Proceedings of the Twenty-Second International FLAIRS Conference (2009)

Mapping Grounded Object Properties across Perceptually Heterogeneous Embodiments

Zsolt Kira

Georgia Institute of Technology
85 Fifth Street N.W.,
Atlanta, GA, 30308, USA
zkira@gatech.edu

Abstract

As robots become more common, it becomes increasingly useful for them to communicate and effectively share knowledge that they have learned through their individual experiences. Learning from experiences, however, is oftentimes embodiment-specific; that is, the knowledge learned is grounded in the robot's unique sensors and actuators. This type of learning raises questions as to how communication and knowledge exchange via social interaction can occur, as properties of the world can be grounded differently in different robots. This is especially true when the robots are heterogeneous, with different sensors and perceptual features used to define the properties. In this paper, we present methods and representations that allow heterogeneous robots to learn grounded property representations, such as that of color categories, and then build models of their similarities and differences in order to map their respective representations. We use a conceptual space representation, where object properties are learned and represented as regions in a metric space, implemented via supervised learning of Gaussian Mixture Models. We then propose to use confusion matrices that are built using instances from each robot, obtained in a shared context, in order to learn mappings between the properties of each robot. Results are demonstrated using two perceptually heterogeneous Pioneer robots, one with a web camera and another with a camcorder.

Introduction

As autonomous robots become increasingly common, it is likely that there will be multiple robots that each learn through experience, that is via embodied interaction with the world. This type of grounded learning, however, ignores social aspects of development and learning. With multiple robots, it is crucial for the robots to be able to share knowledge either through explicit communication or implicit means such as imitation. Such knowledge sharing speeds up learning significantly and can reduce the need for costly human teaching. Experienced robots can also guide other robots through specific developmental trajectories in a way that attempts to optimize learning. It also allows one robot to be the teacher of another robot, reducing need for costly human interaction.

Several problems can prohibit effective sharing of knowledge, however. Knowledge learned via exploration of the world is often embodiment-specific. It is quite common to have some degree of heterogeneity among robots, however, and there can be slight perceptual differences even among two robots of the same model. For example, the camera color models may differ slightly. It is an even greater problem when different types of robots are used. Currently, there is a plethora of robotic systems in use in home environments (e.g. the Roomba and lawn mowing robots), research labs, and in various domains where task allocation to differing robots is necessary.

Symbols are often used to abstract raw sensory readings, and facilitate communication via language. However, even assuming that these symbols are grounded (Harnad, 1990) within each robot, there is the problem of achieving a common grounding among multiple robots, an issue that has been raised as the social symbol grounding problem (Vogt and Divina, 2007). Approaches that ground symbols jointly in the environment by multiple robots at the same time exist (Jung and Zelinsky, 2000), but require that the robots learn in the same environment and under the same conditions.

In order to allow knowledge sharing among such heterogeneous robots, we posit that the robots can first autonomously build models of their differences and similarities, and map symbols from each robot's representation to the other. Note that in order for this to be useful, there must be *some* similarity between the two robots; i.e. if the two robots can only sense non-overlapping features of objects (e.g. sound versus vision), then meaningful communication between them will be much more difficult. The point is to leverage whatever similarity exists between the robots, and to be cognizant when the differences are too substantial for effective knowledge sharing. Mappings between the robots' representations can be built after each robot has learned its respective representations. The building of the models can be performed by leveraging similarity to deal with

---

heterogeneity, specifically by establishing a physically shared context. We have demonstrated such mechanisms in previous work, for example, for learning parameterized models of low-level sensing differences (Kira and Long, 2007). In this paper, we show how data obtained from a shared context at the object level can be used to learn mappings between symbols, representing properties of objects such as color or texture that may be grounded differently in each robot. In this paper we manually select images from the same context; future work will integrate those methods with techniques for building mappings of differences in this paper, leading to an entirely autonomous process.

We use conceptual spaces to anchor sensory data to learned object properties (e.g. color, texture, shape) and concepts (e.g. object categories or specific objects) (Gärdenfors, 2000). Conceptual spaces are geometric spaces that utilize similarity measures and concept prototypes as the basis of categorization. This geometric form of representation has several advantages. It has also been elaborated upon and extended in several other works (e.g. (Aisbett and Gibbon, 2001)), and discussed and implemented to a limited degree in robotic systems (Balkenius and Gärdenfors, 2000),(LeBlanc and Saffiotti, 2007). Most importantly, understanding how different properties and concepts can be mapped between different agents can be intuitively visualized in these spaces.

In this paper, we focus on object properties, specifically color and texture. We represent such color properties as Gaussian Mixture Models in RGB or HSV color space or a metric space representing the output of Gabor filters, and show how they can be learned in a supervised manner. We demonstrate that directly transferring such color properties cannot effectively be done across robots that differ in their video sensing, even if the same space is used. A method for learning mappings between properties across different embodiments, represented as confusion matrices, is proposed. We show that these models can be built using sensory data pairs obtained from shared contexts. Results are demonstrated via experiments using video data from real robots with differing video sensors.

Related Work

The key issue in this paper is related to social symbol grounding, that is finding common symbols for similar concepts across a population of agents. This is related to language formation and has been studied extensively in linguistics and evolutionary or artificial life (Vogt and Divina, 2007),(Steels and Kaplan, 1999). For example, work done by Luc Steels and his colleagues in the area of shared vocabulary development used shared attention to synchronize the two robot's symbols (Steels and Kaplan, 1999). This is a similar concept to ours, although they did not explicitly deal with the issue of robot heterogeneity where robots may have different feature spaces.

Another example of this in robotics includes work by Jung and Zelinsky, who studied two robots that perform the same task (vacuuming) but had different capabilities or roles; one robot swept small pieces and reached into corners, while the other could only vacuum the larger piles and could not reach corners (Jung and Zelinsky, 2000). In that case, a shared ontology was developed by establishing a physically shared context during learning: The two robots followed each other around the room and agreed on symbols for specific locations in the environment. In a similar vein, Billard and Dautenhahn have looked at a situation involving two homogeneous robots where one teacher attempts to share its symbols with another robot via imitation, namely following (Billard and Dautenhahn, 1998).

Conceptual spaces, the representation used here, have been used in robotics in several works. LeBlanc and Saffiotti have looked into the fusion of properties into a single domain, but have so far focused on spaces with identical dimensions (LeBlanc and Saffiotti, 2007). Overall, little work in robotics and elsewhere has focused on sensor heterogeneity, and bridging resulting conceptual differences. This paper presents the first step, namely determining what properties can be mapped from one existing representation to another, across spaces that can differ in their regions and dimensions.

Property Representation and Learning

Abstracting Sensory Data

Sensory data is often abstracted in order to improve learning or to enable communication. In this paper, we use Gärdenfors' conceptual spaces (Gärdenfors, 2000) in order to bridge lower-level representations and symbols. The most basic primitive of the representation is a *dimension* (also referred to as quality or attribute), which takes values from a specific range of possible values (a domain in the mathematical sense, although it is not to be confused with the notion of a domain used in the next paragraph). For example, the hue of an object can be specified as an angle in the range [0, 1]. The values of these dimensions come from perceptual features processed from sensor data. For example, a camera sensor measures physical properties of the world (light), converting them into a digital representation consisting of multiple pixels in the form of an RGB space. A perceptual feature detector can convert regions of the image into an HSV space, and the H (hue) value can make up a dimension. The feature detector returns a set of these, one for each region of the image that it determines is salient.

Gärdenfors posits that there are integral dimensions that cannot be separated in a perceptual sense. For example, the HSV color space can be argued to consist of three integral dimensions. Another example used is pitch and volume that is perceived by the auditory system. A set of such integral dimensions is referred to as a domain. A domain defines a space that consists of all possible values of the integral dimensions. It is useful to abstract and divide these values into specific regions, which define a property. For example, "blue" can be a property that corresponds to some region of the color space. The regions can be arbitrary shapes, although Gärdenfors defines what he calls natural properties consisting of regions with certain characteristics such as convexity. Note that a property corresponds to a region in a single domain.

We can now define a *conceptual space K* as made up of a set of domains. A specific concept in the conceptual space is a set of regions from the domains $D = \{d_1, d_2, ..., d_n\}$ in the conceptual space. A point in the conceptual space is called a knoxel $k =< k_1, k_2, ..., k_n >$, and specifies instances of the concept in the form of vectors. A knoxel can specify points in some of the domains, while leaving others unspecified, in the form of a partial vector. Note that a property is a specific type of concept that utilizes only one of the domains from the conceptual space. In this paper, we focus on mapping properties, possibly located in different domains, and hence will not go into how properties are combined to form concepts.

In order to facilitate communication, symbols are attached to properties and concepts. Each robot maintains a set of symbols X, each of which is grounded to a property (or in general, a concept) via the representation. Symbols correspond to labels or strings, which will be randomly assigned by the robot. A representation can be described as a function that returns the degree to which a specific knoxel k can be categorized as having the corresponding property represented by symbol $x \in \mathrm{X}$; i.e. $R : (k \ , x) \rightarrow [0,1]$. Each property has a prototype for in the form of a knoxel, denoted as k_p. The implementation of properties within the framework of GMMs is described in the next section.

Learning of Properties from Instances

In order to learn a representation for object properties, we will scaffold the robot's learning by first providing it with multiple instances of data that contain a property. Note that no labels are given, and the robot creates its own random labels. Each scene, which can contain multiple properties and concepts, results in a set of knoxels K calculated from the output of the robot's perceptual feature detectors. In this paper, it is assumed that it is known which domain is to be trained for a set of instances. For each property p_i, we use a Gaussian Mixture Model (GMM) to characterize the regions, denoted as G_i. Specifically, each property can be modeled as:

$$P(p_i \mid \theta) = \sum_{j=1} w_j P(p_i \mid \mu_j, \Sigma_j) \quad (1)$$

where w_i is known as the mixing proportions and θ is a set containing all of the mixing proportions and model parameters (mean μ and standard deviation Σ). In this paper, we use a maximum of three clusters per property, as determined by a minimum description length criteria, learned via the Expectation Maximization (EM) algorithm (Bilmes, 1998). Once models are learned, they are used to determine membership in a property. Specifically, given sensory data, the membership for a property is the Gaussian distance function to the nearest property cluster.

Mapping Properties Across Differing Embodiments

As mentioned properties are regions in domains, in our case represented as Gaussian clusters. The same property can be represented as clusters with different characteristics (for example, different standard deviations) or even domains from different sensors (for example, the width of an object as detected by a camera or laser). Given these clusterings of a domain, the problem is to find associations between clusters from each robot (which cluster(s) in one robot belongs to which cluster(s) in another robot).

In order to do this, we use instances from each robot while viewing the same scene and compare properties that they see. This can be established using interaction such as following behaviors that are perceptually driven (assuming robots can detect each other, and determine pose information such as heading) (Kira and Long, 2007). In this paper, this is done manually and in a looser sense; manual selection of images is performed such that both robots see the same object, although not necessarily from the same perspective. Future work will incorporate these behaviors, resulting in a fully autonomous system. Given a scene, each robot processes its sensory data to produce a set of knoxels where property memberships in relevant domains can be calculated. For each pair of properties (one from each robot), statistics described below are maintained in order to determine whether they represent similar physical properties.

Confusion Matrices

The problem of finding mappings between clusters is closely related to comparing different clusterings, which has been dealt with in statistics and machine learning communities (e.g. Fowlkes and Mallows, 1983). This line of research attempts to create measures of similarity between two clusterings. A major representation used in the creation of some of these metrics is the confusion matrix, which is a matrix with k rows (one for each cluster in the first clustering) and k' columns (one for each cluster in the second clustering). Each entry contains the proportion of points that (for the same instance) belong to the cluster represented by the row (in the first clustering) and that belong to the cluster represented by the column (in the second clustering). In other words, it is the intersection of the clusters C_k and C'_k. For the problem of comparing clusterings, the confusion matrix is used to calculate metrics for comparing different clusterings.

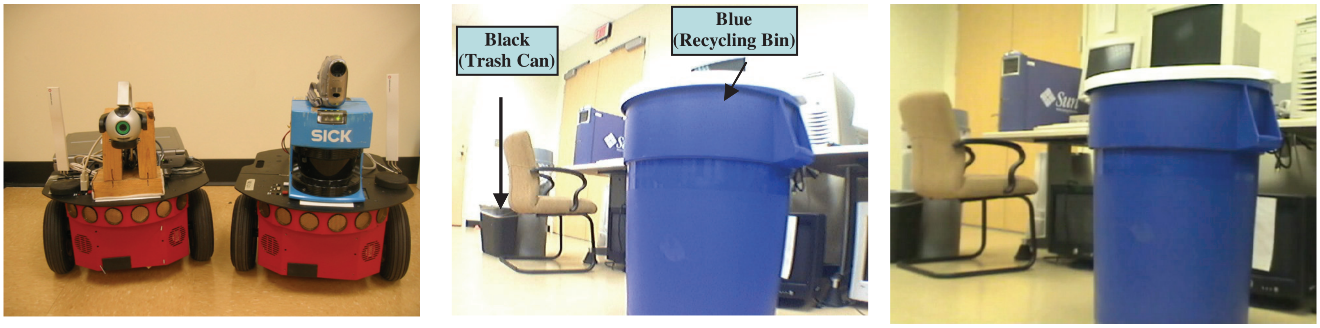

Figure 1 – Pioneer 2DX robots used in the experiments (left) and images of the same scene from each robot (middle and right). The middle image is from the robot with the web camera, while the image on the right is from the robot with the camcorder.

In our case, we seek to map individual clusters to each other, not determine overall similarity between clusterings of the entire space. Hence, instead of calculating such metrics we utilize the confusion matrix to determine pairs of properties that may potentially represent the same physical property. Suppose that there are two clusterings G_i^A and G_j^B defining regions corresponding to properties p_i^A and p_j^B for robot A and B, respectively. Also, each clustering for robot A and B has n_i^A and n_j^B clusters, respectively. Finally, suppose that we have a set of instances I from each robot (obtained using its own sensing) with a sufficiently high membership defined by a threshold for property p_i^A. The confusion matrix $PC^{A,B}$ is then updated with:

$$PC^{A,B}_{(j,k)} = \sum_{i}^{I} \frac{\min(s(i,p_j^A), s(i,p_k^B))}{s(i,p_j^A)} \qquad (2)$$

Here, *s(i,p)* is the Gaussian membership function of instance *i* in property *p*. The *min* function is used to represent the intersection of property memberships, as is used commonly in fuzzy sets. For each property of a robot, the highest values in the corresponding property's row or column will be taken and it will be considered potentially corresponding to the respective property of the other robot. A threshold may be placed on this as well, although we do not in this paper.

In the context of machine learning and statistics literature discussed above, the clusterings that are being compared are always in the same space. In other words, they both utilize the same data, and the data uses the same dimensions. In our case, we are attempting to compare clusters between spaces, where the axes (the dimensions) may differ. Hence, there is an additional correspondence problem in terms of whether an instance in one space corresponds to the same instance in another space. Data gathered in a random context from each robot will be difficult to analyze because of confounding variables such as differences in the environment or perspective. This is why, as discussed previously, some shared context must be established first.

Table 1 – Table of arbitrary property symbols assigned to color categories.

	Brown Objects	Black Objects	Blue Objects	Gray Objects	White Objects
Symbol: Robot A	p_1^A	p_2^A	p_3^A	p_4^A	p_5^A
Symbol: Robot B	p_1^B	p_3^B	p_5^B	p_2^B	p_4^B

Experimental Results

We now describe experiments that have been carried out using image data obtained on two Pioneer 2DX robots, seen in Figure 1. The first robot had odometry, sonar sensors, and a Quickcam Express web camera. The second robot had odometry, laser sensors, and a DCR-HC40 Sony Handycam camcorder. It also had sonar sensors, but data was not recorded from these sensors. The experiments serve to show that direct data transfer across differing embodiments is ineffective and that robots can model their differences in terms of properties using confusion matrices built using data obtained from a shared context.

In order to train color and texture properties, the robots were driven around a laboratory environment for 2-3 runs, resulting in a large amount of stored sensor data. Six to eight objects from the environment per color category were chosen for training. For texture, a single empirically-chosen Gabor filter was used, with the mean and standard deviation of its output comprising the space. Examples of objects include a blue recycling bin, a smaller black trash can, and a blue Sun computer, all of which can be seen in the images on the right portion of Figure 1. In order to avoid bias, property numbers were randomly assigned to the actual color or texture that was used during training. It is important to note that symbolic labels were not given to the robots, they are only added by the authors for clarification of the figures. For each property, all that is given is the domain to be trained, a set of data instances, and segmentation of the target object. In other words, the

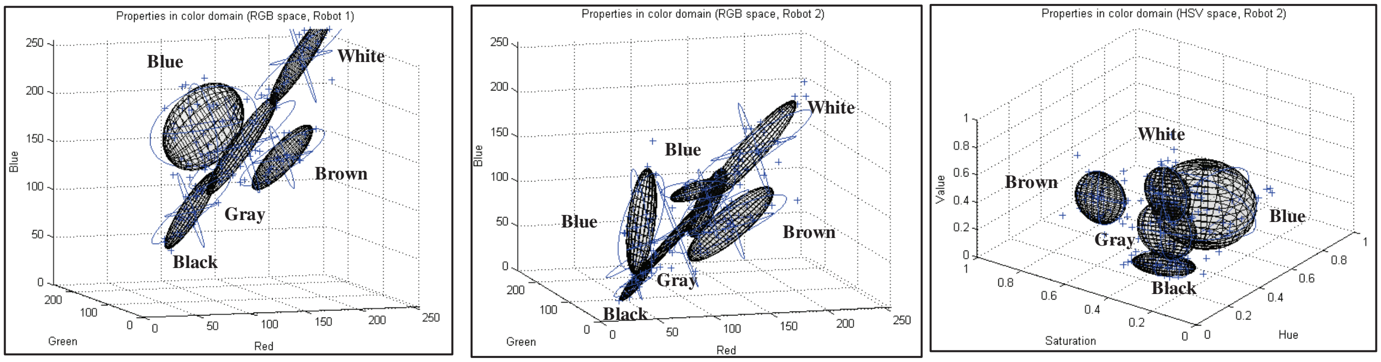

Figure 2 – Color properties, represented as a Gaussian Mixture Model, after training with multiple objects with five colors. Results are shown for two color spaces (RGB and HSV) and the two heterogeneous robots. The resulting models are significantly different for each robot, arising due to heterogeneity in training and sensing.

robots could not simply compare labels to determine which of their properties mapped. Table 1 shows the assignments that were given to both robots. Knowledge of this mapping is not used by the algorithms, and is what must be learned by the robots given instances from a shared context.

Out of all of the images recorded, 45 were chosen per category containing an approximately equal number of instances from each object in the category, resulting in a total of 225 images. 150 of these were randomly chosen for training, while the rest were used in testing the categorization and building the confusion matrices. The same images were used for training of texture properties. The objects were segmented manually in the images; future work will look into automatic segmentation based on color and texture. All processing was performed offline in Matlab, although this is not due processing requirements.

Property Learning, Testing, and Direct Transfer

Figure 2 shows the resulting property regions for both robots and two color spaces (RGB and HSV). As can be seen, despite being trained on the same objects, the representations are quite different. After training the properties using supervised learning, the accuracy of categorizing the color of different views and instances of the objects was tested. The left side of Table 2 shows the resulting accuracy results for both robots, average over five runs (standard deviations are shown). They both achieved a relatively high accuracy given the existence of object brightness changes due to changes in perspective. Interestingly, the results for robot A (that had a cheap web camera) performed similarly to the second robot that had a more expensive camcorder. Overall, the camcorder resulted in colors that were duller and less bright, as can be seen from the results in Figure 2.

In order to show that direct transfer or comparison of properties may not be possible across heterogeneous robots, we directly transferred the learned GMM models from robot A to robot B, and vice-versa, for the same RGB color space. We then tested the resulting categorization success in the same manner as before. In other words, robot A used robot B's learned representation on its own data in order to categorize the testing set. As can be seen from the right side of Table 2, the results were dramatically worse and close to random guessing. Even with training sets consisting of the same objects, the properties of the two robots were incompatible due to sensor heterogeneity.

Table 2 – Color categorization accuracy.

Robot	Own Representation		Transferred Representation	
	#(/ 75)	Percent	#(/ 75)	Percent
Robot A	60.6	80.8 ± 5.0	14.4	19.2 ± 0.7
Robot B	59.8	79.7 ± 2.9	16.4	21.9 ± 2.6

Mapping Properties

We now describe the results of building the confusion matrices based on data instances from the same object. In all of these experiments, we used an RGB color space for robot A and HSV color space for robot B. Table 3 shows the confusion matrix from robot A's perspective. For intuition, each value $PC^{A,B}_{(j,k)}$ in the matrix is modified for each instance in which property j has the largest membership according to robot A's property models. The amount that it is updated by depends on the property membership ascribed to an instance in the same context by robot B. Note that the two matrices may differ (as they do in this case), since the first robot decides which instances to use to update a particular property based on whether its memberships is the highest compared to the other properties.

As can be seen, in both matrices the correct mapping between properties of robot A and properties of robot B can be inferred by their maximal value (in bold). This can be verified using Table 1; for example, in $PC^{A,B}$ the highest value for row p_2^A is in the column corresponding to p_3^B (0.49), which is correct. In some cases, there are other values in the same row that are relatively high. Some of this can be attributed to correlations between properties on the *same* robot. For example, when p_4^B (corresponding to white) had a large membership, p_2^B (corresponding to gray) did as well. This is because some gray objects were light gray and some white objects were dirty or not purely white. If these correlations are divided out by combining both of the robot's learned confusion matrices, the resulting matrix differentiates the mapped properties more profoundly. This can be seen in Table 4. Similar results were obtained for texture properties, where all of the correct mappings were inferred. These results are not shown due to space limitations.

Making Use of the Mappings

Although beyond the scope of this paper, the properties mentioned here can and have been combined in a fuzzy manner to describe entire objects (e.g. the blue trash can), as proposed by (Rickard, 2006). The confusion matrices learned here can then be used in many ways. For example, two robots can determine if two concepts are similar by first determining how many underlying properties are shared and then aligning the concept representations using the mappings. Future work will investigate the use of these representations to enable effective knowledge sharing.

Table 3 - Confusion Matrix $PC^{A,B}$

	p_1^B	p_2^B	p_3^B	p_4^B	p_5^B
p_1^A	**0.48**	0.00	0.00	0.00	0.05
p_2^A	0.00	0.09	**0.49**	0.00	0.04
p_3^A	0.00	0.08	0.00	0.00	**0.60**
p_4^A	0.10	**0.29**	0.00	0.09	0.16
p_5^A	0.17	0.31	0.00	**0.43**	0.07

Table 4 – Combined confusion matrix

	p_1^B	p_2^B	p_3^B	p_4^B	p_5^B
p_1^A	**0.29**	0.00	0.00	0.00	0.00
p_2^A	0.00	0.01	**0.37**	0.00	0.00
p_3^A	0.00	0.02	0.00	0.00	**0.32**
p_4^A	0.03	**0.14**	0.00	0.01	0.05
p_5^A	0.03	0.04	0.00	**0.27**	0.00

Conclusions

This paper has introduced the groundwork for reasoning upon properties of objects and ways in which they can be mapped across different robots. We have shown that robots that differ in their sensing may not be able to directly transfer knowledge in the form of property regions, but that they can learn correct mappings between each robot's properties using instances obtained from a shared context, for both color and texture properties. In this paper, we manually selected images from similar contexts, although one can use behaviors for establishing such shared context (Kira and Long, 2007).

In future work, these two parts will be integrated to allow two robots to build models of their differences in a fully autonomous manner. In doing so, several challenges such as the segmentation of objects from sensory data and ambiguity will have to be dealt with. It would also be interesting to add sensory modalities, such as sonar or laser range finders, and combine the various properties derived from multiple domains in order to represent objects as a whole. Such combination of properties to represent concepts is the ultimate goal, and the mappings learned in this paper can be used in various types of knowledge transfer. For example, robots may be able to infer which concepts are transferable between the robots based on which of their properties are shared.

References

Aisbett, J. & Gibbon, G. (2001), 'A general formulation of conceptual spaces as a meso level representation', Artificial Intelligence 133(1-2), 189--232.

Balkenius, C.; Gärdenfors, P. & Hall, L. (2000),'The Origin of Symbols in the Brain', Technical report, Lund University Cognitive Science.

Billard, A. & Dautenhahn, K. (1998), 'Grounding communication in autonomous robots: an experimental study', Robotics and Autonomous Systems 1-2, 71-81.

Bilmes, J. (1998), 'A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models', International Computer Science Institute 4.

Fowlkes, E. & Mallows, C. (1983), 'A method for comparing two hierarchical clusterings', Journal of the American Statistical Association 78(383), 553-569.

Kira, Z., Long, K., "Modeling Robot Differences by Leveraging a Physically Shared Context", in Proceedings of the Seventh International Conference on Epigenetic Robotics, pp. 53-59, 2007. Sweden: Lund University Cognitive Studies.

Gärdenfors, P. (2000), Conceptual Spaces: The Geometry of Thought, MIT Press.

Harnad, S. (1990), 'The Symbol Grounding Problem', Physica D 42, 335-346.

Jung, D. & Zelinsky, A. (2000), 'Grounded Symbolic Communication between Heterogeneous Cooperating Robots', Auton. Robots 8(3), 269--292.

Kirby, S. (2002), 'Natural Language From Artificial Life', Artificial Life 8(2), 185--215.

LeBlanc, K. & Saffiotti, A. (2007),Issues of Perceptual Anchoring in Ubiquitous Robotic Systems, in 'Proc. of the ICRA-07 Workshop on Omniscient Space'.

Rickard, J. (2006), 'A concept geometry for conceptual spaces', *Fuzzy Optimization and Decision Making* **5**(4), 311--329.

Steels, L. & Kaplan, F. (1999), 'Bootstrapping Grounded Word Semantics', in Briscoe, T., editor, Linguistic evolution through language acquisition: formal and computational models, pages 53-74, Cambridge University Press. Cambridge, UK.

Vogt, P. & Divina, F. (2007), 'Social symbol grounding and language evolution', Interaction Studies 8(1).