A Conceptual Space Architecture for Widely Heterogeneous Robotic Systems[1]

HyunRyong Jung, Arjun Menon, and Ronald C. Arkin
Mobile Robot Laboratory, School of Interactive Computing
Georgia Institute of Technology, Atlanta, GA USA 30332

Abstract. This paper describes the value of the conceptual space approach for use in teams of robots that have radically different sensory capabilities. The formal underpinnings and perceptual processes are described in the context of a biohazard detection task. The architecture is based on the conceptual spaces representation that Gärdenfors suggested as an alternative to more traditional AI approaches.

Keywords. Conceptual space, heterogeneous robot, knowledge representation

1. Introduction

In robotics, a challenging area involves the sharing of knowledge across widely disparate robotic platforms, i.e., when there is no commonality across the sensor space between platforms. Heterogeneous robots need to share their knowledge with each other to achieve a team task efficiently. For our research for the Army, each type of robot is equipped with radically different sensors, so a framework to share sensor data with other widely heterogeneous robots efficiently is essential. However, classical knowledge representations (e.g., symbolic representations and connectionist methods) have several deficits such as the frame and symbol grounding problems, and can exhibit difficulty in computing similarity between concepts.

To address these problems, we use the conceptual space that Gärdenfors [1] suggested as a basis for human and machine cognition. A conceptual space constitutes a metric world in which objects and abstract concepts are represented by quality dimensions. A concept has several domains to distinguish it from other concepts. Thus, a specific concept forms a set of regions across these domains in the conceptual space. Each domain is composed of quality dimensions, and the primary function of the domain is to represent various qualities of situations or objects. As a result, the linkage between a concept and its domains directly grounds the concept in sensory experience. Because the quality dimensions are metric, the similarity can be measured easily. To deal with potential sensor and representation differences, we abstract raw sensory data into natural object properties such as color, features, chemical composition, and so on based on existing MAST mission requirements and sensor capabilities. To represent the regions of a property, a Gaussian Mixture Model is used because a property of a

[1] The work in this paper was funded under the U.S. Army Research Laboratory Micro Autonomous Systems and Technology Collaborative Technology Alliance project (BAE # 329420.)

concept cannot always be represented by a single Gaussian. Table 1 summarizes some of the definitions used in this approach.

Table 1: Conceptual Space Definition Summary

	Definition
Concept	A concept is represented as a set of convex regions in a number of domains.
Domain	A domain is a set of integral dimensions that are separable from all other dimensions.
Property	A property of a concept is a convex region in some domain.
Prototype	Prototypes are the most representative members of a category.
Quality Dimension	Quality dimensions represent various object qualities.
Quality	Sensory input from a sensor system.

Earlier research in our laboratory [2-4,7] focused on limited heterogeneity in the sensors fielded on different robots. In our ongoing research as part of the Army Research Laboratory's Microautonomous Systems Technology Collaborative Technology Alliance, we are extending this previous work to incorporate sensor, power, communication, and computation impoverished platforms with the goal of being able to provide fully distributed team coordinated control for search and rescue, biohazard detection, and other related missions [5].

2. Related Work

Knowledge representation, studied by many AI researchers, constitutes one of the fundamental topics in AI. However, while answering the question—what is the correct knowledge representation for a particular task? — may seem easy, it is more complicated than we expect. Thus, we need to define the concept of knowledge representation. One simple definition [10-11] is that it is the study of how to store knowledge into a form that an agent can reason with. For this application, the agent is a robot that can move and navigate autonomously. To represent knowledge in such systems, several theories of representation, two of which are the symbolic and connectionist paradigms, are widely used in various areas and applications [12-16].

The symbolic paradigm represents the environment with symbols and has a formal syntax [1] [6]. The role of the syntax is to determine what and how symbols must be manipulated. In other words, knowledge can be represented with a set of symbols that are connected based on the principles of syntax. Thus, basic concepts are not modeled in a sensory space per se, but represented by the basic symbols. As a result, learning a new physical property for given symbols and dealing with changes in the meaning of concepts cannot be easily represented in symbolic representation. Therefore, symbolic representation is vulnerable to the frame problem and the symbol grounding problem [1] [6] [17]. In addition, since concepts at one level are represented by symbols, similarity between symbols cannot be easily modeled at the purely symbolic level. Consequently, as similarity plays an important role in learning and concept formation, systems using the symbolic representation can have difficulty learning new grounded symbols.

The central idea of the alternative connectionist approach takes the form of mimicking the human brain, so it is composed of highly interconnected units or neurons [12] [14]. Artificial neuron networks (ANNs) are a specific instance of the connectionist approach. Connectionism appears brain-like, and it is not subject to the symbol grounding problem that typical symbolic representations suffer from. However, since ANNs consist of a large number of simple, highly interconnected neurons, the modification of ANN representations is difficult, so it is often called a “black box model.” Another weakness of connectionism is that while similarity can be represented in ANNs by categorization, it cannot be readily computed as a mathematical value.

3. Overview of Conceptual Spaces

The conceptual space that Gärdenfors [1] suggested is a metric world in which objects and abstract concepts are represented by quality dimensions. A concept has several domains that distinguish it from other concepts. Thus, a specific concept is a set of regions from the domains within the conceptual space. Each domain is composed of quality dimensions, and the primary function of the domain is to represent various qualities of situations or objects. As a result, the direct linkage between a concept and domains via sensory data using the conceptual space approach can eliminate the symbol-grounding problem. Because the quality dimension constitutes a metric world, the similarity between sensed objects and concepts can be measured easily.

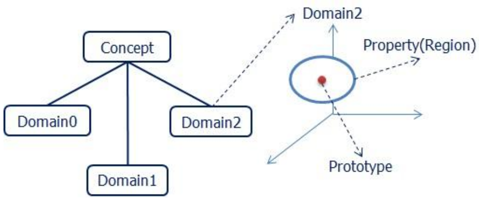

Figure 1: Relationship between conceptual space approach entities

The quality dimensions represent values that can be acquired from sensors. Examples of qualities include temperature, shape, taste, etc. The qualities can not only be explicit features of objects but also abstract non-sensory characteristics such as emotional states. For example, a color domain can be composed of three quality dimensions: hue, saturation, and brightness. A property in the conceptual space is a geometrical structure within the quality dimensions, where a property of a concept forms a convex region in the domain. A concept may also contain salience weights for properties and correlations between the properties. For some concepts, a property can be more important than others, and can be influenced by the task context. Figure 1 illustrates these relationships.

For example, consider how to represent an apple in the conceptual space. The apple is a concept and has diverse properties such as taste, color, shape. Each property can be a certain region in one domain composed of quality dimensions. As Figure 2 illustrates, the color domain of the apple has three quality dimensions: R, G, and B. A property of the apple is a region in the color domain.

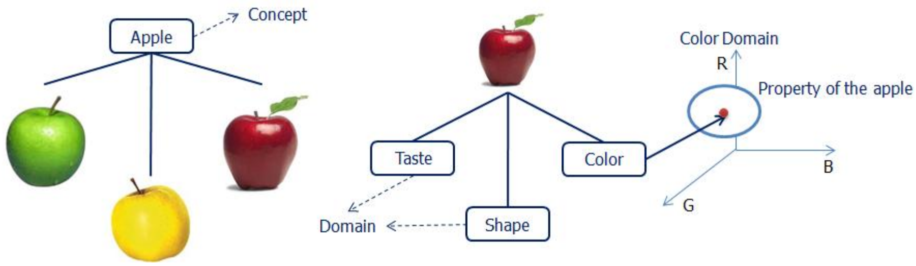

Figure 2. Concept, domain, and property of the apple

Since the property forms a convex region, we can define the most representative member of a domain as its prototype. In this approach, the prototype can be seen as the centroid of objects that are represented. When a robot detects the color of an apple, sensor data from the camera is represented as a point in the quality dimensions, and the similarity can be computed by measuring the Euclidean distance between the point and the prototype of the color domain of the apple.

Consequently, the theory of conceptual spaces can yield a solution to the symbol-grounding problem that traditional methods of knowledge representation struggle with. Moreover, the conceptual space representation provides a natural way of representing similarities, and this ability is one of its major advantages.

3.1. Conceptual Space Definition

A conceptual space is defined as *C*. The conceptual space is composed of a symbol space S_S and a concept space C_S [6]. In the symbol space, several symbols can be defined, and each symbol names a concept. An *i*th concept is denoted by c_i. A concept has properties that are defined as $c(i) = c(P_{i,1}, P_{i,2}, \dots, P_{i,n})$ and each having a range [0, 1]. An *i*th property of a *k*th symbol is denoted by $p(k,i) = P_{k,i}$. A set of concepts $\{c(i), i = 1, \dots, N\}$ is covered by S_S. Note that concepts are regions in conceptual space, but properties are regions in domains. A domain is represented as D_i, and the concept space, C_S is composed of domains. For instance, the concept of a biohazard can be partially defined, (which is used in the test scenarios under development) as $c(1) = c(P_{1,1}, P_{1,2})$, and $P_{1,1}$ is in the color domain D_1, and $P_{1,2}$ is in the temperature domain D_2. Perceptual features are projected to each domain as shown in Figure 3.

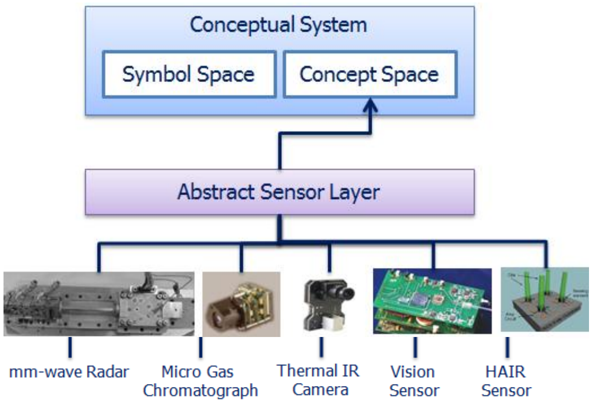

Figure 3. Schematic of conceptual space and abstract sensor layer

A prototype is the centroid of a property and serves as the most representative value of a property. Moreover, since we can categorize a sensed object by finding the closest prototype to the object, it is useful in categorization. The prototypical value is defined as $f(k,j)$ in the domain D_j of the labeled with a *k*th symbol. Figure 4 describes the prototype for the biohazard in the color and chemical domains.

Not all qualities are equally important to a concept, so we need to define the relative importance between properties. The importance of $P_{k,i}$ in domain $D_{j(i)}$ to concept c_k is referred as $\alpha(k, j(i))$. For instance, chemical composition is a primary property in detecting a chemical weapon, since these objects have unique chemical compositions. Thus, the property must have significantly higher importance than others. As Figure 4 illustrates, the chemical composition domain D_2 has much less overlap than the other domains D_1, D_3. Therefore, the chemical domain is the most informative in discriminating a biohazard from a similarly colored and shaped but empty trash can.

3.2. Similarity in Conceptual Space

As objects can be represented as property vectors in conceptual spaces, the definition of similarity of objects is relatively intuitive and easy. The similarity [1] [6] is the distance between objects (and prototypes) and it is one of the main advantages of this representation. Like distance, similarity is a real valued non-negative function and has several properties: The similarity should be maximum when the distance is zero; it should decrease with distance; and be zero when computing the similarity with an inapplicable point. So, we define the similarity *s* between objects *a* and *b* with the following equation:

$$s(a,b) = (1 + d(a,b))^{-1}.$$

As a result, a concept *c* in the symbol space can be computed with the following equation:

$$c(k) = \sum_{i=1}^{n} \alpha(k,i) \cdot s\big(p(k,i), f(k,i)\big).$$

Where $s\big(p(k,i), f(k,i)\big)$ is the similarity between an *i*th prototype and the *i*th property in a *k*th concept; $\alpha(k,i)$ is the importance of the *i*th property in the *k*th domain and *n* is the number of properties in a concept. For instance, one robot detects the color of an object, and the color domain is updated. To calculate the concept of a biohazard, $c(1)$, the temperature, D_1, the chemical composition, D_2, and the color, D_3 are also required.

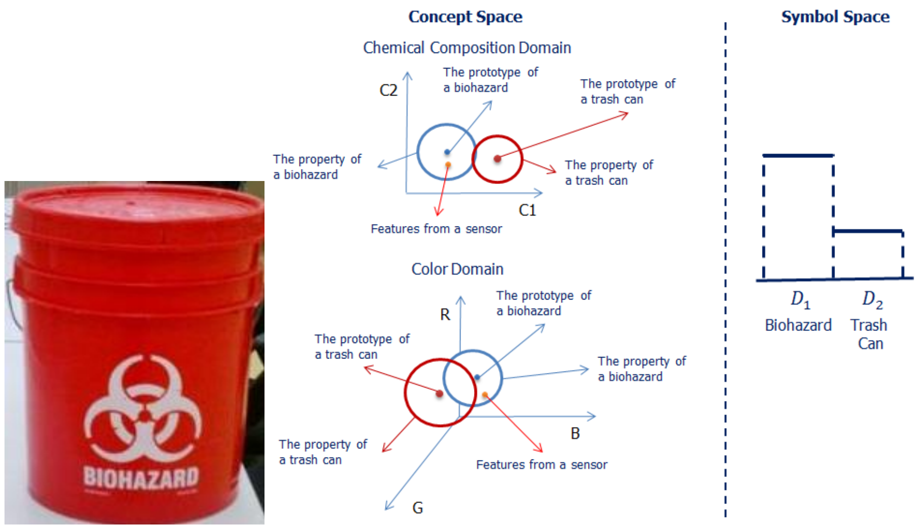

Figure 4. (Left) Potential Biohazard Object. (Right) Notation for prototypes. Note that a similarly colored trashcan may be ambiguous with a biohazard, but the chemical composition which has no overlap provides the basis for disambiguation.

3.3. Abstract Sensor Layer

In this section, the process to convert sensor data to vectors that can represent a property of an object is described, in this case, a bio-weapon. Each robot has a set of m sensors, $S = \{s_1, s_2, \ldots, s_m\}$. We denote the number of sensors as |S|. At time *t*, the robot receives an observation vector $o_{t,i}$ from each sensor, s_i, resulting in a set of measurement or observation vectors, $O = \{o_{t,1}, o_{t,2}, \ldots, o_{t,|S|}\}$. We denote the robots with a superscript, so that s_i^j is sensor i of robot *j*. Sensor data provide a stream of unprocessed information so it is presumed that each robot has a set of p feature detectors, $F = \{f_1, f_2, \ldots, f_p\}$, that further process observations and output perceptual features. We denote specific values of a set of features at time, *t* as F_t, and the specific value of a feature i as $f_{t,i}$. A feature detector is a function, φ, that maps a set of observation vectors into a set of feature vectors. For instance, $f_{i=}\varphi(o_{f_i})$ where $o_{f_i} \in O$ denotes the set of input observations used by the feature detector.

Figure 5 (left) depicts sensors, observations, and perceptual features for a robotic microflyer tasked for this mission. This robot has three sensors, $S^F = \{s_1^F, s_2^F, s_3^F\}$: mm-wave radar, vision sensor, and thermal IR camera. A thermal IR camera provides a color image where each pixel represents a temperature value, and a blob detector takes the thermal image as input and outputs a vector specifying a list of blobs found and their positions. After calculating an average RGB color of the output regions of the blob detector in a thermal image, temperature can be found based on a table lookup. Therefore, the feature detector, $\varphi_{t,2}^F$, contains the computational process to obtain an object's temperature from a thermal image.

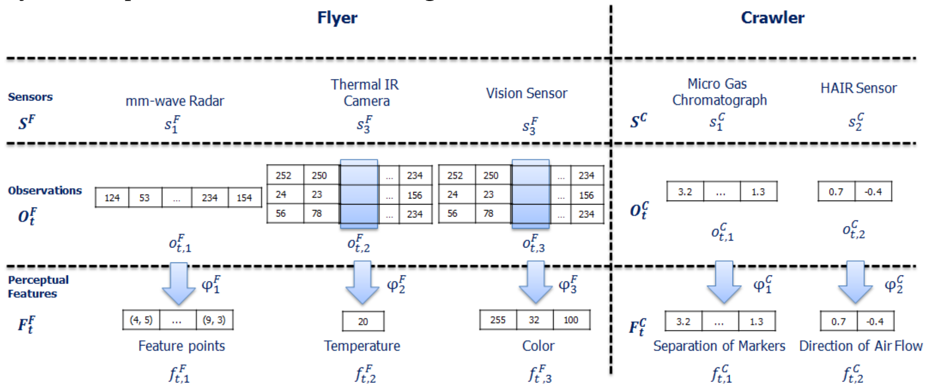

Figure 5. (Left) Flyer robot (Right) Crawler robot

The feature detector, $\varphi_{t,1}^F$, for a mm-wave radar differs from $\varphi_{t,2}^F$ because we need to extract features of a shape from a radar image. Instead of using the blob detector, we will use the line approximation to represent shapes in a radar image. Therefore, the feature, $f_{t,1}^F$, is composed of feature points of a recognized shape so that we can measure Euclidean distance between a detected feature and a prototype of a barrel shape. The feature detector, $\varphi_{t,3}^F$, for a vision sensor is to extract blobs from an image, and then returns an average RGB color of a blob as a feature.

According to the scenario, the crawler has a micro gas chromatograph and a HAIR sensor, where Figure 5 (right) depicts the feature detector of a crawler robot. Since raw sensor data of the micro gas chromatograph can be used in measuring Euclidean distance, the feature detector, $\varphi_{t,1}^C$ is a null function, but $\varphi_{t,2}^C$ will not be used for a property in the conceptual space, since direction of air flow cannot be a property of this particular object (if it were a fan it might be). However, air flow can be combined with the sensor data of the micro gas chromatograph for a robot to move toward the source of a biohazard using chemotaxis for more accurate sensor confirmation.

3.4. Learning Properties from Samples

In conceptual spaces, properties of the concept are regions in a domain. The regions can represent all samples of a concept. To represent the regions of a property, a Gaussian Mixture Model (GMM) [9] is used because a property of a concept cannot be represented by a single Gaussian in some cases. For example, the color of an apple varies (e.g., yellow, red, green), so representing the color with one Gaussian is not effective. The GMM is a parametric probability density function represented as a weighted sum of Gaussian component densities:

$$p(x|\theta) = \sum_{i=1}^{M} w_i \cdot p(x|\mu_i, \Sigma_i)$$

where x is feature vector for a property, and M is the number of Gaussian density functions. w_i is known as the mixing proportions,

$$0 \leq w_i \leq 1, \sum_{i=1}^{M} w_i = 1.$$

θ is a set containing all of the mixing proportions and model parameters,

$$\theta = \{w_i, \mu_i, \Sigma_i\}_{i=1}^{M}.$$

$p(x|\mu_i, \Sigma_i), i = 1, \dots, M$, are the component Gaussian densities,

$$p(x|\mu_i, \Sigma_i) = \frac{1}{\sqrt{(2\pi)^d \Sigma_i}} exp\left(-\frac{1}{2}(x-\mu_i)^T \Sigma_i^{-1}(x-\mu_i)\right).$$

Since each property is modeled as a mixture of Gaussians, a data association problem must be solved. There will be several clusters in the space, and the algorithm must first determine which cluster the data belongs to before updating the parameters of the model. The method used to solve this is Expectation Maximization, which alternates between estimating the association of the points to the clusters and updating the parameters of the clusters given the association.

4. Summary

We have presented the underpinnings of an overall robotic architecture being developed for use in sharing knowledge across heavily constrained microautonomous platforms with respect to power, communication, sensing, and computation. It is inspired by conceptual spaces and can be applied to multi-robot systems equipped with widely heterogeneous sensors. The conceptual space can be used to solve some of the problems that classical knowledge representations have such as symbol grounding. The abstract sensor layer converting raw sensor data to vectors is introduced in order to project the vectors into the conceptual space. Because of the abstract sensor layer, applying the architecture to various multi-robot systems is straightforward. Figure 7 illustrates the overall architecture for heterogeneous robots and how robots share the information that they individually recognize. We are in the process of implementing and testing the architecture on actual robotic platforms for the scenario described.

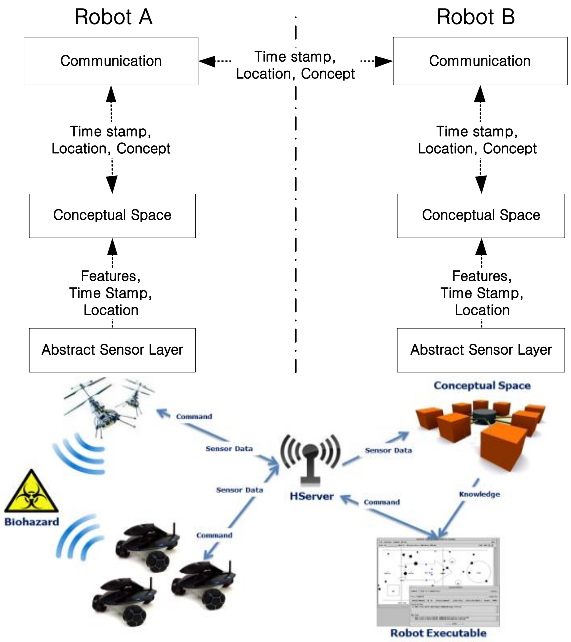

Figure 7: (Top) The architecture of the whole system is composed of three components: communication module, conceptual space, and abstract sensor layer. (Bottom) Overall System.

References

[1] Gardenfors, P., *Conceptual Spaces: The Geometry of Thought*. MIT Press, 2000.

[2] Kira, Z., "Inter-Robot Transfer Learning for Perceptual Classification", *9th International Conf. on Autonomous Agents and Multiagent Systems (AAMAS)*, 2010.

[3] Kira, Z., "Transferring Embodied Concepts between Perceptually Heterogeneous Robots", *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4650-4656, 2009.

[4] Kira, Z., "Mapping Grounded Object Properties across Perceptually Heterogeneous Embodiments", in *Proceedings of the 22nd International FLAIRS Conference*, pp. 57-62, 2009.

[5] Ulam, P., Kira, Z., Arkin, R.C., and Collins, T., 2010. "Mission Specification and Control for Unmanned Aerial and Ground Vehicles for Indoor Target Discovery and Tracking", *SPIE symposium on SPIE Defense, Security, and Sensing*, Orlando, FL, April 2010.

[6] Aisbett, J. and Gibbon, G., "A general formulation of conceptual spaces as a meso level representation," *Artificial Intelligence*, pp. 189-232, 2001.

[7] Kira, Z. Communication and Alignment of Grounded Symbolic Knowledge among Heterogeneous Robots, Ph.D. Dissertation, 2010.

[8] Arkin, R.C., *Behavior-based Robotics*, MIT Press, 1998.

[9] Moore, A. W. Clustering with gaussian mixtures. URL http://www.autonlab.org/tutorials/gmm.html. Tutorial Slides, 2004.

[10] Russell, S and Norvig, P., *Artificial Intelligence: A Modern Approach*, 2nd ed., Prentice Hall, pp. 16, 2002.

[11] Davis, R., Shrobe, H.,and Szolovits, P.,"What is in a Knowledge Representation?," *AI Magazine*, pp. 17–33, Spring 1993.

[12] Zurada, J.M., *Introduction to Artificial Neural Systems*, West Publishing Company, pp. 456-545, 1992.

[13] Naguib, R. and Sherbet, G., *Artificial Neural Networks in Cancer Diagnosis, Prognosis, and Patient Management*, CRC Press, 2001.

[14] Floreano, D., and Mattiussi, C., *Bio-Inspired Artificial Intelligence: Theories, Methods, and Technologies*. MIT Press, pp. 163-267, 2008.

[15] Champandard, A.J., *AI Game Development: Synthetic Creatures with Learning and Reactive Behaviors.* New Riders, 2003.

[16] Buckland, M., *AI techniques for game programming*. Premier-Trade, pp. 231-411, 2002.

[17] Harnad, S., The symbol grounding problem. *Physica D* 42:335–46, 1990.