Scene Interpretation for Self-Aware Cognitive Robots

Melodi Deniz Ozturk and **Mustafa Ersen** and **Melis Kapotoglu** and **Cagatay Koc** and

Sanem Sariel-Talay and **Hulya Yalcin**

Artificial Intelligence and Robotics Laboratory
Istanbul Technical University, Istanbul, Turkey
{ozturkm,ersenm,kapotoglu,kocca,sariel,hulyayalcin}@itu.edu.tr

Abstract

We propose a visual scene interpretation system for cognitive robots to maintain a consistent world model about their environments. This interpretation system is for our lifelong experimental learning framework that allows robots analyze failure contexts to ensure robustness in their future tasks. To efficiently analyze failure contexts, scenes should be interpreted appropriately. In our system, LINE-MOD and HS histograms are used to recognize objects with/without textures. Moreover, depth-based segmentation is applied for identifying unknown objects in the scene. This information is also used to augment the recognition performance. The world model includes not only the objects detected in the environment but also their spatial relations to efficiently represent contexts. Extracting unary and binary relations such as *on*, *on_ground*, *clear* and *near* is useful for symbolic representation of the scenes. We test the performance of our system on recognizing objects, determining spatial predicates, and maintaining consistency of the world model of the robot in the real world. Our preliminary results reveal that our system can be successfully used to extract spatial relations in a scene and to create a consistent model of the world by using the information gathered from the onboard RGB-D sensor as the robot explores its environment.

Introduction

A cognitive robot may face several types of failures during the execution of its actions in the real world. These failures may arise due to the gap between the real-world facts and their symbolic representations used during planning, unexpected events that may change the current state of the world or internal problems (Karapinar, Altan, and Sariel-Talay 2012). The robot should gain experience from these failures and use this experience in its future tasks, which requires tight integration of continual planning, monitoring, reasoning and lifelong experimental learning (Karapinar et al. 2013). Efficient and consistent scene interpretation is a prerequisite for these cognitive abilities. In this work, we propose a consistent world modelling system for this purpose. The system continually monitors the environment to detect and recognize objects and to determine spatial relations among them using visual data from an RGB-D camera during the operation of the robot. As a motivating example to illustrate the stated problem, consider an object manipulation task in the blocks world domain. An example plan constructed for a three-block problem is given in Figure 1. In this toy problem, where all blocks are initially on the table without any other objects on top of them (i.e., satisfying the *clear* predicate), the aim is stacking three blocks on top of each other. The required spatial predicates to be extracted in this domain are *on*, *on_table* and *clear*. During the execution of its plan, the robot may fail in executing action *stack*. Possible reasons for this failure might be a vision problem or unstability of the destination stack. The relation between the context and the failure can be specified as follows:

$$holding(a) \wedge clear(b) \wedge on\_table(c) \wedge on(b,c) \Rightarrow StackFailure \quad (1)$$

where the interpretations in the premise part of this conclusion should be extracted from the scene. To ensure robustness in such cases, the robot needs to continuously monitor the state space for anomalies during action execution, which makes it very important to possess a model of the world consistent with the real environment.

Throughout the paper, we first present background information on the areas of object recognition and scene interpretation. Then, we describe the details of our system for maintaining a knowledge base of objects and for extracting spatial relations among them in order to monitor failures. We

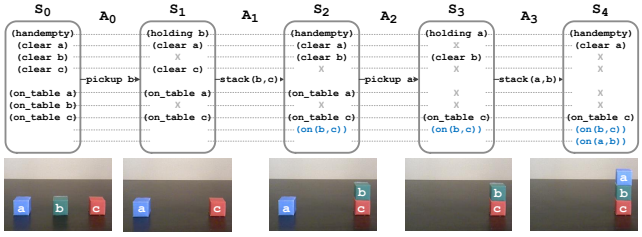


Figure 1: The execution trace for solving the blocks world problem with a three-block case is given. (top) The states (S_i) and the actions (A_j) taken at each state, (bottom) the visual scene observed at each world state are presented.

then give empirical results of our approach followed by the conclusions.

Background

In this section, we give a brief review of the approaches used for recognizing objects and determining spatial relations in a scene. Then, in the following section, we present our system for maintaining a consistent world model with spatial relations.

Object Recognition

There are various approaches for recognition of objects in a scene using different types of visual clues. These approaches can be categorized as 2D object recognition approaches based on local invariant feature descriptors and 3D object recognition approaches based on surface normals computed from the depth map. In the case of 2D color data, local feature descriptors are used to determine patterns in the image which differ from the other pixels in their neighborhood. These distinguishing parts of the image (i.e., keypoints) are generally chosen by considering sharp changes in color intensity and texture. To store the keypoints, descriptors are computed around them which are suitable for measuring their similarities. The idea of using local invariant descriptors became popular when Scale-Invariant Feature Transform (SIFT) (Lowe 1999) was proposed in 1999. SIFT is a keystone in the area, and it is used as the base of the state of the art techniques. It is known to be invariant against geometric transformations such as scale, rotation, translation and affine transformation to a sufficient extent for a lot of applications. It is also claimed to perform well against noises and changes in the illumination. However, SIFT-based approaches are known to have deficiencies in recognizing textureless objects. Information on the 3D shapes of the objects and their colors can be used in order to deal with this problem. By the development of RGB-D sensors, it is possible to get depth information as well as color and texture information for this purpose. To utilize the depth values captured using these types of sensors, different 3D descriptors have been proposed (Aldoma et al. 2012). These descriptors can be divided in two categories: local descriptors and global descriptors. Local descriptors are used to describe the local geometric properties of distinguishing points (i.e., keypoints) whereas global descriptors capture depth-based features globally for a presegmented object without storing local information for extracted descriptors. Among these, LINE-MOD (Hinterstoisser et al. 2012) is unique as it is a linearized multi-modal template matching approach based on weak orientational features which can be used to recognize objects very fast making this approach the most suitable one for real-time robotic applications.

Spatial Relation Extraction

Detecting and representing structures with spatial relations in a scene is known as the scene interpretation problem. While this is a trivial task for humans, interpreting spatial relations by processing visual information from artificial vision systems is not a totally solved problem for autonomous agents (Neumann and Möller 2008). In the recent years, some approaches have been proposed to solve this problem. Some of these works use 2D visual information for extracting qualitative spatial representations in a scene (Falomir et al. 2011; Sokeh, Gould, and Renz 2013). In these works, some topological and orientational relations among objects are determined in the scene. In another work, Sjöö et al. have proposed a method for determining topological spatial relations *on* and *in* among the objects, and this information is used to guide the visual search of a robot for the objects in the scene (Sjöö, Aydemir, and Jensfelt 2012). Object recognition approach used in their work is based on matching SIFT (Lowe 1999) keypoints on a monocular image of the environment.

Moreover, there are studies on using semantic knowledge for scene interpretation. In one of these studies, challenges are identified for using high-level semantic knowledge to reason about objects in the environment (Gurău and Nüchter 2013). In another study, a system is proposed for reasoning about spatial relations based on context (e.g., nearby objects, functional use etc.) in previously unseen scenes (Hawes et al. 2012). In another work, proximity-based high-level relations (e.g., relative object positions to find objects that are generally placed together) are determined by comparing Euclidean distance between pairs of recognized objects in the scene (Kasper, Jäkel, and Dillmann 2011). This system relies on 3D data obtained using an RGB-D sensor and an AR-Toolkit marker acting as a reference coordinate system. Finally, Elfring et al. have proposed a remarkable approach for associating data from different sources into a semantically rich world model (Elfring et al. 2013). Their approach is based on using probabilistic multiple hypothesis anchoring.

Our proposed work differs from the previous studies in three ways. First, a 3D model of the world is created by combining object recognition and segmentation results with reasoning in a knowledge base. Second, spatial relations are determined for a higher level task of detecting failures after action executions. Third, the object recognition system used in this work is more generic as it can deal with textureless objects that do not have any distinguishing texture information.

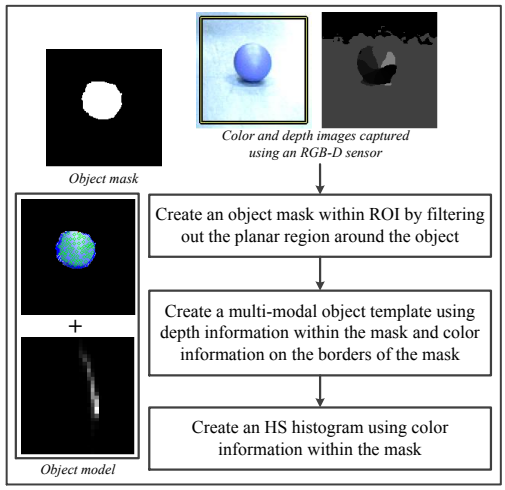
Consistent World Modelling

We propose a visual scene interpretation system to represent world states symbolically for monitoring action execution in cognitive robotic tasks. Our system involves two main procedures, namely, object recognition to detect and label objects in the scene, and scene interpretation to maintain a consistent world model to represent some useful spatial relations among objects for manipulation scenarios.

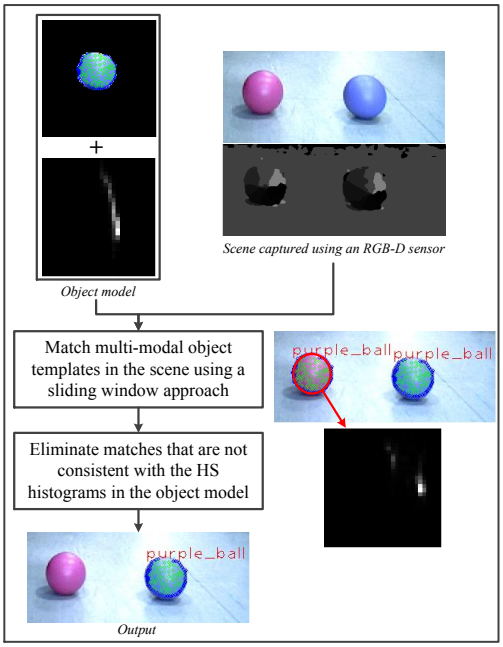
3D Object Recognition

In our system, we use LINE-MOD (Hinterstoisser et al. 2012) as a basis for recognizing objects in the scene. LINE-MOD is a multi-modal template matching approach primarily designed for textureless objects where each template encodes the surface normals and the color gradients on the

object from a viewpoint. We have used the beta version of LINE-MOD which is available in OpenCV. In this version, color information on the surface of the object is not taken into account while constructing templates, and the color gradients are extracted only around the boundaries. To exploit the color information in a more efficient way, a histogram is generated to model each template in Hue-Saturation-Value (HSV) color space (Ersen, Sariel-Talay, and Yalcin 2013). V (value) channel is not considered while constructing these histograms as it is directly dependent on the intensity of the light source in the environment (Figure 2(a)).



(a) Model creation



(b) Recognition

Figure 2: The phases of (a) modelling and (b) recognizing an object.

Object recognition is performed by matching multi-modal LINE-MOD templates in the scene and then verifying the matches by considering their corresponding HS histograms (Figure 2(b)). Similarity threshold for template matching is specified as 80% by taking into account the noisy data captured using an ordinary RGB-D sensor and a moderate value (i.e., 0.4) is preferred as the threshold for correlation of the histograms to enhance robustness against the changes in the illumination.

Scene Interpretation

In the physical world, there is uncertainty in the data gathered through sensors due to different factors like varying illumination conditions or dynamic environments. This makes bare object recognition results unreliable for cognitive robotic applications. To handle this issue, we have devised a temporal scene interpretation system that combines information from a sequence of frames rather than relying only on the current scene to maintain a consistent knowledge base (KB) of objects ($o_i \in \mathcal{O}$). Moreover, spatial relations among existing objects are included in the KB in order to keep a more comprehensive world model. The proposed scene interpretation system is used in our Pioneer 3-AT robot (Figure 3).

To maintain a consistent KB representing the world, it is beneficial to use different sources or forms of sensory data ($s_j \in \mathcal{S}$). For this reason, we combine the recognition results with depth-based segmentation. This enables the system to gather more information about the objects that cannot be detected by the recognition system until they are recognized correctly. Moreover, this is also useful for detection of unknown objects in the robot's environment. For this purpose, the 3D point cloud data acquired from the RGB-D sensor is segmented using Euclidean clustering. To distinguish real objects from clutter, the obtained segments are filtered considering both their sizes and whether they lie on the ground. They are then included in the world model as yet-undefined objects. Figure 4 shows an example in which LINE-MOD (Figure 4(a)), HS histograms (Figure 4(b)) and segmentation outputs (Figure 4(c)) contribute to the KB (Figure 4(d)). As seen in Figure 4(d), three objects are recognized and included in the world model with their color information.

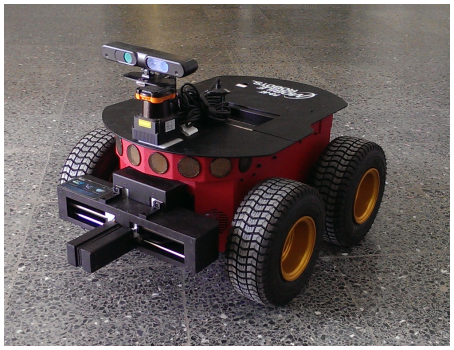
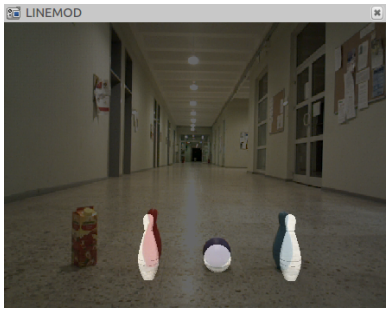
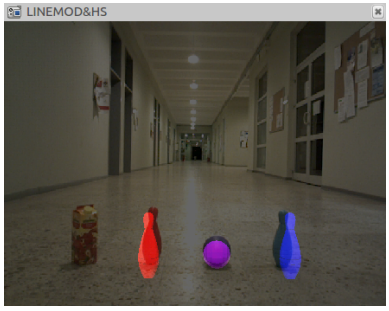


Figure 3: Our Pioneer 3-AT robot with a laser scanner and Asus Xtion Pro Live RGB-D sensor mounted on top of it.

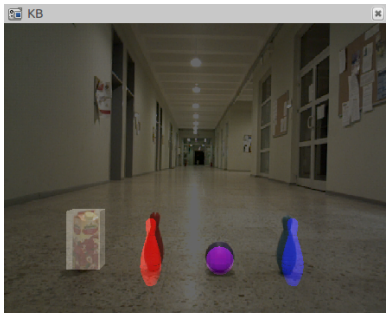


(a)



(b)

(c)



(d)

Figure 4: Construction of the KB: (a) LINE-MOD results, (b) LINE-MOD&HS results, (c) segmentation results and (d) KB fusing all these information.

However, the leftmost object cannot be recognized as it is not modelled, and only the segmentation result is taken into account while adding it to the KB as an unknown object.

The certainty of each object o_i's existence in the environment at time step t is represented in the KB as a confidence value $0 \leq c_i^t \leq 1$ which is initially defined as 0 and updated as follows whenever a new observation is taken:

$$c_i^t = c_i^{t-1} + \begin{cases} -0.1, & \text{if } \sum_{s_j} c_{ij}^t = 0 \wedge \sum_{s_j} f_{ij}^t > 0 \\ \sum_{s_j} w_j \cdot f_{ij}^t \cdot c_{ij}^t, & \text{if } \sum_{s_j} c_{ij}^t > 0 \wedge c_i^{t-1} < 1 \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

where c_{ij}^t is the matching similarity of the recognition of o_i by the source s_j. Similarity value for each extracted segment is taken as 1 since the segmentation algorithm does not have a scale to measure reliability. w_j is the weight of the source s_j which is empirically determined. According to our experiments, object recognition using the color data (i.e., LINE-MOD&HS) gives more reliable results than using only LINE-MOD and thus its w_j is greater. The field of view coefficient, $0 \leq f_{ij}^t \leq 1$, represents the expectation that object o_i is detected regarding its location and the field of view of source s_j. It is determined based on the boundaries of the corresponding field of view of a source. These boundaries (as illustrated in Figure 5 for the RGB-D sensor) are determined experimentally and define the visually reachable area from the sensor's point of view. Establishing the RGB-D sensor on the robot as the origin, the area indicated in dark blue shows the limited region in which the vision algorithms are observed to detect objects with the highest probability. In the area indicated by light blue, visual detection performance is observed to decrease starting from the curve separating the two regions onwards to the end of the second region. Any information about the objects detected in the latter area is considered to be more unreliable as the location of the recognition gets closer to the outer boundary of the field of view. No considerable recognition is expected outside of these regions. Using these boundaries, f_{ij}^t is calculated as follows:

$$f_{ij}^t = \frac{max\_dist - dist_{ij}}{max\_dist - def\_dist} \quad (3)$$

where $dist_{ij}$ is the distance of the object from the RGB-D sensor, def_dist is the curve separating the two regions onwards to the end of the second region, and max_dist is the outer boundary of the field of view. These regions are also used to expect the detection of objects whose models are registered to the KB previously and that are likely to be seen because they lie in the field of view.

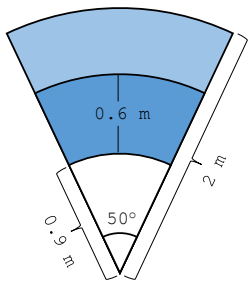


Figure 5: The field of view of our Pioneer 3-AT robot.

An example is given to illustrate how a recognized object's information is added into the KB (Figure 6). First, the object is detected by only LINE-MOD, and its color information is associated by LINE-MOD&HS. When the object is taken out of the scene and thus cannot be detected by the aforementioned methods any longer, its confidence is gradually decreased over time until it is decided that the object does not exist in the world any longer for some external reason. This case is illustrated in Figure 7 where the confidence on the recognition is indicated by the transparency of the color: as the confidence value increases, the representing color gains solidity.

According to Equation 2, if the expected objects in the field of view of the robot are not detected ($\Sigma_{s_j} c_{ij}^t = 0 \wedge \Sigma_{s_j} f_{ij}^t > 0$), their confidence values are decreased. However, the confidence values of the objects that are not expected to be observed remain the same as in the following equation:

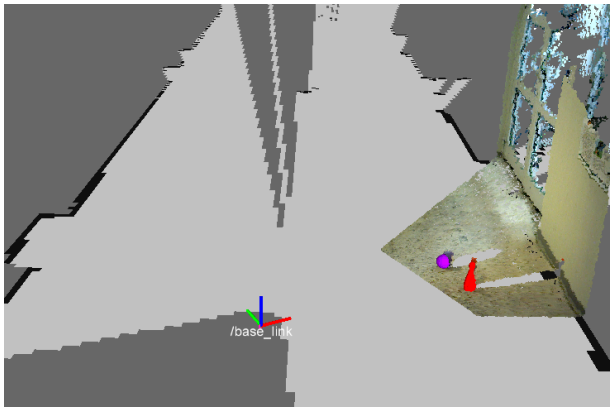
$$\forall o_i, (\sum_{s_j} f_{ij}^t = 0) \Rightarrow c_i^t = c_i^{t-1} \tag{4}$$

An example scenario is presented in Figure 8 where the generated map of the environment, the registered objects in the KB and the 3D point cloud of the scene from the robot's instant view are overlapped. In this scenario, the confidence values of the previously registered objects to the KB that are not located in the robot's field of view remain the same when the robot turns left.

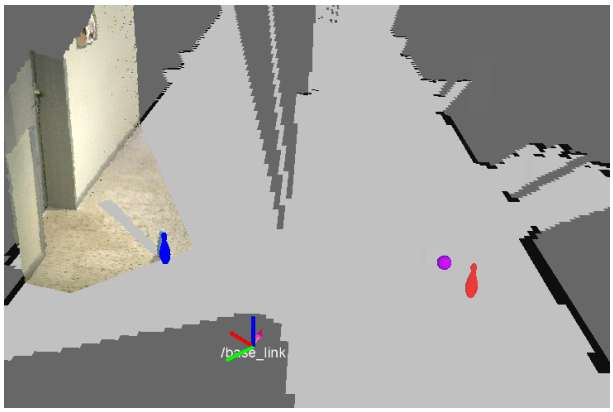
If two object types are recognized in overlapping 3D regions in the environment, they are assumed to belong to a single object, and the matching similarity (c_{ij}^t) is used to decide on the type of the object. Every different object type recognized for a single item in the world is kept in a weighted list: as one object type gets recognized more often, its weight increases. The maximum weight value determines the interpreted type. Objects only detected by the segmentation algorithm keep no type information until the object gets recognized by one of the vision algorithms. A similar

Figure 6: Adding a newly recognized object to the KB.

Figure 7: Updating the KB when an in-view object is removed from the scene.



(a)



(b)

Figure 8: As the robot turns left and observes new objects, the world model is updated while previously recognized out of view objects are preserved in the KB.

weighted list is used for recognitions of different colors regarding a single object to determine its most probable color.

In addition to type, color and confidence, the location of an existing object is adjusted with every new instance of its detection. Upon every new detection of an existing object in the KB, the position information regarding that recognition is taken into account to determine a more correct estimation for the location of the object. This adjustment helps in correcting misplacements or tolerating small object movements.

$$\mathbf{loc}_i^t = \frac{c_i^{t-1} \cdot \mathbf{loc}_i^{t-1} + \sum_{s_j} (w_j \cdot f_{ij}^t \cdot c_{ij}^t \cdot \mathbf{loc}_{ij}^t)}{c_i^t} \tag{5}$$

where $\mathbf{loc}_i^{t-1}$ is the registered location vector of o_i at timestep $t-1$ in the KB, and $\mathbf{loc}_{ij}^t$ is the newly found vector coordinates for the same object.

The world model is also updated when the robot acts upon its environment. If an object is being moved by the robot, while it is sensed that it is still in the robot's gripper, its location is updated in accordance to the location of the gripper. If the gripper is sensed empty (i.e., the robot puts the object

down or drops it before reaching its intended destination), it is assumed that the object preserves its last known location, until further information can be acquired by the robot from its sensors.

Determining Spatial Object Relations Extracting spatial predicates of and relations among objects is essential for accurate creation of the world model and interpretation of the scenes. Uncertainties are also taken into account for the updates of relational predicates. The following are the predicates considered for object manipulation tasks:

The *on_ground/on_table* relation: If the distance between the bottom surface of an object and the ground/table is observed to be within a certain threshold and no other objects can be detected under this object, it is determined to be on the ground or table. Surfaces are determined by plane segmentation.

The *on* relation: If two objects' projections on the ground intersect, and the distance between the bottom surface of the higher object and the top surface of the lower object is within a certain threshold, the object at the higher position is determined to be *on* the lower object. The following formula is used to check the *on* relation:

$$\forall o_i, o_k, \neg(DC_{xy}(o_i, o_k) \vee EC_{xy}(o_i, o_k)) \wedge UP(o_i, o_k) \Rightarrow on(o_i, o_k) \quad (6)$$

where *DC(disconnected)* and *EC(externally connected)* are topological predicates of RCC8 (Randell, Cui, and Cohn 1992) and *UP* is a directional predicate which can be considered as the 3D expansion of *N(north)* from cardinal direction calculus (Frank 1991).

As the objects on the top can obscure the ones on the bottom from some points of view, once an *on* relation is established and as long as the top object's position does not change enough to be considered as *on_ground*, the bottom object is kept in the KB even if it cannot be detected by the recognition algorithms any longer, which is formulated as follows:

$$\forall o_i, o_k, on(o_i, o_k) \wedge \neg on\_ground(o_i) \wedge \sum_{s_j} c_{kj}^t = 0 \Rightarrow c_k^t = c_k^{t-1} \quad (7)$$

Every *on* relation has a *stability* property, which is to be used for failure detection. This value is computed as follows:

$$stability(on(o_i, o_k)) = \prod_{dim=\{x,y\}} \frac{size_{i,dim}/2 - offset_{i,dim}}{size_{i,dim}/2} \quad (8)$$

where $size_{i,dim}$ denotes the size of the top object o_i in the dimensions x and y parallel to the ground. The $offset_{i,dim}$ denotes the offset of o_i (i.e., unsupported part by the bottom object). If the top object's complete area is supported by the bottom object, the *on* relation is assumed to have a stability of 1. Otherwise, the stability value decreases in correlation with the percent of the top object that is unsupported.

In accordance with the robot's movements, the *on* relation is considered invalid if the top object is picked up by the robot successfully. Any object that has no other objects on top of it is considered to have the *clear* relation, which indicates that the object is free to be picked up by the robot.

The *near* relation: The *near* predicate related to the proximity of two objects is computed taking the relative sizes of the objects into account as follows:

$$\begin{aligned} \forall o_i, o_k, (size_{i,x} + size_{k,x} \geq dist_{ik,x}) \\ \wedge (size_{i,y} + size_{k,y} \geq dist_{ik,y}) \\ \wedge (size_{i,z} + size_{k,z} \geq dist_{ik,z}) \\ \wedge (\neg on(o_i, o_k) \wedge \neg on(o_k, o_i)) \\ \Rightarrow near(o_i, o_k) \end{aligned} \quad (9)$$

where $dist_{ik,x}$ denotes the distance between o_i and o_k in dimension x. If the distance between any two objects' centers of mass is less than or equal to the sum of the two objects' sizes in all three dimensions, the two objects are considered to be *near* to each other. Note that *on* relation has a greater importance than *near* relation, and these relations are considered to be mutually exclusive to each other.

Experimental Evaluation

The proposed system is evaluated in real time for different possible situations using the real-world data captured by the RGB-D sensor mounted on top of our Pioneer 3-AT robot. We have selected a set of objects having different shapes and colors as illustrated in Figure 9 for these experiments. These objects involve two plastic bowling pins with different colors, two small plastic balls with different colors, a bigger beach ball and two paper boxes with different sizes and colors. First, we have evaluated the overall object recognition performance. Then, we have tested the scene interpretation performance of the system on an object manipulation scenario involving the objects shown in Figure 9.

In the first set of experiments, the overall recognition performance has been evaluated by comparing the results of LINE-MOD and our approach combining LINE-MOD with HS histograms. The evaluation has been performed on a scene involving only the objects of interest in 10 different configurations for each object by changing its position. The

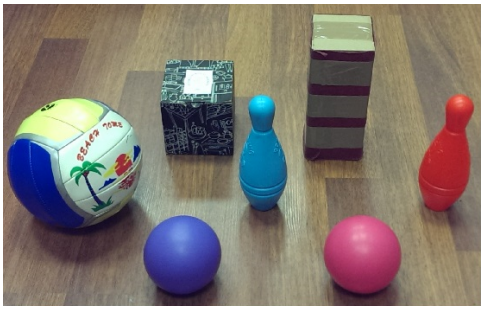


Figure 9: The objects used in the experiments.

results are illustrated in Table 1 and Table 2 as confusion matrices. As expected, LINE-MOD is generally successful for distinguishing objects by using differences in their geometrical shapes. However, it has problems with similar shaped objects but with different colors. For example, it cannot distinguish different colored balls from each other. Furthermore, it sometimes confuses different shaped objects as well due to quantization errors (e.g., the black box is confused with the similar sized balls in Table 1). Our approach based on checking HS histogram correlations on the results obtained using LINE-MOD leads to much better results in these situations as presented in Table 2. The only observed problem with our approach is that false negatives may occur as some correct results are eliminated by checking color correlation in different illumination conditions.

In the second set of experiments, we have tested the success of our system to maintain the consistency of the world model using 10 trials for each criteria in varying conditions (Figure 10). First, we have checked whether the model can successfully detect undefined objects using the segmentation output. We have seen that unknown objects can be detected with 90% success rate. Segmentation only fails in detecting objects when they touch each other. Next, we have evaluated the consistency of knowledge base updates concerning newly added objects to and removed objects from the scene. The average success of the former one is 90% due to recognition and/or segmentation errors, while the latter one is observed to be 100% successful in all trials. As the object manipulation update (i.e., holding an object, changing its location and putting it down) and out of view object preservation operations are not affected by the uncertainty factor in object recognition or segmentation, the system performed 100% success in making the necessary updates.

In the final set of experiments, the performance of our system for extracting spatial relations *on* (with its stability), *clear*, *on_ground* and *near* has been tested. The system is evaluated in 30 different scenes where 10 scenes involve objects having *on* relation, 10 scenes involve objects having *near* relation and the other scenes involve no pairwise object relations. The results are presented in Table 3. As given in these results, our system can be used to successfully detect relations for all the objects used in on-ground object manipulation scenarios. The highest error rates are observed for the *on* relation and determining its stability where the former is mainly caused by object recognition problems (e.g., recognition of two black boxes on top of each other instead of the striped box) and the latter is due to alignment problems. The success in determining the *on* relation also accounts for the errors in the *clear* and *on_ground* relations. *near* relation is observed to be determined without any errors.

Table 1: Confusion matrix for LINE-MOD recognition.

	blue pin	*red pin*	*purple ball*	*pink ball*	*beach ball*	*black box*	*striped box*	*not found*
blue pin	10	10	7	7	0	0	0	0
red pin	10	10	5	6	0	0	0	0
purple ball	0	0	10	10	3	0	0	0
pink ball	0	0	10	10	2	0	0	0
beach ball	0	0	10	10	10	0	0	0
black box	0	0	3	3	1	10	0	0
striped box	0	0	2	2	0	6	10	0

Table 2: Confusion matrix for LINE-MOD&HS recognition.

	blue pin	*red pin*	*purple ball*	*pink ball*	*beach ball*	*black box*	*striped box*	*not found*
blue pin	10	0	0	0	0	0	0	0
red pin	0	10	0	2	0	0	0	0
purple ball	0	0	9	0	1	0	0	1
pink ball	0	0	0	9	0	0	0	1
beach ball	0	0	0	0	10	0	0	0
black box	0	0	0	0	1	10	0	0
striped box	0	0	0	1	0	2	10	0

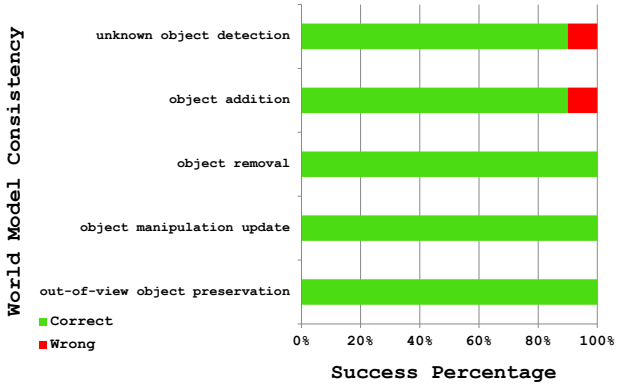


Figure 10: Performance of the proposed system on maintaining a consistent world model.

Table 3: Performance of the proposed system on determining spatial relations.

	Precision	Recall	F-Score
on	90.00%	90.00%	90.00%
stable	88.89%	100.00%	94.12%
clear	98.00%	98.00%	98.00%
on_ground	98.00%	98.00%	98.00%
near	100.00%	100.00%	100.00%

Conclusions

We have presented an approach for temporal scene interpretation and automated extraction of spatial relations to be used in a lifelong experimental learning framework for ensuring robust task execution in cognitive robotic applications. Our approach is based on using visual information extracted from the scenes captured as our ground robot Pioneer 3-AT explores its environment. This information is used to build a knowledge base with locations of objects used in manipulation scenarios and spatial relations among them in the physical world. First, we have shown how our system can be used to recognize objects with different geometric shapes and colors. Then, we have given the details of the vi-

sual scene interpreter for specifying spatial relations among the objects of interest and creating a consistent world model. The results of the conducted experiments on our system indicate that the system can be used to successfully model the environment with objects and spatial relations among them by combining recognition and segmentation results from observed scenes as the robot performs its actions. In our future studies, we plan to integrate temporal reasoning into spatial reasoning in order to detect the possible causes of failures from previous states (e.g., an unstable stack of blocks causing a failure when stacking another block on top of them).

Acknowledgments

This research is funded by a grant from the Scientific and Technological Research Council of Turkey (TUBITAK), Grant No. 111E-286. We thank Prof. Muhittin Gokmen for his recommendations on vision algorithms. We also thank Dogan Altan and Mehmet Biberci for their helpful comments.

References

Aldoma, A.; Marton, Z.-C.; Tombari, F.; Wohlkinger, W.; Potthast, C.; Zeisl, B.; Rusu, R. B.; Gedikli, S.; and Vincze, M. 2012. Tutorial: Point cloud library: Three-dimensional object recognition and 6 DOF pose estimation. *IEEE Robotics and Automation Magazine* 19(3):80–91.

Elfring, J.; van den Dries, S.; van de Molengraft, M. J. G.; and Steinbuch, M. 2013. Semantic world modeling using probabilistic multiple hypothesis anchoring. *Robotics and Autonomous Systems* 61(2):95–105.

Ersen, M.; Sariel-Talay, S.; and Yalcin, H. 2013. Extracting spatial relations among objects for failure detection. In *Proc. of the KI 2013 Workshop on Visual and Spatial Cognition*, 13–20.

Falomir, Z.; Jiménez-Ruiz, E.; Escrig, M. T.; and Museros, L. 2011. Describing images using qualitative models and description logics. *Spatial Cognition and Computation* 11(1):45–74.

Frank, A. U. 1991. Qualitative spatial reasoning with cardinal directions. In *Proceedings of the 7th Austrian Conference on Artificial Intelligence*, 157–167.

Gurău, C., and Nüchter, A. 2013. Challenges in using semantic knowledge for 3D object classification. In *Proc. of the KI 2013 Workshop on Visual and Spatial Cognition*, 29–35.

Hawes, N.; Klenk, M.; Lockwood, K.; Horn, G. S.; and Kelleher, J. D. 2012. Towards a cognitive system that can recognize spatial regions based on context. In *Proc. of the 26th AAAI Conference on Artificial Intelligence (AAAI'12)*, 200–206.

Hinterstoisser, S.; Cagniart, C.; Ilic, S.; Sturm, P. F.; Navab, N.; Fua, P.; and Lepetit, V. 2012. Gradient response maps for real-time detection of textureless objects. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 34(5):876–888.

Karapinar, S.; Altan, D.; and Sariel-Talay, S. 2012. A robust planning framework for cognitive robots. In *Proc. of the AAAI-12 Workshop on Cognitive Robotics*, 102–108.

Karapinar, S.; Sariel-Talay, S.; Yildiz, P.; and Ersen, M. 2013. Learning guided planning for robust task execution in cognitive robotics. In *Proc. of the AAAI-13 Workshop on Intelligent Robotic Systems*, 26–31.

Kasper, A.; Jäkel, R.; and Dillmann, R. 2011. Using spatial relations of objects in real world scenes for scene structuring and scene understanding. In *Proc. of the 15th IEEE Intl. Conference on Advanced Robotics (ICAR'11)*, 421–426.

Lowe, D. G. 1999. Object recognition from local scale-invariant features. In *Proc. of the 7th IEEE Intl. Conference on Computer Vision (ICCV'99)*, 1150–1157.

Neumann, B., and Möller, R. 2008. On scene interpretation with description logics. *Image and Vision Computing (Cognitive Vision Special Issue)* 26(1):82–101.

Randell, D. A.; Cui, Z.; and Cohn, A. G. 1992. A spatial logic based on regions and connection. In *Proc. of the 3rd Intl. Conference on Principles of Knowledge Representation and Reasoning (KR'92)*, 165–176.

Sjöö, K.; Aydemir, A.; and Jensfelt, P. 2012. Topological spatial relations for active visual search. *Robotics and Autonomous Systems* 60(9):1093–1107.

Sokeh, H. S.; Gould, S.; and Renz, J. 2013. Efficient extraction and representation of spatial information from video data. In *Proc. of the 23rd Intl. Joint Conference on Artificial Intelligence (IJCAI'13)*, 1076–1082.