Incremental Object Part Detection toward Object Classification in a Sequence of Noisy Range Images

Stefan Gächter, Ahad Harati, and Roland Siegwart
Autonomous Systems Lab (ASL)
Swiss Federal Institute of Technology, Zurich (ETHZ)
8092 Zurich, Switzerland
{gaechter, harati, siegwart}@mavt.ethz.ch

***Abstract*— This paper presents an incremental object part detection algorithm using a particle filter. The method infers object parts from 3D data acquired with a range camera. The range information is uniquely quantized and enhanced by local structure information to partially cope with considerable measurement noise and distortion. The augmented voxel representation allows the adaptation of known track-before-detect algorithms to infer multiple object parts in a range image sequence. The appropriateness of the method is successfully demonstrated by an experiment.**

I. Introduction

In recent years, a novel type of range camera to capture 3D scenes emerged on the market. One such camera is depicted in figure 1. The measurement principle is based on time-of-flight using modulated radiation of an infrared source. Compared with other range sensors [1], range cameras have the advantage to be compact and at the same time to have a measurement range of several meters, which makes them suitable for indoor robotic applications. Further, range cameras provide an instant single image of a scene at a high frame rate though with a lower image quality in general [2], [3]. The 3D information acquired with a range camera is strongly affected by noise, outliers and distortions, because of its particular measurement principle using a CMOS/CCD imager [4], [5], [6]. This makes it difficult to apply range image algorithms developed in the past. Hence, the goal of this paper is to present an object part detection method adapted to range cameras.

Object parts are quite proper features for object classification based on geometric models [7], [8], [9]. This approach can account for different views of the same object and for variations in structure, material, or texture for the objects of the same kind, since more or less the decomposition of the objects into its parts remains unchanged. The majority of the currently available approaches in this field are appearance based, which makes them very sensitive to the mentioned variations.

In general, range image algorithms depend on the robust estimation of the differential properties of object surfaces [10]. Given the noisy nature of images from range cameras, this can only be obtained with high computational cost [11], [12]. However, the detailed reconstruction of object surface geometry is not necessary for part based object classification as far as the parts are detected. On the other hand, object parts can be represented properly by bounding-boxes [9], because the overall structure of an object part is more important and informative than the details of its shape or texture. For example, the concept of a chair leg is more about its stick like structure than wether it is wooden or metallic, of light or dark color, round or square.

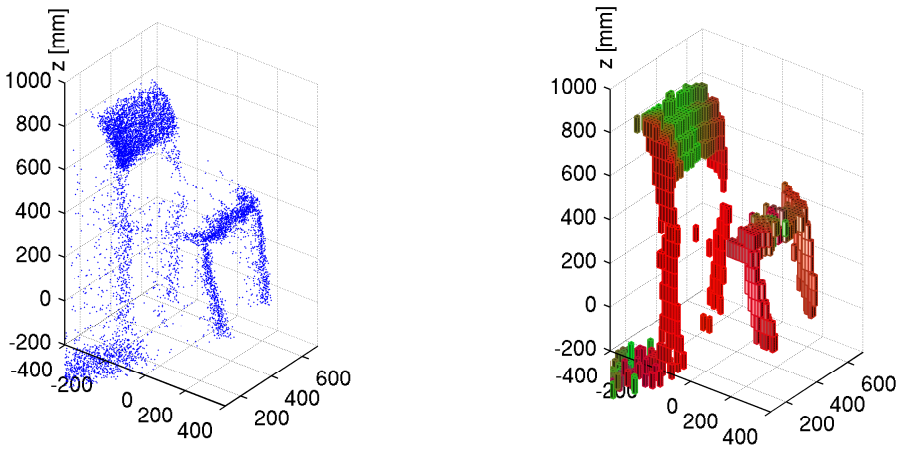
Fig. 1. Range Camera SR-3000.

However, segmentation of range images into object parts remains the most challenging stage. Because of the low signal-to-noise ratio of the mentioned sensor, this is an ill posed problem. Using an incremental algorithm and several range images can improve the performance. In fact, it is possible to skip segmentation and track hypothetical parts in the scene. This is a common approach in radar applications, where a target has to be jointly tracked and classified in highly noisy data [13], [14]. Hence, for each part category, a classifier is considered which incrementally collects the evidences from the sequence of range images and tracks the hypothetical parts. Therefore, the object part detection becomes the sequential state estimation process of multiple bounding-boxes at potential object part poses in the three-dimensional space. This is realized in the framework of a particle filter [15], [14], which can cope with different sources of uncertainty, among them scene registration errors.

To reduce the computational burden and at the same time partially remove the outliers, five sequential input point clouds are quantized into a voxel space and voxels containing less than three points are neglected. The voxel representation is enhanced with the shape factor, a local structure measure of linear, planar, or spherical likeliness which is then used for obtaining particle weights in the resampling phase.

The contribution of this work lies in bringing well es-

This work was partially supported by the EC under the FP6-IST-045350 robots@home project.



(a) Point Set (b) Voxel Set

Fig. 2. 3D point cloud and its quantized version of a sequence of five registered range images at time $k = 20$. The voxel color indicates the shape factors: red for *linear* like, green for *planar* like, and blue for *spherical* like local structures. Refer to the experimental section for the parameters used and the computational details.

tablished algorithm from classification, tracking and state-estimation to the framework of object classification. In addition, to the best of our knowledge, this is a first work which addresses object part detection using a range camera. The presented work here paves the way toward incremental object classification based on parts in the field of indoor mobile robotics. The approach presented here is quite general in handling different object parts with simple geometry. However, through out this paper, a chair leg is chosen as an example part to demonstrate the method.

II. Related Work

Part extraction from range images is a long standing issue in structure based object recognition and classification. Seminal work has been done by [16], where algorithms are presented that infere objects from surface information. Object parts are represented by surface patches. Others authors used the same representation, for example [17]. In the present work, bounding-boxes are adopted, which is a more abstract volumetric representation than commonly used parametrical models based on surfaces [18], [19], [20]. In addition, the quantization achieved by the voxel representation within the five-step time-span is related to occupancy grids [21], [22], but less computationally intensive.

In [23], a method to capture local structure in range images is presented in order to classify natural terrain. In the present work, local structure is captured in the same way with shape factors, and varieties of them – commonly used in 3D image processing [24] – are studied. However, shape factors are calculated based on the voxel representation here.

The object part detection algorithm evolves from the work done in [25], [26]. They developed a method for joint detection and tracking of multiple objects described by color histograms. Color-based tracking is a well researched topic in the vision community [27], [28], [29], [30]. Here, these techniques are taken as inspiration to detect object parts in quantized point clouds using shape factor as color.

III. Range Image Quantization

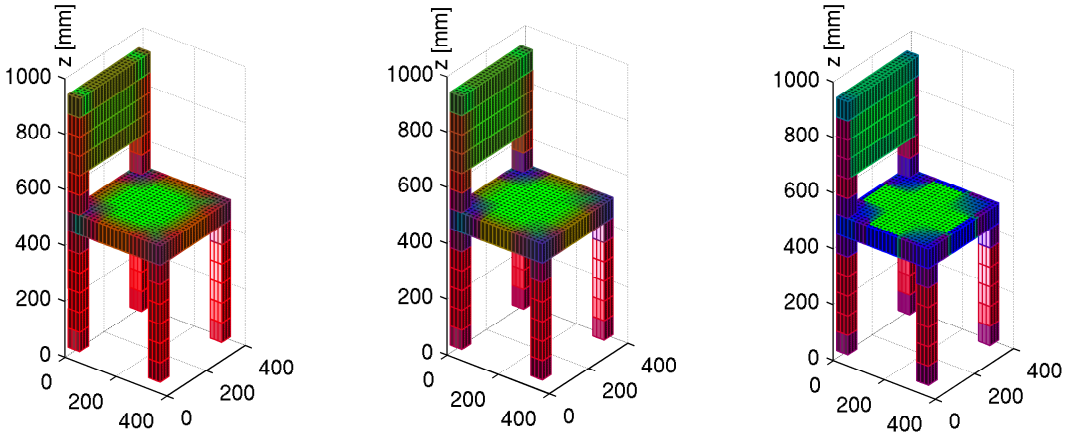
One of the smallest range cameras among different manufactures [31], [32], [33], [34], [35] is the SR-3000 made by [36], see figure 1. For the work presented here, the SR-2 of the same manufacturer is used, which exhibits similar measurement performance for the application in question. The camera has a resolution of 124×160 pixels with approximately a maximum measurement distance of $7.5m$. The intrinsic and extrinsic camera parameters are calibrated based on the methods explained in [37] and [6] respectively.

Despite the calibration, the range image remains affected by noise, outliers and distortions. Mainly the reasons are limited imager resolution and low emission power. Finally, the outcome is similar to what represented in figure 2(a) for a sample chair in the scene.

Therefore, to deal with high data volume and partially filter outliers, point clouds are quantized into voxels. In each time step, five successive 3D point clouds registered in a global frame [38] are considered. Commonly cubic voxels are used in such a scenario. Here, voxel shape is considered elongated along vertical direction to better capture stick like structures, which may be potential chair legs, see figure 2(b). This stage is not aimed to remove others parts – like seat or back – but to locally enforce the geometry of the desired part.

IV. Object Part Detection

The structural variability of objects is strongly related to the number and type of parts and their physical relationship with one another. Such relationships can be encoded within a probabilistic grammar in order to perform object classification [9]. Towards such an approach and considering the range camera as the sensing system, object parts are modeled as probabilistic bounding-boxes.



(a) Maximum (b) Sum (c) Volume

Fig. 3. Test voxel set colored according to the three different shape factor computation methods: normalization by the *maximum*, *sum*, or reasoning on the spanned *volume* of the eigenvalues. The voxel color indicates the shape factors: red for *linear* like, green for *planar* like, and blue for *spherical* like local structures.

A bounding-box is a cuboid defined by the center point and span length. Its probabilistic nature results from the incremental estimation process with particle filters. In this work, the particles encode the hypothetical positions and extensions of multiple object parts, i.e. of bounding-boxes. The evolution of the particles over time enable the simultaneous detection and tracking of the object parts. Particles at those positions object parts are observed survive, whereas the others die off. The goodness of part observation – the particle's weight – is calculated based on the shape factors found in the image regions defined by the bounding-boxes associated with each particle. Thus in addition to the bounding-box, an object part is modeled by a unique distribution of shape factors.

A. Shape Factor

The shape factors characterizes the local part structure by its linear, planar, or spherical likeliness. They are calculated for each voxel from its surrounding spatial voxel distribution by the decomposition of the distribution into the principal components – a set of ordered eigenvalues and -vectors.

In the literature, different methods are presented on how to compute the shape factors. In [39], [40] a tensor representation is proposed for structure inference from sparse data. A structure tensor can be expressed as a linear combination of a linear, planar, and spherical basis tensor, where the relative magnitudes for a local region can be defined as the shape factors for the linear r_l, planar r_p, and spherical r_s case, respectively:

$$\begin{aligned} r_l &= \frac{\lambda_1 - \lambda_2}{\lambda_1}, \\ r_p &= \frac{\lambda_2 - \lambda_3}{\lambda_1}, \\ r_s &= \frac{\lambda_3}{\lambda_1}, \end{aligned} \tag{1}$$

where λ_i are the ordered eigenvalues $\lambda_1 \geq \lambda_2 \geq \lambda_3$ of the tensor decomposition. For the present work, the voxel distribution decomposition is done equivalently. In (1), the shape factors are normalized by the maximum eigenvalue so that each lies in the range $\in [0, 1]$ and their sum is one, $r_l + r_p + r_s = 1$. Another normalization scheme is to use the sum of the eigenvalues [24]:

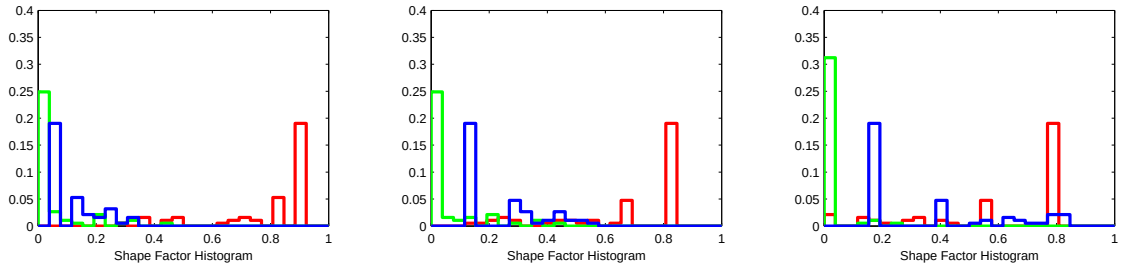
$$\begin{aligned} r_l &= \frac{\lambda_1 - \lambda_2}{\lambda_3 + \lambda_2 + \lambda_1}, \\ r_p &= \frac{2(\lambda_2 - \lambda_3)}{\lambda_3 + \lambda_2 + \lambda_1}, \\ r_s &= \frac{3\lambda_3}{\lambda_3 + \lambda_2 + \lambda_1}. \end{aligned} \tag{2}$$

The shape factors can also be defined by reasoning on the volume spanned by the eigenvalues:

$$\begin{aligned} r_l &= \frac{\lambda_1}{\sqrt{\lambda_2\ \lambda_3}}, \\ r_p &= \frac{\sqrt{\lambda_2\ \lambda_1}}{\lambda_3}. \end{aligned} \tag{3}$$

The two shape factors tend toward zero, if the structure is spherical. However, the normalization of these factors is not evident, but it is possible to find a suitable normalization function as discussed in [38]. The advantage is then to have additional parameters to adapt the shape factors' saliency. Which of the shape factor computation methods is used, depends largely on their ability to characterize voxels distinctively according to the present object structure at hand.

Figure 3 depicts a test voxel set of a chair, where for each voxel of size $15 \times 15 \times 70mm$ the shape factor was computed according to the three presented methods. The computation was done with a neighborhood window – defining the scale of the local structure – of size $11 \times 11 \times 3$ voxels. As it is visible, the first method, where the maximum eigenvalue is



(a) Maximum (b) Sum (c) Volume

Fig. 4. Shape factor histograms of the front chair leg in the test voxel set, where the shape factors have been computed according to the three different computation methods: normalization by the *maximum*, *sum*, or reasoning on the spanned *volume* of the eigenvalues. The color indicates the shape factor distribution: red for *linear* like, green for *planar* like, and blue for *spherical* like local structures.

used for normalization, favors linear like structures, whereas the third method, where the spanned volume is used as criterion, favors spherical like structures. A balanced result is obtained with the second method, where the sum of the eigenvalues is used for normalization. This is also the method employed later on, because it gives the best result for real data.

B. Histogram as Feature Vector

The shape factor distribution in the region of interest defined by the bounding-box is approximated by a histogram to obtain a unique feature vector that models an object part. This approach is inspired by the work done in [27], where color histograms are used to track objects. In the present application, histograms have the advantage to be robust against the structural variability of object parts: rotation, partial occlusion, and scale have little effect on the model. In addition, the computational cost of histograms is modest.

The bins of the shape factor histogram are populated with the three different shape factors, where for each N_b bins are used. It is not necessary to account for all possible combinations, as commonly done for color histogram computation, because the shape factors sum up to one. Thus, $N_t = 3 \times N_b$ bins are sufficient. All voxels in the bounding-box volume are considered so that empty voxels retain some spatial information in the otherwise spatially invariant histogram.

The bin index $b \in \{1, \ldots, N_t\}$ is associated with the shape factor vector $\mathbf{r}(\mathbf{u})$ for each voxel position $\mathbf{u}$. The bounding-box volume in which the shape factor information is gathered is defined as $V(\mathbf{x})$. Within this region the shape factor distribution $\mathbf{q}(\mathbf{x}) = \{q(n;\mathbf{x})\}_{n=1\ldots N_t}$ is estimated by standard bin counting:

$$q(n;\mathbf{x}) = K \sum_{\mathbf{u} \in V(\mathbf{x})} \delta(b(\mathbf{x}) - n), \tag{4}$$

where δ is the Kronecker delta function; K is a normalization constant ensuring $\sum_{n=1}^{N_t} q(n;\mathbf{x}) = 1$. Hence, the feature vector consists of N_t elements, each representing a certain shape factor likeliness. Figure 4 depicts the three shape factor histograms of the front chair leg of the test voxel set in figure 3. It is clearly visible that the linear shape factor dominates indicating the stick like structure of the object part.

C. Support Vector Classifier

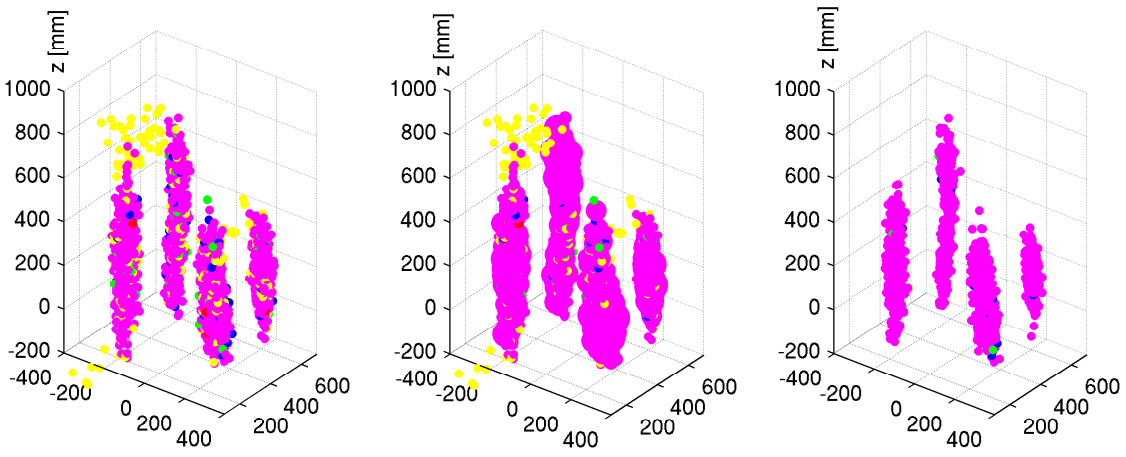
It is now possible to generate for each hypothetical object part – encoded by the particles – a feature vector. In order to judge, if an object part in question is likely to belong to a certain object part class it is necessary to evaluate an importance weight. This can be done by computing a similarity measure based on the distance between a template feature vector and the generated feature vector. This is commonly done in color based tracking [27]. However, template matching might not be discriminative enough to cover an entire object part class. Using a classifier built on large amount of training data results often in a better detection performance. Given the high dimension of the feature vector – the shape factor histogram – a suitable training method is the support vector machine (SVM). In the present work, a support vector classifier with a linear kernel is trained using the framework provided by [41]. The details of SVM are omitted here, but can be found for example in the just mentioned reference.

D. Incremental State Estimation

The aim is to detect incrementally object parts modeled by a bounding-box in a sequence of voxel images. The detection algorithm has to typically handle multiple object parts of the same type. Thus, the problem can be stated formally as follows:

$$P(R_k = r|\mathbf{z}_{1:k}) = \int p(R_k = r, \mathbf{x}_{1:r,k}|\mathbf{z}_{1:k}) d\mathbf{x}_{1:r,k}. \tag{5}$$

The probability P of having r parts present at time k is the marginal of the joint probability of the object part states $\mathbf{x}_{1:r,k} = [\mathbf{x}_{1,k}, \ldots, \mathbf{x}_{r,k}]$ and their number $R_k = r$ given a voxel image sequence $\mathbf{z}_{1:k} = z_1, \ldots, z_k$. Thus, the object part state – the position and extension of the bounding-box – is modeled by a random vector $\mathbf{x}_k$. The object part number R_k is modeled as a Markov system, where the state values are a discrete number $r = \{1, \ldots, M\}$ with M being the maximum number of parts expected.



(a) A priori Density (b) Observation Density (c) A posteriori Density

Fig. 5. Particle distributions at time $k = 20$ after prediction (a), during the update (b), and after the resampling (c) step. The color indicates the number of hypothetical parts encoded by a particle: red for 1, green for 2, blue for 3, magenta for 4, and yellow for 5 states.

The solution of (5) can be found in a recursive prediction and update procedure using a particle filter with augmented particle state $\mathbf{y}_k^{(n)} = [R_k^{(n)}; \mathbf{x}_{1:r,k}^{(n)}]$, where $n = 1, \ldots, N$, see [25]. The particle filter approximates the posterior density $p(\mathbf{y}_k|\mathbf{z}_{1:k})$ by a weighted set of N random samples. The evolution of each particle state through time is defined by the transition probability function $p(\mathbf{y}_k|\mathbf{y}_{k-1})$ – the relation among particle states over time – and the observation likelihood function $p(\mathbf{z}_k|\mathbf{y}_k)$ – the relation between particle state and observation.

The particle filter discussed in the following is a sampling-importance-resampling filter as presented in [42], that has been extended to multiple targets in [25] with the multiple-model approach as proposed in [14]. Multiple-model means in the current context that the filter deals with a particle state of continuous and discrete values. The particle filter has the same structure as in [25], but the transition and observation model have been adapted where necessary to perform object part detection with sensory information from a range camera.

1) Transition Model: The transition model of the object part state is the linear model $\mathbf{x}_k = \mathbf{x}_{k-1} + \mathbf{v}_{k-1}$, where $\mathbf{v}_{k-1}$ is the process noise assumed, for simplicity, to be white, zero-mean Gaussian with covariance matrix $\mathbf{C}_u$. The transition probability density function is $p(\mathbf{x}_k|\mathbf{x}_{k-1}) = \mathcal{N}(\mathbf{x}_{k-1}, \mathbf{C}_u)$. This transition model helps to account for errors in the image registration process mainly caused by the viewpoint dependency of the range image quality which is typical for range cameras.

The model of the object part number R_k is the Markov system defined by the transition matrix $\mathbf{T}_u$ of dimension $M \times M$, where $t_u^{i,j}$ is the probability of change in the number of parts. For example, the probability to observe one part $j = 1$ when currently none is observed $i = 0$ is $t_u^{i,j}$. The Markov system can be extended to any number of parts, but with increasing number of parts the number of particles has to be adjusted accordingly to obtain a similar detection performance.

The Markov system defines three possible cases how the number of parts can evolve over time: the number remains *unaltered*, *increases*, or *decreases* from time step $k-1$ to k. For each case, particles are drawn differently from the transitional prior $p(\mathbf{x}_k|\mathbf{x}_{k-1})$.

- $R_k^{(n)} > R_{k-1}^{(n)}$: If the number of parts increases, the current state has to be augmented by additional states. For the r_{k-1} parts that continue to exist, particles $\mathbf{x}_{i,k}^{(n)}$ are sampled from the transitional prior $p(\mathbf{x}_k|\mathbf{x}_{k-1})$. For the $r_k - r_{k-1}$ new hypothetical parts, particles are sampled from the initialization density $\tilde{p}(\mathbf{y}_{k-1}|\mathbf{z}_{k-1})$, which describes the probability of having a part with state $\mathbf{y}_{k-1}$, when only the observation $\mathbf{z}_{k-1}$ is available. Instead of sampling uniformly from the entire observation space, the sampling is constraint to where only an observation at time $k-1$ took place; a hypothetical object part position is selected from the possible voxel position with equal probability. Thus, the particles evolve only in the neighborhood of the observed structure in the scene. Objects have a well defined geometry and it is not possible that two parts of the same kind can be observed at the same position. Therefore, a minimum distance constraint between parts is imposed during the sampling. Because the parts are represented by bounding-boxes, the constraint is verified by an intersection check as presented in [43]. Apart from the design of the observation likelihood, these two measures enable the object part detection in the three dimensional space.
- $R_k^{(n)} = R_{k-1}^{(n)}$: If the number of parts remains unaltered, the r_k particles $\mathbf{x}_{i,k}^{(n)}$ are sampled from the transitional prior $p(\mathbf{x}_k|\mathbf{x}_{k-1})$.
- $R_k^{(n)} < R_{k-1}^{(n)}$: If the number of parts decreases, r_k hypothetical parts are select at random from the possible r_{k-1} with equal probability. For the selected parts, r_k particles $\mathbf{x}_{i,k}^{(n)}$ are sampled from the transitional prior

$p(\mathbf{x}_k|\mathbf{x}_{k-1})$.

2) Observation Model: The observation likelihood function generates the importance weights used to incorporate the measurement information $\mathbf{z}_k$ in the particle set $\{\mathbf{y}_k^{(1)}, \ldots, \mathbf{y}_k^{(N)}\}$. Given the problem at hand – the parts have to be detected from various view angles out of sparse and noisy data – the observation model is a non-linear function $\mathbf{z}_k = g(\mathbf{x}_k, \mathbf{w}_k)$ of the part state $\mathbf{x}_k$ and measurement noise $\mathbf{w}_k$. Instead of using a generative observation model, as it is common in a Bayesian estimation framework, a discriminative one is selected [44], i.e. the learned support vector machine presented previously.

In the detection framework, the observation likelihood function is defined by the comparison of the probability that an object part is present with that an object part is absent. This is equivalent to the likelihood ratio of the classification probabilities computed with the learned classifier. Assuming that the classification can be done independently for each hypothetical object part, the likelihood ratio is then

$$L(R_k) = \prod_{i=1}^{r} \frac{p(\mathbf{z}_k|\mathbf{x}_{i,k})}{1 - p(\mathbf{z}_k|\mathbf{x}_{i,k})}. \tag{6}$$

Considering the classification probability $p(\mathbf{z}_k|\mathbf{x}_{i,k})$ as a discriminative measure $a_{i,k}$ in the range $\in [0, 1]$, the likelihood ratio can be expressed as

$$L(R_k) = \exp\left(-\frac{1}{b}\sum_{i=1}^{r}(1 - 2a_{i,k})\right), \tag{7}$$

where b is a parameter to adjust the observation sensitivity, which has to be determined experimentally. With this definition, the likelihood ratio takes large values in the 3D space where object parts are present and correctly identified.

Thus, the unnormalized importance weight $\tilde{\pi}_k^{(n)}$ for each particle with state $\mathbf{y}_k^{(n)} = [R_k^{(n)}; \mathbf{x}_{1:r,k}^{(n)}]$ is computed as:

$$\tilde{\pi}_k^{(n)} = \begin{cases} 1, & \text{if} \quad R_k^{(n)} = 0 \\ L(R_k^{(n)}), & \text{if} \quad R_k^{(n)} > 0. \end{cases} \tag{8}$$

The likelihood ratio defined above has a pivoting point for a probability equal 0.5. Further, particles with large number of hypothetical object parts but having a classification probability only slightly greater than 0.5 are favored over particles with small number of parts but having a high probability. Hence, the object part detection algorithm has an inherent tendency for exploration.

V. Experiment

The above discussed incremental object part detection method is exemplified by the detection of chair legs, a linear object structure in vertical direction. It is reasonable to assume – especially for indoor robotic applications – that the pose of the range camera with respect to the ground plane can be inferred. With this knowledge, the object part representation of the class of chair legs can be simplified: a bounding-box defined by its center point position $\mathbf{s} = [s_x, s_y, s_z]^\mathsf{T}$ and span length $\mathbf{t} = [t_x, t_y, t_z]^\mathsf{T}$. For other object part classes, such as chair seat and table plate, the rotation around the z-axis has to be considered in addition.

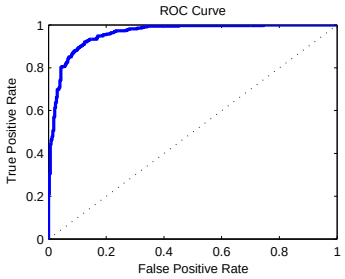
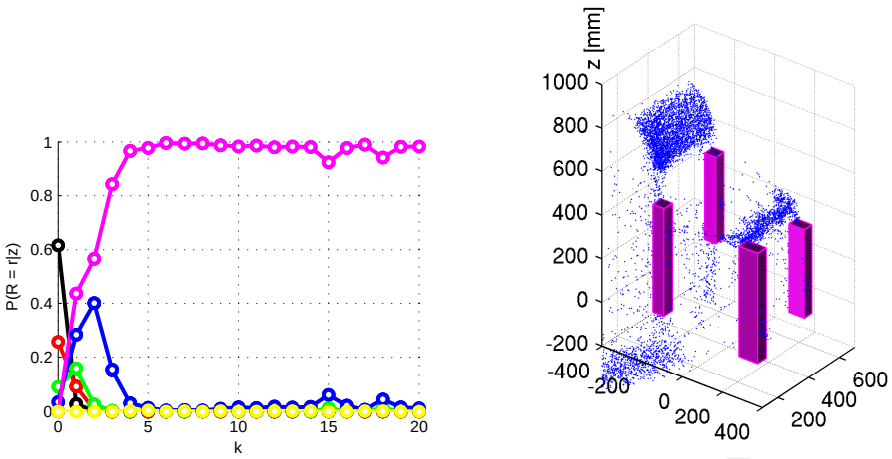


Fig. 6. Receiver operating characteristic (ROC) curve of the support vector classifier for linear like object structures.

The object part detection method is applied to a series of about 300 range images taken of a chair by moving the camera from the bottom to the top. At each time step k, the range image is transformed into a 3D point cloud and roughly aligned with the reference frame. The reference frame results from an initial setup calibration [38]. The aligned point cloud is quantized and added to a voxel set that is accumulated over the last five images. The voxels have the dimension of $15 \times 15 \times 70\,mm$ to accommodate for linear object structures in vertical direction. Gross outliers are removed discarding voxels with less than three measurement points associated. For each voxel in the set, the shape factor is computed according to (2) using a neighborhood window of dimension $21 \times 21 \times 15$ voxels. The sum is used for normalization, because it results in balanced shape factors for the linear and planar structures present. Thus, at each time step k a discrete, sparse 3D image results, where with each voxel in the image a triplet of shape factor is associated. From a sequence of such images, the object parts are inferred.

The particle filter uses $N = 1000$ samples for a maximum number of parts $M = 5$. This rather low number of particles becomes possible, because the particles are constraint in the 3D space to the neighborhood of where an observation took place. The particle's state consists of the number of hypothetical parts, their bounding-box positions and span lengths: $\mathbf{y}_k^{(n)} = [R_k^{(n)}; \mathbf{s}_{1:r,k}^{(n)}; \mathbf{t}_{1:r,k}^{(n)}]$. The probabilities in the transition matrix $\mathbf{T}_u$ are conservatively chosen to keep the gathered knowledge, but also because the particle filter implementation has an inherent exploration behavior. Hence, the diagonal entries of the matrix have a probability of 0.7, the entries from r to $r + 1$ parts have a probability of 0.1, and the remaining entries have a probability of 0.05. The covariance matrix $\mathbf{C}_u$ for the transition model has the diagonal entries of 100, 100, and $2500\,mm^2$. Importance is given to the vertical direction to accommodate for the high part position uncertainty in this dimension.

A training set of 3100 samples was generated to train the support vector classifier by computing the shape factor histograms of randomly selected bounding-boxes in a stream of voxel images. The samples were manually labeled so that a classifier was trained that can detect linear like object



(a) Part Presence Probability

(b) Estimated Parts

Fig. 7. Evolution of the part presence probability over time (a). Corresponding estimated object parts at time $k = 20$ (b). The color indicates the number of hypothetical parts encoded by a particle: black for *none*, red for 1, green for 2, blue for 3, magenta for 4, and yellow for 5 states.

structures. However, of all the labeled samples, only an equal number of good and bad ones are used to not bias the learning; a set of 767 for training and 328 for verification. The performance of the resulting linear support vector classifier is reasonably good as can be see in figure 6. A not perfect classifier might be even of advantage to alleviate ambiguities when more than one object part of the same class are present.

The behavior of the particle filter can be observed in figure 5, where for time step $k = 20$ the a priori, observation, and a posteriori particle density are depicted. It is visible in 5(c) that after the resampling particles with good classification probability survived. Almost all of the particles indicate four states, or that four legs are detected. The weights are represented in figure 5(b) by the size of the particles. Thus, particles at the chair's leg positions, see also figure 2(b), are bigger than the ones at the chair's back position. Further, the exploratory behavior defined by the transition matrix is visible in figure 5(a). Not only particles indicating four states are present, but also particles with more or less number of states.

The probability of the number of object parts present is computed according to (5). In case of the particle filter, the probability is approximated by $P(R_k = r|\mathbf{z}_{1:k}) \approx \sum_N \delta(R_k^{(n)}, r)/N$, where N is the number of particles. The probabilities for a sequence of range images are depicted in figure 7(a). In figure 7(b), the estimated object parts at time step $k = 20$ are depicted overlaid over the original 3D point clouds. The mean bounding-box is computed for particles having a part presence probability over 0.5. As it can be seen from figure 7(a), this is only the case for particles representing the four legs. Note that initially particles for different number of parts compete with each other, but that finally only the four state particles survive. In this sequence, no false positive occurred.

VI. Conclusion

This paper presented an algorithm for object part detection using a particle filter. The algorithm can handle multiple parts of the same class and different uncertainties. The experiment showed that the approach can estimate the chair leg position and extension in the current range image given the measurement history. Thus, the noisy and sparse information of the object part structure is successfully accumulated over time. Hence, the basic scheme has been proven to be suitable for incremental object part detection.

However, it needs further testing and improvements for its robust application in robotics. First, the detection has to be extended to multiple classes of object structures by training the corresponding support vector classifiers. Then further investigation can be done on recently introduced support vector classifiers [44].

Finally, in the prediction phase of the particle filter, instead of constraining the new samples to the occupied voxels, more informative constraints can be utilized by considering plausible object configurations. We are currently integrating the presented algorithm into a part based object classification system.

Acknowledgement

Thanks to Jacek Czyz for providing further insights into his particle filter implementation and to Luciano Spinello for the inspiring discussions on SVM.

References

[1] F. Blais, "Review of 20 years of range sensor development," *Journal of Electronic Imaging*, vol. 13, no. 1, pp. 231–243, January 2004.

[2] S. May, K. Pervoelz, and H. Surmann, *Vision Systems - Applications*. I-Tech, 2007, ch. 3D Cameras: 3D Computer Vision of wide Scope, pp. 181–202.

[3] C. Beder, B. Bartczak, and R. Koch, "A comparison of pmd-cameras and stereo-vision for the task of surface reconstruction using patchlets," in *Proceeding of the Second International ISPRS Workshop, BenCOS 2007, held with IEEE CVPR 2007*, 2007, pp. 1–8.

[4] O. Gut, "Untersuchungen des 3D-Sensors SwissRanger," Master's thesis, Institute of Geodesy and Photogrammetry - Swiss Federal Institute of Technology, Zurich, 2004, http://www.geometh.ethz.ch/publicat/diploma/gut2004/ (14.9.2007).

[5] S. A. Gudmundsson, H. Aanaes, and R. Larsen, "Environmental effects on measurement uncertainties of time-of-flight cameras," in *International Symposium on Signals, Circuits and Systems (ISSCS 2007)*, vol. 1, 2007, pp. 1–4.

[6] S. May, B. Werner, H. Surmann, and K. Pervölz, "3D time-of-flight cameras for mobile robotics," in *Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006.

[7] R. A. Brooks, "Symbolic reasoning among 3-D models and 2-D images," *Artificial Intelligence*, vol. 17, pp. 285–348, 1981.

[8] L. Stark and K. W. Bowyer, *Generic Object Recognition using Form and Function*, ser. Series in Machine Perception and Artificial Intelligence, H. W. P. Bunke, Ed. World Scientific, 1996.

[9] M. A. Aycinena, "Probabilistic geometric grammars for object recognition," Master's thesis, Massachusetts Institute of Technology - Department of Electrical Engineering and Computer Science, 2005.

[10] P. J. Besl, *Surfaces In Range Image Understanding*. Springer-Verlag Inc., New York, 1988.

[11] G. Danuser and M. Stricker, "Parametric model fitting: From inlier characterization to outlier detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 3, pp. 263–280, March 1998.

[12] H. Wang and D. Suter, "Robust adaptive-scale parametric model estimation for computer vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 11, pp. 1459–1474, November 2004.

[13] Y. Bar-Shalom and X. R. Li, *Estimation and Tracking: Principles, Techniques, and Software*. Artech House, 1993.

[14] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter - Particle Filters for Tracking Applications*. Artech House, 2004.

[15] Y. Boers and H. Driessen, "A particle-filter-based detection scheme," *Signal Processing Letters, IEEE*, vol. 10, no. 10, pp. 300–302, October 2003.

[16] R. B. Fisher, *From Surfaces to Objects - Computer Vision and Three Dimensional Scene Analysis*. John Wiley & Sons Ltd., Chichester, Great Britain, 1989, http://homepages.inf.ed.ac.uk/rbf/BOOKS/FSTO/ (14.9.2007).

[17] N. S. Raja and A. K. Jain, "Obtaining generic parts from range images using a multi-view representation," *CVGIP: Image Understanding*, vol. 60, no. 1, pp. 44–64, 1994.

[18] Q.-L. Nguyen and M. D. Levine, "Representing 3-d objects in range images using geons," *Computer Vision and Image Understanding*, vol. 63, no. 1, pp. 158–168, 1996.

[19] H. Rom and G. Medioni, "Part decomposition and description of 3D shapes," in *Proceedings of the 12th IAPR International Conference on Pattern Recognition*, vol. 1, 1994, pp. 629–632.

[20] W. H. Field, D. L. Borges, and R. B. Fisher, "Class-based recognition of 3D objects represented by volumetric primitives," *Image and Vision Computing*, vol. 15, no. 8, pp. 655–664, August 1997.

[21] J. P. Jones, "Real-time construction of three-dimensional occupancy maps," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '93)*, vol. 1, 1993, pp. 52–57.

[22] P. Payeur, P. Hebert, D. Laurendeau, and C. Gosselin, "Probabilistic octree modeling of a 3d dynamic environment," in *Robotics and Automation, 1997. Proceedings., 1997 IEEE International Conference on*, vol. 2, 1997, pp. 1289–1296.

[23] N. Vandapel, D. Huber, A. Kapuria, and M. Hebert, "Natural terrain classification using 3-d ladar data," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '04)*, vol. 5, 2004, pp. 5117–5122.

[24] C.-F. Westin, S. Peled, H. Gudbjartsson, R. Kikinis, and F. A. Jolesz, "Geometrical diffusion measures for MRI from tensor basis analysis," in *Proceedings of the 5th Annual Meeting of the International Society for Magnetic Resonance Medicine (ISMRM)*, 1997, p. 1742.

[25] J. Czyz, B. Ristic, and B. Macq, "A color-based particle filter for joint detection and tracking of multiple objects," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '05)*, vol. 2, 2005, pp. 217–220.

[26] Y. Boers and H. Driessen, "Hybrid state estimation: A target tracking application," *Automatica*, vol. 38, December 2002.

[27] P. Pérez, C. Hue, J. Vermaak, and M. Gangnet, "Color-based probabilistic tracking," in *Proceedings of the European Conference Computer Vision (ECCV)*, ser. LNCS 2350, A. H. et al., Ed. Springer-Verlag, 2002.

[28] K. Nummiaro, E. Koller-Meier, and L. J. Van Gool, "Object tracking with an adaptive color-based particle filter," in *Proceedings of the 24th DAGM Symposium on Pattern Recognition*, 2002, pp. 353–360.

[29] E. Maggio and A. Cavallaro, "Hybrid particle filter and mean shift tracker with adaptive transition model," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '05)*, vol. 2, 2005, pp. 221–224.

[30] H. Wang, D. Suter, K. Schindler, and C. Shen, "Adaptive object tracking based on an effective appearance filter," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 9, pp. 1661–1667, 2007.

[31] Canesta Inc., USA, http://www.canesta.com/ (13.9.2007).

[32] PMDTechnologies GmbH, Germany, http://www.pmdtec.com/ (13.9.2007).

[33] Matsushita Electric Industrial Co. Ltd, Japan, http://biz.national.jp/Ebox/security/tomozure/index.html/ (13.9.2007).

[34] Sharp Co., Japan, http://www.sharp.co.jp/corporate/news/060323-a.html/ (13.12.2006).

[35] 3DV Systems, Israel, http://www.3dvsystems.com/ (13.9.2007).

[36] MESA Imaging AG, Switzerland, http://www.swissranger.ch/ (13.9.2007).

[37] T. Kahlmann, F. Remondino, and H. Ingensand, "Calibration for increased accuracy of the range imaging camera SwissRanger," in *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, ISPRS Commission V Symposium*, vol. XXXVI, no. 5, 2006, pp. 136–141.

[38] S. Gächter, "Incremental object part detection with a range camera," Autonomous Systems Lab, Swiss Federal Institute of Technology, Zurich (ETHZ), Switzerland, Tech. Rep. ETHZ-ASL-2006-12, 2006.

[39] C.-F. Westin, "A tensor framework for multidimensional signal processing," Ph.D. dissertation, Department of Electrical Engineering - Linköping University, Sweden, 1994.

[40] G. Medioni, M.-S. Lee, and C.-K. Tang, *A Computational Framework for Segmentation and Grouping*. Elsevier, 2000.

[41] C.-W. Hsu, C.-C. Chang, and C.-J. Lin, "A practical guide to support vector classification," Department of Computer Science - National Taiwan University, Tech. Rep. July 18, 2007, 2007.

[42] M. Isard and A. Blake, "CONDENSATION - conditional density propagation for visual tracking," *International Journal of Computer Vision*, vol. 29, no. 1, pp. 5–28, 1998.

[43] T. Akenine-Möller and E. Haines, *Real-Time Rendering*, second edition ed. A K Peters, 2002.

[44] C. Shen, H. Li, and M. J. Brooks, "Classification-based likelihood functions for bayesian tracking," in *Proceedings of the IEEE International Conference on Video and Signal Based Surveillance (AVSS'06)*, 2006, p. 33.