Learning Spatial Relations Between Objects From 3D Scenes

Severin Fichtl, John Alexander, Frank Guerin
Computing Science
University of Aberdeen, Aberdeen AB24 3UE, Scotland
Email: f.guerin@abdn.ac.uk

Wail Mustafa, Dirk Kraft, Norbert Krüger
Maersk Mc-Kinney Moller Institute
Niels Bohrs Allé 1, DK-5230 Odense M, Denmark
Email: norbert@mmmi.sdu.dk

I. Introduction

Ongoing cognitive development during the first years of human life may be the result of a set of developmental mechanisms which are in continuous operation [1]. One such mechanism identified is the ability of the developing child to learn effective preconditions for their behaviours. It has been suggested [2] that through the application of behaviours involving more than one object, infants begin to learn about the relations between objects.

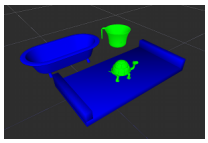
We consider a precondition to be a learnt decision rule by which some features of the environment are used to predict the successful outcome of a behaviour. This can be used as a planning operator to allow a robot to sequence learnt actions to achieve a goal. The limited scope of this definition allows us to approach the problem computationally. This concept of a precondition is loosely related to the notion of an affordance [3] used as a planning operator, which has been well studied within the field of developmental robotics (see e.g. [4], [5])

Learning a precondition for a motor action from raw sensor data is challenging as it may take many thousands of examples to learn an effective rule. For this reason we first perform an abstraction to convert data into a form which simplifies the learning procedure.

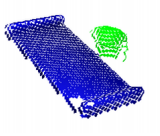
In this work, we learn a limited number of abstractions which can then be used to form preconditions for motor actions. These abstractions take the form of spatial relations amongst objects. We consider three "classes" of spatial relation: The objects either are *separated* from, *on-top* of, or *inside* each other. We have tackled this same problem in previous work [6]. Here we report on recent improved results using a novel application of histograms to visually recognise a spatial relation between objects in the environment. Using this histogram based approach we are able to report a very high rate of success when the system is asked to recognise a spatial relation.

II. Learning Spatial Relations

For learning spatial relations between objects we used data from a sophisticated 3D vision system inside the physically realistic simulator RobWorkSim [7]. In our experiments we use 4 different household objects (see Figure 1a). Using these objects we are able to design combinations of object pairs accounting for each of the three classes of spatial relation.



(a) Coffee cup, turtle, tray and bathtub.



(b) Texlet representation of a scene.

Fig. 1: Objects in the simulated environment.

To collect samples, we placed pairs of objects in the simulated environment and used a Kinect-based vision system to create a representation of the scene and segment the objects.

For the *separated* class, two objects were placed randomly in the scene with the distance between the objects always greater than 0. The objects pairings were: the coffee cup next to the tray, the turtle next to the tray and the turtle next to the bathtub. For the *on-top* class, the tray was placed randomly in the scene and then either the coffee cup or the turtle was placed randomly on-top of the tray. The objects parings were: the coffee cup on-top of the tray and the turtle on-top of the tray. For the *inside* class the bathtub was placed randomly in the scene and the turtle was placed randomly inside the bathtub. See Figure 2 for an illustrated example of each of the three classes.

Our system uses Kinect-based vision [8] to extract information about objects in the scene. Kinect produces a depth map which describes the distance from the camera to each point of the surfaces visible to the camera system. It is important to note that within our simulator, the Kinect device includes realistic noise, allowing for more realistic data about the depth to the objects the scene. Using the picture of the scene and the depth map, our vision system calculates a 3D point cloud. Based on this 3D point cloud and the colour information of the scene, our vision system creates surface patches (called textlets) as shown in Figure 1b. These texlets describe the surfaces within the scene with additional information, such as the position, the orientation and colour of the surface [9].

In the scene, the objects used were each a uniform colour and the colour of each object was distinct. This allowed for a system of purely colour based segmentation. Although this is a simplification it is justified given the scope of this work. After segmentation, each object is assigned its unique set of texlets. A texlet representation of a scene with two segmented objects can be seen in Figure 1b.

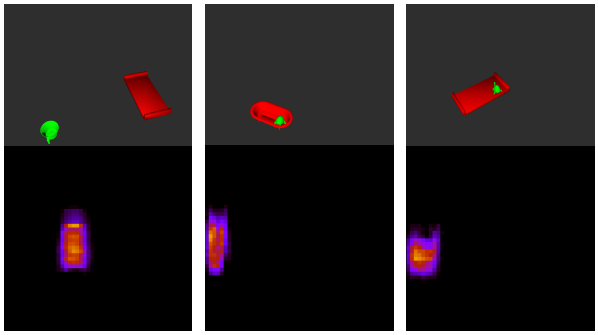


Fig. 2: Three texlet representations and the corresponding histograms.

We create 2D histograms which store relevant information about the spatial relations between objects. To extract this information from the sets of texlets, we calculate two distance measures between all texlets of one object to all texlets of another object: The absolute distance of a pair of texlets along the xy plane and the relative distance of the first object's texlets to the second object's on the z-axis, such that if the first texlet is above the second texlet, the distance is positive, otherwise negative. Since we always consider pairs of objects, if the first object has n texlets and the second has m texlets, this results in a nm vector for each measure considered. These distances are used to fill the histograms, such that the the x-axis is the absolute xy distance and the y-axis of the histogram is the relative distance between textlets along the z-axis. The histograms have 50 bins per axis, experientially this value produced the best results. The x-axis runs from 0mm to 2000mm, and the y-axis from $\pm$150mm. Example histograms are given in Figure 2.

For learning to differentiate the spatial relations between objects, we used a Random Forest (RF) classifier [10]. The RF had 100 trees, an overlap of 0.9 and 400 inputs per tree. For each class we used 5400 samples to build a training set. The training set contained samples from all configurations of each class in equal numbers (e.g. from *on-top* there were 2700 sample from coffee cup and tray and 2700 from the turtle and tray.) Similarly, in the validation set each class was represented equally with 1404 samples per class[1]. After training, the system classified 99.95% of the 4212 validation samples correctly. Although our result is not directly comparable, we achieved a higher success rate than [11] with similar data.

We tested the system on its ability to generalise to novel objects: We introduced a 6-sided die either next to or on-top of wedge. The system performed well, classifying 95% of the samples correctly. Results are shown in Table II

III. Discussion and Related Work

The most closely related work on learning spatial relations between objects in a 3D space is [11] who use a support vector machine based approach. In this approach the support vectors are picked from for their ability to differentiate the point cloud into two objects. This has the effect that the subset of points considered by the classifier are on the edges of the object. Relations are then learnt based upon the relative positions of clusters of the support vectors. For any classification based approach to be successful, it requires that similar relations have a similar representation; at the level of point clouds/textlets the representation of a relation can be very different. In the case of [11], the scene is reduced to clusters with xzy coordinates. We feel that our histogram based approach allows for a more generic representation of the scene — we maintain a higher proportion of the important information about the relations between objects.

[1]For some classes, the object pairings were not equally represented.

TABLE I: Known objects

	True Pos.	False Pos.	True Neg.	False Neg.
Separated	1404	1	2807	0
On-top	1404	0	2808	0
Inside	1403	0	2808	1

TABLE II: Novel objects

	True Pos.	False Pos.	True Neg.	False Neg.
Separated	1421	105	1347	31
On-top	1262	30	1422	190
Inside	0	86	2807	0

Acknowledgment

This work was supported by the EU Cognitive Systems project XPERIENCE (FP7-ICT-270273) and Leverhulme Grant F/00 152/AL.

References

[1] Frank Guerin, Dirk Kraft, and Norbert Krüger. A survey of the ontogeny of tool use: from sensorimotor experience to planning. *IEEE Transactions on Autonomous Mental Development*, 5(1):18–45, 2013.

[2] J. Piaget. *The Construction of Reality in the Child*. London: Routledge & Kegan Paul, 1937. (French version 1937, translation 1955).

[3] James J. Gibson. *The Ecological Approach To Visual Perception*. Lawrence Erlbaum Associates, 1986.

[4] E. Ugur, E. Oztop, and E. Sahin. Goal emulation and planning in perceptual space using learned affordances, 2011.

[5] Lucas Paletta and Gerald Fritz. Reinforcement learning of predictive features in affordance perception. In Erich Rome, Joachim Hertzberg, and Georg Dorffner, editors, *Towards Affordance-Based Robot Control*, volume 4760 of *Lecture Notes in Computer Science*, pages 77–90. Springer Berlin Heidelberg, 2008.

[6] Severin Fichtl, John Alexander, Dirk Kraft, Jimmy Alison Jorgensen, Norbert Krüger, and Frank Guerin. Learning object relationships which determine the outcome of actions. *Paladyn*, (Special Issue on Advances in Developmental Robotics):1 – 12, 2013.

[7] Jimmy A Joergensen, Lars-Peter Ellekilde, and Henrik G Petersen. RobWorkSim - an Open Simulator for Sensor based Grasping. *Robotics (ISR), 2010 41st International Symposium on and 2010 6th German Conference on Robotics (ROBOTIK)*, pages 1–8, June 2010.

[8] Søren Maagaard Olesen, Simon Lyder, Dirk Kraft, Norbert Krüger, and Jeppe Jessen. Real-time extraction of surface patches with associated uncertainties by means of Kinect cameras. *Journal of Real-Time Image Processing*, pages 1–14, 2012.

[9] N. Pugeault, F. Wörgötter, and N. Krüger. Visual primitives: Local, condensed, and semantically rich visual descriptors and their applications in robotics. *International Journal of Humanoid Robotics (Special Issue on Cognitive Humanoid Vision)*, 7(3):379–405, 2010.

[10] N Pugeault and R Bowden. Spelling it out: Real-time ASL fingerspelling recognition. In *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pages 1114–1119, 2011.

[11] Benjamin Rosman and Subramanian Ramamoorthy. Learning spatial relationships between objects. *Int. J. Rob. Res.*, 30(11):1328–1342, September 2011.