Cogn Process (2017) 18:169–182
DOI 10.1007/s10339-017-0791-z

RESEARCH REPORT

A computational framework for attentional object discovery in RGB-D videos

Germán Martín García[1] · Mircea Pavel[1] · Simone Frintrop[2]

Received: 26 June 2015 / Accepted: 12 January 2017 / Published online: 2 February 2017

Abstract We present a computational framework for attention-guided visual scene exploration in sequences of RGB-D data. For this, we propose a visual object candidate generation method to produce object hypotheses about the objects in the scene. An attention system is used to prioritise the processing of visual information by (1) localising candidate objects, and (2) integrating an inhibition of return (IOR) mechanism grounded in spatial coordinates. This spatial IOR mechanism naturally copes with camera motions and inhibits objects that have already been the target of attention. Our approach provides object candidates which can be processed by higher cognitive modules such as object recognition. Since objects are basic elements for many higher level tasks, our architecture can be used as a first layer in any cognitive system that aims at interpreting a stream of images. We show in the evaluation how our framework finds most of the objects in challenging real-world scenes.

Handling Editor: John K. Tsotsos (York University).
Reviewers: Markus Vincze (Vienna University of Technology), Neil Bruce (University of Manitoba).

✉ Germán Martín García
martin@ais.uni-bonn.de

Mircea Pavel
pavelm@iai.uni-bonn.de

Simone Frintrop
frintrop@informatik.uni-hamburg.de

[1] Institute of Computer Science VI, University of Bonn, Bonn, Germany

[2] Computer Vision Group, Department of Informatics, University of Hamburg, Hamburg, Germany

Keywords RGB-D object discovery · Computational visual attention · 3D inhibition of return

Introduction

A computational cognitive system that aims at interpreting a scene by means of visual data is confronted with two big challenges: the first one is how to process in reasonable time the huge amount of perceptual input that continuously arrives via its sensors, e.g. a camera; the second is the complexity of the task itself: how should pixels be grouped into units that are semantically meaningful? An important prerequisite for scene interpretation is the detection of objects in the scene: several findings emphasise the central role of object perception in human vision (Feldman 2003; Pylyshyn 2001).

Biological systems have developed attention mechanisms to cope with the first problem. The biological solution has been to prioritise the processing of perceptual input according to its relevance. For example, the human visual system has attention mechanisms by which only a fraction of all the visual input is processed (Pashler and Sutherland 1998). Concerning the second challenge—how visual data is interpreted in the human visual system—there are several findings from psychology and cognitive science: first, authors such as Scholl (2001) argue that segmentation processes exist on all levels of the visual system that group regions into perceptually coherent units; second, Rensink (2000) proposes a model where these so-called proto-objects are combined by visual attention to form coherent objects; third, it is known that Gestalt principles play an important role in object perception (Wagemans et al. 2012). Furthermore, authors such as Pylyshyn argue that in the human visual system visual

elements are individuated before their properties or categories are known (Pylyshyn 2001).

In this paper, we propose a computational model that combines these individual findings from human vision into one coherent framework. It can serve as a basic architecture that can be used by autonomous systems, e.g. robots, in order to process visual data obtained from a moving camera. Our framework has at its core an attention system to prioritise the processing of visual data. The attention system delivers regions of interest based on saliency computation, and refines these regions using segmentation processes to provide accurate boundaries of potential objects. The resulting object candidates are ranked according to several criteria such as convexity. Finally, a spatial inhibition of return (IOR) mechanism inhibits attended regions to enable a visual exploration of the scene. In contrast to other works (Itti et al. 1998; Palomino et al. 2011), we root the IOR mechanism in spatial 3D coordinates which corresponds to human vision (Posner et al. 1985; Wang et al. 2016) and enables us to deal with camera motion.

An overview of the proposed system is shown in Fig. 1: a camera moves around a scene providing a continuous stream of RGB-D data. In the lower processing stream, the depth information is used to build a 3D map of the scene with the KinectFusion algorithm (Newcombe et al. 2011). In the upper stream, an attention system computes a saliency map (1) and a segmentation of the image (2). Based on these two, object candidates are generated (3). Information about already attended objects is stored in the 3D map, raycasted to the current camera pose (4), and used to inhibit already attended objects (5).

This paper integrates two of our previous approaches into one coherent system for visual scene exploration. In Martín García and Frintrop (2013) we have presented the spatial IOR mechanism, and in Martín García et al. (2015), we introduced a method based on saliency and segmentation to produce visual object candidates. The new system integrates these two approaches and we show in the experiments how the IOR mechanism lets us retrieve most of the objects in the scene with a small number of object candidates per frame.

Related work

Since the related literature spans two different fields, we first present the related work on computational attention systems that have addressed the IOR problem, and second, the relevant work on the topic of object discovery.

Related work in computational attention systems

Many computational attention systems have been built during the last two decades, first for the purpose of mimicking and understanding the human visual system (see survey in Heinke and Humphreys 2004), and second to improve technical systems in terms of speed and quality (see an extensive review in Frintrop et al. 2010). The general structure of attention systems is based on psychological models such as the feature integration theory (FIT) (Treisman and Gelade 1980), which states that visual features are computed in parallel in separate areas of the brain, and by means of focused attention the features are bound

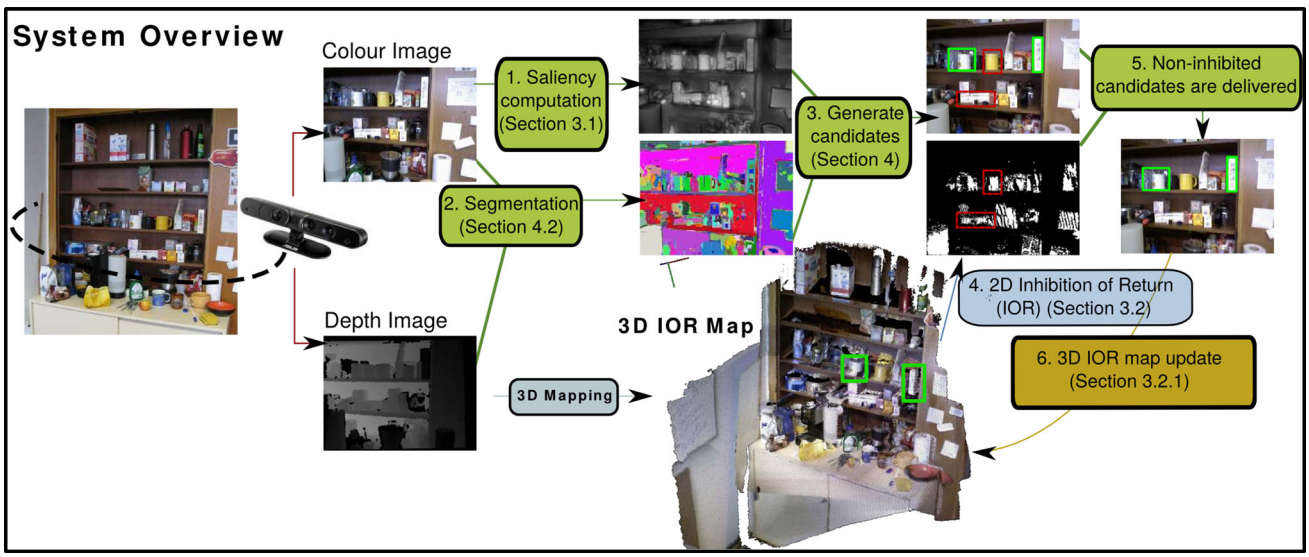

Fig. 1 General structure of our system. An RGB-D sensor records a sequence of a scene. Object candidates are generated based on colour and depth data and projected into the 3D scene map. Spatial inhibition of return, rooted in the 3D map, enables the inhibition of already attended object candidates, naturally resulting in scene exploration

together. Koch and Ullman (1987) proposed a model where those features are fused into a saliency map that encodes where attention should be allocated. One of the first computational attention systems that was implemented based on this model was the renowned Itti–Koch model (1998), where feature channels are computed in parallel, image pyramids enable a multi-scale contrast computation, and feature contrasts are computed by Difference-of-Gaussians.

One component of attention systems is the inhibition of return mechanism: a mechanism that inhibits attention from returning to already attended areas. It was discovered by Posner et al. (1985) as taking place in the human visual system, operating in spatial coordinates and not retinotopical ones, and was hypothesised to enable visual exploration. Already Itti et al. (1998) proposed a computational implementation of IOR that consisted in zeroing values in the saliency map for the regions that had already been the target of attention. Their method however only worked on single images. While IOR is simple on individual images, image sequences introduce the challenge of establishing correspondences between objects over time. In this context, Backer et al. (2001) performed object-centred IOR by tracking the attended objects. However, their approach operated on simple artificially rendered scenes instead of real world data and on 2D images instead of 3D data as we do. In the work of Palomino et al. (2011) the authors implemented IOR by visually tracking the objects that are the target of attention. In contrast to all these approaches that perform inhibition in image coordinates, our attention system implements an IOR mechanism that operates on spatial world coordinates, similarly as in human vision.

Object discovery and its cognitive background

The problem of object discovery consists in finding the potential objects that are present in an image, before their category or identity is known. It defines a strategy for understanding images, by which first, object candidate regions are generated, and second, the candidates are recognised. This strategy is opposed to the older sliding-window approach, where bounding boxes at every location and scale of the image are exhaustively examined by an object classifier (Viola and Jones 2004). Interestingly, this inversion (to first propose object candidates and then identify them) has a parallel in the cognitive science literature in the work of Pylyshyn (2001): Pylyshyn postulates that the human visual system requires a mechanism that visually individuates the elements in the environment before their properties or categories are known.

The task of discovering objects in images is a chicken-and-egg problem: how to look for an object before knowing how it looks like and which features it has? Two big scientific communities, computer vision and robotics, have developed different approaches to solve this problem. Computer vision approaches usually operate on colour images and generate a pool of object candidates, also known as object proposals, based on various types of image features which are combined by a machine learning method (Alexe et al. 2012; Manén et al. 2013). The idea is to generate promising candidate regions as a pre-processing for recognition, whose number is significantly smaller than the number of sliding windows used by default. Since usually around 1000 to 10,000 candidates are generated, these approaches are less useful for systems which have to operate in real time and which potentially aim to interact with the objects. In the robotics community, it is therefore preferred to generate a smaller set of object candidates. Attention systems are a popular approach because of their ability to focus on the relevant parts of the input: an image symmetry operator was used in conjunction with Gestalt principles in Kootstra and Kragic (2011) to generate object candidates. More recently, an attentional 2.5D symmetry operator on depth data was proposed by Potapova et al. (2014) to generate fixation points on the centre of objects; the approach then uses features such as 3D convexity in order to produce object candidates. Other groups integrate several views of the scene into a single 3D reconstruction of the environment where the discovery takes place (Herbst et al. 2011; Karpathy et al. 2013). Some approaches use information about changes over time to segregate objects from background (Herbst et al. 2011) or interact with possible object candidates to determine what is an object (Schiebener et al. 2014). While these are good approaches to resolve ambiguities, it is certainly desirable to be able to find objects also without or before interaction, and if possible already from a single view without the need to regard a scene over a longer time.

In the following we will describe our attention system and how the IOR mechanism is implemented ("The attention system: saliency computation and spatial inhibition of return" section). Section "Saliency-based object discovery" will cover the object candidate generation based on the attention system and RGB-D data. Finally, in "Evaluation" section we will evaluate our two main contributions: the object candidate generation and the IOR mechanism.

The attention system: saliency computation and spatial inhibition of return

In this section, we describe our bottom-up computational attention system VOCUS2 (Frintrop et al. 2015). Attention is known to have two main components: bottom-up and top-down. Bottom-up attention is driven by the intrinsic

properties of the scene and is commonly modelled by saliency computation. On the other hand, top-down attention takes into account extrinsic factors such as the task at hand, the internal state of the agent, etc. We concentrate here on bottom-up attention since top-down information is not available. In the first part of this section, we summarise how VOCUS2 (Frintrop et al. 2015) computes saliency for a given input image. Then, we address how we incorporate the spatial IOR mechanism into the attention system. Extending VOCUS2 with an IOR mechanism that operates on spatial 3D coordinates is a contribution of this paper.

Saliency computation

The architecture of the VOCUS2 saliency system (Frintrop et al. 2015) follows the renowned model of Itti and colleagues (1998). Contrast is computed by Difference-of-Gaussians filters: weighted average feature values are computed for centre and surround regions; then, the centre value is subtracted from the surround and vice versa. The method relies on an opponent colour space, which has a correspondence to the opponent theory of human perception Hurvich and Jameson (1957). It uses three feature channels: intensity, red/green and blue/yellow. All three feature channels are treated equally as opposed to Itti et al. (1998), where the two colour channels are fused into a single one, which is later fused with intensity and orientation. By keeping the two colour channels separated until the fusion into the final saliency map all three channels get the same relevance.

The second and most important difference to the Itti model is that we introduce the so-called twin pyramids to allow for arbitrary centre–surround contrast ratios. In the Itti model, a Gaussian Pyramid is computed for each of the feature channels; centre–surround contrast is computed by subtracting layers of the pyramid, approximating in this way a Difference-of-Gaussians filter. The problem with this approach is that centre–surround ratios are restricted to powers of 2: if two consecutive layers of the pyramid are subtracted we have a 1:2 ratio, if the second next layer is subtracted we have a 1:4 ratio, etc. Instead, we compute two pyramids for each feature channel: one centre pyramid and one surround pyramid; each of them is computed with a different smoothing factor σ, which means we can compute contrasts of arbitrary centre–surround ratios.

The results in Frintrop et al. (2015) show that VOCUS2 computes precise saliency maps competitive with other state-of-the-art methods, and it is also very fast: less than 40 ms for a standard 640 × 480 image. An example saliency map is shown in Fig. 4. In "Saliency-based object discovery" section we explain how saliency is used as a cue to generate object candidates. In the rest of this section, we explain an extension to the VOCUS2 system to implement an inhibition of return mechanism in spatial coordinates to perform shifts of attention at the object candidate level.

Inhibition of return in spatial coordinates

How to shift the focus of attention is a classical problem in computational attention systems. Always choosing the most salient region as the focus of attention would result in an attention system that always selects the global maximum as the target of attention. As in human vision (Posner et al. 1985), computational IOR helps exploring a scene by inhibiting those regions that have already been attended. When working on singles images, it is often performed by simply zeroing the region of the saliency map that was already attended (Itti et al. 1998). However, this is not enough when facing a sequence of frames from a given scene where correspondences between the visual elements should be established.

We propose a mechanism for IOR that operates on spatial coordinates. We use the KinectFusion algorithm (Newcombe et al. 2011) to obtain a 3D map that lets us store the IOR information in 3D coordinates, so that it becomes independent of the camera pose. The KinectFusion algorithm represents the 3D environment in the form of a voxel grid. This voxel grid is the discretisation of a truncated signed distance function (TSDF), i.e. every voxel in the grid stores the distance to the closest surface. At zero crossings of this function, we can expect to find the actual points of the surface.

We extend this voxel grid in order to store the IOR information. In particular, we want to store whether a point in space should be inhibited (an IOR flag), and for how long (an IOR weight). This extended voxel grid is what we will refer to in the following as the 3D IOR map. The result of raycasting this 3D IOR map to a particular pose of the camera will be called the 2D, or raycasted, IOR map.

3D IOR map update

We need a mechanism to store when a particular region of the scene has been attended, whether it should already be inhibited, and for how long. Initially, the scene has not yet been explored and all its regions could potentially be the target of attention. Thus, the IOR weights and flags are set to 0 for every voxel in the grid. Then, as the system is exposed to more frames of the sequence, object candidates are produced ("Saliency-based object discovery" section). For each frame, the pixel-precise masks of the object candidates are projected to the 3D IOR map in order to obtain the voxels that should be updated. The IOR weights of the corresponding objects voxels are increased by one. When a certain threshold *IOR_LIMIT* is reached, the IOR flag is activated, and the IOR weight is multiplied to a

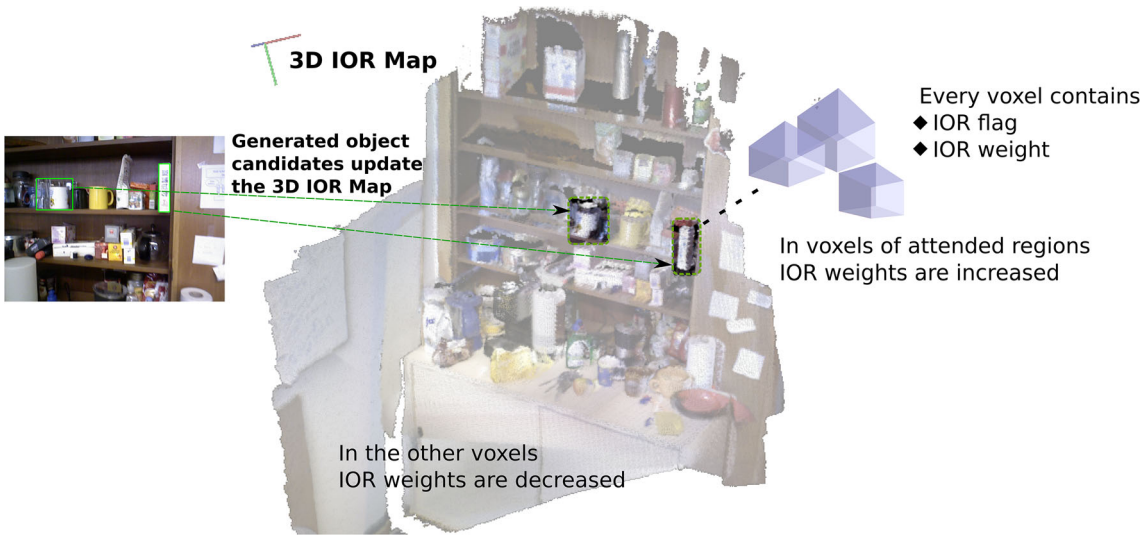

Fig. 2 IOR Map update process: the object candidates generated on a particular frame are projected to the 3D map. At the voxels in the map corresponding to the object candidates, the IOR weights are increased. Everywhere else, the IOR weights are decreased

factor mf of its value: $IOR\_WEIGHT := IOR\_WEIGHT \cdot \text{mf}$. This means that once the IOR activation threshold is reached, it will take more time for the inhibition to die out than it took to reach it. This is done to prevent the inhibition effect from quickly vanishing and the attention being allocated again on the same objects. Meanwhile, the IOR weights of the voxels that were not part of any object candidate are decreased by one. When the weights reach zero, the IOR flag is again deactivated. To sum up, regions in space that are the target of attention increase their IOR weight, and those that are not, decrease them. The IOR map update procedure is illustrated in Fig. 2, while the IOR weight evolution is depicted in Fig. 3.

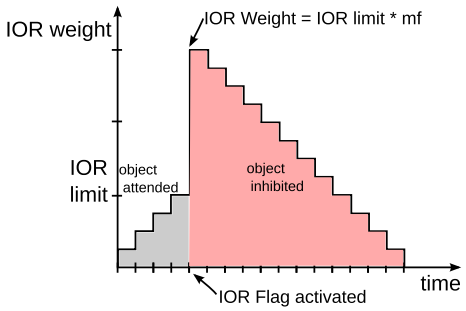

Fig. 3 Illustration of how the IOR weight evolves over time when an object has been attended long enough to activate the IOR flag. In the *grey area*, the object is attended until the IOR weight is high enough to trigger the IOR flag. The *red area* depicts the time when the object is inhibited (colour figure online)

2D IOR map

In order to use the inhibition information within our attention system, we need to obtain a 2D map from the 3D data. Since our 3D IOR Map is embedded in the voxel grid of KinectFusion, it is possible to raycast a 2D IOR map $I(x, y)$ for any given camera pose. The result of such an operation is a binary map containing white pixels for locations that should be inhibited, and black pixels for those that should not. We show in Fig. 4 an example image (left), together with the saliency map (middle) computed from it, as well as the 2D IOR map that has been raycasted (right). The white pixels in the 2D IOR map indicate the locations where attention has been allocated up to that point in time. In principle, such a 2D IOR map can be used to inhibit points or regions in the saliency map. We show in "Inhibition of object candidates" section how we use it to directly inhibit the object candidates.

Saliency-based object discovery

Our approach for object discovery uses saliency as a cue to estimate the location of objects, and segmentation as a way to refine their boundaries. The idea to combine saliency and segmentation has a cognitive motivation in the psychological work of Rensink (2000): in human perception, so-called *proto-objects* are detected by segmentation processes that bundle parts of the visual field; such processes are believed to exist on all levels of the visual system (Scholl 2001). Second, these proto-objects are combined by focused attention to form coherent objects.

An overview of our approach is shown in Fig. 5. It operates on RGB-D data, but if only colour data are available the method works also well. The combination gives however the best performance, since both modalities are complementary. From the colour image, we compute a

Fig. 4 *Left* original frame from the Coffee Machine Sequence. *Middle* the saliency map. *Right* the raycasted 2D IOR Map

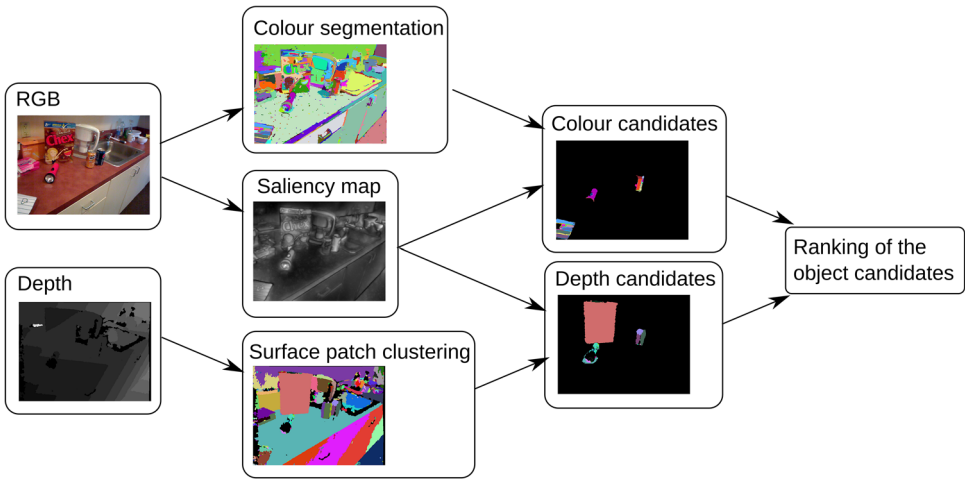

Fig. 5 Overview of the proposed method for object discovery. A pool of object candidates from both *colour* and *depth* data generates complementary candidates (colour figure online)

saliency map. In parallel, we compute a segmentation of the colour and depth images ("Segmentation" section). By using saliency to select segments, we obtain a set of object candidates ("Detection of salient blobs" section). Since in many applications it is preferable or even required to restrict the processing to a small number of candidates, e.g. to meet real-time requirements, it is important to rank the candidates according to their quality to be able to select the k best ones. The ranking strategy is discussed in "Ranking of the candidates" section.

Detection of salient blobs

First, we extract salient regions from the colour image which we will need later on to select segments that form the object candidates. For the extraction of salient blobs from the saliency map $\mathrm{sal}(x, y)$, we first determine the set of local maxima $\{l_1, \ldots, l_n\}$. A local maximum is here a (collection of) pixel(s) which is larger than all neighbouring pixels. For each local maximum in the saliency map $l = (x_l, y_l)$, where (x_l, y_l) are the pixel coordinates of the point, we do seeded region growing (Adams and Bischof 1994) to obtain a salient region s_l. The region growing recursively investigates all neighbours of l_i and adds them to the salient blob s_l, as long as the saliency of the pixel is above some percentage of the saliency of the seeding point $\mathrm{sal}(l_x, l_y)$. Thus, for every candidate point $p = (x_p, y_p)$, we compute whether $\mathrm{sal}(x_l, y_l) \geq \mathrm{sal}(x_p, y_p) \geq \mathrm{sal}(x_l, y_l) \times t$, with $0 < t < 1$. This procedure is repeated for different values of t (we use 0.6 and 0.7), and the complete set of salient regions $\{s_1, \ldots, s_m\}$ is stored for the next step.

Segmentation

In parallel to the salient region extraction, the original image and depth data are segmented. We use colour segmentation and depth segmentation separately to produce object candidates (as we showed in Martín García et al. (2015), both modalities work best when used independently). The process by which salient regions select segments works in the same way for each of the segmentation methods: for each salient region s, we pick the segments which overlap at least o per cent with s. We set this overlap to $o = 30\%$ with respect to the segment.

Image segmentation: colour candidates

We chose the algorithm by Felzenszwalb and Huttenlocher (2004) for segmenting colour images into perceptually coherent segments. The authors proposed a method that constructs a graph based on the pixel neighbourhoods, and

iteratively merges groups of pixels into regions, keeping a trade-off between the internal variability of the regions and the difference between neighbouring components. Therefore, it relies mainly on one parameter, k, that determines the scale of observation. We set it in all our experiments to 200, to slightly over-segment the images. The candidates that we obtain with this method are shown exemplarily in the first row of Fig. 6.

Surface clustering: depth candidates

To segment the depth maps, we use a method similar to Richtsfeld et al. (2012): we cluster neighbouring points into uniform planar patches without discontinuities based on their surface normals. Normal clustering starts at the point with lowest curvature and greedily assigns neighbouring points as long as they fit to the initial plane model. The algorithm iteratively creates planar surface patches until all points belong to some plane or are identified as noise. An example of the segments and the candidates obtained with this method is shown in Fig. 6, second row.

Late fusion of colour and surface patches: colour + depth candidates

Our proposed method for object discovery consists in a late fusion of the best candidates from *colour* and *depth*. An early fusion approach of both modalities would imply producing a segmentation that incorporates both colour and depth. As we show in Martín García et al. (2015), the late fusion approach is preferable to the early one.

Inhibition of object candidates

At this point we can use the 2D IOR map to inhibit those candidates that correspond to regions that have already been attended. To decide which candidates to inhibit, we simply compute the intersection of the 2D IOR map IOR(x, y) and the object candidate binary mask o as: $Z(x,y) = \mathrm{IOR}(x,y) \cap o(x,y)$. We inhibit an object candidate if a certain percentage of its pixels are marked as inhibited in the 2D IOR map: $\sum_{x,y} Z(x,y) / \sum_{x,y} o(x,y) \geq \theta$;

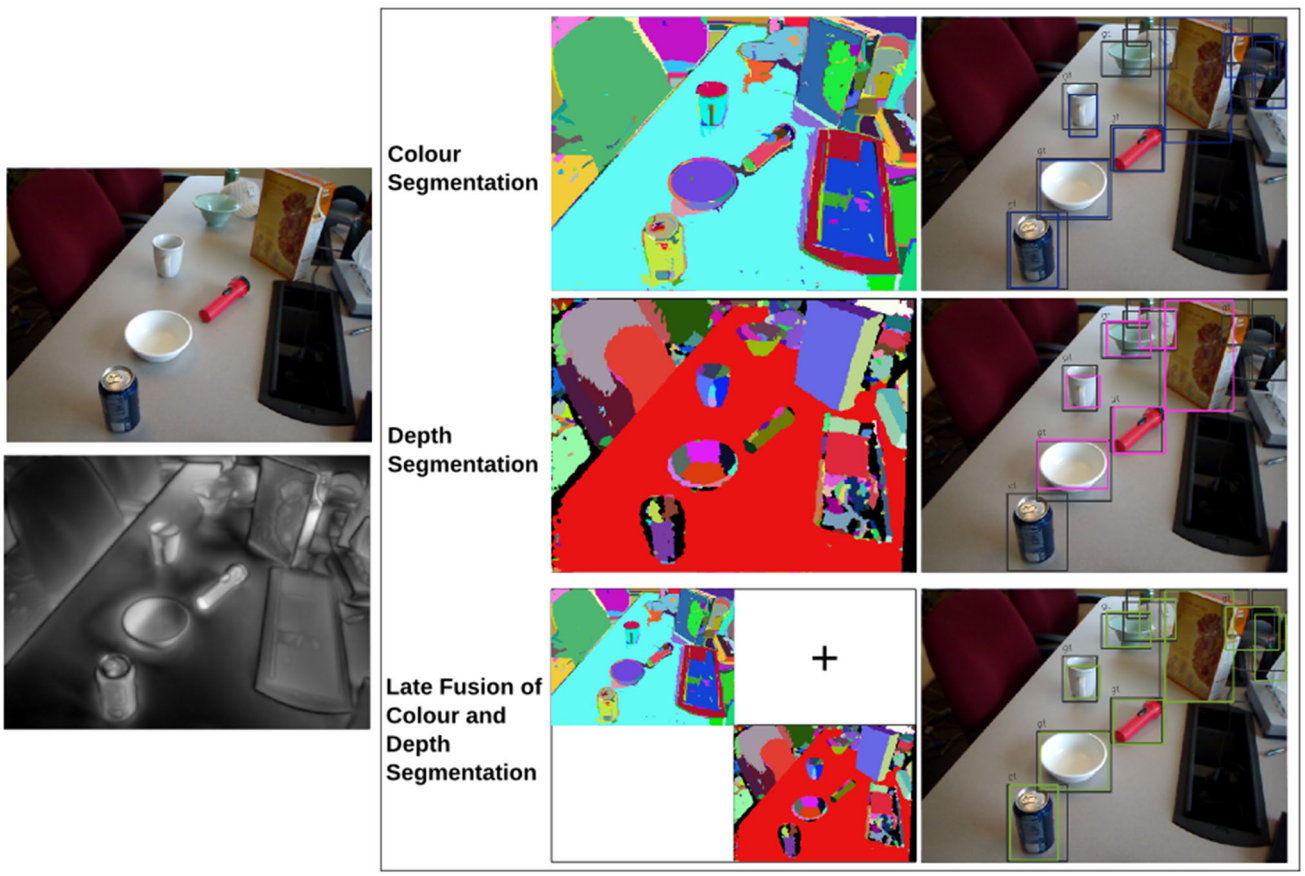

Fig. 6 *Left side* the original image; below, the saliency map. *right* the successful candidates for each of the segmentation methods: colour-based (*top*), depth-based (*middle*), and the late fusion of both (*bottom*). The first column shows the corresponding segmentations; the second displays the ground truth (*gray*) and the candidates' (in colour) bounding boxes

we set $\theta = 0.3$ in our experiments. Otherwise, the candidate goes to the next stage: the ranking of the candidates.

Ranking of the candidates

A critical issue is how to rank the object candidates to be able to select the best ones first. As mentioned before, this is important especially in robotics applications to meet real-time constraints and to select the most promising candidates for interaction.

In Martín García et al. (2015) we investigated three different approaches for ranking the object candidates. Here, we use the one that gave the best results. The ranking approach ranks object candidates using several features extracted from the candidate mask: (1–7) Hu's image moments (Hu 1962), which are invariant to rotation and scale; (8) a 3D convexity measure (described below); (9) the object proposal area normalised to the image area; (10) the average saliency of the proposal; (11) the perimeter of the object candidate mask normalised to the image area; (12) the normalised average depth of the proposal.

Convexity is known to be among the Gestalt cues that influence the figure-ground segregation processes (Kanizsa and Gerbino 1976). It has been used in computational systems as a cue to segment objects in image data (Fowlkes et al. 2007; Kootstra and Kragic 2011) as well as in 3D data (Karpathy et al. 2013; Potapova et al. 2014). Our 3D convexity feature is computed following the approach of Potapova et al. (2014): given an object's point cloud $\{p_i\}$, V is the corresponding object's convex hull, and v_j is a set of visible faces from the current viewpoint. The convexity measure κ is calculated as the mean of the shortest distances from the object points to the visible surfaces of the object's 3D convex hull:

$$\kappa = \frac{1}{n} \sum_{p_i} d_{\min}(p_i, V), \tag{1}$$

where n is the number of object points and $d_{\min}(p_i, V)$ is the shortest distance from the point to any visible face

$$d_{\min}(p_i, V) = \min_j d(p_i, v_j). \tag{2}$$

The lower the convexity measure, the more convex the object candidate is. Given this set of features, we trained a support vector machine (SVM) (Chang and Lin 2011) to classify between object/non-object. Training was done on the ground truth annotated scenes of the Washington Dataset Lai et al. (2011). For every feature vector, the SVM outputs the probability of the object candidate being object/non-object. This probability is used as a ranking score to sort candidates. To train the SVM, the Washington dataset was divided into two parts. One part was used for training and the other for testing and vice versa.

Evaluation

We divide our evaluation according to the two main contributions of this paper. In "Object discovery evaluation" section, we compare our object discovery method against other state-of-the-art competitors. Here, we will measure the quality of the object candidates when compared to the annotated ground truth. Then, in "IOR evaluation" section, we evaluate the proposed IOR mechanism. Here, we will show that taking very few candidates per frame is enough to obtain a high recall of globally discovered objects.

To evaluate our object discovery method we use two publicly available datasets, the Washington Dataset (Lai et al. 2011), and the Coffee Machine sequence, which we introduced in Martín García and Frintrop (2013). The latter is a challenging scene for object discovery with high clutter, and a total of 80 distinct objects appearing throughout the sequence and up to 48 objects per frame. It lasts for 436 frames, and has manually annotated ground truth for every 30th frame. The Washington dataset (Lai et al. 2011) contains eight sequences recorded with a Kinect camera on household environments, and is intended to test object recognition algorithms. Thus, it contains labelled ground truth where different object instances appear, and serves to evaluate our generic object candidates. Note, however, that not all the objects that appear are labelled.

In the Washington Dataset, the ground truth is provided as bounding boxes, so, in order to measure the overlap we fit a bounding rectangle on each object candidate mask we generate. On both datasets, we measure precision as the number of correct object candidates over the total number generated in a given frame, and distinguish two types of recall: global recall, being the number of distinct objects that were detected over the whole sequence, and frame recall (or simply recall), the number of ground truth objects that were retrieved in a single frame. We consider object candidates as correct if they satisfy the Pascal criterion, i.e. intersection-over-union ratio is greater than 0.5 (Everingham et al. 2007).

Object discovery evaluation

In this section, we compare our method of object discovery to other state-of-the-art methods: the one of Potapova et al. (2014), the Objectness measure of Alexe et al. (2012), and the method of Manén et al. (2013). The method of Potapova et al. (2014) relies on both colour and depth cues to produce object candidates, while the methods of Alexe et al. (2012) and Manén et al. (2013) use only colour. Our

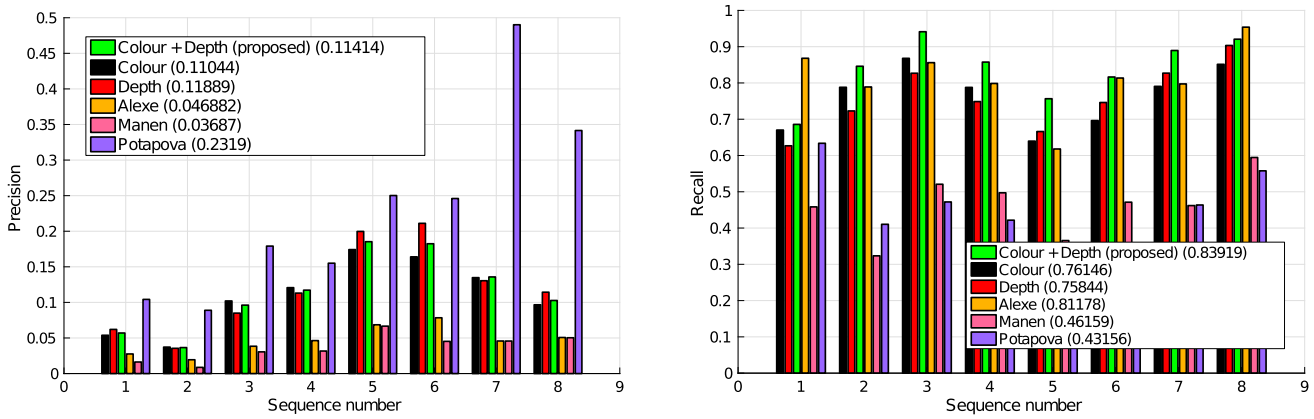

Fig. 7 Average precision (*left*) and recall (*right*) values on the Washington dataset. *Numbers in parenthesis* denote the average recall/prec. over all sequences

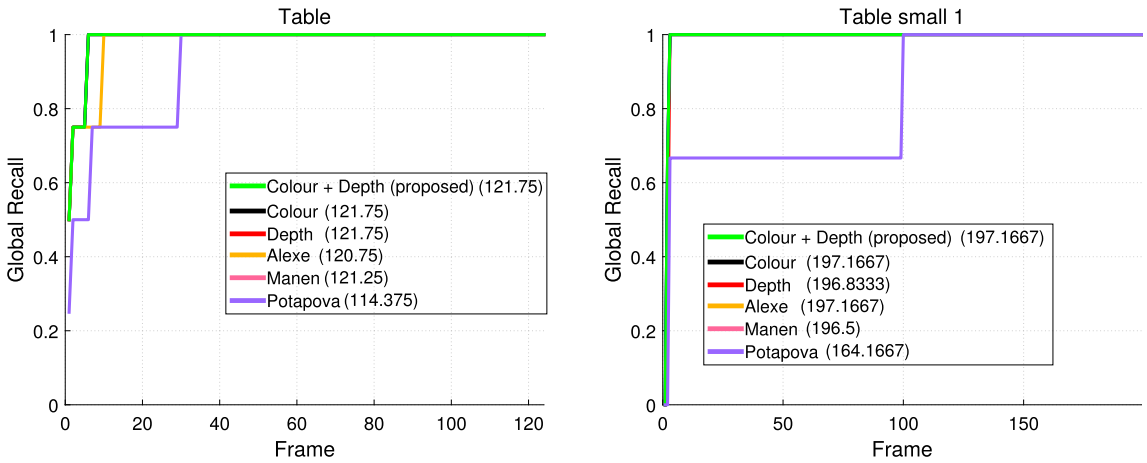

Fig. 8 Global recall over time for two sequences of the Washington dataset. In *parenthesis*, the area under curve (AUC) values

methods are denoted as *Colour*, *Depth* and *Colour* + *Depth* (C + D) respectively, depending on the segmentation method that was used.

Washington dataset

First, we show our results on the Washington Dataset (Lai et al. 2011). In Fig. 7, we show average precision and recall obtained in each of the sequences (1–8) by all methods. The results show that, in terms of recall, our method C + D clearly outperforms all the other methods with an average recall of 84%. The second best is the objectness measure of Alexe et al. (81%).

In terms of precision, the method of Potapova et al. (2014) turns out to be the best: it produces very few object hypotheses but these are mostly correct. Our proposed methods achieves a precision of about 10%, whereas the methods of Alexe and Manén only reach 4 and 3% respectively. The generally low precision values come partly from the fact that few objects are present in the scenes, and not all objects are labelled as ground truth in this dataset.

In Fig. 8 we can see the global recall plot as it evolves over time. This metric shows the percentage of objects that have been discovered throughout the sequence, as time passes. We show the results for two sequences of the dataset but very similar results are obtained for the rest. The plots reflect that the dataset is relatively easy, and after a few frames, all the objects in the sequence are found by all the methods we applied.

A complementary view for these results is shown in Fig. 9, where precision and recall values are plotted over the number of object candidates. It averages over all eight sequences in the Washington dataset. This is useful for deciding how many object candidates to generate. There, one can see that by taking about 20 candidates from method C + D, approximately 75% of the objects are detected in each frame.

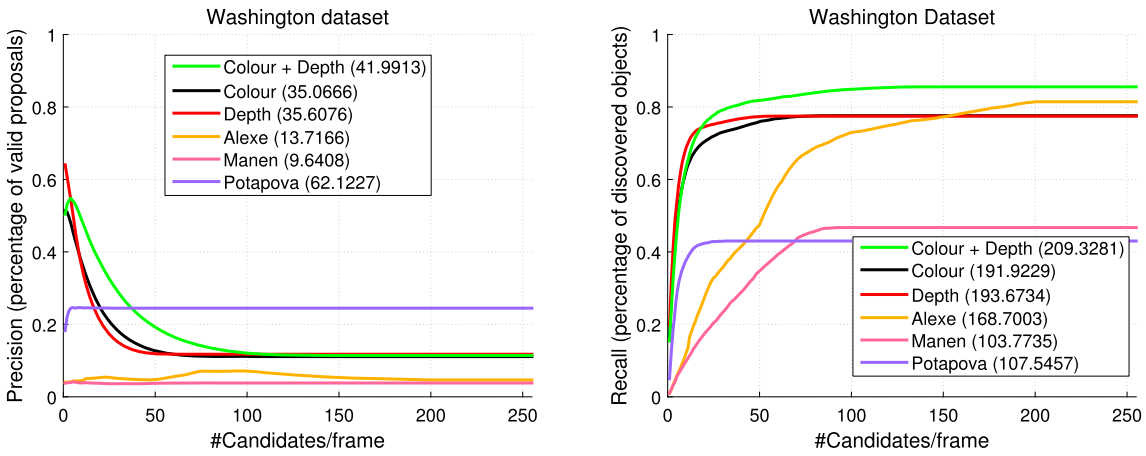

Fig. 9 Precision and recall values over number of proposals/candidates on the Washington dataset. In *parenthesis*, the area under curve (AUC) values

Table 1 *F*-score values for all the methods in the Washington and Coffee Machine Sequence datasets. The three first columns correspond to our methods

	Colour + Depth	Colour	Depth	Alexe	Manen	Potapova
F-score						
Washington	0.20	0.19	0.21	0.09	0.07	0.30
Coffee	0.32	0.28	0.34	0.15	0.19	0.27

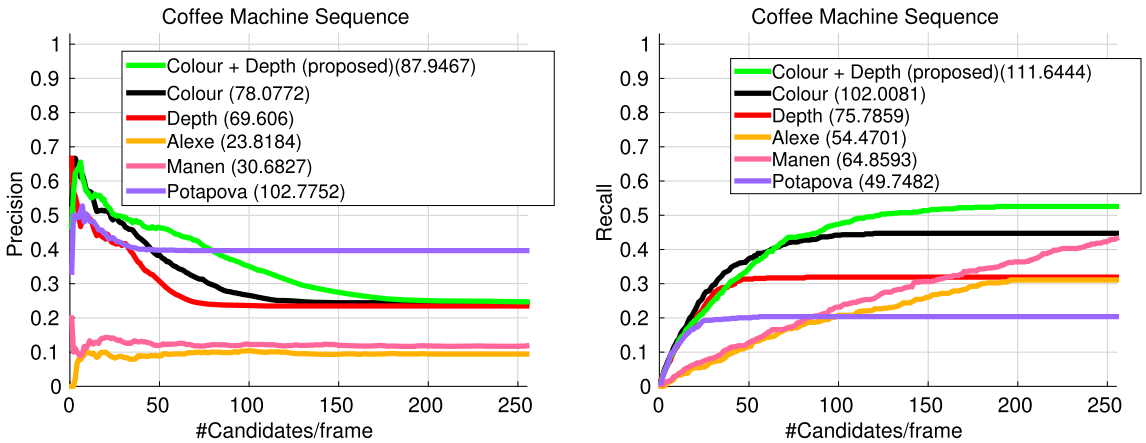

Fig. 10 Coffee Machine sequence results. *Left* precision over number of candidates. *Right* recall over number of candidates. In *parenthesis*, the area under curve (AUC) values

Finally, in Table 1 we show the *F*-score values obtained by each of the methods. The highest value is achieved by the method of Potapova (0.30), followed by our proposed approach (0.20). The relatively low value that we obtain comes from the fact that few objects are labelled in this dataset, and so our precision decays as we take more candidates: by considering the same number of candidates that the method of Potapova et al. (2014) produces (about 12), we would obtain an *F*-score of 0.56.

Coffee Machine sequence

The high recall obtained by most methods in the Washington dataset shows that the benchmark is relatively easy. Thus, we evaluate all the methods in a sequence (Martín García and Frintrop 2013) that contains many more objects (on average 36 per frame, some frames have up to 48 objects) and plenty of clutter.

As before, we show in Fig. 10 the precision and recall values over the number of candidates. The

Fig. 11 Candidates that matched an object out of the top 20 object candidates on several frames of the Coffee Machine Sequence

difficulty of the sequence is reflected in lower recall values than for the Washington dataset for all the methods. Despite this difficulty, most of the methods achieve a considerably higher precision (compare the AUC values in Figs. 9 and 10). This is due to the fact that in this dataset, all objects were labelled for the ground truth. Finally, the *F*-score values are shown in Table 1: in this dataset our proposed method obtains the highest score (0.32).

The top 20 candidates of our method C + D are shown in Fig. 11 for several frames. There, one can see examples of objects that were successfully retrieved (that satisfy the Pascal criterion w.r.t the ground truth).

IOR evaluation

In this section, we evaluate the IOR mechanism in terms of how well it serves for visual scene exploration: our purpose is to show that with a few object candidates we can still detect most of the objects in the scene by the end of the sequence. Thus, we constraint our object discovery method to producing a very small number of object candidates, ranked according to the SVM score. The experiment is illustrated in Fig. 12 for a few frames of the Coffee Machine Sequence.

Propagation of ground truth annotations

We manually annotated our Coffee Machine Sequence on every 30th frame. That means, we created greyscale masks for every 30th frame, where every object kept a consistent greyscale level (or ID) throughout the sequence. This was already enough to evaluate our object candidates in "Object discovery evaluation" section, however, in order to test the IOR mechanism we require ground truth available in every frame: the IOR mechanism takes place from frame to frame, and so, its effect would be "lost" if we evaluated the results on every 30th frame.

We developed a method to automatically propagate the sparse ground truth annotations to the unlabelled frames. The method proceeds as follows. We run the KinectFusion algorithm a first time in order to build the 3D map of the scene. The idea is that we want to use every annotated frame to generate interpolated ground truth for the closest frames before and after it. For example, manually annotated ground truth frame 90 will be used to automatically generate the ground truth of frames 75–105. Thus, we run KinectFusion another two times: once backwards, generating ground truth for the 15 frames before every annotated one; and once

Fig. 12 Illustration of the IOR experiment: on top some frames from the Coffee Machine Sequence; the *green bounding boxes* show the candidates that successfully matched an object. The *bottom row* shows the 2D IOR map at those frames. The *red arrows* depict candidates that have been attended long enough to activate the inhibition flags (colour figure online)

forwards, generating the ground truth of the 15 frames following every annotated one.

So, for every frame for which ground truth exists, we project the annotated ground truth masks to the 3D map, and store the object labels in the corresponding voxels. For every frame for which no ground truth exists, the object labels are raycasted according to the current camera pose to form a raycasted ground truth map. The results of this method are shown in Fig. 13 for some frames of the sequence.

Results

We show the results of our experiments in terms of global recall over time in Fig. 14. Here, we are interested in seeing how many objects of the scene we are able to retrieve with as few object candidates as possible. First, we show the effects of altering the multiple factor parameter mf, i.e. , the parameter that controls how long does the IOR effect last. Higher values for this factor have the effect that once the IOR activation value is reached, it takes longer to die out (see "3D IOR map update" section for details). We show the results obtained when generating 20 candidates per frame for our C + D object discovery method, for three different values of mf: 2, 4 and 6 (green, black and pink curves respectively in Fig. 14). The global recall values achieved were 86% for mf = 2 and 4, and 91% for mf = 6.

In the second part of this set of experiments we compare the results of running our object discovery method with IOR and without it. We chose the method that performed best in terms of global recall in the previous experiment: 20 candidates per frame method with IOR and mf = 6. We compare it to two different configurations without IOR: generating 20 candidates (red curve) and 255 candidates per frame (blue curve). As the results show, fixing the number of candidates on 20, using the IOR mechanism makes a big difference in terms of global recall: 91 (value reached by the pink curve) versus 73% (value reached by the red curve). Furthermore, the results were only 2% behind w.r.t. the method generating 255 candidates per frame.

In short, the results show that by using the IOR mechanism we can rely on a much smaller number of candidates per frame (20 as opposed to 255) and still retrieve most of the objects in the scene (91%). A small number of candidates is beneficial because it means less queries for recognition are required and/or less interactions with the potential objects.

Conclusion

In this paper, we have presented a computational framework for visual scene exploration that sequentially processes the video frames of a 3D scene and produces visual object candidates. An attention mechanism guides the processing of visual information by generating visual object candidates. Exploration of new areas of the

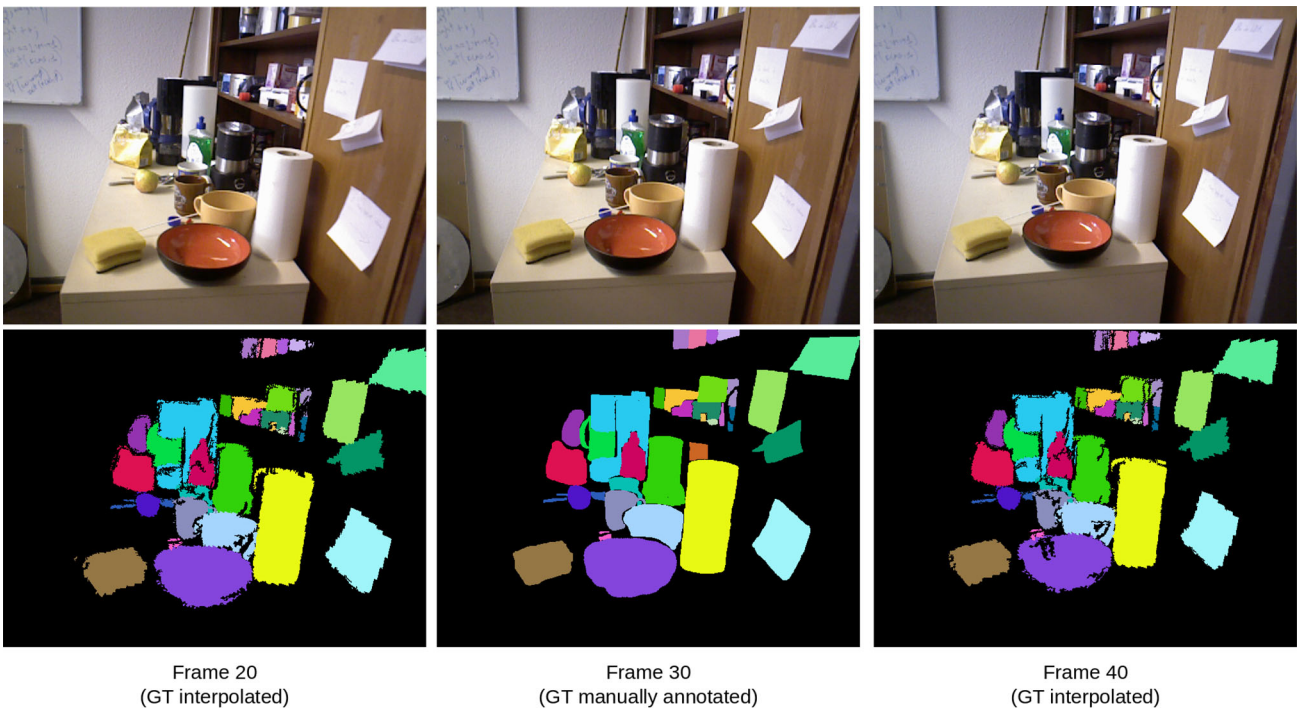

Fig. 13 Manually annotated and interpolated ground truth for the Coffee Machine sequence

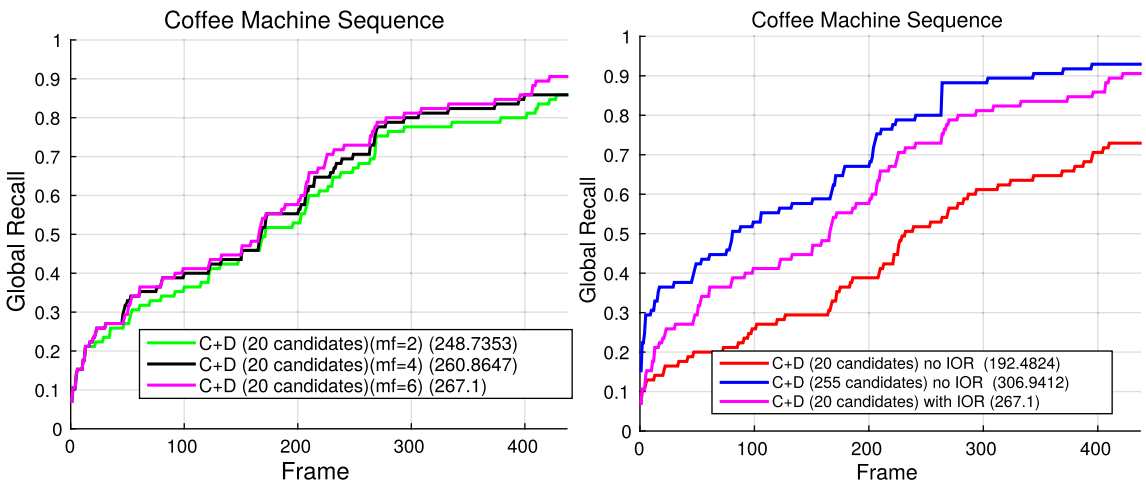

Fig. 14 Global recall over time on the Coffee Machine Sequence: *left*, results obtained by generating 20 candidates per frame with the IOR mechanism and different values for the mf parameter; *right*, results with 20 candidates per frame with (*pink*) and without IOR (*red*), and with 255 candidates without IOR (*blue*) (colour figure online)

environment is allowed by an inhibition of return mechanism implemented in spatial coordinates. Our aim was not to model how biological systems work, but rather to draw inspiration from findings about the human visual system in order to build a technical system that can successfully generate visual object candidates for understanding 3D scenes. A principal element of our system is attention: we believe it has a key role in biological systems as a way to prioritise the processing of sensory input. The same challenge is faced by technical systems: interpreting visual data is extraordinary complex, and the amount of incoming data in a video stream can be overwhelming.

The results of our experiments show that on images containing a great deal of clutter, our object discovery method is able to find most of the objects. The evaluation shows that the IOR mechanism can be applied when sequentially exploring a scene, and with as few as 20 candidates per frame, is able to retrieve up to 90% of the objects that are present.

References

Adams R, Bischof L (1994) Seeded region growing. IEEE Trans Pattern Anal Mach Intell 16(6):641–647

Alexe B, Deselaers T, Ferrari V (2012) Measuring the objectness of image windows. IEEE Trans Pattern Anal Mach Intell 34(11):2189–2202

Backer G, Mertsching B, Bollmann M (2001) Data- and model-driven gaze control for an active-vision system. IEEE Trans Pattern Anal Mach Intell (PAMI) 23(12):1415–1429

Chang CC, Lin CJ (2011) Libsvm: a library for support vector machines. ACM Trans Intell Syst Technol 2(3):1–27

Everingham M, Van Gool L, Williams C, Winn J, Zisserman A (2007) The Pascal visual object classes challenge 2007 results. http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc2007/. Accessed 17 Feb 2015

Feldman J (2003) What is a visual object? Trends Cogn Sci 7(6):252–256

Felzenszwalb PF, Huttenlocher DP (2004) Efficient graph-dased image segmentation. Int J Comput Vis (IJCV) 59(2):167–181

Fowlkes CC, Martin DR, Malik J (2007) Local figure-ground cues are valid for natural images. J Vision 7(8):2

Frintrop S, Rome E, Christensen HI (2010) Computational visual attention systems and their cognitive foundations: A survey. ACM Trans Appl Percept 7(1):6

Frintrop S, Werner T, Martín García G (2015) Traditional saliency reloaded: a good old model in new shape. In: Proceedings of CVPR

Heinke D, Humphreys GW (2004) Computational models of visual selective attention. A review. In: Connectionist models in cognitive psychology, vol 4. Psychology Press, pp 273–312

Herbst E, Henry P, Ren X, Fox D (2011) Toward object discovery and modeling via 3-D scene comparison. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)

Hu MK (1962) Visual pattern recognition by moment invariants. IRE Trans Inform Theory 8(2):179–187

Hurvich L, Jameson D (1957) An opponent-process theory of color vision. Psychol review 64(6):384

Itti L, Koch C, Niebur E (1998) A model of saliency-based visual attention for rapid scene analysis. IEEE Trans Pattern Anal Mach Intell 20(11):1254–1259

Kanizsa W, Gerbino W (1976) Convexity and symmetry in figure-ground organization. In: Henle M (ed) Vision and artifact. Springer, New York, pp 25–32

Karpathy A, Miller S, Fei-Fei L (2013) Object discovery in 3D scenes via shape analysis. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)

Koch C, Ullman S (1987) Shifts in selective visual attention: towards the underlying neural circuitry. Matters of intelligence. Springer, Berlin

Kootstra G, Kragic D (2011) Fast and bottom–up object detection, segmentation, and evaluation using Gestalt principles. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)

Lai K, Bo L, Ren X, Fox D (2011) A large-scale hierarchical multi-view rgb-d object dataset. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)

Manén S, Guillaumin M, Van Gool L (2013) Prime object proposals with randomized Prim's algorithm. In: IEEE International Conference on Computer Vision (ICCV)

Martín García G, Frintrop S (2013) A computational framework for attentional 3D object detection. In: Proceedings of the Annual Conference of the Cognitive Science Society (CogSci)

Martín García G, Potapova E, Werner T, Zillich M, Vincze M, Frintrop S (2015) Saliency-based object discovery on RGB-D data with a late-fusion approach. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)

Newcombe RA, Izadi S, Hilliges O, Molyneaux D, Kim D, Davison AJ, Kohli P, Shotton J, Hodges S, Fitzgibbon A (2011) KinectFusion: real-time dense surface mapping and tracking. In: Proceedings of IEEE International Symposium on Mixed and Augmented Reality (ISMAR)

Palomino AJ, Marfil R, Bandera JP, Bandera A (2011) A novel biologically inspired attention mechanism for a social robot. EURASIP J Adv Signal Process 2011:4

Pashler HE, Sutherland S (1998) The psychology of attention, vol 15. MIT Press, Cambridge

Posner MI, Rafal RD, Choate LS, Vaughan J (1985) Inhibition of return: neural basis and function. Cogn Neuropsychol 2(3):211–228

Potapova E, Varadarajan KM, Richtsfeld A, Zillich M, Vincze M (2014) Attention-driven object detection and segmentation of cluttered table scenes using 2.5D symmetry. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)

Pylyshyn ZW (2001) Visual indexes, preconceptual objects, and situated vision. Cognition 80(1–2):127–158

Rensink R (2000) The dynamic representation of scenes. Visual Cogn 7:17–42

Richtsfeld A, Morwald T, Prankl J, Zillich M, Vincze M (2012) Segmentation of unknown objects in indoor environments. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)

Schiebener D, Ude A, Asfour T (2014) Physical interaction for segmentation of unknown textured and non-textured rigid objects. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)

Scholl B (2001) Objects and attention: the state of the art. Cognition 80:1–46

Treisman AM, Gelade G (1980) A feature integration theory of attention. Cogn Psychol 12:97–136

Viola P, Jones MJ (2004) Robust real-time face detection. Int J Comput Vision 57(2):137–154

Wagemans J, Elder JH, Kubovy M, Palmer SE, Peterson MA, Singh M, von der Heydt R (2012) A century of gestalt psychology in visual perception: I. perceptual grouping and figure-ground organization. Psychol Bull 138:1172–1217

Wang A, Liu X, Chen Q, Zhang M (2016) Effect of different directions of attentional shift on inhibition of return in three-dimensional space. Atten Percept Psychophys 78(3):838–847