Decision Making Under Uncertain Segmentations

Joni Pajarinen and Ville Kyrki

***Abstract*— Making decisions based on visual input is challenging because determining how the scene should be split into individual objects is often very difficult. While previous work mainly considers decision making and visual processing as two separate tasks, we argue that the inherent uncertainty in object segmentation requires an integrated approach that chooses the best decision over all possible segmentations. Our approach over-segments the visual input and combines the segments into possible objects to get a probability distribution over object compositions, represented as particles. We introduce a Markov chain Monte Carlo procedure that aims to produce exact, independent samples. In experiments, where a 6-DOF robot arm moves object hypotheses captured by an RGB-D visual sensor, our approach of probability distribution based decision making outperforms an approach which utilises the traditional most likely object composition.**

I. INTRODUCTION

Segmentation is one of the key components of many vision systems. Most current approaches search for the single best segmentation. However, the quality of a particular segmentation result is strongly application dependent, to the extent that ambiguities are unavoidable without prior knowledge of object models [1].

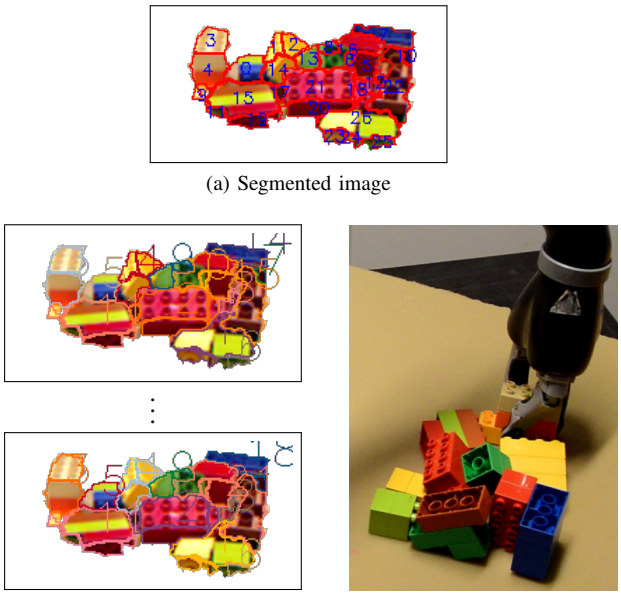
In this paper, we argue that optimal segmentation should consider the application even further so that the segmentation result should be considered together with its intended use. Furthermore, we claim that a single optimal result does not necessarily lead to optimal utility in the application, but that better utility can be obtained by considering the distribution of possible segmentations. The difference can be significant in cases where the segmentation will be used as a basis for decisions with variable benefits, which can be found often in robotics [2] and other vision systems such as autonomous cars [3] or security cameras [4].

The approach is in contrast to most state-of-the-art segmentation approaches which search for a single optimal result, for example by minimising an energy functional [5]. Optimising the method for a particular application is then performed by modifying the energy functional or its parameters [6]. Our approach avoids the difficulty of modifying the segmentation criterion by considering the utility function directly.

This work was supported by the Academy of Finland, decision 271394.

J. Pajarinen and V. Kyrki are with the Department of Electrical Engineering and Automation, Aalto University, Finland

E-mail: {Joni.Pajarinen, Ville.Kyrki}@aalto.fi



(a) Segmented image

(b) Object compositions

(c) Robot moves object

Fig. 1. Overall system. (a) Segment an image. (b) Create probability distribution over object compositions from segments. (c) Instead of using a single object composition for decision making, the robot chooses the action which maximises the utility w.r.t. the whole probability distribution over object compositions.

The major contributions of this paper are threefold: (i) We present the novel idea that instead of the single most likely segmentation, it is preferable to make decisions based on the distribution of segmentations; (ii) we present a Markov chain Monte Carlo procedure that produces approximately exact, independent samples from the distribution; and (iii) we present experiments with a robotic system that demonstrate improved decisions with the proposed approach. The operation of the system, illustrated in Fig. 1, consists of a preliminary over-segmentation of an RGB-D image (Fig. 1a), creation of a distribution of object composition hypotheses based on the initial segmentations (Fig. 1b), and maximisation of the expected utility of an action over the distribution to choose a robotic action (Fig. 1c).

II. RELATED WORK

Image segmentation is a widely and actively studied research field [7], [8], [9], [10], [11]. Most of the earlier work concentrates on segmenting grey and color 2D images [7], [8]. With new cheap 3D-sensors research on segmenting 3D images has seen high activity lately [12], [13], [14], [11], especially in robotics, where 3D information can be a necessity.

We are not aware of previous work on using the complete probability distribution over object compositions for decision making. Previous research exists on utilising a probability distribution over segmentations to find the best segmentation [15], [16]. The robotic system presented in [16] utilises a probability distribution over segmentations to gather information about the most likely segmentation. The goal in [16] is to reduce segmentation uncertainty through active exploratory actions such as moving the camera and poking objects. [17] localises 3D objects in an RGB-D image by matching the best 2D segmentation hypotheses to the depth map inside a bounding box. In the task of localising objects, [17] shows the usefulness of using several 2D hypotheses instead of only one. In another line of work, [18] uses several different *temporal* segmentations of RGB-D video to optimise labelings of human activities. Many robotic systems base their decisions on segmented images. In [19], in the task of clearing a pile of objects, a robotic system creates at each time step a segmentation hypothesis about possible objects and attempts to verify an object hypothesis by interaction (poking and pushing). In specific applications, object segmentation is not always needed for decision making. For example, grasps can be selected based on features [20] computed directly from an image.

A common technique for finding the best object composition is through graph cuts [9], [21], [11]. In this paper, we estimate the probability distribution over object compositions using a Markov chain Monte Carlo (MCMC) procedure. Prior work on applying MCMC to finding a single best segmentation exists; for example, in the task of segmenting humans from video frames using human shape models [22], or for segmenting 2D images [23]. Because of the inherent ambiguity in segmentation the authors also present in [23] a technique for selecting a fixed amount of distinct 2D segmentations, instead of only the most likely segmentation.

This paper is based on our earlier work [24] which formalises planning manipulation actions over different object compositions over several time steps as a partially observable Markov decision process (POMDP). In [24] the idea of sampling over compositions along the lines of Algorithm 1 is also briefly presented. This paper extends the work by analysing the convergence of the Markov chain as well as presenting new procedures that aim to generate exact, independent samples. These give theoretical and practical insights needed to apply the idea in a variety of contexts. Moreover, the experimental set-up in this paper is different, aiming to analyse directly if decision making over the distribution of compositions is preferable to the alternative of using the best single composition.

III. UNCERTAIN SEGMENTATION

Many systems first segment an image, then form an object model from the segmentation, and finally, based on this model, decide on the next action. Usually, the goal in segmentation is to find the most likely object composition $\mathbf{h}^*$ from the segmented image [15], [12], [13], [14], [11], [16]:

$$\mathbf{h}^* = \arg\max_{\mathbf{h}} P(\mathbf{h}) \,. \tag{1}$$

This composition can be used directly to find the action with highest application specific preference, that is, the action a^* which maximises the application specific utility $U(\mathbf{h}, a)$:

$$a^* = \arg\max_{a} U(\mathbf{h}^*, a) \,. \tag{2}$$

In this paper, we instead propose to use the action which maximises the expected utility:

$$a^* = \arg\max_{a} \sum_{\mathbf{h}} P(\mathbf{h}) U(\mathbf{h}, a) \,, \tag{3}$$

where $P(\mathbf{h})$ is a probability distribution over object compositions. In some applications, e.g. when comparing different segmentation algorithms, when a human operator requires a single hypothesis, or under heavy computational constraints, a constant utility function may be desirable and then Equations 2 and 3 yield the same action. In many other applications, Eq. 3 yields a better solution than Eq. 2. For example, an autonomous vehicle usually does not care about immobile objects further away from the road but the vehicle should assign a high cost for failing to detect pedestrians close to the road. Similarly, a robot pouring *hot* coffee into a cup for a human should assign a high cost to spilling coffee and take into account the whole distribution of segmentations. In Section IV, in the robotic task of moving toys away from a table, we demonstrate that the approach based on Eq. 3 outperforms an approach based on Eq. 2.

To generate object compositions in a computationally efficient way, we first segment the image into pixel patches (= segments, see Fig. 1a), and then combine the segments into object compositions which consist of object hypotheses (see Fig. 1b). Previous work on segmenting an image and then combining the segments into a single object composition immediately [11], or through interaction [15], [16], exists. We instead maintain a probability distribution over object compositions and make decisions based on the probability distribution.

More formally, denote with $\delta_{i,j}$ whether segments i and j are directly (physically) connected: $\delta_{i,j} = 1$ and $\delta_{i,j} = 0$ denote direct and no direct connection, respectively. Denote with δ all possible direct connections. Moreover, denote with $c_{i,j} = 1$ when segments are part of the same object and with $c_{i,j} = 0$ when not. Denote with $\mathbf{h} = (h_1, \ldots, h_N)$ an object composition, where h_i is an object hypothesis. An object hypothesis h_i consists of a set of directly connected segment pairs. All segments belonging to the same object hypothesis are either directly or indirectly connected, that is, $c_{i,j} = 1$ for all segment pairs i, j which are part of the same object hypothesis. Note that our sampling procedure in Algorithm 1 in Section III-A uses these direct and indirect connections to estimate the probability of a segment pair connection.

$\mathbf{h}$ has worst case dimensionality of $2^{N^2/2}$ w.r.t. the number of segments N. In practice, the dimensionality may be lower because segments with only "air" between them

cannot be directly connected. Often, in real-world scenes, the dimensionality of $\mathbf{h}$ is a product of the dimensionality of disconnected groups of segments $\prod_i 2^{N_i^2/2}$, where N_i is the number of segments in a segment group. For exact computation this is still intractable, and therefore we use an approximate particle representation for the probability distribution over object compositions: $P(\mathbf{h}) = \sum_i w_i \mathbf{h}_i$, where $\sum_i w_i = 1$ and $w_i \geq 0 \; \forall i$.

A. Markov chain Monte Carlo

In order to generate the particle based probability distribution over object compositions which can be used as a basis for decision making, we utilise Gibbs sampling (also known as Glauber dynamics) [25], [26], [27]. We randomly sample direct connections one connection at a time. We will first discuss how a new Markov chain state is sampled, then show that the proposed Markov chain is ergodic and converges to a unique distribution for non-deterministic connection probabilities, and finally present a sampling procedure that aims to generate exact, independent samples.

Our sampling technique for generating a new Markov chain state takes advantage of the fact that evaluating the probability for a single segment connection is fast because we only need to consider local segment connections. The sampling technique consists of two steps: 1) select randomly two segments i and j which may be directly connected, 2) sample the direct connection from the probability distribution, which is estimated by assuming the direct connection is disabled and by keeping other direct connections fixed to their current values. When i and j are indirectly connected, that is, part of the same object through some other connections, the probability for the direct connection between i and j depends only on the prior probability of i and j being part of the same object because connecting i and j would not change which object hypothesis other segments would belong to. When i and j are not already part of the same object, the probability for the direct connection depends on the probabilities between the segment sets U and V which connecting i and j would connect into the same object hypothesis. Fig. 2 illustrates this.

Algorithm 1 defines formally how to sample a new object composition $\mathbf{h}^*$, when given the current object composition $\mathbf{h}$ and two random numbers w and q. The algorithm first samples a possible direct connection, then on lines 4 and 5 determines the segment sets U and V which the direct connection would connect. On lines 6, 7, the algorithm computes the probability for the segment sets U and V to belong to the same object hypothesis when i and j are connected and when not. Assuming an uniform direct connection prior, line 9 computes the direct connection probability, and line 9 finally samples the direct connection.

a) Ergodicity of the Markov chain.: When the connection probability $P(c_{i,j})$ for any two segment patches is non-deterministic $0 < P(c_{i,j}) < 1$, the Markov chain generated by Algorithm 1 is ergodic and converges to an unique distribution. Because $P(c_{i,j})$ is non-deterministic the probabilities on lines 6, 7, and 8 are non-deterministic, and

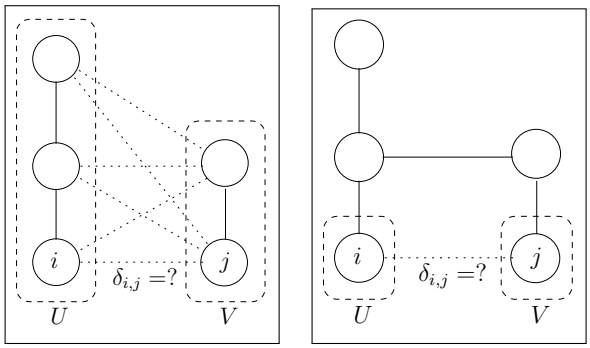


Fig. 2. Effect of indirect connections on the connection probability between segments i and j. A circle denotes a segment and a solid line denotes a connection between segments. Dotted lines denote which connection probabilities are used for sampling the connection between i and j. U and V denote the sets of segments directly or indirectly connected to i and j, respectively. (a) Because i and j are not indirectly connected we have to consider the connection probabilities between segments that will become part of the same object, that is, we have to take into account the connection probabilities between all segments in the sets U and V. (b) Because i and j are already indirectly connected we consider only the probability of the direct connection between i and j.

1 $\mathbf{h}^*$ = `Sample(`$\mathbf{h}$,w,q`)`
Input: Composition $\mathbf{h}$, random values w and q
Output: New composition $\mathbf{h}^*$
2 $\delta_{i,j} \leftarrow$ The wth direct connection
3 $\hat{\mathbf{h}} \leftarrow \mathbf{h}$ so that $\delta_{i,j} = 0$
4 $U \leftarrow \begin{cases} i & \text{if } c_{i,j} = 1 \text{ in } \hat{h} \\ i \cup \{u | c_{i,u} = 1\} & \text{if } c_{i,j} = 0 \text{ in } \hat{h} \end{cases}$
5 $V \leftarrow \begin{cases} j & \text{if } c_{i,j} = 1 \text{ in } \hat{h} \\ j \cup \{v | c_{j,v} = 1\} & \text{if } c_{i,j} = 0 \text{ in } \hat{h} \end{cases}$
6 $P(\hat{h} | \delta^*_{i,j} = 1) \leftarrow \frac{1}{Z(U,V,\hat{h}_k)} \prod_{u \in U} \prod_{v \in V} P(c_{u,v} = 1)$
7 $P(\hat{h} | \delta^*_{i,j} = 0) \leftarrow \frac{1}{Z(U,V,\hat{h}_k)} \prod_{u \in U} \prod_{v \in V} P(c_{u,v} = 0)$
8 $P(\delta^*_{i,j} = 1 | \hat{h}) \leftarrow \frac{P(\hat{h} | \delta^{i,j}_{k+1} = 1)}{P(\hat{h}_k | \delta^{i,j}_{k+1} = 1) + P(\hat{h}_k | \delta^{i,j}_{k+1} = 0)}$
9 $\delta^*_{i,j} \leftarrow \begin{cases} 0 & \text{if } P(\delta^*_{i,j} = 1 | \hat{h}) \leq q \\ 1 & \text{if } P(\delta^*_{i,j} = 1 | \hat{h}) > q \end{cases}$

Algorithm 1: Sample new object composition.

because we randomly select the direct connection to consider, Algorithm 1 enables or disables any direct connection with non-zero probability. Therefore, the Markov chain is ergodic and converges to an unique distribution in the limit. Note that because of the inherent uncertainty in segmentation the condition $0 < P(c_{i,j}) < 1$ usually applies, e.g. in the experiments in Section IV.

b) MCMC procedure.: We would like to have our MCMC approach produce automatically independent samples from the correct distribution. Our MCMC approach first aims to get an exact sample, that is, a sample from the correct probability distribution, then continue sampling until having enough independent samples ($\mathbf{H}$ in Algorithm 2). Algorithm 2 shows the proposed MCMC approach. Because the Markov chain state is a discrete combination of binary

$\mathbf{H}$ =Compositions (N_{ESS},N_{START},$|\mathbf{H}|$)
Input: ESS target N_{ESS}
Output: Compositions $\mathbf{H}$
$\{\mathbf{h_1}, T\} \leftarrow$ CFTP (N_{START})
$t \leftarrow 1$, $\mathbf{H} \leftarrow \mathbf{h_1}$
while $((\text{ESS}_{min}(\mathbf{H}) < N_{\text{ESS}})$ AND
$(T < T_{\text{MAX}}))$ **do**
while $t < T$ **do**
$\mathbf{h_{t+1}} \leftarrow$ Sample ($\mathbf{h_t}$,w,u)
$\mathbf{H} \leftarrow \{\mathbf{H}, \mathbf{h_{t+1}}\}$
$t \leftarrow t + 1$
end
$T \leftarrow 2T$
end
$\mathbf{H} \leftarrow$ Prune $\mathbf{H}$ evenly to size $|\mathbf{H}|$

Algorithm 2: Sample a set of object compositions.

$\{\mathbf{H_T}, T\}$ = CFTP (N_{START})
Input: # of start compositions N_{START}
Output: Composition $\mathbf{H_T}$ at time T
$\mathbf{H}_{\text{INIT}} \leftarrow \{\{\mathbf{h_1}|\delta = 1\}, \{\mathbf{h_2}|\delta = 0\},$
$\{\mathbf{h_3}, \ldots, \mathbf{h}_{\mathbf{N}_{\text{START}}}|\delta = \text{random}\}\}$
$T \leftarrow 1$
repeat
$T \leftarrow 2T$, $\mathbf{H_T} \leftarrow \mathbf{H}_{\text{INIT}}$
$w_T, \ldots, w_{T/2} \leftarrow$ Random
$u_T, \ldots, u_{T/2} \leftarrow$ Random
for $t \leftarrow T$ ***to*** *1* **do**
$\mathbf{H_{T-t+1}} \leftarrow \emptyset$
foreach $\mathbf{h_{T-t}} \in \mathbf{H_{T-t}}$ **do**
$\mathbf{H_{T-t+1}} \leftarrow \mathbf{H_{T-t+1}} \cup$
Sample ($\mathbf{h_{T-t}}$,w_t,u_t)
end
end
until $|\mathbf{H_T}| = 1$

Algorithm 3: Coupling from the past (CFTP).

variables, each variable denoting whether two segments are directly connected, we could use the coupling from the past (CFTP) [28] technique to get exact samples. The basic idea of CFTP is to run Markov chains starting from each possible state with the same random numbers, starting further back in time, until the chains collapse (see [28] for why). For monotone [28] and anti-monotone [29] Markov chains only two starting states are needed. However, our chain is not monotone nor anti-monotone. Because of the large number of states we start CFTP from a limited dispersed set of states: the all connected, all disconnected, and from a fixed number of randomly selected states. We can make the collapsed sample more likely to be exact by increasing the number of starting states: when the starting states cover the whole state space the collapsed sample will be exact [28]. Algorithm 3 shows the CFTP procedure we use. In the experiments, we used 100 (N_{START} in Algorithm 2 and Algorithm 3) starting states.

After CFTP, we start the actual sampling from the collapsed sample, and double the sampling horizon until having enough independent samples. We use the minimum of the effective sample size (ESS) [30] over all possible direct connections (N_{ESS} in Algorithm 2) as an lower bound estimate for the number of independent samples. Sampling stops when the estimate for independent samples is large enough. We also use a hard limit on the number of generated samples (T_{MAX} in Algorithm 2).

IV. EXPERIMENTS

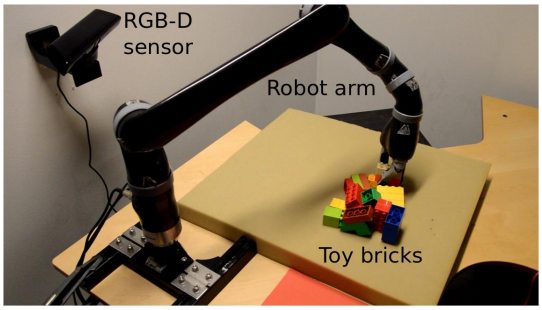
We compare decision making based on the most likely object composition to decision making based on the probability distribution over object compositions in a robotic manipulation task. Note that in the experiments, the goal is not to find correct segmentations but to make correct decisions. The experiments test whether maximising the expected utility instead of making decisions based only on the most likely segmentation increases performance in practice. We are not aware of previous approaches that make decisions based on maximising the expected utility over object compositions. Fig. 3 shows the overall experimental setup. In the setup, a Kinect RGB-D sensor captures images of the scene and a 6-DOF robotic Kinova Jaco arm tries to move as many toy bricks away from the table as possible. Because we do not assume any prior information in advance, including geometric or colour information, and because the bricks are in a pile, segmenting the bricks correctly is difficult. For clearly separated known objects one could possibly use standard segmentation methods. At each time step, the RGB-D sensor captures an RGB-D image of the scene, and the system segments the RGB-D image into patches and computes a prior probability for each patch pair to belong to the same object using the approach in [11]. In more detail, [11] groups neighbouring pixels into clusters and fits planes and B-splines onto the patches to get parametric models (see Fig. 1a for examples of segmented patches). [11] computes for each patch pair a set of features based on the texture, distance of the patches from each other, and other properties. Finally, [11] inputs the computed features into a support vector machine (SVM), trained with a labeled set of household items which differ from the items in our experiments, and scales the output into a probability indicating whether the patches belong to the same object. We use the approach of [11] to compute prior probabilities $P(c_{i,j})$ for all segment/patch pairs, where $P(c_{i,j} = 1)$ defines the prior probability for i and j to be part of the same object. Note that, in place of the segmentation approach we currently use [11], our approach can use also other approaches to oversegment and estimate patch pair probabilities.

We use $P(c_{i,j})$ in the MCMC procedure in Algorithm 2 and compute a probability distribution over object compositions (see Fig. 1b for examples). The number of MCMC samples should be chosen according to the computational budget. In the experiments, we used $|\mathbf{H}| = 2000$ samples. Because the minimum lower bound estimate ESS underestimates the real

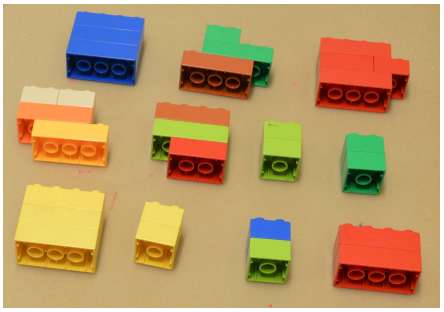
ESS it should be lower than the number of samples: we used a target ESS of $N_{\text{ESS}} = 200$, a tenth of the number of samples. The number of CFTP starting states influences the independence of the first sample w.r.t. the starting state. We used $N_{\text{START}} = 100$ CFTP starting states in the experiments. The hard limit for the number of samples generated was $T_{\text{MAX}} = 131072$. For computing grasps, and for estimating the grasp success probability (without taking grasp history into account), we used the top-down grasping approach in [24]. The approach in [24] estimates the grasp parameters, grasp spread (distance between opposing fingers), robot hand rotation, and the grasp centroid, from the point cloud representing the object. We labeled grasps valid when the spread did not exceed 4cm.

In the experiments, we shook a box containing toy bricks shown in Fig. 3b and emptied the bricks into a specific area on a table. Fig. 4 shows the 10 random scenes for each method ordered so that the first scene produced highest utility and the last scene the lowest utility. The goal was to move a toy brick in each time step away from the table. We now describe the experimental application using the terms introduced for the general framework in Section III. In the application, action a specifies the object to grasp and move away. The robot is rewarded 1 for a successfully grasped and moved object and 0 otherwise. The utility function $U(\mathbf{h}, a)$ is thus directly proportional to the probability of successfully grasping a in object composition $\mathbf{h}$. We compare two methods. The first one, called "Best composition", corresponds to Eq. 2. "Best composition" finds first the most likely object composition $\mathbf{h}^*$, and then finds the action a that maximises the utility function $U(\mathbf{h}^*, a)$. In this application, Eq. 2 tries to grasp the object which has the highest grasp success probability in the most likely object composition. The second method, called "Best object", corresponds to Eq. 3. "Best object" tries to maximise the expected utility $\sum_{\mathbf{h}} P(\mathbf{h})U(\mathbf{h}, a)$ over all possible object compositions. In this application, Eq. 3 tries to grasp the object which has the highest grasp success probability weighted by the probability of the object to exist in an object composition.

c) Results & discussion.: Fig. 5 shows the number of successful moves (a maximum of six moves per scene) in 10 experimental runs for each method. One complete object movement operation, including image processing, segmenting, generating the particle based probability distribution, and moving an object, took on the average 79.9s of which our MCMC approach took 8.8s (11%). The time needed for MCMC depends on the number of particles and CFTP starting states and can be adjusted. The "Best object" method performed significantly better than "Best composition" (the p-value was 0.029 in the Mann-Whitney U test [31]). To qualitatively compare the methods we recorded decisions by both, although only one method operated the robot arm in each scene, that is, we ran one method and at the same time output the decisions which the other method would have made for the same object compositions. In the scenes in Fig. 4a, , even though graspable objects were still available, "Best object" would have finished execution early 3 times



(a) Overall experimental setup



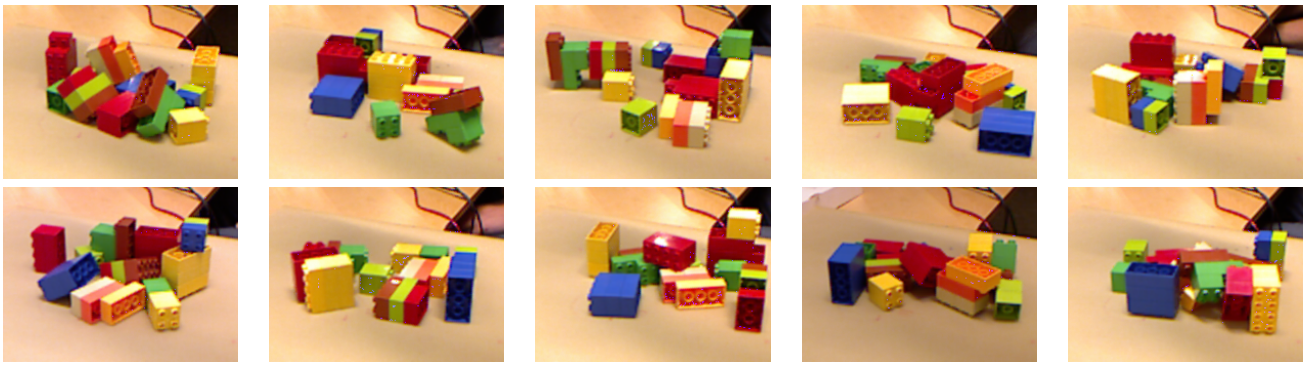
(b) Toy bricks used

Fig. 3. In the experiments, we use an RGB-D sensor for visual input and a 6-DOF Kinova Jaco arm for grasping randomly placed toy bricks.

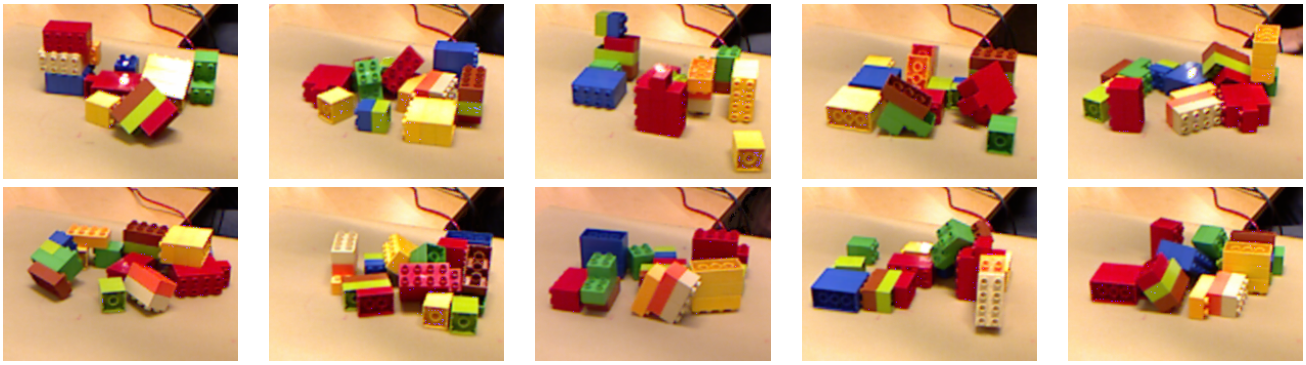
and "Best composition" finished early 10 times, that is, in every scene, and in the scenes in Fig. 4b "Best object" finished execution early 3 times and "Best composition" would have finished early 23 times. The most likely composition was often missing graspable objects that were part of other object compositions. Fig. 6 shows an example of one such situation. Fig. 6a shows under-segmentation happening sometimes. In general, it is better to over-segment too heavily than under-segment but this applies to all over-segmentation approaches including the over-segmentation approach utilised by the two comparison methods. Grasps were sometimes successful even when the segmentation of the grasped object did not correspond to a real object. For example, the robot sometimes grasped the segmented top of an object and moved the complete object successfully. The robot can achieve higher performance because our utility function did not include unnecessary constraints. For applications, such as moving fragile objects, the utility function can penalize grasping an incorrectly segmented object if this could lead to dropping the object.

V. CONCLUSIONS

In this paper, we argued that image segmentation should be considered in the context it is applied in. When used as part of a vision system most current approaches try to determine the best segmentation (Eq. 1) and then use the result for decision making. However, usually the best segmentation is application specific (Eq. 2). Moreover, in applications with non-uniform utility the probability distribution over possible segmentations (Eq. 3) performs better than a single segmentation. To make computations with the complete probability



(a) Random scenes in "Best composition" evaluations, ordered according to experimental success from best to worst



(b) Random scenes in "Best object" evaluations, ordered according to experimental success from best to worst

Fig. 4. Cropped kinect RGB images of the 20 randomly generated scenes.

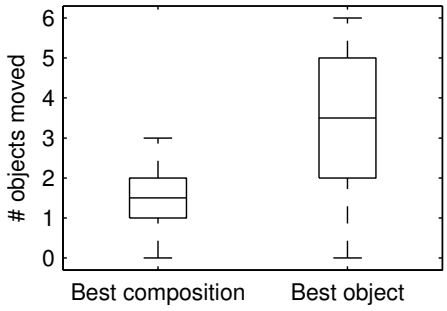


Fig. 5. Results for the robot moving toy bricks in random scenes (please, see the text for further details). The box plot in the figure describes the number of successful moves in 10 experimental runs for each method. The "Best object" method performed significantly better than "Best composition" (the p-value was 0.029 in the Mann-Whitney U test [31]).

distribution feasible, we provided an MCMC procedure for estimating the distribution.

In a task of moving toys from a table with a robot arm, our probability distribution based approach outperformed an approach based on the most likely object composition. Analysis showed that the most likely composition did not always contain objects which could have been moved and which were present in other less likely compositions.

The ideas in this paper could be applied in many different applications that use image segmentation for decision making. For example, a robot pouring hot coffee into a cup for a human should take into account all possible segmentations of the scene to prevent accidents. In the future, support for modelling moving objects could be added to the proposed approach using particle filtering.

References

[1] J. Malik, S. Belongie, T. Leung, and J. Shi, "Contour and texture analysis for image segmentation," *International Journal of Computer Vision*, vol. 43, no. 1, pp. 7–27, 2001.

[2] K. S. Fu, R. C. Gonzalez, and C. G. Lee, *Robotics: Control, Sensing, Vision, and Intelligence*. McGraw-Hill, 1987.

[3] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2012, pp. 3354–3361.

[4] I. Barbosa, M. Cristani, A. Del Bue, L. Bazzani, and V. Murino, "Re-identification with RGB-D Sensors," in *Computer Vision ECCV 2012. Workshops and Demonstrations*, ser. Lecture Notes in Computer Science. Springer, 2012, vol. 7583, pp. 433–442.

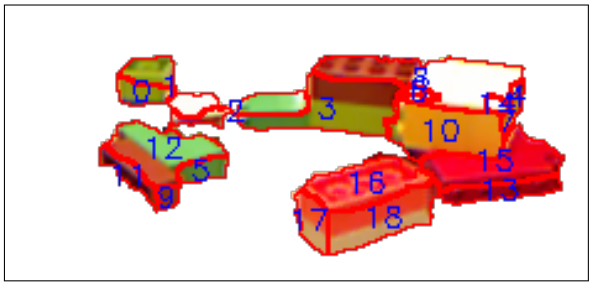
[5] Y. Boykov and G. Funka-Lea, "Graph cuts and efficient N-D image segmentation," *International Journal of Computer Vision*, vol. 70, no. 2, pp. 109–131, 2006.

[6] S. Vicente, V. Kolmogorov, and C. Rother, "Graph cut based image segmentation with connectivity priors," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2008, pp. 1–8.

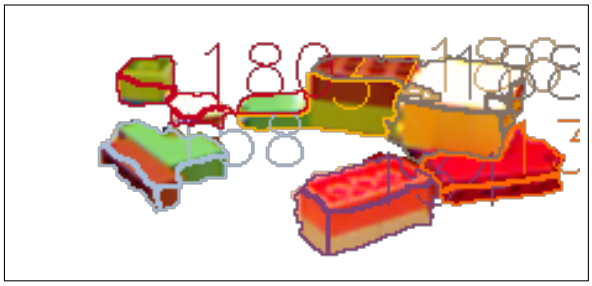
[7] R. M. Haralick and L. G. Shapiro, "Image segmentation techniques," *Computer vision, graphics, and image processing*, vol. 29, no. 1, pp. 100–132, 1985.

[8] N. R. Pal and S. K. Pal, "A review on image segmentation techniques," *Pattern recognition*, vol. 26, no. 9, pp. 1277–1294, 1993.

[9] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.



(a) Segmented patches



(b) Most likely composition

Fig. 6. "Best object" is able to move an object when "Best composition" fails. Time step 6 in the sixth scene in Fig. 4b: (a) Segmented patches, (b) the most likely object composition (probability 0.271). "Best object" grasps successfully an object hypothesis consisting of patches 0 and 1. However, "Best composition" finishes because segments 0, 1, and 2 form in (b) object hypothesis 180, which can not be grasped.

[10] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *International Journal of Computer Vision*, vol. 59, no. 2, pp. 167–181, 2004.

[11] A. Richtsfeld, T. Mörwald, J. Prankl, M. Zillich, and M. Vincze, "Learning of perceptual grouping for object segmentation on RGB-D data," *Journal of visual communication and image representation*, vol. 25, no. 1, pp. 64–73, 2014.

[12] K. Lai, L. Bo, X. Ren, and D. Fox, "A large-scale hierarchical multi-view RGB-D object dataset," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2011, pp. 1817–1824.

[13] A. K. Mishra, A. Shrivastava, and Y. Aloimonos, "Segmenting "simple" objects using RGB-D," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2012, pp. 4406–4413.

[14] A. Richtsfeld, T. Mörwald, J. Prankl, M. Zillich, and M. Vincze, "Segmentation of unknown objects in indoor environments," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2012, pp. 4791–4796.

[15] D. Beale, P. Iravani, and P. Hall, "Probabilistic models for robot-based object segmentation," *Robotics and Autonomous Systems*, vol. 59, no. 12, pp. 1080–1089, 2011.

[16] H. van Hoof, O. Kroemer, and J. Peters, "Probabilistic segmentation and targeted exploration of objects in cluttered environments," *IEEE Transactions on Robotics*, vol. 30, no. 5, pp. 1198–1209, 2014.

[17] B. Kim, S. Xu, and S. Savarese, "Accurate localization of 3D objects from RGB-D data using segmentation hypotheses," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2013, pp. 3182–3189.

[18] H. S. Koppula, R. Gupta, and A. Saxena, "Learning human activities and object affordances from RGB-D videos," *The International Journal of Robotics Research*, vol. 32, no. 8, pp. 951–970, 2013.

[19] D. Katz, M. Kazemi, J. A. Bagnell, and A. Stentz, "Clearing a pile of unknown objects using interactive perception," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2013, pp. 154–161.

[20] D. Fischinger, M. Vincze, and Y. Jiang, "Learning grasps for unknown objects in cluttered scenes," in *Proceedings of the IEEE International Conference on of Robotics and Automation (ICRA)*. IEEE, 2013, pp. 609–616.

[21] Y. Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in ND images," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, vol. 1. IEEE, 2001, pp. 105–112.

[22] T. Zhao and R. Nevatia, "Bayesian human segmentation in crowded situations," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2. IEEE, 2003.

[23] Z. Tu and S. Zhu, "Image segmentation by data-driven Markov chain Monte Carlo," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 657–673, 2002.

[24] J. Pajarinen and V. Kyrki, "Robotic manipulation in object composition space," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2014.

[25] G. Casella and E. I. George, "Explaining the Gibbs sampler," *The American Statistician*, vol. 46, no. 3, pp. 167–174, 1992.

[26] D. J. C. MacKay, *Information theory, inference and learning algorithms*. Cambridge university press, 2003.

[27] D. A. Levin, Y. Peres, and E. L. Wilmer, *Markov chains and mixing times*. American Mathematical Society, 2009.

[28] J. G. Propp and D. B. Wilson, "Exact sampling with coupled Markov chains and applications to statistical mechanics," *Random structures and Algorithms*, vol. 9, no. 1-2, pp. 223–252, 1996.

[29] O. Häggström and K. Nelander, "Exact sampling from anti-monotone systems," *Statistica Neerlandica*, vol. 52, no. 3, pp. 360–380, 1998.

[30] A. Gelman, J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari, and D. B. Rubin, *Bayesian data analysis*. CRC press, 2013.

[31] H. B. Mann and D. R. Whitney, "On a test of whether one of two random variables is stochastically larger than the other," *The Annals of Mathematical Statistics*, vol. 18, no. 1, pp. 50–60, 1947.