Anticipation and Attention for Robust Object Recognition with RGBD-Data in an Industrial Application Scenario

Narunas Vaskevicius Kaustubh Pathak Andreas Birk

***Abstract*— An extension based on attention and anticipation of a robot vision pipeline for object recognition in RGBD images from low-cost sensors like MS Kinect or ASUS Xtion is presented. This work originated in research on an industrial application scenario, namely shipping-container unloading, but it is applicable to advanced manipulation tasks in unstructured environments in general where the perception must be very robust while being as fast as possible. For these scenarios, we build on our previous work that proved to be competitive in cluttered scenes in table-top scenarios and which forms the backbone of our RGBD object recognition. It is further enhanced by two main contributions. First, a simple but very effective form of anticipation as top-down expectations of the evolution of the scene due to the actions of the robot is used to speed up the processing. Second, attention is used as a mechanism for further speed-up by focusing processing only on certain regions of interest of the scene based also on an anticipation mechanism. The method is analyzed in experiments using real-world data from an industrial demonstration set-up.**

I. INTRODUCTION

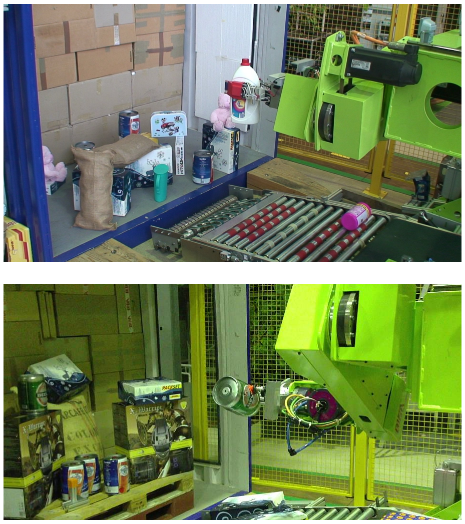


Fig. 1. The demonstration set-up for shipping container unloading, located at the Institute of Production & Logistics (BIBA) in University of Bremen. The two images show objects being autonomously unloaded using two different grippers - Velvet Fingers gripper [1] (top), and a suction gripper (bottom).

The work presented here deals with challenging scenarios for object recognition with especially high demands on robustness and processing speed. This research originates from work on object recognition and localization in single RGBD images in the context of the development of a system for the fully automated unloading of heterogeneous goods from shipping containers [3]. The hardware set-up for such a container unloading demonstrator is shown in Fig. 1. But the presented work is applicable for perception and manipulation tasks in general, especially in unstructured environments like underwater applications where RGBD data from stereo cameras is an interesting option as basis for perception (Fig. 2).

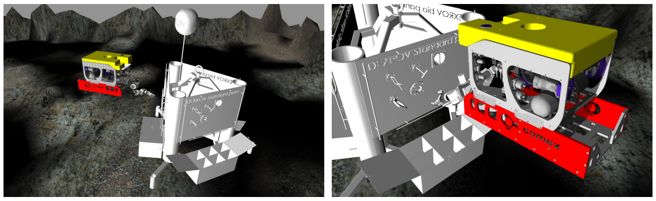


Fig. 2. Scenario with underwater manipulation tasks to be carried out in the context of the DexROV project [2] where intelligent support functions on the vehicle itself are executed to aid the operation by a human controller on-shore who is connected with severe communication latencies.

The research presented here builds upon our previous work in the context of the ICRA 2011 ”Solutions in Perception Challenge“ for which we developed a successful object recognition and localization pipeline [4]. It was shown in our follow-up work [5], [6] that an extended version of this approach is suitable for a challenging industrial scenario in terms of robustness and localization precision. However, improved robustness comes with a cost of a higher computational time. The cycle time is a critical parameter for an industrial system, therefore here, we investigate an attention and anticipation framework with the goal of achieving faster processing without sacrificing any of the robustness.

Following main contributions are made in this paper. We use anticipation as a top-down generation of expectations about the dynamics of the scene with the aim to avoid, when possible, costly bottom-up processing that is replaced by a computationally inexpensive reprojection test. To this end, the system anticipates that all recognized objects except the one that is manipulated remain static – which is verified in a top down process by a fast reprojection test of the objects' models onto their anticipated poses. In addition, attention is used, i.e., the selection of regions of interest (RoI), onto which the bottom-up processing is focused. A form of anticipation is also used here, namely the expectation that manipulated objects are likely to have occluded other objects and that hence new information will become available in the regions where they were placed in the scene.

Sec. II provides an overview of the related work as well as brief explanation of our approach to attention and anticipation in the context of container unloading. An important part of the attention and anticipation framework is the reprojection test and therefore is explained in more detail in Sec. III. The overall framework is then described in Sec. IV. The performance improvements due to attention and anticipation are evaluated in Sec. V. Finally, Sec. VI

The authors are with the Department of Computer Science and Electrical Engineering, Jacobs University Bremen, 28759 Bremen, Germany. {n.vaskevicius, k.pathak, a.birk}@jacobs-university.de

The research leading to the results presented here has received funding from the European Community's Seventh Framework Programme (EU FP7) and Horizon 2020 (EU H2020) within the projects “Cognitive Robot for Automation Logistics Processes (RobLog)” and “Effective Dexterous ROV Operations in Presence of Communications Latencies (DexROV)”.

concludes the paper.

II. Related Work

A. Anticipation

Anticipation is a core element of human perception [7] which is also relevant for artificial systems as it can be very beneficial in increasing processing speed. Anticipation can be seen as a top-down process that generates hypotheses to predict the future. This can be, for example, based on a tracking of objects and an extrapolation of the determined trajectories to predict their future locations [8], e.g., for obstacle avoidance [9]. We are interested in the prediction of the physical behavior of objects, especially as direct or indirect consequences of interactions with the unloading system. One line of approach in this context uses machine learning on time-series of raw sensor and motor data, i.e., so to say raw past experiences, to predict likely sensor data given a state and a series of possible future motor commands. Examples include Artificial Neural Networks to predict future sensor-motor relations with a relative simple robot, namely a Khepera, in a simple environment [10], [11] or Bayes filtering on vision/motor-data of an autonomous mobile robot to anticipate places [12].

The approaches using machine learning on raw sensor/motor-data can only lead to very short prediction time horizon and is often quite unreliable. More complex anticipation is possible using Gibson's notion of affordance [13] [14], i.e., the quality of an object that allows an agent to perform an action. This can be used to predict opportunities for interaction [15] [16], which is somewhat comparable to procedural reasoning [17].

The anticipation strategy used in this work is very simple but also very effective - as shown in Sec. V, it is robust with good classification rates while leading to a significant speed-up of the perception cycles by using only fast top-down processing.

The default perception cycle [4] almost always leads to the recognition and localization of multiple objects of which only one is manipulated. The other recognized objects should remain static - which is a valid assumption. But occasionally one or several ones of these supposed static objects get perturbed during the unloading. Ignoring these perturbations, i.e., trying an unloading of multiple objects recognized/localized in one RGBD snapshot, would lead to a decrease in the robustness of the unloading system. Though these perturbations are rare, even the reduction of a few percent in the recognition/localization rates is undesirable in an industrial application scenario. Our default perception system (Fig. 3) hence operated with full cycles on an RGBD snapshot for each single object to be unloaded - including all the robust but also computationally expensive bottom-up processing steps.

The anticipation used here uses the hypotheses that all recognized/localized objects that are not manipulated remain static. But these hypotheses are not taken for granted, they are validated in a top down process (Fig. 4) that acquires a new RGBD image and projects the object models onto their anticipated poses and does a kind of cross-correlation to check whether they are indeed there. This reprojection test can be done in a fast manner and as experiments presented in section V show it can successfully detect whether there are unexpected dynamics in the scene or not.

B. Attention

Attention as a process to focus processing is a second mechanism that is very helpful to speed-up the perception pipeline. Attention methods have been intensively researched in the last decade. A detailed survey on computational visual attention systems – including also the cognitive aspects – can, for example, be found in [18].

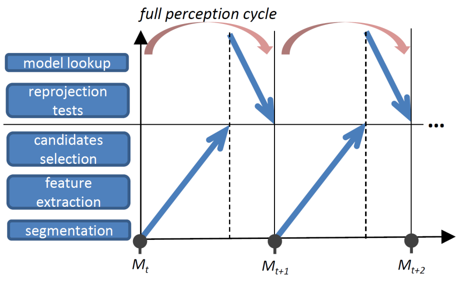


Fig. 3. The object recognition cycle in our system consists of bottom-up (segmentation, feature extraction, candidates selection) and top-down (model lookup, reprojection tests) perception steps. The default pipeline uses a sequence that consists of the full perception cycles, i.e., a new snapshot of RGBD data M_t is taken each time and fully processed with all bottom-up and top-down modules.

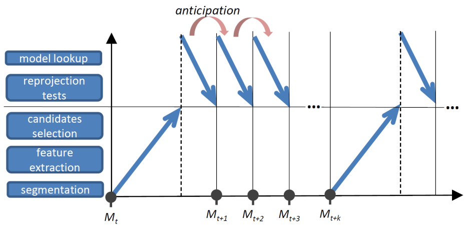


Fig. 4. The use of anticipation is based on the fact that most objects tend to remain static during the unloading - hence a fast top-down process can be used to verify this hypothesis for each already recognized object with a computationally inexpensive reprojection test.

One very popular way to employ attention in perception is to use saliency, i.e., to use a process on pixel-level that uses neighborhood information to assess the presence of objects or proto-objects. The concept of saliency has a strong biological background [19]. The basic idea is to use a fast process to select regions of interest on which subsequently more computationally intensive processes are applied for the actual object recognition or for other perceptual tasks. But, especially for strongly bio-inspired methods, it can be challenging to get a fast performance [20], [21], [22] due to the relatively simple but massively parallel saliency processing in natural systems. One option is to also employ parallelism for saliency computations in artificial systems, e.g., by implementing them with CUDA on a graphics card [23] or by even embedding them in silicon directly in a chip [24]. Another option is to use processes that are not necessarily biologically plausible but give a good trade-off between detecting (proto-)objects and computation speed [25], [26], [27], [28], [29], [30].

However, there are also other options than saliency for employing attention in object recognition. For example, attention can be implemented through the use of spatial context to generate expectations about likely objects, e.g., via GPS information in urban scenarios [31]. Another option is to use expectations across different sensor modalities, e.g., be priming visual processing by sound [32]. A further line of research uses input from Human-Robot-Interaction, e.g., in form of cues from dialog or gesture processing modules [33], [34], [35]. Attention can even arise as an emergent side-effect, e.g., through the dual use of a stereo camera for object search and navigation on a mobile platform [36].

Our attention mechanism is based on the fact that the regions of the (re)moved objects, for which their models and hence their projected contours are known, are the areas where most new information can be gathered. Especially, objects that are removed

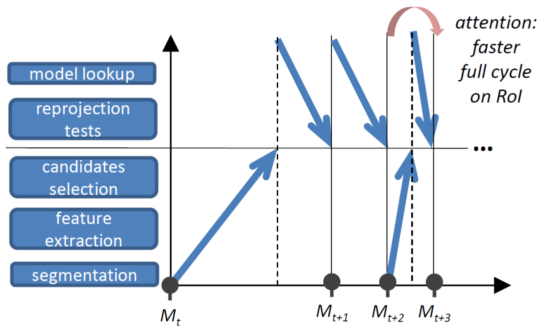


Fig. 5. When one or more objects are removed, new information becomes available mainly in the areas which have now become unoccluded due to this removal. This allows us to focus attention by concentrating the perception processes to the corresponding regions of interest (RoI), hence speeding up the computations of the different perception steps in a full cycle.

are likely to have occluded other objects about which new information will then be available and moved objects need to have their poses redetermined. Hence the locations of the (re)moved objects can serve as regions of interest for doing a complete but faster perception cycle on them (Fig. 5).

III. REPROJECTION TEST

The reprojection test used for the hypotheses verification in the textured object recognition module [4] is also a crucial part of the anticipation step. In this section, we provide details of how it is applied within the recognition module. How it can be utilized in the anticipation and attention components will be shown in the subsequent section (Sec. IV).

In the following, we will use the notation of [4] with a summary provided for the reader's convenience in Appendix. In general, scalars are in normal small letters, vectors in bold small letters, and matrices in bold capitals. For quantities resolved in different frames, we use the left superscript/subscript notation of [37]. Right subscripts are used for indexing or for denoting vector components. If required, the k-th time index is indicated by array indexing, for example $A[k]$.

In a nutshell, the reprojection test consists of two steps, namely, projection and verification. Firstly, the anticipated state of the observed scene is used to generate expected sensor data. For example, in the context of object hypotheses verification, the state is represented by the 3D models of objects and their locations with respect to the sensor as hypothesized by the recognition module. In the context of change detection in a scene, it can be a previously built map of the environment or a previous observation of the same scene. Independently of the state representation, a sensor model is used to simulate the collection of data. The verification step is then used to check the consistency between the simulated data and the real measurements for the identification of correct hypotheses or changes in the scene.

During the training phase of the recognition module [4] an object database is built using object modeling approach described in [38]. In addition to cues (e.g. visual features, color histograms, etc.) needed for the recognition, the database contains 3D object models as aggregated colored pointclouds $\mathcal{P}_o$ (Fig. 6). This representation is used in the reprojection test during the hypotheses verification step and during attention and anticipation stages as will be described in the following section.

1) Projection: The set of colored points $\mathcal{P}_o$ can be transformed to the current sensor frame $\mathcal{F}_s$ using the computed hypothesis about its 3D pose ${}^{s}_{o}\mathbf{T}$. Using the camera matrix, these points can be projected onto the image plane. We use the depth buffer for projecting only those parts of the object which are visible from the position of the sensor. Thus for each object, a virtual RGBD image ${}^{s}\mathbf{M}_o$ is generated as follows. A 3D colored point in $\mathcal{P}_o$ is denoted as the tuple ${}^{o}\tau$. It can be transformed to frame $\mathcal{F}_s$ by

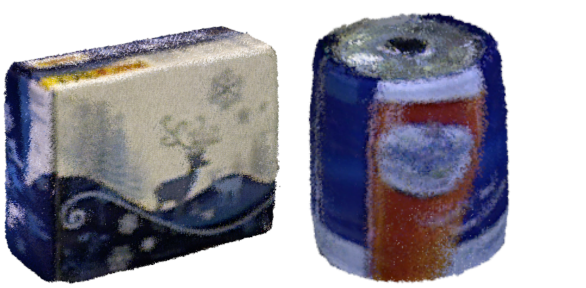


Fig. 6. Two example object representations stored in the database. Pointclouds of a parcel and a beer barrel are shown on the left and right respectively.

$$ {}^{s}\tau \triangleq \langle {}^{s}_{o}\mathbf{T}\, {}^{o}\tau_{\mathbf{p}}, {}^{o}\tau_{\mathbf{c}} \rangle, \text{ where} \tag{1a} $$

${}^{o}\tau_{\mathbf{c}}$ and ${}^{o}\tau_{\mathbf{p}}$ are respectively the color and point components of ${}^{o}\tau$. All the colored points in the object model, which after the above transform project to the same pixel coordinates ${}^{s}\mathbf{m}$, can be denoted as

$$ {}^{s}\mathcal{P}({}^{s}\mathbf{m}) \triangleq \left\{ {}^{s}\tau \mid \; {}^{s}\underline{\mathbf{m}} \simeq \mathbf{C}\, {}^{s}\tau_{\mathbf{p}} \right\}, \tag{1b} $$

where $\mathbf{C}$ is $3x3$ camera intrinsic matrix of the sensor. Then the virtual RGBD scan of the object model from the sensor frame can be constructed pixelwise as,

$$ {}^{s}\mathbf{M}_o({}^{s}\mathbf{m}) = \underset{{}^{s}\tau \in {}^{s}\mathcal{P}({}^{s}\mathbf{m})}{\operatorname{argmin}} \left\| {}^{s}\tau_{\mathbf{p}} \right\| \tag{1c} $$

2) Verification: Let the mask $\mathbb{M}_o \triangleq \{{}^{s}\mathbf{m} \mid \exists\, {}^{o}\tau \in {}^{o}\mathcal{P} : \; {}^{s}\underline{\mathbf{m}} \simeq \mathbf{C}\, {}^{s}_{o}\mathbf{T}\, {}^{o}\tau_{\mathbf{p}}\}$ be the set of discrete image coordinates obtained by projecting the object model $\mathcal{P}_o$.

For each pixel $\mathbf{m}$ in the mask $\mathbb{M}_o$ of the projected model consistency is checked using the indicator functions $\mathbf{1}_d\{\cdot\}$ and $\mathbf{1}_c\{\cdot\}$. The depth similarity at pixel $\mathbf{m}$ is evaluated by comparing the range values at the same pixel in the simulated and the real RGBD images:

$$ \mathbf{1}_d(\mathbf{m}, \tau) \triangleq \begin{cases} 1, & \left| \| {}^{s}\mathbf{M}(\mathbf{m})_{\mathbf{p}} \| - \| \tau_{\mathbf{p}} \| \right| < \varepsilon_d \\ 0, & \text{else.} \end{cases} \tag{2} $$

In the above definition, ${}^{s}\mathbf{M}$ is the actual current RGBD image from the sensor as opposed to ${}^{s}\mathbf{M}_o$ which is a virtual RGBD image, also in the same sensor frame. Color consistency is checked using a small window $\mathbb{B}_w$ of size $2w + 1$ around the pixel $\mathbf{m}$ in the real range image:

$$ \begin{aligned} &\mathbb{B}_w(\mathbf{m}) \triangleq \{\mathbf{b} \mid \; |\mathbf{m}_u - \mathbf{b}_u| \leq w \wedge |\mathbf{m}_v - \mathbf{b}_v| \leq w\}, \\ &\mathbf{1}_c(\mathbf{m}, \tau) \triangleq \begin{cases} 1, & \text{If } \exists\, \mathbf{b} \in \mathbb{B}_w(\mathbf{m}), \text{ such that, } \left\| {}^{s}\mathbf{M}(\mathbf{b})_{\mathbf{c}} - \tau_{\mathbf{c}} \right\|_c < \varepsilon_c \\ 0, & \text{else,} \end{cases} \end{aligned} \tag{3} $$

where, $\| \cdot \|_c$ is a color similarity metric, and $\mathbf{m}_u, \mathbf{m}_v$ are (u, v) coordinates of pixel $\mathbf{m}$. We have used *CIE $L^*A^*B^*$* space where the perceptual difference between colors can be approximated by the Euclidean distance between the color vectors.

As it is shown in Fig. 3 and will be discussed in the following section, one of the first stages in our perception cycle is a segmentation step which aims to *over-segment* the scene. Let us denote $\mathcal{S} = \{\mathbb{S}_1, \mathbb{S}_2, \dots \mathbb{S}_n\}$ to be the set of patches obtained during the segmentation. Using the definitions from above, the following quantities are defined to determine the consistency between real and modeled data:

$$\mathbf{s}_d \triangleq \frac{\sum_{\mathbf{m}\in\mathbb{M}_o} \mathbf{1}_d\{\mathbf{m}, {}^s\mathbf{M}_o(\mathbf{m})\}}{|\mathbb{M}_o|} \tag{4a}$$

$$\mathbf{s}_c \triangleq \frac{\sum_{\mathbf{m}\in\mathbb{M}_o} \mathbf{1}_c\{\mathbf{m}, {}^s\mathbf{M}_o(\mathbf{m})\}}{|\mathbb{M}_o|} \tag{4b}$$

$$f(\mathbb{S}) \triangleq \sum_{\mathbf{m}\in\mathbb{M}_o\cap\mathbb{S}} \mathbf{1}_d\{\mathbf{m}, {}^s\mathbf{M}_o(\mathbf{m})\} \wedge \mathbf{1}_c\{\mathbf{m}, {}^s\mathbf{M}_o(\mathbf{m})\} \tag{4c}$$

$$\mathbb{S}^\star \triangleq \underset{\mathbb{S}\in\mathcal{S}}{\operatorname{argmax}} f(\mathbb{S}), \tag{4d}$$

$$\mathbf{s}_o \triangleq \frac{f(\mathbb{S}^\star)}{|\,\mathbb{S}^\star|}. \tag{4e}$$

The quantities introduced in equations (4a) and (4b) respectively are distance and color consistency measures.

As already mentioned, the segments $\mathbb{S} \in \mathcal{S}$ are assumed to be sub-segments of the objects, i.e., we assume *over-segmentation*. If the hypothesis about the object's location is correct then there must exist a segment $\mathbb{S}^\star$ with a high consistency in the overlap between reprojected model and the segments in the real image $\{\mathbb{S} \in \mathcal{S} \mid \mathbb{M} \cap \mathbb{S} \neq \emptyset\}$. This requirement is expressed in equation (4e), where function $f(\cdot)$ measures overlap consistency by comparing colors and ranges between simulated and real data. Using this function we can calculate the last quantity $\mathbf{s}_o$ needed for the consistency test. It measures the coverage rate of the segment with the highest consistent overlap.

Based on definitions (4a), (4b) and (4e) the final consistency test is done using the following inequality:

$$(w_c \cdot \mathbf{s}_c + (1 - w_c) \cdot \mathbf{s}_d) \cdot \mathbf{s}_o \geq \theta_c, \tag{5}$$

where scalar $0 \leq w_c \leq 1$ is the weight factor for the color consistency measure and the threshold $0 \leq \theta_c \leq 1$ is the lowest allowed total consistency for the hypothesis to be considered correct. Thus, if the inequality (5) holds, then the object is considered to be in the scene at location ${}^s_o\mathbf{T}$. Segments with consistencies $f(\mathbb{S})/|\mathbb{S}|$ above a specified threshold are then used to construct the mask of the detected object.

IV. ATTENTION AND ANTICIPATION FRAMEWORK

As mentioned earlier, the goal of using anticipation and attention here is to achieve faster processing without sacrificing robustness. Through the use of anticipation, the system keeps track of the objects that have been removed or moved – either deliberatively through manipulation or unintentionally through unexpected dynamics in the scene, e.g., as side-effects of the manipulation. This can be checked in a fast top-down process through reprojection tests (Fig. 4). The locations of perturbed or removed objects form the regions of interest for the next perception cycle (Fig. 5).

The attention and anticipation framework was embedded in a perception pipeline described in [4], [5] and it can be explained with the help of Fig. 7.

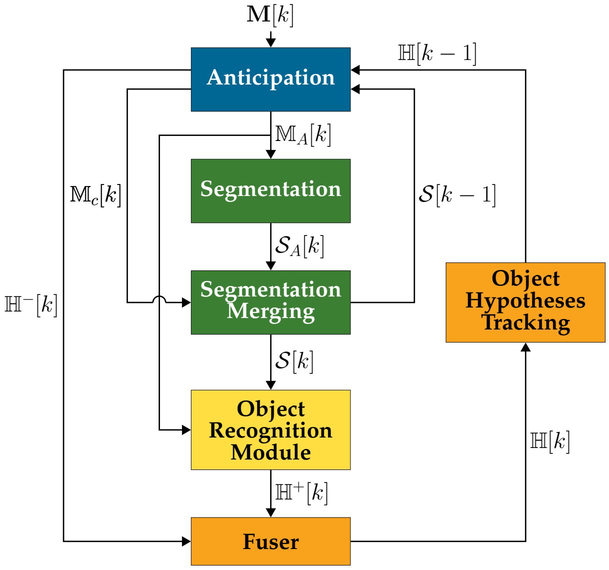


Fig. 7. The modified recognition pipeline with the anticipation module. Refer to Appendix for an explanation on the notation.

Anticipation/Attention: This module takes as its input: 1) a set of object hypotheses $\mathbb{H}[k-1]$ from the previous scene observation time-instant $k-1$; 2) the previous segmentation of the scene $\mathcal{S}[k-1]$; 3) a new RGBD image $\mathbf{M}[k]$. It uses the reprojection test described in Sec. III for finding out the changes in the scene. This consists of two steps:

- Projecting the previous RGBD image $\mathbf{M}[k-1]$ and segmentation $\mathcal{S}[k-1]$ resolved in sensor frame $\mathcal{F}_{s[k-1]}$ into the frame $\mathcal{F}_{s[k]}$ of $\mathbf{M}[k]$ using (1). The segmentation is projected by projecting corresponding pointclouds of segments $\mathbb{S} \in \mathcal{S}[k-1]$. If, after unloading an object, the sensor always returns to an observation pose for taking the next scan, then the observation frames for k and $k-1$ are identical (i.e. $\mathcal{F}_{s[k-1]}$ and $\mathcal{F}_{s[k]}$ are the same) and the projection is trivial.
- Next, equations (4) are applied, with ${}^s\mathbf{M}_o$ replaced by $\mathbf{M}[k-1]$ and the projected segments $\mathbb{S} \in \mathcal{S}[k-1]$. Segments with high value of consistencies $f(\mathbb{S})/|\mathbb{S}|$ are assumed to have stayed unchanged and are used to find a consistent mask $\mathbb{M}_c[k]$. Those with low values of consistency are used to find the *attention-mask* $\mathbb{M}_A[k]$, where attention is now focused for further processing.

Each of the hypotheses from $\mathbb{H}[k-1]$ are also projected into the current frame. By finding the projected mask intersection of the recognized objects from $\mathbb{H}[k-1]$ with $\mathbb{M}_c[k]$, it is possible to determine the hypotheses set $\mathbb{H}^-[k]$ of objects which have remained static in the scene and do not need to be re-recognized – the anticipation module, hence, forwards them directly to a fuser component. The fuser is responsible for combining object hypotheses coming from different recognition modules and ensures that more confident hypotheses are taken in case of overlap or contradiction. The forwarded hypotheses are treated as hypotheses coming from a different recognition module. In most of the cases these hypotheses don't overlap with the hypotheses detected in the attention regions and hence they are simply concatenated. Let us denote all segments from $\mathcal{S}[k-1]$ which belong to the objects in the set $\mathbb{H}[k-1] - \mathbb{H}^-[k]$ as $\Delta\mathbb{M}$. Then, for added robustness, the attention mask $\mathbb{M}_A[k]$ is extended as $\mathbb{M}_A[k] \leftarrow \mathbb{M}_A[k] \cup \Delta\mathbb{M}$.

Segmentation: The RGBD raster of the attention region $\mathbb{M}_A[k]$ is divided into contiguous clusters, i.e. segments or patches, which are homogeneous with respect to certain geometric and/or color-based criteria. In this work, the segmentation was done using the Mean-Shift [39] algorithm extended to operate in RGBD space. The segmenter basically *over-segments* the scene and the resultant *atomic-patches* form the basis for downstream perception pipeline components. An example

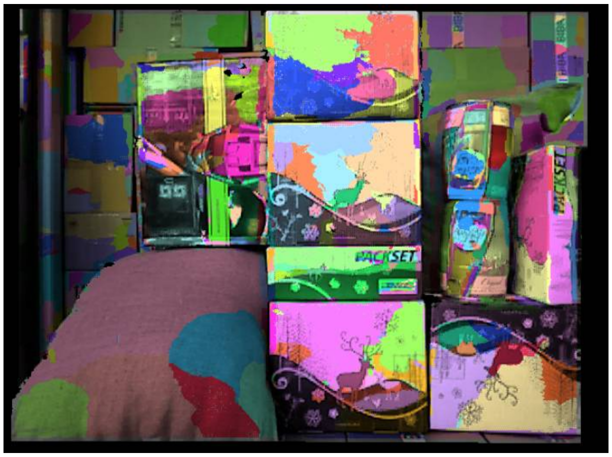


Fig. 8. Example segmentation of a full scene. Parcels and the beer barrels are neatly stacked, making the segmentation difficult.

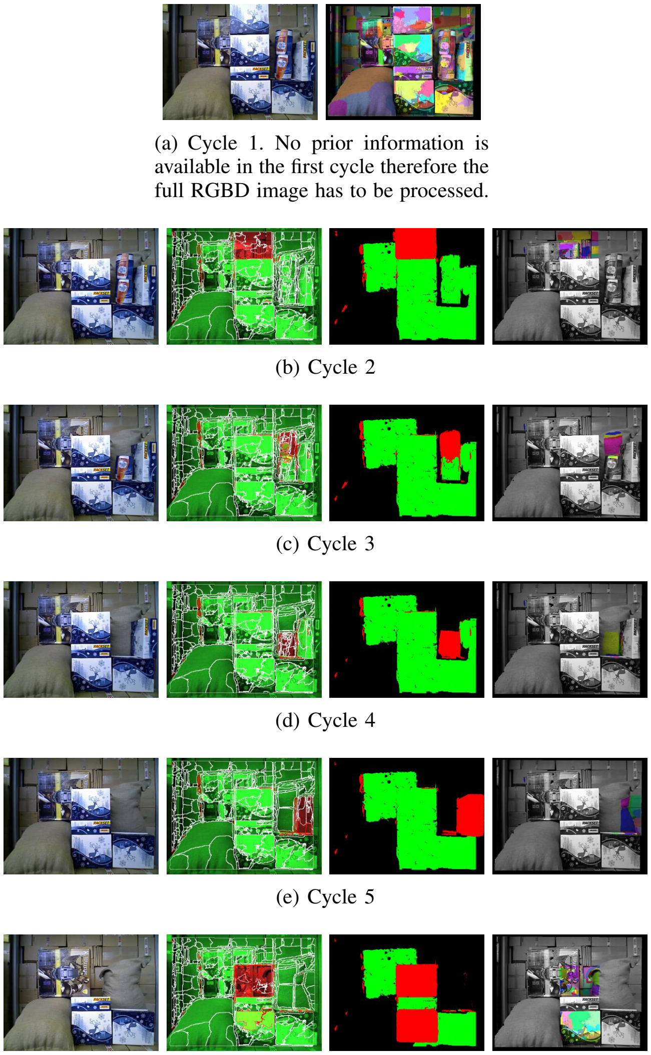
segmentation of a full scene is shown in Fig. 8.

Segmentation Merging: The masked portion $\mathbb{M}_A[k]$ in $\mathbf{M}[k]$ is now reprocessed by the segmentation module – this makes the process efficient, since the segmentation needs to be redone only on the areas of $\mathbf{M}[k]$ which changed. The resulting partial segmentation $\mathcal{S}_A[k]$ is now merged with the unaltered segmentation corresponding to $\mathbb{M}_c[k]$ by the segmentation-merging module to produce a complete segmentation $\mathcal{S}[k]$. This and $\mathbb{M}_A[k]$ are used by the OR module to do an incremental recognition of possible new objects to produce an incremental hypotheses set $\mathbb{H}^+[k]$. The latter is then fused with $\mathbb{H}^-[k]$ to produce a full set $\mathbb{H}[k]$ which is sent to the hypothesis tracker.

Hypothesis Tracking: This is currently a simple module which caches sets of hypotheses according to the cycle-time k and can be queried for past sets $\mathbb{H}[k-n]$.

V. PERFORMANCE IMROVEMENTS DUE TO ANTICIPATION AND ATTENTION

To evaluate the performance of anticipation and attention, we use the scene depicted in the Fig. 9(a). The paper is also accompanied by a video demonstrating several autonomous unloading cycles on additional scenes. The video is also available on Jacobs Robotics YouTube channel[1]. The scene presented here contains 12 textured objects in total, with 9 being visible in the first unloading cycle. The dynamics of the scene during the complete unloading period is demonstrated in Fig. 9 and Fig. 10. Each row of subfigures corresponds to an unloading cycle, with a color image taken from the scene-observation pose being in the left. It is followed by the patch consistency image, where a green-yellow-red color map is used to indicate the different consistency levels. The green color is used for highly consistent regions and red for inconsistent patches. The yellow shade indicates the clarity of the classification decision – it's impact is high near the decision threshold and decreases with the higher distance from the threshold. The next column is used for the illustration of the object movement classification. Green areas correspond to the masks of the objects which were not perturbed during the previous unloading cycle, whereas the red color regions represent the attention mask, i.e. the (re)moved or newly appeared objects. Finally, the last column shows the segmentation of the attention region.



(a) Cycle 1. No prior information is available in the first cycle therefore the full RGBD image has to be processed.

(b) Cycle 2

(c) Cycle 3

(d) Cycle 4

(e) Cycle 5

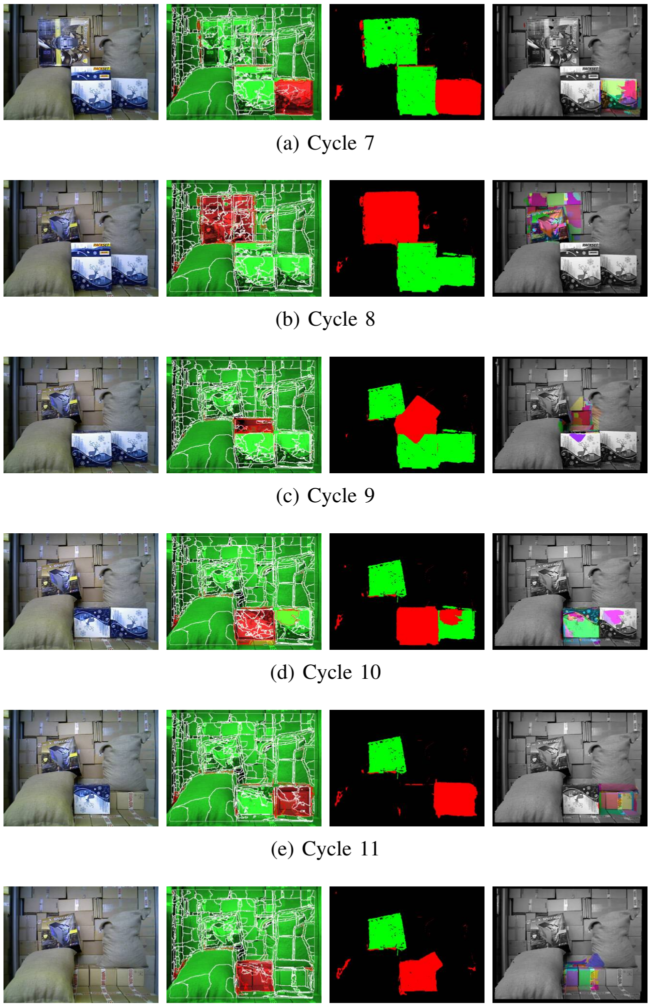
(f) Cycle 6

Fig. 9. Anticipation and attention visualization during the first 6 cycles of the unloading.

As can be seen in the third column of Fig. 9 and Fig. 10, the anticipation-attention algorithm correctly identified all the changes in the scenes with the only false positives in cycle 6 and 10. The parameters (e.g. θ_c from inequality (5)) are tuned conservatively in order to be sensitive to even the slightest changes in the scene. Therefore false positives (regions classified incorrectly as changed) are expected from time to time. This only slightly reduces efficiency, but ensures robustness, which is of high importance to industrial applications. Table I summarizes the performance of the anticipation during the unloading of the container in terms of the ability to classify correctly whether recognized objects were perturbed. In this experiment only one object was falsely classified as perturbed (cycle 6) due to the aforementioned reasons.

Table II and Table III provide quantitative comparison of the recognition pipeline running with anticipation-attention enabled and disabled. In the case of anticipation-attention being enabled, the processing of the full RGBD scan is done only in the first cycle. The subsequent scans are processed only in the attention regions, thus leading to a drastic drop in the computation time. For the scenario used in this work, the overall average runtime has decreased by

[1] https://www.youtube.com/JacobsRobotics



(a) Cycle 7

(b) Cycle 8

(c) Cycle 9

(d) Cycle 10

(e) Cycle 11

(f) Cycle 12

Fig. 10. Anticipation and attention visualization during the last 6 cycles of the unloading

	#Cases	Correctly Classified	Percentage
manipulated/perturbed	12	12	100 %
unperturbed	44	43	98 %

TABLE I

Accuracy of the classification of recognized objects

65%. The time gains for the segmentation and the recognition were 91% and 47% respectively as can be computed from the corresponding columns of Table II. Thus by spending a mere 5% of the perception time on anticipation-attention we got an overall boost in performance of 65%. The reprojection test, which is a major part of the anticipation component, spent on average 170 ms. per object. This can be considerably reduced by parallel computing, since the operations involved are very simple and mostly independent.

Decrease in the computation time does not reduce the accuracy of the recognition as can be inferred from Table III. Both cases had almost the same precision, however recognition with anticipation and attention enabled had a slightly better recall, which can be expected, since it performs a more concentrated search of the objects, thus causing less false negatives.

Anticipation	Average Runtime, s			
	Anticipation	Segmentation	Recognition	Overall
✓	3.08	2.50	15.72	21.31
×	-	27.96	32.96	60.92

TABLE II

The effect of anticipation and attention on perception runtime. First row contains average runtimes when anticipation-attention was enabled, the second - when it was disabled

Anticipation	Recognition				
	TP	FP	FN	Precision	Recall
✓	54	3	2	0.95	0.96
×	52	2	4	0.96	0.93

TABLE III

The effect of anticipation and attention on recognition performance. TP stands for true-positives, FN for false negatives, etc.

VI. Conclusions

We presented an extension to a RGBD robot perception system to increase processing speed while maintaining robustness. We introduced a simple but efficient form of anticipation that uses top down processes in form of reprojection tests to check whether the expected changes of the scene due to the manipulation of an object have taken place or whether there are any unexpected dynamics due to unintended side-effects. Furthermore, we use attention for further speed up by directing the bottom up processing to regions of interests where expected or unexpected changes in the scene have taken place. The method is tested on real world data from an advanced demonstration set-up, namely a challenging industrial application scenario in form of container unloading, but it is applicable to advanced perception and manipulation tasks in general.

Appendix

TABLE IV

Notations

${}^i\mathbf{p} \in \mathbb{R}^3$	Position vector of a spatial point resolved in the reference frame $\mathcal{F}_i$.
${}^i\mathbf{M}$	An RGBD image taken from camera-frame $\mathcal{F}_i$.
${}^i\mathbf{m} \in \mathbb{R}^2$	Image pixel coordinates of a point in an image taken from the camera-frame $\mathcal{F}_i$.
${}^i\underline{\mathbf{m}} \in \mathbb{R}^3$	The homogeneous coordinates for ${}^i\mathbf{m}$.
$\mathbb{M}$	A mask consisting of a set of pixels ${}^i\mathbf{m}$.
$\mathbb{M}_A$	An attention mask consisting of a set of pixels representing the attention region.
$\mathbb{S}$	A Segment consisting of a set of homogeneous and connected pixels ${}^i\mathbf{m}$.
$\mathcal{S}$	A segmentation consisting of a set of segments - $\{\mathbb{S}_1, \mathbb{S}_2, \ldots \mathbb{S}_n\}$.
$\mathcal{S}_A$	A segmentation of the attention region.
$\mathbf{C}$	The camera intrinsic matrix.
$\mathcal{P}$	A colored pointcloud - a set of 3D points with color.
$\mathbb{H}$	A list of detected objects together with their descriptions (IDs, labels), confidences, masks and locations.

References

[1] V. Tincani, M. G. Catalano, E. Farnioli, M. Garabini, G. Grioli, G. Fantoni, and A. Bicchi, "Velvet fingers: A dexterous gripper with active surfaces," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, 2012, pp. 1257–1263.

[2] J. Gancet, P. Weiss, G. Antonelli, M. F. Pfingsthorn, S. Calinon, A. Turetta, C. Walen, D. Urbina, S. Govindaraj, C. A. Mller, X. Martinez, T. Fromm, B. Chemisky, G. Indiveri, G. Casalino, P. A. D. Lillo, E. Simetti, D. D. Palma, A. Birk, A. Tanwani, I. Havoutis, A. Caffaz, L. Guilpain, and P. Letier, “Dexterous undersea interventions with far distance onshore supervision: the dexrov project,” in *10th IFAC Conference on Control Applications in Marine Systems (CAMS)*. International Federation of Automatic Control, 2016.

[3] T. Stoyanov, N. Vaskevicius, C. A. Mueller, T. Fromm, R. Krug, V. Tincani, R. Mojtahedzadeh, S. Kunaschk, R. Mortensen Ernits, D. Ricao Canelhas, M. Bonilla, S. Schwertfeger, M. Bonini, H. Halfar, K. Pathak, M. Rohde, G. Fantoni, A. Bicchi, A. Birk, A. Lilienthal, and W. Echelmeyer, “No More Heavy Lifting: Robotic Solutions to the Container Unloading Problem,” *Robotics and Automation Magazine*, 2016, in press.

[4] N. Vaskevicius, K. Pathak, A.-E. Ichim, and A. Birk, “The Jacobs Robotics approach to object recognition and localization in the context of the ICRA’11 Solutions in Perception Challenge,” in *IEEE Conf. on Robotics and Automation*, St. Paul, MN, USA, May 2012.

[5] N. Vaskevicius, C. Mueller, M. Bonilla, V. Tincani, T. Stoyanov, G. Fantoni, K. Pathak, A. Lilienthal, A. Bicchi, and A. Birk, “Object recognition and localization for robust grasping with a dexterous gripper in the context of container unloading,” in *Automation Science and Engineering (CASE), 2014 IEEE International Conference on*, Aug 2014, pp. 1270–1277.

[6] R. Krug, T. Stoyanov, M. Bonilla, V. Tincani, N. Vaskevicius, G. Fantoni, A. Birk, A. Lilienthal, and A. Bicchi, “Velvet fingers: Grasp planning and execution for an underactuated gripper with active surfaces,” in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, May 2014, pp. 3669–3675.

[7] G. Hesslow, “Conscious thought as simulation of behaviour and perception,” *Trends in Cognitive Sciences*, vol. 6, no. 6, pp. 242–247, 2002.

[8] E. Datteri, G. Teti, C. Laschi, G. Tamburrini, G. Dario, and E. Guglielmelli, “Expected perception: an anticipation-based perception-action scheme in robots,” in *Intelligent Robots and Systems, 2003. (IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, vol. 1, 2003, pp. 934–939 vol.1.

[9] M. Mora and J. Tornero, “Path planning and trajectory generation using multi-rate predictive artificial potential fields,” in *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, 2008, pp. 2990–2995.

[10] F. Alnajjar, A. Hafiz, I. Bin Mohd Zin, and K. Murase, “Vision-sensorimotor abstraction and imagination towards exploring robot’s inner world,” in *Neural Networks, 2008. IJCNN 2008. (IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on*, 2008, pp. 2418–2424.

[11] V. Stephan and H.-M. Gross, “Neural anticipative architecture for expectation driven perception,” in *Systems, Man, and Cybernetics, 2001 IEEE International Conference on*, vol. 4, 2001, pp. 2275–2280 vol.4.

[12] G. Zhang and H. Suh, “Integration of a prediction mechanism with a sensor model: An anticipatory bayes filter,” in *Robotics and Automation, 2009. ICRA ’09. IEEE International Conference on*, 2009, pp. 3620–3625.

[13] J. J. Gibson, *The Theory of Affordances*. Erlbaum, 1977, pp. 67–82.

[14] ——, *The Ecological Approach to Visual Perception*. Boston: Houghton Mifflin, 1979.

[15] L. Paletta, G. Fritz, F. Kintzler, J. Irran, and G. Dorffner, “Learning to perceive affordances in a framework of developmental embodied cognition,” in *Development and Learning, 2007. ICDL 2007. IEEE 6th International Conference on*, 2007, pp. 110–115.

[16] G. Fritz, L. Paletta, R. Breithaupt, and E. Rome, “Learning predictive features in affordance based robotic perception systems,” in *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, 2006, pp. 3642–3647.

[17] F. Ingrand and O. Despouys, “Extending procedural reasoning toward robot actions planning,” in *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, vol. 1, 2001, pp. 9–14 vol.1.

[18] S. Frintrop, E. Rome, and H. I. Christensen, “Computational visual attention systems and their cognitive foundations: A survey,” *ACM Trans. Appl. Percept.*, vol. 7, no. 1, pp. 1–39, 2010.

[19] L. Itti and C. Koch, “Computational modelling of visual attention,” *Nature Reviews Neuroscience*, vol. 2, no. 3, pp. 194–203, 2001, 10.1038/35058500.

[20] L. Itti, C. Koch, and E. Niebur, “A model of saliency-based visual attention for rapid scene analysis,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 20, no. 11, pp. 1254–1259, 1998.

[21] D. Walther and C. Koch, “2006 special issue: Modeling attention to salient proto-objects,” *Neural Netw.*, vol. 19, no. 9, pp. 1395–1407, 2006.

[22] H. Xiaodi and Z. Liqing, “Saliency detection: A spectral residual approach,” in *Computer Vision and Pattern Recognition, 2007. CVPR ’07. IEEE Conference on*, 2007, pp. 1–8.

[23] X. Tingting, W. Hao, Z. Tianguang, K. Kuhnlenz, and M. Buss, “Environment adapted active multi-focal vision system for object detection,” in *Robotics and Automation, 2009. ICRA ’09. IEEE International Conference on*, 2009, pp. 2418–2423.

[24] K. Joo-Young, K. Minsu, L. Seungjin, O. Jinwook, K. Kwanho, O. Sejong, W. Jeong-Ho, K. Donghyun, and Y. Hoi-Jun, “A 201.4gops 496mw real-time multi-object recognition processor with bio-inspired neural perception engine,” in *Solid-State Circuits Conference - Digest of Technical Papers, 2009. ISSCC 2009. IEEE International*, 2009, pp. 150–151,151a.

[25] W. Hou, X. Gao, D. Tao, and X. Li, “Visual saliency detection using information divergence,” *Pattern Recognition*, vol. 46, no. 10, pp. 2658–2669, 2013.

[26] T. N. Vikram, M. Tscherepanow, and B. Wrede, “A saliency map based on sampling an image into random rectangular regions of interest,” *Pattern Recognition*, vol. 45, no. 9, pp. 3114–3124, 2012.

[27] W. Maier and E. Steinbach, “Surprise-driven acquisition of visual object representations for cognitive mobile robots,” in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, 2011, pp. 1621–1626.

[28] C. Changhyun and H. I. Christensen, “Cognitive vision for efficient scene processing and object categorization in highly cluttered environments,” in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, 2009, pp. 4267–4274.

[29] P. E. Forssen, D. Meger, K. Lai, S. Helmer, J. J. Little, and D. G. Lowe, “Informed visual search: Combining attention and object recognition,” in *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, 2008, pp. 935–942.

[30] C. Siagian and L. Itti, “Rapid biologically-inspired scene classification using features shared with visual attention,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 2, pp. 300–312, 2007.

[31] K. Amlacher, G. Fritz, P. Luley, A. Almer, and L. Paletta, “Geo-contextual priors for attentive urban object recognition,” in *Robotics and Automation, 2009. ICRA ’09. IEEE International Conference on*, 2009, pp. 1214–1219.

[32] C. Beltran-Gonzalez and G. Sandini, “Visual attention priming based on crossmodal expectations,” in *Intelligent Robots and Systems, 2005. (IROS 2005). 2005 IEEE/RSJ International Conference on*, 2005, pp. 1060–1065.

[33] O. Sugiyama, T. Kanda, M. Imai, H. Ishiguro, and N. Hagita, “Natural deictic communication with humanoid robots,” in *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, 2007, pp. 1441–1448.

[34] B. Moller, S. Posch, A. Haasch, J. Fritsch, and G. Sagerer, “Interactive object learning for robot companions using mosaic images,” in *Intelligent Robots and Systems, 2005. (IROS 2005). 2005 IEEE/RSJ International Conference on*, 2005, pp. 2650–2655.

[35] L. Shuyin, M. Kleinehagenbrock, J. Fritsch, B. Wrede, and G. Sagerer, “”biron, let me show you something”: evaluating the interaction with a robot companion,” in *Systems, Man and Cybernetics, 2004 IEEE International Conference on*, vol. 3, 2004, pp. 2827–2834 vol.3.

[36] M. Vincze, W. Wohlkinger, S. Olufs, P. Einramhof, and R. Schwarz, “Towards bringing robots into homes,” in *Robotics (ISR), 41st International Symposium on*, 2010, pp. 1–6.

[37] J. J. Craig, *Introduction to Robotics*. Addison-Wesley, 1989.

[38] R.-G. Mihalyi, K. Pathak, N. Vaskevicius, T. Fromm, and A. Birk, “Robust 3d object modeling with a low-cost rgbd-sensor and ar-markers for applications with untrained end-users,” *Robotics and Autonomous Systems (RAS)*, vol. 66, pp. 1–17, 2015.

[39] D. Comaniciu and P. Meer, “Mean shift: a robust approach toward feature space analysis,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, May 2002.