b). In order to track each object robustly at time n, the 3D registration problem in this case is to register the two model distributions in Fig. 4(a) to the integrated target distribution in Fig. 4(b). For each model, the target data has numerous outliers, which are points belonging to another object; therefore, a robust registration method is necessary in this case.

Many 3D point set registration methods are based on the iterative closest point (ICP) method [3], and they have been successfully implemented and applied in many applications. [15] showed that ICP-based registration methods have the same effect of minimizing the KL-divergence between two



(a) Two model data at time $t-1$ (b) A scene data at time t

Fig. 4. 3D point set registration task of the *mergence* case



(a) Registration result of a cup (b) Registration result of a hand

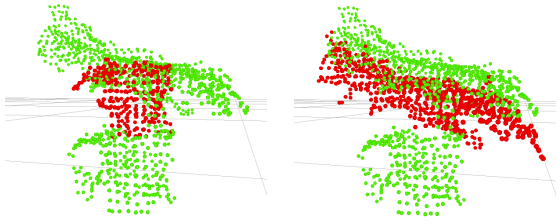
Fig. 6. 3D point set registration results using L2-distance of the two models

GMMs of $T(O_m,\mathbf{\Theta})$ and O_s. [14] proposed a GMM-based registration method using the L2 distance of GMMs as a cost function between the transformed model and the target data. The L2 estimator is more robust against outliers than the KL-divergence estimator and the maximum likelihood estimator (MLE). Another advantage of the L2 distance is its closed-form expression for GMMs. The L2 distance of two GMMs $(\mathcal{G}_a,\mathcal{G}_b)$ can be expressed as (10) with the property of a Gaussian function of (1), $\int \phi(\mathbf{x}|\mu_1,\Sigma_1)\phi(\mathbf{x}|\mu_2,\Sigma_2)d\mathbf{x} = \phi(\mathbf{0}|\mu_1-\mu_2,\Sigma_1+\Sigma_2)$.

$$
\begin{aligned}
d_{L2} &= \int \left(\sum_{i=1}^{k_a} w_{ai}\phi(\mathbf{x}|\mu_{ai},\Sigma_{ai}) - \sum_{i=1}^{k_b} w_{bi}\phi(\mathbf{x}|\mu_{bi},\Sigma_{bi})\right)^2 d\mathbf{x} \\
&= \sum_{i=1}^{k_a}\sum_{j=1}^{k_a} w_{ai}w_{aj}\phi(0|\mu_{ai}-\mu_{aj},\Sigma_{ai}+\Sigma_{aj}) \\
&- 2\sum_{i=1}^{k_a}\sum_{j=1}^{k_b} w_{ai}w_{bj}\phi(0|\mu_{ai}-\mu_{bj},\Sigma_{ai}+\Sigma_{bj}) \\
&+ \sum_{i=1}^{k_b}\sum_{j=1}^{k_b} w_{bi}w_{bj}\phi(0|\mu_{bi}-\mu_{bj},\Sigma_{bi}+\Sigma_{bj})
\end{aligned}
\tag{10}
$$

The numerical calculation of (10) consists of three forms of discrete Gaussian transforms [12]. Apparently, however, the performance in terms of the computational time depends mainly on the number of Gaussians, k_a and k_b. Hence, a reduction of the size by GMM simplification is necessary for the implementation of the algorithm in real-time.

Fig. 5 shows the GMM-based registration results with the KL-divergence distance while Fig. 6 shows the results when the L2-distance is used with the same variance value of the Gaussians without the simplification process. Obviously, the



(a) Registration result of a cup (b) Registration result of a hand

Fig. 5. 3D point set registration results using KL-divergence of the two models

KL-divergence measure is more efficient to reflect the global effects of the points, as it tries to maximize the likelihood of the model matching to the scene and thus places the model in the center of the scene, while L2-distance reflects local effects more than a global influence and the registration results show that it is more robust against most of the outliers.

Another advantage of GMM-based 3D registration is its closed expression of the gradient of the cost function. In this research, we used rigid transformation, which is defined by the rotation matrix $\mathbf{R}$ and the translation vector $\mathbf{t}$. Let $\mathbf{P}$ denotes a $m\times 3$ matrix of the 3D point set P. The rigid transformed model at time t can then be expressed as follows:

$$
\mathbf{P}_m^t = T(\mathbf{P}_m^{t-1},\mathbf{\Theta}) = \mathbf{P}_m^{t-1}\mathbf{R}^T + \mathbf{t} \tag{11}
$$

The gradient of the cost function (10) can be derived by the chain rule $\frac{\partial F}{\partial \mathbf{\Theta}} = \frac{\partial F}{\partial \mathbf{P}_m^t}\frac{\partial \mathbf{P}_m^t}{\partial \mathbf{\Theta}}$. The first derivative $\frac{\partial F}{\partial \mathbf{P}_m^t}$ is the partial derivative of the cost function with respect to each point. The derivatives of the first and the third terms of (10) are zero due to the rigid transformation. The partial derivative of the cost function at each point is determined as follows:

$$
\frac{\partial F}{\partial \mu_{i,d}^m} = -2w_i^m\sum_{j=1}^{n} w_j^s \frac{\partial}{\partial \mu_{i,d}^m}\phi(0|\mu_i^m-\mu_j^s,\Sigma_i^m+\Sigma_j^s). \tag{12}
$$

The second derivative can be simply obtained by the linear form of (11). The gradient of the cost function can be expressed as

$$
\begin{aligned}
\frac{\partial F}{\partial \mathbf{t}} &= \frac{\partial F}{\partial \mathbf{P}_m^t}^T \mathbf{1}_m \\
\frac{\partial F}{\partial r_i} &= \mathbf{1}_d^T\left(\left(\frac{\partial F}{\partial \mathbf{P}_m^t}^T \mathbf{P}_m^{t-1}\right)\otimes\left(\frac{\partial \mathbf{R}}{\partial r_i}\right)\right)\mathbf{1}_d,
\end{aligned}
\tag{13}
$$

where $\mathbf{1}_m$ is a m dimensional column vector of all ones, and $\otimes$ denotes the element-wise multiplication. The main part of the gradient calculation is the first partial derivative of the cost function at each point of the model (12). This is a similar form of the Gaussian transform between a point and a GMM; thus, it can also be obtained using the same process used to calculate the cost function (10).

In order to optimize the transformation parameter, any gradient descent optimization algorithm can be used with the help of the gradient of (13). In this research, we used the Limited-Memory Broyden Fletcher Goldfarb Shannon (L-BFGS) minimization algorithm, which is based on a quasi-Newton algorithm for large-scale numerical optimization problems [1]. Moreover, this minimizer allows one to set the constraints of the search space, which helps to find the local

[1]This is implemented in the vision-numerics library (vnl) in http://vxl.sourceforge.net/

minima around the initial point set because the global minima is occasionally not the correct position when the true object is merged with a relatively large object.

V. Gaussian Sum Filtering

Bayesian filtering is a probabilistic approach to estimate probability distribution of a hidden variable from the observation sequences in the iterative way based on the Markov property and Bayes' theory. In the cases of the predictive and filtering distribution can be approximated as Gaussian Mixtures, the filtering methods are called Gaussian Sum Filtering (GSF) in [2]. In particular, [17] proposed several Gaussian Sum Particle Filtering (GSPF) methods for the cases of nonlinear functions and non-Gaussian noises in the DSS model of (5). In this research, GSF method can be applied because of the assumption that the dynamic motion of an object is piece-wise linear function and it can be estimated in the 3D registration process. This chapter illustrates GSF method with the GMM approximation of the measurement distribution additional to the predictive and filtering distributions.

A. Time update

The time update step is to estimate the predictive distribution $\overline{\mathscr{G}}_n$ from the prior distribution $\mathscr{G}_{n-1}$, and the relationship can be described as follows.

$$\begin{aligned} & p(\mathbf{x}_n|\mathbf{y}_{0:n-1}) \\ &= \int p(\mathbf{x}_{n-1}|\mathbf{y}_{0:n-1})p(\mathbf{x}_n|\mathbf{x}_{n-1})d\mathbf{x}_{n-1} \\ &= \int \sum_{i=1}^{k_{n-1}} w_{(n-1)i}\phi(\mathbf{x}_{n-1}|\mu_{(n-1)i},\Sigma_{(n-1)i})p(\mathbf{x}_n|\mathbf{x}_{n-1})d\mathbf{x}_{n-1} \\ &\triangleq \sum_{i=1}^{\bar{k}_n} \overline{w}_{ni}\phi(\mathbf{x}_n|\overline{\mu}_{ni},\overline{\Sigma}_{ni}) \end{aligned} \tag{14}$$

Before observing a new measurement, the number of Gaussians at time n, $\bar{k}_n$ and each weight value, $\overline{w}_{ni}$ can be thought as the same number of k_{n-1} and $w_{(n-1)i}$, respectively. Then, each Gaussian in the predictive distribution can be approximated as,

$$\phi(\mathbf{x}_n|\overline{\mu}_{ni},\overline{\Sigma}_{ni}) \approx \int \phi(\mathbf{x}_{n-1}|\mu_{(n-1)i},\Sigma_{(n-1)i})p(\mathbf{x}_n|\mathbf{x}_{n-1})d\mathbf{x}_{n-1}, \tag{15}$$

where each covariance $\Sigma_{(n-1)i}$ approaches to zero as shown in [2].

With the piece-wise linear function between time $n-1$ and n (11), and the model noise $\mathbf{u}_{n-1}$ follows Gaussian noise with 3D covariance matrix $\mathbf{Q}_{n-1} \in \mathbb{R}^{3\times3}$, the time update step in GSF follows the Extended Kalman Filtering (EKF) method, and mean and covariance values of each predictive Gaussian can be obtained by the following equations:

$$\begin{aligned} \overline{\mu}_{ni} &= \tilde{\mathbf{f}}(\mu_{(n-1)i}) = \mathbf{R}\mu_{(n-1)i} + \mathbf{t} \\ \overline{\Sigma}_{ni} &= \mathbf{F}_{(n-1)i}\Sigma_{(n-1)i}\mathbf{F}_{(n-1)i}^T + \mathbf{Q}_{n-1}, \quad where, \\ \mathbf{F}_{(n-1)i} &= \frac{\partial\tilde{\mathbf{f}}(\mathbf{x})}{\partial\mathbf{x}}|_{\mathbf{x}=\mu_{(n-1)i}} = \mathbf{R}. \end{aligned} \tag{16}$$

B. Measurement update

When a new measurement $\widehat{\mathscr{G}}_n$ is obtained, the filtering distribution, $\mathscr{G}_n$, is updated from the predictive distribution, $\overline{\mathscr{G}}_n$, that is calculated in the time update step by calculating the posterior distribution of (8). Following GMM representation, the three distributions have a relationship of (17).

$$\begin{aligned} & p(\mathbf{x}_n|\mathbf{y}_{0:n}) \\ &= C_n p(\mathbf{y}_n|\mathbf{x}_n)p(\mathbf{x}_n|\mathbf{y}_{0:n-1}) \\ &= C_n \sum_{i=1}^{\widehat{k}_n}\sum_{j=1}^{\bar{k}_n} \widehat{w}_{ni}\overline{w}_{nj}\phi(\mathbf{x}_n|\widehat{\mu}_{ni},\widehat{\Sigma}_{ni})\phi(\mathbf{x}_n|\overline{\mu}_{nj},\overline{\Sigma}_{nj}) \\ &\triangleq \sum_{i=1}^{k_n} w_{ni}\phi(\mathbf{x}_n|\mu_{ni},\Sigma_{ni}) \end{aligned} \tag{17}$$

By means of the property of Gaussian function, the product of two Gaussians in (17) produces another Gaussian, which results in the mixture of $\widehat{k}_n \times \bar{k}_n$ Gaussians with parameters in (18).

$$\begin{aligned} & p(\mathbf{x}_n|\mathbf{y}_{0:n}) \approx \sum_{i=1}^{\widehat{k}_n}\sum_{j=1}^{\bar{k}_n} C_n w_{nij}\phi(\mathbf{x}_n|\mu_{nij},\Sigma_{nij}), \quad where \\ & \Sigma_{nij} = (\widehat{\Sigma}_{ni}^{-1} + \overline{\Sigma}_{nj}^{-1})^{-1} \\ & \mu_{nij} = \Sigma_{nij}\widehat{\Sigma}_{ni}^{-1}\widehat{\mu}_{ni} + \Sigma_{nij}\overline{\Sigma}_{nj}^{-1}\overline{\mu}_{nj} \\ & w_{nij} = \widehat{w}_{ni}\overline{w}_{nj}\frac{|\Sigma_{nij}|^{1/2}|\widehat{\Sigma}_{ni}+\overline{\Sigma}_{nj}|^{1/2}}{|\widehat{\Sigma}_{ni}|^{1/2}|\overline{\Sigma}_{nj}|^{1/2}}\phi(0|\widehat{\mu}_{ni}-\overline{\mu}_{nj},\widehat{\Sigma}_{ni}+\overline{\Sigma}_{nj}) \\ & C_n = \frac{1}{\sum_{i=1}^{\widehat{k}_n}\sum_{j=1}^{\bar{k}_n} w_{nij}} \end{aligned} \tag{18}$$

Because the number of Gaussians grows recursively as $k_n = k_{n-1} \times \widehat{k}_n$, the GMM simplification process is needed to limit the size of Gaussians to the given number. In this research, HC method [11] with L2 distance is used for the simplification process, and the number of Gaussians is determined proportional to the size of point set constructing an object with a simplification ratio $\lambda \in \mathbb{R}$ ranging between 0 to 1.

$$k_n = \frac{k_{n-1} + \lambda\widehat{k}_n}{2} \tag{19}$$

As an example result of the proposed filtering method, the integrated point set of two objects, O_n in Fig. 4(b) can be separated correctly to O^h and O^c as shown in Fig. 7. The each point P_i in the integrated point set is identified by comparing the value of filtering distribution functions at that point.

$$p_i \in \begin{cases} O_n^h & \text{for} \quad p^h(\mathbf{x}_n|\mathbf{y}_{0:n})|_{\mathbf{x}_n=p_i} > p^c(\mathbf{x}_n|\mathbf{y}_{0:n})|_{\mathbf{x}_n=p_i} \\ O_n^c & \text{for} \quad p^h(\mathbf{x}_n|\mathbf{y}_{0:n})|_{\mathbf{x}_n=p_i} < p^c(\mathbf{x}_n|\mathbf{y}_{0:n})|_{\mathbf{x}_n=p_i} \end{cases} \tag{20}$$

VI. Experiments and Results

The purpose of the proposed method is to represent any arbitrary time-varying objects' shape and position and track them robustly. The proposed 3D GMM-based representation is inherently and sufficiently flexible to describe any shape of objects, but the expressiveness and the robustness depends on how much one can reduce the size of Gaussians. Therefore, in order to evaluate the robustness (tracking accuracy)

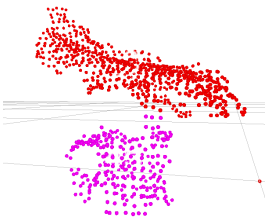


Fig. 7. The result of correcting the false segments in the Fig. 4(b)

according to the proposed filtering methods, we performed several experiments in dynamic situations. Fig. 8 shows six movements of two hands translating or rotating a white box in three dimensions. Two hands change their shapes and the box changes its position and orientation at every moment, and they are in contiguity with each other.

Table I shows the details of the six experiments. Each

TABLE I

AVERAGE NUMBER OF POINTS AT EACH FRAME OF THE SIX TEST DATA ACCORDING TO THE SAMPLING DISTANCE

Task	Time [s]	# of frames	Sampling distance [m]			
			0.01	0.015	0.02	0.025
translation in x	46.78	520	2092.28	975.05	551.09	352.52
translation in y	40.85	448	2181.45	1024.47	573.39	370.06
translation in z	37.77	416	2295.05	1062.85	595.43	380.43
rotation in x	39.40	432	2271.93	1064.02	610.74	396.70
rotation in y	44.27	480	2534.13	1183.50	673.85	433.80
rotation in z	47.68	533	2307.17	1086.04	617.30	399.00

action was repeated ten times, and it took around 40 seconds in total. The human hands and the object were observed by two Microsoft Kinect cameras, and each instance of captured point set data were merged with a common 3D coordinate. This massive point set data was then down-sampled with a constant sampling distance by using a VoxelGrid filter in [25]. This experiment was performed with four different sampling distances ranging from 0.01m to 0.025m because the size of the measurement GMM, $\widehat{k}_n$, is a substantial control parameter for the tracking performance. The filtered point set data is passed to the tracking algorithm every 90ms, and the measurement GMM is then constructed with a diagonal

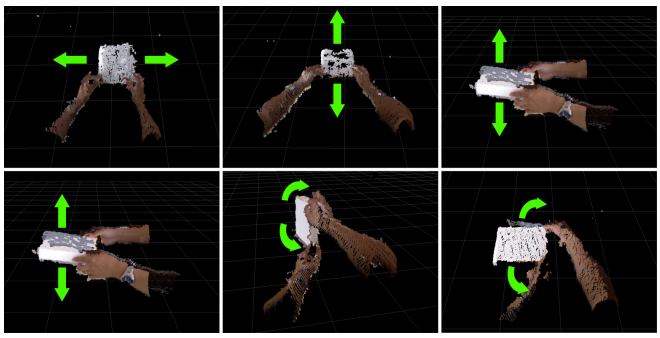


Fig. 8. Six hand motions with a white box: translation in x, y, and z-direction in the first row from left to right, and rotation in x, y, and z-axis in the second row

Fig. 9. Illustrations of the tracking results in the sequence (from top to down) of rotating in z-axis. The first column shows the original captured point set data. The figures on the second column are initial segmentation results. The third column illustrates Gaussian mixture models as a set of 3D ellipsoids. The tracking results of the proposed algorithm are depicted in the figures on the fourth column

covariance of the σ^2 value: σ is the value of corresponding sampling distance.

In order to evaluate the proposed method, the tracking accuracy was calculated by counting the number of points matching to each object as (21). The accuracy demonstrates the ratio of correctly segmented points to total points for all frames. For the evaluation, we used a white box for the moving object and it can be easily distinguished from the human hand via the color data. In this way, the ground truth object of each point was found using RGB data of each point.

$$accuracy = \left(1 - \frac{\sum_t \left(\sum_i^{n_t^H} w(p_i^H) + \sum_i^{n_t^B} 1 - w(p_i^B)\right)}{\sum_t \left(n_t^H + n_t^B\right)}\right) \times 100. \quad (21)$$

Here, p is the average value of the RGB data and $w(p)$ is 1 if p is close to color white. In these experiments, the threshold value to divide the human hand color and white was 125 in grayscale. n_t^H and n_t^B are the numbers of points of the human hand and the white box at time t, respectively. The computation device is an Intel i7 2.8GHz CPU and RGB-D point set data, size of 640 $\times$ 480, is captured at an average of 30Hz frequency.

A. The tracking results and errors

Fig.9 shows the selected snapshots of the test movement of z-axis rotation, with the sampling distance of 0.015m and the simplification ratio (λ) of 0.15. The figure on the first column are original 3D RGB-D data, and the corresponding segmentation results are shown on the second column. Initially, two hands and the white box are departed from each other such that the three objects are segmented correctly as

shown on the first row of Fig. 9. From the second row to the forth row shows the sequence of the test motions with multiple contacts between the three objects. The figures on the third column illustrates GMM of each object, and the final results of the proposed tracking algorithm are depicted in the forth column. The tracking errors are induced by the falsely identified points: a part of the right hand (id:23) contacting to the box as in the third row of Fig. 9. The size of error points are influenced by the filtering methods and the simplification process.

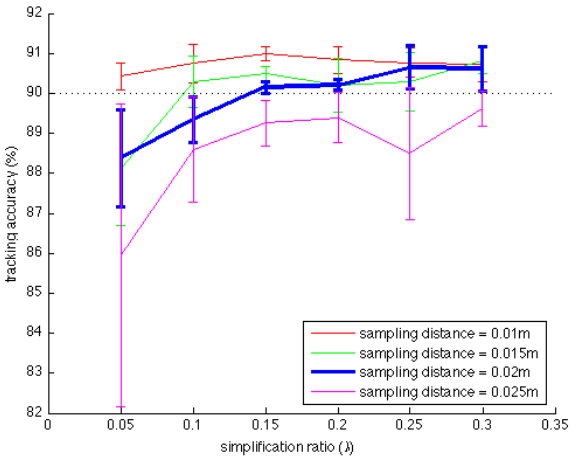
B. The tracking accuracy according to the representation

Fig. 10 and Fig. 11 show the tracking accuracies and computation times for the six tests (three translational and three rotational motions). In order to find the optimal control parameters of the sampling distance and simplification ratio, the requirement of the computable frames per second was set to a minimum of 5 FPS. In Fig. 11, the available parameter values are 0.025m for the sampling distance with any simplification ratio and 0.02m for the sampling distance with a simplification ratio of less than 0.15. Among these values the highest tracking accuracy can be obtained by the parameters of 0.02m for the sampling distance and 0.15 for the simplification ratio, thus achieving 90% accuracy for the three translations and 87% accuracy for the three rotations.

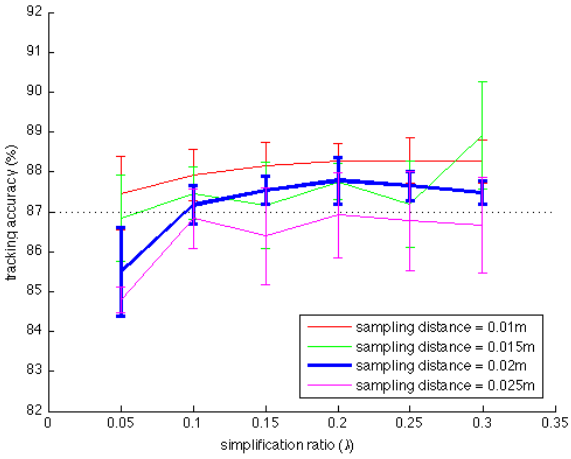
VII. Conclusion and Further Works

In this paper, we presented a novel tracking method for multiple moving objects from 3D point set data. In particular, this method adopted Gaussian mixture model to represent any arbitrary objects without prior knowledge. The flexibility of the model-free approach suffers from the false segmentations due to the contacts and occlusions among multiple moving objects. The proposed method enhanced the robustness of the tracking task by suggesting the GMM-based 3D registration and the Gaussian Sum Filtering for estimating GMM probability distribution of the true position of the objects. In addition, GMM simplification method was applied to improve real-time performance, and the tracking performance was examined by the various experiments in dynamic situations. The results showed that the tracking accuracy increases up to 91% using as more Gaussians as 30% of the number of points, while the real-time computation is not possible in the setting. As investigating the trade-off relation between the tracking accuracy and the computational efficiency according to the control parameters, this method is able to perform over 87% for the tracking accuracy with 5 FPS for the computation time. The optimal parameters were simplification ratio of 0.15 in the cases of about 600 points at every time frame that is reduced by down-sampling with 0.02m sampling distance from the original point data set.

Although the results showed the feasibility of the algorithm, there are some supplement points for further works. First, in order to enhance computational time for the real-time task, the simplification of GMMs should be optimized and GPU processing is required. Second, proper number of Gaussians and their spatial structure should be considered



(a) Translation tests



(b) Rotation tests

Fig. 10. Tracking accuracy results for the six tests

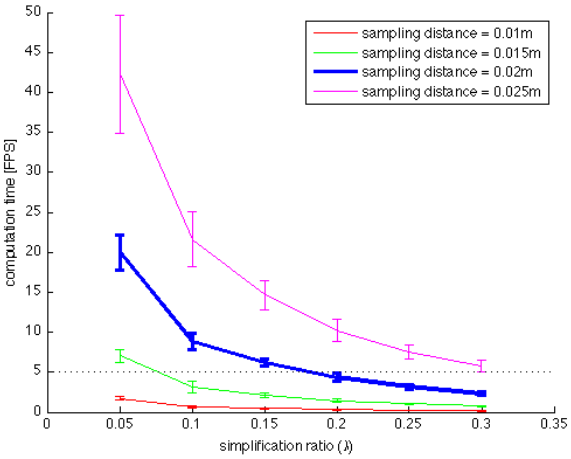


Fig. 11. Computation time results for the six tests

according to the shape of objects. Third, the proposed object representation and tracking methods were tested in the 'contact' case of multiple objects as shown in Fig. 1(a), but it needs to be extended to the cases of occlusion (Fig. 1(b)), split, addition, and removal of multiple objects. These further research can achieve on-line structure modeling of any articulated objects and robustly tracking them in real-time without prior-knowledge. That is, a robot can learn new objects and the related skills in an unstructured environment merely by observing human demonstration.

VIII. Acknowledgements

This work is supported in part within the DFG excellence initiative research cluster "Cognition for Technical System–CoTeSys" and financially supported by ERASMUS MUNDUS Build on Euro-Asian Mobility program of European Commission.

References

[1] E.E. Aksoy, A. Abramov, J. Dörr, K. Ning, B. Dellen, and F. Wörgötter. Learning the semantics of object–action relations by observation. *The International Journal of Robotics Research*, 30(10):1229–1249, 2011.

[2] B.D.O. Anderson and J.B. Moore. *Optimal filtering*, volume 11. Prentice-hall Englewood Cliffs, NJ, 1979.

[3] P.J. Besl and N.D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on pattern analysis and machine intelligence*, 14(2):239–256, 1992.

[4] J.A. Bilmes. A gentle tutorial of the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models. *International Computer Science Institute*, 4:126, 1998.

[5] N. Blodow, L.C. Goron, Z.C. Marton, D. Pangercic, T. Ruhr, M. Tenorth, and M. Beetz. Autonomous semantic mapping for robots performing everyday manipulation tasks in kitchen environments. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4263–4270. IEEE, 2011.

[6] L. Chang, J.R. Smith, and D. Fox. Interactive singulation of objects from a pile. In *International Conference on Robotics and Automation (ICRA)*, pages 3875–3882. IEEE, 2012.

[7] 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 (Methodological)*, pages 1–38, 1977.

[8] H. Dindo and G. Schillaci. An adaptive probabilistic approach to goal-level imitation learning. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4452–4457. IEEE, 2010.

[9] V. Garcia, F. Nielsen, and R. Nock. Hierarchical gaussian mixture model. In *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, 2010.

[10] J. Goldberger, H.K. Greenspan, and J. Dreyfuss. Simplifying mixture models using the unscented transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(8):1496–1502, 2008.

[11] J. Goldberger and S. Roweis. Hierarchical clustering of a mixture model. *Advances in Neural Information Processing Systems*, 17(NIPS-2004):505–512, 2005.

[12] L. Greengard and X. Sun. A new version of the fast gauss transform. *Documenta Mathematica*, 3:575–584, 1998.

[13] X. Huang, I. Walker, and S. Birchfield. Occlusion-aware reconstruction and manipulation of 3d articulated objects. In *IEEE International Conference on Robotics and Automation (ICRA), 2012*.

[14] B. Jian and B. Vemuri. Robust point set registration using gaussian mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8):1633–1645, 2011.

[15] B. Jian and B.C. Vemuri. A robust algorithm for point set registration using mixture of gaussians. In *10th IEEE International Conference on Computer Vision, (ICCV 2005)*, volume 2, pages 1246–1251. IEEE, 2005.

[16] Y. Jiang, M. Lim, C. Zheng, and A. Saxena. Learning to place new objects in a scene. *International Journal of Robotics Research (IJRR)*, 31(9):1021–1043, 2012.

[17] J.H. Kotecha and P.M. Djuric. Gaussian sum particle filtering. *IEEE Transactions on Signal Processing*, 51(10):2602–2612, 2003.

[18] M. Krainin, P. Henry, X. Ren, and D. Fox. Manipulator and object tracking for in-hand 3d object modeling. *The International Journal of Robotics Research*, 30(11):1311–1327, 2011.

[19] Z.C. Marton, L. Goron, R. Rusu, and M. Beetz. Reconstruction and verification of 3d object models for grasping. *Robotics Research*, pages 315–328, 2011.

[20] J.R. Medina, M. Lawitzky, A. Mortl, D. Lee, and S. Hirche. An experience-driven robotic assistant acquiring human knowledge to improve haptic cooperation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2011*, pages 2416–2422. IEEE, 2011.

[21] A.T. Miller and P.K. Allen. Graspit! a versatile simulator for robotic grasping. *Robotics & Automation Magazine, IEEE*, 11(4):110–122, 2004.

[22] L. Montesano, M. Lopes, A. Bernardino, and J. Santos-Victor. Learning object affordances: From sensory–motor coordination to imitation. *IEEE Transactions on Robotics*, 24(1):15–26, 2008.

[23] I. Oikonomidis, N. Kyriazis, and A. Argyros. Efficient model-based 3d tracking of hand articulations using kinect. *Procs. of BMVC, Dundee, UK (August 29–September 10 2011)[547]*, 2011.

[24] R.B. Rusu, N. Blodow, Z.C. Marton, and M. Beetz. Close-range scene segmentation and reconstruction of 3d point cloud maps for mobile manipulation in domestic environments. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2009)*, pages 1–6. Ieee, 2009.

[25] R.B. Rusu and S. Cousins. 3d is here: Point cloud library (pcl). In *2011 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1–4. IEEE, 2011.

[26] R. Schnabel, R. Wahl, and R. Klein. Efficient ransac for point-cloud shape detection. In *Computer Graphics Forum*, volume 26, pages 214–226. Wiley Online Library, 2007.

[27] K. Zhang and J.T. Kwok. Simplifying mixture models through function approximation. *IEEE Transactions on Neural Networks*, 21(4):644–658, 2010.