Active 3D Segmentation through Fixation of Previously Unseen Objects

Mårten Björkman
celle@csc.kth.se
Danica Kragic
danik@csc.kth.se

CSC - CAS/CVAP
Royal Institute of Technology (KTH)
Stockholm, Sweden

Abstract

We present an approach for active segmentation based on integration of several cues. It serves as a framework for generation of object hypotheses of previously unseen objects in natural scenes. Using an approximate Expectation-Maximisation method, the appearance, 3D shape and size of objects are modelled in an iterative manner, with fixation used for unsupervised initialisation. To better cope with situations where an object is hard to segregate from the surface it is placed on, a flat surface model is added to the typical two hypotheses used in classical figure-ground segmentation. The framework is further extended to include modelling over time, in order to exploit temporal consistency for better segmentation and to facilitate tracking.

1 Introduction

Recent results of object recognition and classification show impressive performance in rather complex environments. Apart from the underlying techniques, their performance is based on the type and complexity of *a-priori* information used for training and model estimation. Similarly, in the area of object tracking, *a-priori* information about shape or appearance facilitates reliable estimation. However, to date there has been relatively little work on generating hypotheses about previously unseen objects in general scenes (*object discovery*), or work that performs tracking of previously unseen objects, when no prior information is available. The former is closely related to attention [11, 15] and segmentation [7, 14, 19] which have classically been performed in the spatial domain. The latter has recently been demonstrated in notable contributions of [2, 6]. However, none of the above mentioned techniques demonstrates an integrated system where segmentation and tracking are performed in 3D.

Our goal is to develop techniques that enable not only detection and tracking of objects, but also *object attribution* in terms of 3D properties, such as the shape and size. In addition, we want to understand natural scenes and relations between objects. Since objects are assumed previously unseen, there are no models of appearance or shape to direct segmentation. This means that objects have to be discovered and attributed in a bottom-up manner. We define an object as something that occupies a portion of 3D space, and resembles some continuity in appearance and shape.

General scene understanding and multiple object detection relates to the classical problem of figure-ground segmentation. It has traditionally been viewed as a process involving

BMVC 2010 doi:10.5244/C.24.119

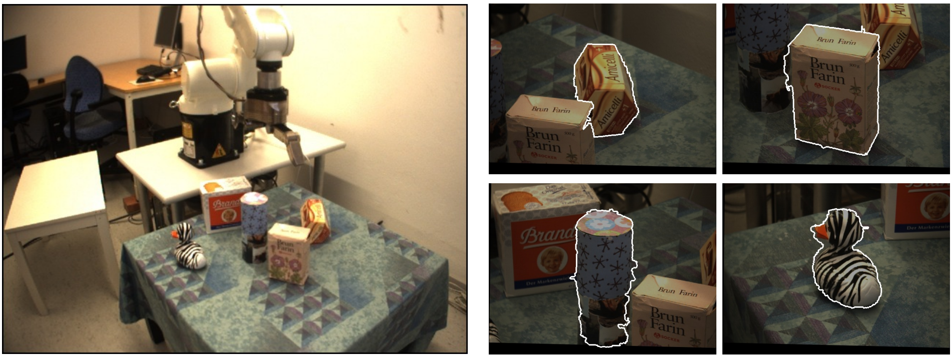

Figure 1: A wide field view in which object hypotheses are searched using attention (left) and segmentations of each found hypothesis in the foveated view (right).

two different components; an object-like foreground (figure) surrounded be a less discriminant background (ground) [20]. The problem is more complicated when the background involves many different objects, some of which may be more discriminant than the object of interest or when objects are in close contact. To resolve this, different strategies have been applied in computational systems, *e.g.* imposing symmetry constraints on the objects or assuming objects to be placed on a uniformly coloured table top [3, 12]. When an object is placed on a multi-coloured table top, with no appearence model associated, the segmented foreground region often covers, not just the object itself, but also parts of the table.

Tracking by modelling distributions of foreground and background pixels has been popular in particular for tracking of people and vehicles. Raja *et al.* [18] use Gaussian Mixture Models updated with Expectation-Maximization (EM) for tracking of people, whereas Perez *et al.* [16] relies on histograms and particle filters to better handle clutter and occlusions. Comaniciu and Meer [7] initially used mean-shift density maximisation [8] to find colour clusters for segmentation and later adopted the same technique to the spatial domain for tracking with weighting based on colour histogram similarities. Our first contribution is an approximate EM algorithm that, while marginalising over all unknown labels, also takes the dependencies between neighbouring labels into consideration.

Level set methods have successfully been applied for joint object segmentation and tracking, representing a tracked object by a closed contour. Chockalingam *et al.* [6] use mixtures of unimodal fragments in feature-spatial domain for representation. To cope with large motions and deformations, fragments are guided by a Lucas-Kanade tracker prior to level set segmentation. Bibby and Reid [2] instead use colour histograms for representation and performs segmentation with pixel-wise posteriors. By only updating pixels close to the boundary, a foreground object can be tracked at a very high rate. In this paper we propose the use of fixation for unsupervised initialisation and 3D cues for better segmentation. The only, to our knowledge, previous work in this domain is that of Mishra and Aloimonos [14], which suffers from a significantly larger computational cost.

In addition, we propose a flat surface model to improve the segmentation task. This helps us cope with situations when an object is hard to segregate from the surface it is placed on. A flat surface assumption is reasonable, given that most objects in indoor scenes are in fact placed on flat, or at least locally flat, surfaces. We will show that even if no such plane is visible in the scene, foreground segmentation is still possible, since the flat surface model will become just another background model. A further extension is that we let the segmentation evolve over time, this in order to provide more information and gradually improve segmentation, which also facilitates tracking.

2 Scene part modelling and estimation

We propose a segmentation approach based on three different models, each described by a set of parameters; the foreground model θ_f, the background model θ_b and that of the flat surface θ_s. Each pixel has an associated label $l_i \in L = \{l_f, l_b, l_s\}$, depending on which component it belongs to. The model parameters $\theta = \theta_f \cup \theta_b \cup \theta_s$ and the labellings of all pixels $\mathbf{l} = \{l_i\}$ are unknown and need to be estimated from measurements $\mathbf{m} = \{m_i\}$ at each pixel. In the current implementation the measurements $m_i = (p_i, c_i)$ consists of image point positions, binocular disparities $p_i = (x_i, y_i, d_i)$, and colour representation in HSV space $c_i = (h_i, s_i, v_i)$.

Maximum a Posteriori (MAP) estimation of labels $\mathbf{l}$ and model parameters θ has frequently been used for segmentation, which is the case in *e.g.* GrabCut [19]. By alternating between keeping $\mathbf{l}$ and θ fixed, a local maximum of $P(\mathbf{m}, \mathbf{l}|\theta)$ is searched. For cases with only two possible states per label, the MAP estimate can be found exactly using graph-cuts [10]. With more than two labels the problem becomes NP-hard, however efficient and accurate approximation methods exist [5]. For histogram based modelling of likelihoods, it has been shown [21] that objective functions can be expressed in terms of the unknown labelling, which often leads to global optimality using a non-iterative optimisation procedure.

The reason for not using MAP estimation here is that it only delivers model parameters based on one particular set of labels, the set that happens to maximise the posterior. The MAP estimate might be an extreme case, which is not representative of the overall distribution. What frequently occurs in segmentation over time, especially for non-textured regions, are instabilities where the result alternates between two competing local maxima, which leads to radical and unwanted shifts in model parameters.

2.1 Approximated Expectation-Maximisation

Instead of letting a single labelling determine the model parameters, parameters can be estimated by applying marginalisation over all possible labellings, similar to what is done in Expectation-Maximisation (EM). With EM the maximum likelihood estimate of θ is computed iteratively for $P(\mathbf{m}|\theta) = \sum_{\mathbf{l}} P(\mathbf{m}, \mathbf{l}|\theta)$, with iterations that involve two steps. In the first step (E-step) the conditional distribution $w(\mathbf{l}) = P(\mathbf{l}|\mathbf{m}, \theta')$ is computed using the current estimate θ' and in the second step (M-step) a new estimate is found by maximising an objective function $Q(\theta|\theta') = \sum_{\mathbf{l}} w(\mathbf{l}) \log P(\mathbf{m}, \mathbf{l}|\theta)$. Unfortunately, this summation becomes intractable if neighbouring labels are dependent, something we assume in our case, since the total number of labellings is 3^N, where N is the number of pixels.

To make the summation computationally tractable we introduce an approximation. The approximation treats the labels as independent in one of the two steps. We do this by replacing $w(\mathbf{l})$ with the product of all the conditional marginals for each unobserved label, $w(l_i) = P(l_i|\mathbf{m}, \theta')$. Since a measurement m_i depends only on its associated label l_i, the second step becomes a maximisation of

$$Q_1(\theta|\theta') = \sum_i \sum_{l_i \in L} w(l_i) \log P(m_i, l_i|\theta), \tag{1}$$

where the joint probability at each point is given by $P(m_i,l_i|\theta) = P(m_i|l_i,\theta)P(l_i|\theta)$. With a summation done over $3N$ labels, instead of all labellings, this computation becomes feasible.

The conditional marginals $w(l_i)$ are computed with loopy belief propagation [22], where dependencies between 4-neighbours are taken into consideration. However, in order to do this we need to rewrite the problem into an energy minimisation problem. From Bayes' rule and the fact that m_i depends only on l_i, we have

$$P(\mathbf{l}|\mathbf{m},\theta) = \frac{\prod_i P(m_i|l_i,\theta)}{\prod_i P(m_i|\theta)} P(\mathbf{l}|\theta) = \frac{\prod_k P(m_k|l_k,\theta)P(l_k|\theta)}{\prod_k \sum_{l\in L} P(m_k|l_k=l,\theta)} \cdot \prod_i \prod_{j\in N_i} P(l_i,l_j). \tag{2}$$

The network of image points can be viewed as a Markov Random Field (MRF), where the first factor in (2) represents cliques of one point each and the second involves pairs of points and their dependencies. The energy functions used for belief propagation are given by the negative logarithms of these two factors. Like many others [4, 19] we use the Potts model [9, 17] to define the joint probabilities of neighbours $P(l_i,l_j)$. The effect of these dependencies is a smoothing factor that captures the spatial continuity in typical scenes and penalises solutions that are too fragmented.

2.2 Scene part modelling

Foreground points are represented by a 3D ellipsoid and assumed to be normally distributed in spatial-disparity space, $P(p_i|l_i = l_f,\theta) = n(p_i; p_f, \Delta_f)$. Here $n(x;\bar{x},\Delta)$ denotes a normal distribution with mean $\bar{x}$ and covariance Δ. The spatial distributions of points in the background and on the flat surface are assumed to be uniform, *i.e.* $P(x_i,y_i|l_i = l_s,\theta) = P(x_i,y_i|l_i = l_b,\theta) = 1/N$. The background points are represented by a normal distribution in disparity space, $P(d_i|l_i = l_b,\theta) = n(d_i;d_b,\Delta_b)$, while disparities on the flat surface are assumed to depend linearly on the spatial coordinates, *i.e.* $P(d_i|l_i = l_s,\theta) = n(d_i;a_sx_i + b_sy_i + d_s,\Delta_s)$. The last assumption is motivated by the fact that a plane in metric 3D space will be a plane also in spatial-disparity space.

For each scene part we represent pixel colour distributions by 2D histograms using hue and saturation; $p(h_i,s_i|l_i = l_f,\theta) = H_f(h_i,s_i)$, $p(h_i,s_i|l_i = l_s,\theta) = H_s(h_i,s_i)$ and $p(h_i,s_i|l_i = l_b,\theta) = H_b(h_i,s_i)$. With these color histograms included in the set of model parameters, the complete set is given by

$$\theta_f = \{p_f,\Delta_f,c_f\},\ \theta_s = \{a_s,b_s,d_s,\Delta_s,c_s\},\ \theta_b = \{d_b,\Delta_b,c_b\},$$

where c_f, c_b and c_s are the color histogram bins stacked into vectors. All these parameters are estimated by maximising (1) above, using the joint measurement conditionals that can be summarised as

$$\begin{aligned}
P(m_i|l_i = l_f,\theta) &= n(p_i;p_f,\Delta_f)\,H_f(h_i,s_i),\\
P(m_i|l_i = l_s,\theta) &= N^{-1}n(d_i;a_sx_i + b_sy_i + d_s,\Delta_s)\,H_s(h_i,s_i),\\
P(m_i|l_i = l_b,\theta) &= N^{-1}n(d_i;d_b,\Delta_b)\,H_b(h_i,s_i).
\end{aligned}$$

2.3 Adaptations for tracking

To facilitate tracking and exploit the continuous stream of new image data, the parameter updates are extended to be time dependent. This is done by including a transition probability term $P(\theta|\theta^t) = p(\theta_f|\theta_f^t)\ p(\theta_s|\theta_s^t)\ p(\theta_b|\theta_b^t)$, where θ^t is the estimate from the previous frame. The resulting objective function

$$Q_2(\theta|\theta') = \sum_i \sum_{l_i \in L} w(l_i) \log P(m_i, l_i|\theta) + \log P(\theta|\theta^t)$$

is applied similar to $Q_1(\theta|\theta')$ in Section 2.1. In a tracking scenario, we assume that the parameters between consecutive frames will change only slightly. The only exception is the spatial position of the foreground object, where the previous position in $P(\theta|\theta^t)$ is replaced by the one predicted by a Kalman filter.

In order to model the consistency of covariance matrices, here the spread of points for the different scene parts, we use the following function

$$g(\Delta;\Delta^t,S) = \left(\frac{1}{2\pi|\Delta|}\right)^{S/2} \exp\,(-\frac{S}{2}\sum_i \lambda_i \mu_i^\top \Delta^{-1} \mu_i),$$

where μ_i and λ_i are the eigenvectors and eigenvalues of Δ^t, and S is the strength of the dependency. This equation can be interpreted as the probability of drawing S points with a variance of Δ^t, when the underlying distribution has in fact a variance of Δ.

With Λ_f being the expected variance over time of the foreground position and σ_c^2 the variance for each colour bin, the transition probability of the foreground can be summarised as $p(\theta_f|\theta_f^t) = n(p_f; p_f^t, \Lambda_f)\, n(c_f; c_f^t, \sigma_c^2 I)\, g(\Delta_f; \Delta_f^t, S_f)$. The transition probabilities of the surface and background models, $p(\theta_s|\theta_s^t)$ and $p(\theta_b|\theta_b^t)$, can be expressed in a similar manner. This means that the set of parameters governing the transitions is $(\Lambda_f, \Lambda_s, \Lambda_b, \sigma_c^2, S_f, S_s, S_b)$. For the experiments in Section 4 we set the expected variances to Λ_f = diag{10000, 10000, 4}, Λ_b = 100 and Λ_s = diag{0.0001, 0.0004, 1}. We used hue-saturation histograms with 10×10 bins each, with an expected variance of σ_c^2 = 0.0001 per bin. Time consistency values for the covariance matrices were set to $S_f = S_b = S_s = N$, i.e. the number of image points. These are the only free parameters that are kept fixed and never updated.

3 Initialisation through fixation

Critical to any segmentation and tracking system is the initialisation phase. Since there is no implicit way of knowing what should be considered as foreground, a targeted region has somehow to be pointed out, either manually or through some other mean. Unlike systems for off-line image manipulation [19], an autonomous system does not have the luxury of a human operator in the loop. Object detection has been proposed as a mean for initialisation [2, 16], but this implies you have some model of what to detect, which is not possible when working with previously unseen objects. However, in stereo the situation is somewhat different. While image points are densely packed, 3D points appear in clusters. These clusters may serve as bottom-up cues for object detection, regardless of appearance and shape.

Initialisation involves associating the measured 3D points to the three scene parts and determine an initial estimate of the model parameters for these parts. For foreground object targeting, we introduce two assumptions: 1) an object of interest is in the centre of view, and 2) the size of this object is approximately known (10 cm for all objects in the experiments). With these assumptions a high-density 3D point cluster is found using mean-shift filtering [8] in the spatial-disparity domain. Points within this cluster become associated to the foreground. Using random sampling among the remaining points a plane is found in 3D. Points within one disparity value from this plane are assumed to belong to the flat surface model, while all points that remain are initiated to the background. When all points have been distributed among the three scene parts, model parameters can be estimated and the approximated expectation-maximisation loop in Section 2.1 started.

As an experimental platform we use an Armar III robotic head [1], a head with two sets of stereo cameras; a foveal set and a wide field set. To guide the head towards areas of interest, object hypotheses are detected in the wide field view by a modified version of Itti *et al.*'s model of attention [11]. For each such detected hypothesis, the head performs a rapid saccade, placing it in fixation in the centre of the foveal view. This is the starting point for the segmentation process and the motivation for the assumptions mentioned above. The fixation system itself relies on SIFT features [13] that are matched and mean-shift filtered in 3D. Note that even if our particular stereo head changes the vergence angle to constantly keep an object of interest at zero disparity, vergence is not a necessity for segmentation to succeed. However, by placing objects of interest at zero disparity, we guarantee a maximum overlap between the narrow ($\sim 12^{\circ}$) foveal views.

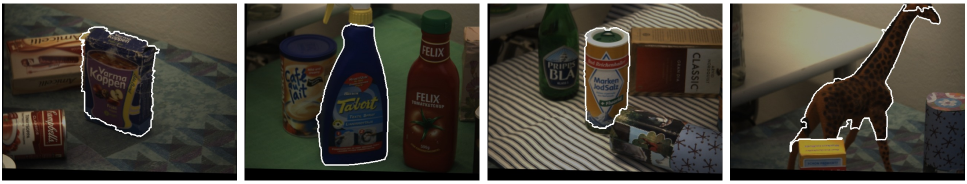

Figure 2: Examples of segmentations from the proposed system.

4 Experimental results

We performed a large series of experiments using the procedure described in previous section. Some examples of the segmentations achieved after 10 updates are shown in Figure 2. The last image in the figure shows a special case where the 3D ellipsoid model turns out to be unsuitable as a representation of the object shape. Even with additional updates the legs of the giraffe are never included in the segment. In Figure 3 the segmentation of a toy cat are shown for the 1st, 5th and 10th updates, including the labelling of pixels on the second row. White areas indicate image points where the foreground model is most probable, whereas black areas represent the background. The remaining grey areas are associated to the flat surface model. Note that some surface pixels behind the cat have been mislabelled as background. The reason is the limited disparity range (64) within which stereo matches are searched. Even if the range is wide compared to what typical matching methods can handle, it is narrower than what often is the case in reality. This is another motivation for fixation, in particular for narrow field cameras.

4.1 The benefit a disparities and a flat surface model

The benefit of the flat surface model, as a complement to the figure and ground models, can be seen on the first row of Figure 4, where the surface model has been disabled. Both the box and the table include different shades of blue and without a flat surface model the foreground model will grow and eventually cover not just the object, but also the top of the table. The reverse occurs when no disparities are used, as shown on the second row. Due to the similarity in colour, the foreground and background priors, $p(l_f|\theta)$ and $p(l_b|\theta)$, have a large influence on the end result. Since the table covers such a large part of the image, the background will start to dominate, until the similarly coloured foreground parts disappears completely. These two cases are examples of instabilities that may occur when there is no strong colour cue to rely on and the success of the initialisation becomes a critical factor.

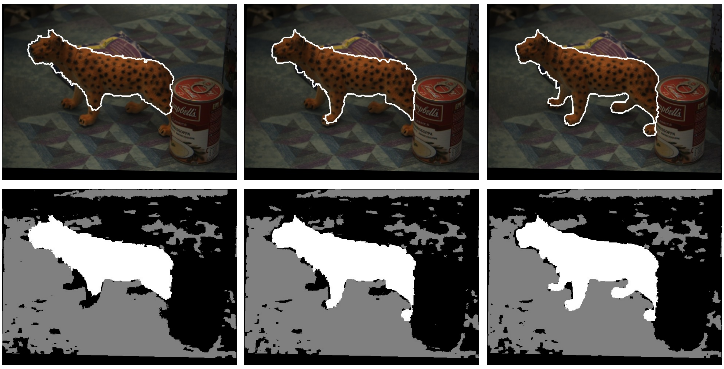

Figure 3: The 1st, 5th and 10th updates of a segmented toy cat. White areas in the bottom row show the foreground, grey the flat surface and black the remaining background points.

When introducing a new assumption to an existing system, there is always a risk that the system will fail, if these assumptions turn out to be invalid. The addition of a third scene part model, that of a flat surface, can be thus questioned, especially when there is no distinct such part in the scene. The behaviour of the presented system under such conditions is illustrated by the images in Figure 5. From the labellings to the right can be concluded that the flat surface and background models have changed order. The system tries to find some imaginary plane and since the "thickness" of the plane can be arbitrary (given by the estimated parameter Δ_s) it finds a plane that intersects all objects, but the object of interest, leaving the segmentation of this object intact.

To further analyse the benefit of a flat surface model, we ran the system under different conditions, with different objects, table tops and illumination, and annotated silhouettes of about 800 fixated objects by hand. Stereo images were collected and a comparison between alternative methods was done off-line. Figure 6 summarises the average F1 scores and distribution of scores for four different alternatives. Results using the full system are

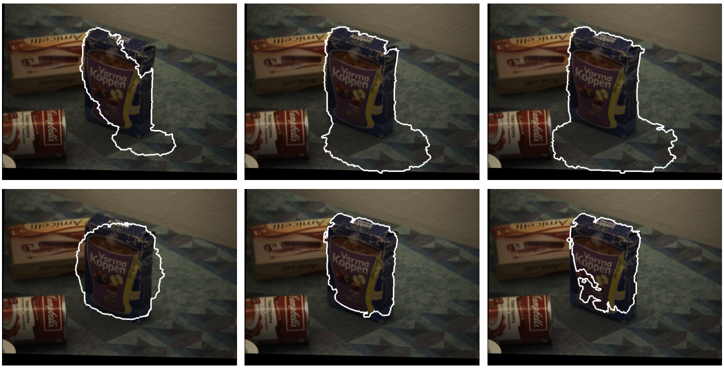

Figure 4: The 1st, 5th and 10th updates of a segmented box, without a flat surface model (upper) and without disparities altogether (lower).

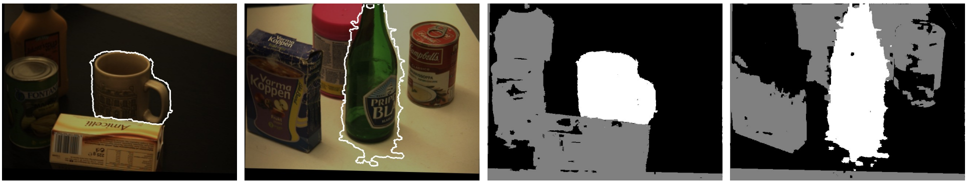

Figure 5: Segmentations of two objects without a textured table top. Images to the right show the foreground (white), flat surface (grey) and background (black) labelling.

	Full system	No surface model	No disparities	GrabCut[19]
Mean F1 score	0.797	0.646	0.616	0.615
F1 > 0.9	45.9%	21.9%	22.1%	10.2%
0.9 > F1 > 0.7	28.7%	23.8%	21.1%	35.7%
0.7 > F1 > 0.5	13.5%	20.8%	24.0%	21.9%
0.5 > F1 > 0.3	9.2%	25.4%	18.2%	16.9%
F1 < 0.3	2.7%	8.1%	14.6%	15.3%

Figure 6: F1 scores obtained using the full system (1st column), without a flat surface model (2nd), without binocular disparities (3rd) and using a modified version of GrabCut (4th).

shown in the first column, without a flat surface model in the second and without disparities in the third. The last column shows the results using a modified version of GrabCut [19]. Instead of letting a human operator frame the objects, a disc with a radius of one fifth of the image height was used. The same disc was applied for the results in the third column. The difference between the last two columns is primarily attributed to the fact that we use a colour model invariant to illumination, whereas GrabCut does not. From the results it can be concluded that disparities become particularly important as cues when combined with a flat surface model, which is true even if 50% of our examples included non-textured table tops.

4.2 Segmentation while tracking

Tracking is in important tool for a mobile robot that, while moving around an object and keeping it in fixation, collects information about the object and the rest of the scene. Areas occluded in one view can be revealed and further information be accumulated, information necessary for action planning and recognition.

The proposed system was adapted for tracking scenarios, with a tracking loop running in $25/3$ Hz. For the sequence of object segmentations shown in Figure 7 we move a table, on which objects are placed, in front of the camera system, while tracking the object currently in fixation[1]. By exploiting the consistency over time, we use a single approximate EM iteration per update. The delay between each image in the sequence is 8 frames, which is equivalent to about a second. As can be seen on the first row, the maximum image motion is about 40 pixels per update, assuming 640×480 pixel images. Since the camera control is limited to a PI controller, there are visible lags for frames with significant acceleration. For frames in which no segmentations appear in the figure, the system has gone from one fixation point to another and is waiting for fixation to stabilise.

The system was implemented on a 2.67 GHz Intel QX6700 CPU, with belief propagation running on a NVidia 8800 GTS GPU. During tracking most time is spent on stereo match-

[1] The full sequence as a movie can be downloaded from http://www.csc.kth.se/∼celle/segm_bmvc.avi

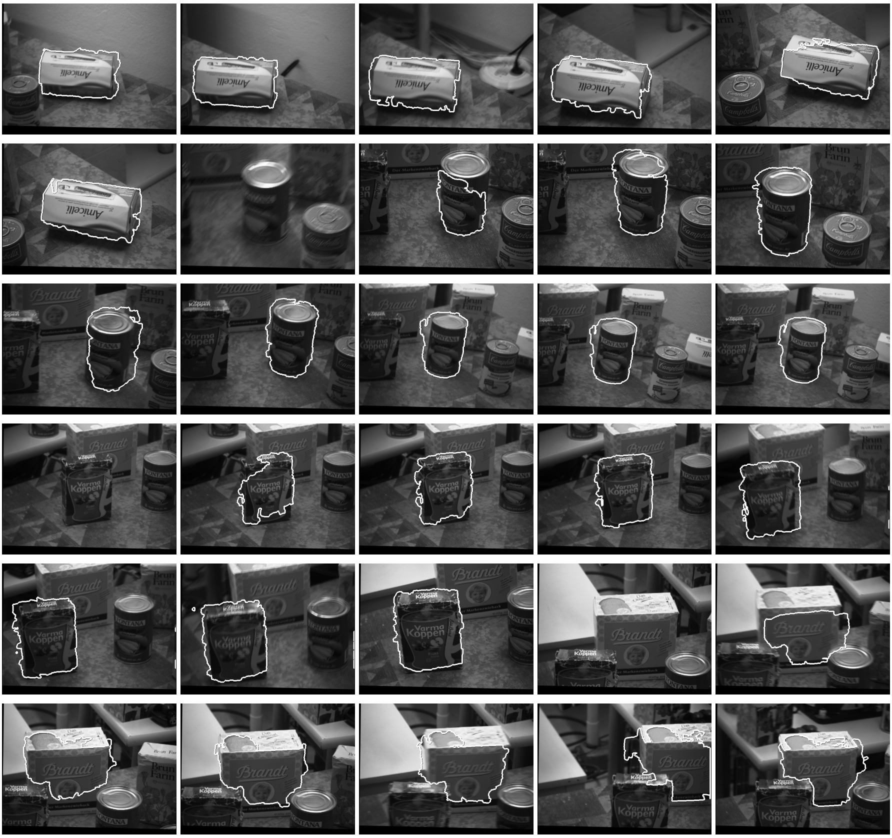

Figure 7: Sequence of segmentations obtained using the proposed system, while tracking objects of interest detected by an attention system. With the system running in around 8 Hz, the delay between each images is about a second.

ing using OpenCV (35 ms), 3D position and colour statistics collection (25 ms) and belief propagation (20 ms). Belief propagation is particularly well suited for implementation on GPUs and about 15 times faster than implementations on a CPU. The passing of messages within a regular network of labels is highly deterministic and there are ways to divide the problem into steps, so that each step involves messages that are independent and can thus be easily parallalised. This is in contrast to graph-cut methods, that are less deterministic both in terms of local operations and number of required passes. Whereas belief propagation can be interrupted after a given number of passes, graph-cuts are useful only once a minimum cut has been found. In our implementation we limit number of the passes to 10 for each approximate EM iteration.

5 Conclusions

We have presented an approach for active object segmentation and tracking based on image point positions, disparities and colour cues. It uses an approximate Expectation-Maximisation algorithm for object modelling and labelling that, while marginalising over all unknown labels, takes the dependencies between neighbouring labels into consideration. The benefits of EM can thus be exploited, without a fragmentation of the segmentation, that would have been the case if dependencies were ignored. Instead of relying on a human operator, fixation is used for unsupervised initialisation of the modelling loop. We have further shown the strength of performing segmentation in 3D space, rather than in image space, especially when combined with a flat surface model that eases the problem of segregating objects from the surfaces they are placed on. The complete system runs in an active framework in which objects can be discovered, segmented and tracked at a rate of about 8 Hz.

The representation of colours should be investigated in greater detail. If the same representation is to be used for long-term object modelling, invariance to illumination is a necessity. This was the motivation for the hue-saturation histograms used in our study. However, for discrimination between objects in a particular scene such invariance becomes less relevant. It would be possible to use a colour representation that varies in invariance depending on its use. The same can be said about the 3D modelling of object shape. It may be possible to vary the level of abstraction, depending on purpose and shape of the object. For manipulation a simple 3D ellipsoid might be satisfactory, while for classification it may not. This leads to a broader question, the question on how to abstract a representation of a scene, when an overwhelming amount of data has been extracted and parts of the scene do not need to be represented with the same level of detail.

References

[1] T. Asfour, K. Regenstein, P. Azad, J. Schröder, and R. Dillmann. Armar-III: A humanoid platform for perception-action integration. In *Proc. International Workshop on Human-Centered Robotic Systems (HCRS)*, 2006.

[2] C. Bibby and I. Reid. Robust real-time visual tracking using pixel-wise posteriors. In *European Conference on Computer Vision (ECCV08)*, pages 831–844, October 2008.

[3] M. Björkman and J-O. Eklundh. Foveated figure-ground segmentation and its role in recognition. In *Proc. British Machine Vision Conference (BMVC05)*, pages 819–828, September 2005.

[4] Y. Boykov and M-P. Jolly. Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images. In *Proc. IEEE International Conference on Computer Vision (ICCV01)*, volume I, pages 105–112, 2001.

[5] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, 2001.

[6] P. Chockalingam, N. Pradeep, and S. Birchfield. Adaptive fragments-based tracking of non-rigid objects using level sets. In *Proc. IEEE International Conference on Computer Vision (ICCV09)*, October 2009.

[7] D. Comaniciu and P. Meer. Robust analysis of feature spaces: Color image segmentation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR97)*, pages 750–755, June 1997.

[8] K. Fukunaga and L. Hostetler. The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on Information Theory*, 21 (1):32–40, January 1975.

[9] D. Geman, S. Geman, C. Graffigne, and P. Dong. Boundary detection by constrained optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7): 609–628, July 1990.

[10] D.M. Greig, B.T. Porteous, and A.H. Seheult. Exact maximum a posteriori estimation for binary images. *Journal of the Royal Statistical Society. Series B (Methodological)*, 51(2):271–279, 1989.

[11] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20 (11):1254–1259, 1998.

[12] W.H. Li and L. Kleeman. Interactive learning of visually symmetric objects. In *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS09)*, pages 4751–4756, 2009.

[13] D. Lowe. Object recognition from local scale-invariant features. In *Proc. IEEE International Conference on Computer Vision (ICCV99)*, volume 2, pages 1150–1157, September 1999.

[14] A. Mishra and Y. Aloimonos. Active segmentation. *International Journal of Humanoid Robotics*, 6:361–386, 2009.

[15] A. Oliva, A. Torralba, M.S. Castelhano, and J.M. Henderson. Top-down control of visual attention in object detection. In *International Conference on Image Processing*, pages 253–256, 2003.

[16] P. Perez, C. Hue, J. Vermaak, and M. Gangnet. Color-based probabilistic tracking. In *European Conference on Computer Vision (ECCV02)*, pages 661–675, June 2002.

[17] R. Potts. Some generalized order-disorder transformation. *Proceedings of the Cambridge Philosophical Society*, 48:106–109, 1952.

[18] Y. Raja, S.J. McKenna, and S. Gong. Tracking and segmenting people in varying lighting conditions using colour. In *Proc. IEEE International Conference on Face and Gesture Recognition*, pages 228–233, 1998.

[19] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics*, 23:309–314, 2004.

[20] E. Rubin. *Visuell wahrgenommene Figuren*. Gyldenalske Boghandel, Copenhagen Denmark, 1921.

[21] S. Vicente, V. Kolmogorov, and C. Rother. Joint optimization of segmentation and appearance models. In *Proc. IEEE International Converence on Computer Vision (ICCV09)*, pages 755–762, 2009.

[22] Y. Weiss. Correctness of local probability propagation in graphical models with loops. *Neural Computation*, 12:1–41, 2000.