# Category-Based Intrinsic Motivation

**Rachel Lee*** rlee1@swarthmore.edu
**Ryan Walker*** rwalker1@cs.swarthmore.edu
**Lisa Meeden*** meeden@cs.swarthmore.edu
**James Marshall**** jmarshall@slc.edu

*Swarthmore College
Computer Science Department

**Sarah Lawrence College
Computer Science Department

## Abstract

A goal of epigenetic robotics is to design a control architecture that implements an ongoing, autonomous developmental process which is unsupervised, unscheduled, and task-independent. The developmental process we are currently exploring contains three essential mechanisms: categorization, prediction, and intrinsic motivation. In this paper we describe a hybrid approach that uses Growing Neural Gas for categorization, neural networks for prediction, and Intelligent Adaptive Curiosity for intrinsic motivation. We apply this system to a physical robot operating in a dynamic visual environment and analyze the types of categories it forms.

## 1. Introduction

In a realistic environment, a robot is flooded with a constant stream of perceptual information. In order to use this information effectively for determining actions, a robot must have the ability to categorize its experience. Based on these categories, a robot must be able to predict how the environment will change as a result of its actions. Most importantly, this process of development should be driven by an intrinsic motivation to explore the categories of its experience where it can make the most learning progress. The developmental process we are currently exploring contains these three essential mechanisms: categorization, prediction, and intrinsic motivation.

Psychologists Ryan and Deci define intrinsic motivation as "the inherent tendency to seek out novelty and challenges, to extend and exercise one's capacities, to explore, and to learn" (Ryan and Deci, 2000, p. 70). In order to determine what to explore, an organism must compare the incoming stimuli from the environment to its internal memory to discover differences and similarities. This collative process, a term coined by the psychologist Berlyne, is necessary to evaluate the degree of novelty or incongruity of the current stimuli with respect to the organism's past experiences or expectations (Berlyne, 1966). By categorizing its experience, an organism can more effectively decide which aspects of its environment are novel and should be explored. In a recent survey of developmental robotics, Lungarella et. al. state that categorization "is of such fundamental importance for cognition and intelligent behavior that a natural organism incapable of forming categories does not have much chance of survival" (Lungarella et al., 2003, p. 161).

Although intrinsic motivation and categorization are clearly intertwined, recent work in epigenetic robotics has focused primarily on intrinsic motivation alone (Barto et al., 2004, Marshall et al., 2004, Schmidhuber, 2006). One approach to intrinsic motivation, known as Intelligent Adaptive Curiosity (IAC), employs a limited form of categorization that divides the sensorimotor space into a set of similarity-based regions (Oudeyer et al., 2007). However, no abstractions of the sensorimotor data are formed, and this top-down approach may take a long time to create categories that accurately reflect the structure of the sensorimotor space. In addition, category formation is triggered by the number of exemplars and not by the uniqueness of the exemplars, which can lead to an excessive number of similar categories. Finally, IAC's memory grows linearly with each additional experience.

In this paper we propose a hybrid system that combines IAC's approach to intrinsic motivation with a mechanism known as Growing Neural Gas (GNG), which discovers relevant categories in sensorimotor data (Fritzke, 1995). GNG's bottom-up approach to category formation quickly matches the structure of the sensorimotor space and only forms new categories when the existing ones are sufficiently different from the current data. We call this system Category-Based Intrinsic Motivation (CBIM). We apply CBIM to a physical robot operating in a dynamic visual environment and analyze the types of categories it forms. First, we summarize GNG, IAC, and introduce our hybrid system CBIM. Next, we describe the physical robot and the experiment. Finally, we present the results and discuss implications and future work.

### 1.1 Growing Neural Gas

GNG is an unsupervised learning method for dimensionality reduction (Fritzke, 1995). Given some high dimensional distribution of data, such as the sensorimotor data of a robot, a GNG will find a topological structure that closely matches the given distribution.

A GNG consists of a network of units and edges that are used to characterize the topological space in which its input vectors reside. Each unit contains a model vector that characterizes a portion of the overall distribution. Taken together, the units and edges of the GNG serve as a representative summary of the given distribution. The dimensionality of the network itself is not fixed in advance. The resulting graph is able to expand or contract as necessary by adding or deleting units and edges.

A given input vector is matched to the nearest and next-nearest GNG units based on Euclidean distance. This distance is also used as a measure of error, which the GNG stores in the nearest unit. All units connected to the nearest unit are moved toward the input vector by a fraction of the error. In this way, the GNG dynamically adapts to slight variations in the input signal that do not require the addition of new units.

Each edge in the GNG is assigned an age that is initially set to 0. If the nearest unit and the next-nearest units are not connected, an edge is placed between them; if they are connected, the age of the edge between them is reset to 0. Edges throughout the GNG above a given age threshold are pruned; if this results in isolated units, those units are also removed from the GNG.

A GNG begins with two units that are assigned random initial model vectors. In the original GNG (Fritzke, 1995), a new unit is added after a fixed number of time steps determined by the user. This unit's model vector is placed between the unit with the greatest accumulated error and its neighbor with the greatest accumulated error. In an alternative implementation called an Equilibrium GNG (Provost et al., 2006), units are only added when the average error of the GNG's units exceeds a given threshold. This approach makes it possible to grow the GNG in response to new data that doesn't fit the current topology of the network, but prevents the addition of unnecessary units when the incoming data is similar to existing model vectors.

Because a GNG is able to autonomously grow and adapt over time, it is a suitable categorization mechanism for the open-ended learning system we propose in this work.

### 1.2 Intelligent Adaptive Curiosity

IAC is a method for implementing intrinsic motivation. IAC has been successfully tested on a Sony AIBO robot operating on a baby play mat with various toys that can be bitten, swatted, and observed (Oudeyer et al., 2007). Using IAC as its control mechanism, the AIBO clearly exhibited a developmental progression, first learning about simpler aspects and later focusing on more complex aspects of its environment.

The key idea of IAC is that the drive to learn is based on maximizing learning progress. This is achieved by creating a memory of all the experiences encountered by the robot and subdividing this memory into similarity-based regions. Each region contains an "expert" that is trying to learn to predict the effect of taking actions in particular sensory situations. More formally, the expert is trying to map the sensorimotor information at time $t$ to the sensory outcome at time $t+1$: $SM(t) \longrightarrow S(t+1)$. Each region monitors the errors of the expert over time and generates a measure of learning progress, which is essentially the change in the current mean error rate with respect to an earlier mean error rate.

On each time step the robot consults this memory in order to determine which action to take. First the robot senses the world. Next, it generates a set of candidate actions, either by enumerating all possibilities or, if the space of actions is continuous, by generating a random sample of possible actions. Then it concatenates each candidate action with the current sensory information and probes the memory to find all matching regions. With some high probability it selects the candidate action associated with the region with the maximal learning progress. Otherwise it chooses a random region from the matched set. It then executes the selected action, observes the outcome, and uses this data to train the expert associated with the selected region.

When a region's sensorimotor context is predictable, initially its expert will make good progress and be chosen frequently. As the expert succeeds in learning, its progress will slow, and the learning progress of other regions will surpass it. In this way, IAC guides the robot to explore its environment in a sensible way, focusing on those aspects where it can make the best gains, and ignoring aspects that have already been learned or are unlearnable.

Although each region of IAC is in a limited sense a category, there is no abstraction taking place. Every experience the robot has had ($SM(t)$ paired with $S(t+1)$) is explicitly stored within the appropriate region. Each region is limited to a fixed maximum size (usually 250 exemplars). Once this maximum is exceeded the region is split into two new sub-regions. If the robot continues to experience very similar situations it will repeatedly form additional regions, even though they may not represent any significant differences from existing regions. This excess region formation will limit the effectiveness of IAC as it is applied to richer, more complex domains.

### *1.3 Category-based Intrinsic Motivation*

CBIM is an open-ended learning system that combines GNG's flexibility and power of abstraction with IAC's notion of region-based maximal learning progress. We use the Equilibrium GNG, which only adds units based on accumulated error. Each IAC region is associated with one GNG unit. Each GNG unit model vector is determined by all of the sensorimotor exemplars that have been mapped to it. Each region stores a fixed number of exemplars, only enough to calculate learning progress; the oldest exemplars are removed as newer exemplars are encountered.

Unlike IAC, the growth of CBIM's memory is bounded by the complexity of the robot's sensory and motor capabilities as well as the environment because categories in CBIM are formed based on error and not simply on the quantity of experience. If the robot repeatedly experiences very similar situations, the associated GNG model vectors will adjust slightly to each experience. However, a new model vector will only be created when the error across the GNG grows too high. Then the new unit will be added at precisely the point in the GNG where the model vectors are least representative of the robot's experiences. In this way, CBIM's categories mirror the robot's experiences, growing to handle new information or shrinking to remove spurious categories that are not consistent with later experiences.

In the original IAC model, the region experts were implemented as $k$-nearest neighbors (with $k$=1). In CBIM, the region experts are implemented as feed-forward neural networks with a single layer of weights. Every time a sensorimotor vector is mapped to a particular GNG unit, in the associated region the weights of the neural network expert are updated using standard backpropagation with $SM(t)$ as the input and $S(t+1)$ as the target. By using neural networks as the experts, CBIM incorporates another form of abstraction not found in IAC. Each neural network expert makes generalizations of the sensorimotor data in the process of learning to predict the outcomes of actions. These generalizations are likely to provide more robust behavior throughout the developmental process.

## 2. Experiments

In order to demonstrate the viability of CBIM, we designed a physical environment for a robot to do open-ended learning. For these experiments we used a Rovio, which is a consumer-level robot equipped with a camera. We wrote a Python interface using the open source Rovio API. The Python interface allowed us to control the Rovio through Pyrobot, a robotics control platform that implements a common API for both real and simulated robots (Blank et al., 2006). Image processing was also handled in Pyrobot.

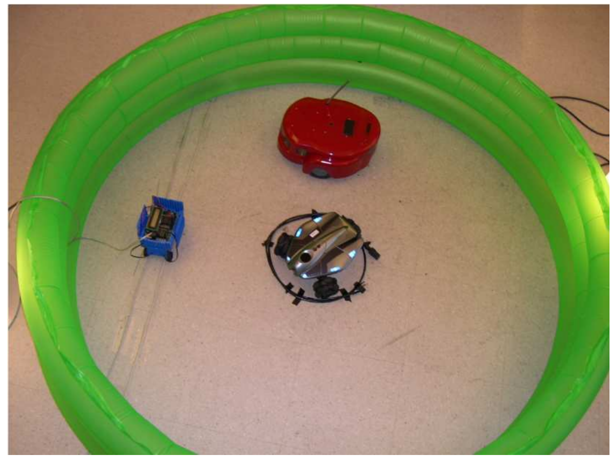

Figure 1: The experimental environment with the developing Rovio robot in the center, a larger static red robot, and a smaller moving blue robot.

The environment consisted of the Rovio in the center of a green inflatable pool as shown in Figure 1. This provided a uniform backdrop, limiting the robot's vision to the arena, thus simplifying the visual stimulus. In addition to the green background, two robots were placed within the arena to serve as other objects of focus. A large, inactive red robot was placed to one side of the environment, but close enough to completely fill Rovio's somewhat narrow field of vision when the Rovio looked directly at it. A smaller blue robot continuously moved back and forth on a track, using sensors to move from wall to wall. The Rovio was placed in the middle of this environment, capable only of rotating left or right. The Rovio could also choose not to move at all.

The developing Rovio robot experienced the world through vision, receiving sensory input extracted from its camera images. The Pyrobot vision system was configured to filter images from the Rovio's camera to find the particular colors associated with each object in the environment: the green walls, the blue robot, and the red robot. However, due to variability in lighting conditions, the filters were not completely accurate. For example, a particular object might be present in the image, but its color might not be recognized by any of the filters. Each color channel was further filtered into a blob—a bounding box surrounding the largest mass of the respective color in the image. Figure 2 shows an image of the blue robot from the Rovio's camera that has been

Figure 2: A camera image from the Rovio robot to which a blue blobify filter has been applied.

filtered for blue and then blobified.

The filter results were summarized in the sensory data as follows. Binary inputs indicated whether each color channel was active, meaning that an object of the specified color was present in the current image. Additionally, the robot received information about the color channel it *chose* to focus on. In other words, on each time step the robot could only attend to one of the color channels. This choice was part of its action decision. The area and relative position of the largest blob in this chosen channel were provided. The area was scaled to a value between 0 and 1, normalized by the size of the entire camera image. The relative position was represented as 0 for left, 0.5 for centered, and 1 for right. In summary, the Rovio had access to five sensory inputs:

S(t) $= (red, green, blue, blobArea, blobPosition)$

It was frequently the case in this environment that multiple color channels were active simultaneously. For instance, if the robot was facing the upper-left section of the environment (see Figure 1), it often had all three color channels active: it could see the red robot on its right, the blue robot on its left and the green background in between. Because the green background was almost always present in the camera image, the green channel tended to be active most of the time. Only when the Rovio was looking directly at the red robot, which tended to fill its entire camera image, would the green background not be seen. On rare occasions (about 1% of the time) none of the color channels were active due to the color variability caused by lighting conditions.

The Rovio had two output commands: which color channel to focus on and how much to rotate. The color channel choice was a value between 0 and 1 that was divided into three equal bins. For values in the range $[0.0, 0.33]$ the choice was red; for values in the range $[0.34, 0.66]$ the choice was green; for values in the range $[0.67, 1.0]$ the choice was blue. Rotation was a value between 0 and 1 that was divided into seven equal bins, ranging from a hard left, to staying still, to hard right. Thus the Rovio's motor action consisted of two values:

M(t) $= (channelFocus, rotation)$

For CBIM, this framework results in sensorimotor vectors of 7 dimensions. Therefore each GNG unit contains a 7-dimensional model vector. Each IAC region expert tries to predict the mapping from 7-dimensional sensorimotor vectors to 5-dimensional sensory vectors.

The goal of the experiment is for the robot to categorize its world, learning and making progress in predicting how the objects in this environment behave. Each object in the environment is brightly and evenly colored, so as to provide clear visual stimulus for the Rovio. Because each object in the environment has a unique color, each color channel can view only one object, eliminating possible confusion between objects.

In addition to its distinguishing color, each object offers unique learning opportunities with varying levels of predictability. The green walls offer a constant, large background making them quite predictable. For example, if on the current time step only the green channel is active and the Rovio chooses to focus on green and make a small turn to either the left or the right, it is highly likely that on the next time step the green channel will remain active and its area and relative position will be nearly identical to the previous time step. The red, static robot is also predictable, but is visible in only a few positions. For example, if the Rovio is directly facing the red robot with only the red channel active and chooses to focus on red and turn right, on the next time step it is likely that both the red and green channels will be active, and the red blob will be positioned to the left and be half the size it was previously. The blue robot is much harder to predict because it is constantly moving.

In this environment the robot must learn to predict the relative position and size of the objects in its visual field. What it will see at the next time step depends both on what it is currently seeing and what action it chooses to take. Over time, the developing robot should focus on all three objects, associating each object with a particular color channel.

Experiments lasted for 5000 time steps and took approximately 2 hours. We conducted 10 CBIM experiments using the Rovio. The IAC parameter settings were: 10 randomly generated candidate actions; 15% chance of selecting a random action; mean error rate was smoothed over 15 time steps; learning progress was calculated by comparing mean error rates separated by 10 time steps; and experts were feed-forward neural networks with a single layer of weights using a learning rate of 0.5 and no mo-

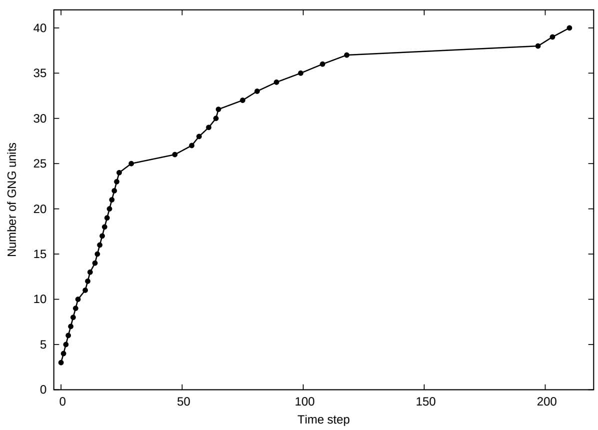


Figure 3: The number of units in the GNG of a typical CBIM run. In this run, after 210 time steps no additional units were inserted.

mentum. The Equilibrium GNG parameter settings were: error threshold of 0.5 to add a new unit; winning unit learning rate of 0.2; and neighbor unit learning rate of 0.006.

At the beginning of each experiment, the Rovio was positioned in an initial pose, facing away from the red and blue robots in the environment. This was done so as not to bias the early category formation toward either red or blue. If the developing robot initially decides to turn right, it will see the blue robot first and form categories to cover this situation. If instead, it initially decides to turn left, it will see the red robot first and form a different sequence of categories. The Rovio rapidly explored the world early in the experiment, forming many categories in the first few hundred time steps. The Rovio then attended to particular features of the environment. Common behaviors included investigating the bounds of the red robot, attempting to track the blue robot, and focusing on the area in which the blue robot occupied the same field of view as the red robot. Each experiment varied in the order that CBIM created categories and in the amount of time the developing robot spent focusing on each color channel. The next section analyzes the results in depth.

## 3. Results

Because categorization is of such fundamental importance to CBIM, we will first focus our analysis on how the GNG evolves over time. Then we will discuss the role that intrinsic motivation plays in the robot's choice of actions in its vision-based environment.

### *3.1 GNG categories*

Although results differ from run to run, there are clear GNG formation trends that directly correspond to the robot encountering new sensory experiences in its environment. Recall that the GNG begins with two units which are assigned random model vectors. Once the robot senses the world for the first time, these initial units immediately accumulate enough error to trigger the formation of new units. At the beginning of the run, only the environment's green background is within the robot's camera view. Therefore, the first new GNG units are added to reflect that the green channel is active. As the robot begins to turn, it will either turn to the left and encounter the red robot or turn to the right and encounter the blue robot. As soon as one of these robots is in view, again the GNG immediately responds by constructing new units to represent that a new color channel has been activated for the first time. As the run continues, the robot will eventually see both the red and blue robot simultaneously. The first time this occurs, the GNG creates new units to represent this unfamiliar event.

Early on, much of the incoming sensorimotor data is novel, thus the bulk of new GNG units are added in the first 100 time steps and taper off rather quickly after that. Figure 3 shows how fast the GNG grows at the start of one particular run. In this case the GNG ceases to add more units by time step 210; in other runs, the last unit is added between time steps 250 and 400. This is a clear improvement on the linear growth of IAC regions.

Figure 7 shows a series of two-dimensional representations of the GNG model vectors and edges at different time steps during the run. To create these plots, a principal component analysis was performed on the final configuration of the 7-dimensional GNG model vectors from the last time step (step 5000). The projections of the model vectors onto the first two principal components were then plotted for each of the time steps shown, with respect to the computed eigenbasis. The plots show the evolution of the model vectors and edges of the GNG in more detail.

In Figure 7, the GNG model vectors are represented by the large points that are connected via edges. The clusters of small points show the sensorimotor inputs presented to the GNG during the run, with the input on the current time step indicated by a small circle. Each cluster of points represents a similar set of sensorimotor contexts experienced by the robot. The labels represent which color channels are active.

At time step 0, the two random initial model vectors happen to both represent having the red and blue channels active simultaneously. All of the initial experiences of the robot have only the green channel active, and new GNG model vectors are added to try to reduce the error between the existing units and the new sensory data. By time step 5, it is clear

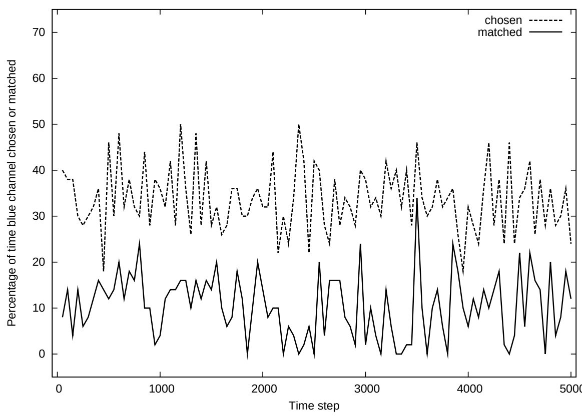


Figure 4: Results of a typical random controller.

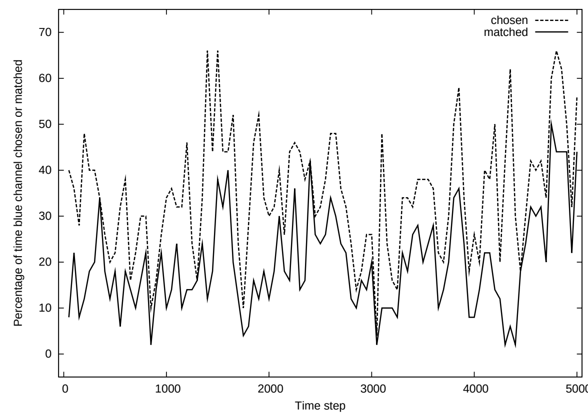


Figure 5: Results of a typical CBIM controller.

that the GNG is growing to try to accommodate the repeated green channel exemplars near the center. By time step 25, a number of GNG units are now covering the green channel exemplars. At time step 35, the robot has turned enough to see both the blue robot and red robot simultaneously (with the green walls) for the first time. The GNG is again growing to accommodate this new event. Because the GNG always adds units between existing units, it can create intermediate model vectors that may not be representative of any of the data encountered so far. By time step 65, the model vectors at the exterior of the GNG are well matched with the data, while those at the center are not. Near the end of the run, at time step 4000, many of these intermediate units and edges have been removed, and the structure of the GNG more closely matches the underlying data representation.

In the original IAC model, regions are sub-divided based only on the quantity of exemplars. Initially every experience is grouped within a single IAC region. Once this region grows beyond the size limit, it splits into two sub-regions. It can take quite a long time before the repeated splitting of IAC regions begins to accurately reflect the sensorimotor data. In contrast, CBIM's GNG regions are formed based on encountering novel experiences. Each unfamiliar event encountered by the robot is immediately marked by a new category, and only novel experiences will trigger the formation of categories.

### *3.2 Intrinsically motivated behavior*

In order to demonstrate that CBIM categorizes its sensorimotor space appropriately and uses these categories to effectively select learning experiences, a series of control experiments were executed for comparison. In the control experiments, actions were simply selected randomly without the use of categorization, prediction, or intrinsic motivation. We will refer to occasions when the Rovio selects a color channel while an object of the corresponding color is in its camera view as a matched selection. Figure 4 shows the results of a typical control run. The correlation coefficient between the percentage of time when the blue channel is chosen and when the blue channel is actually matched is only 0.17.

However, in Figure 5, the channel choice and matched selection for CBIM is much more tightly coupled, with a correlation coefficient of 0.57. When this run is divided into thirds, the correlation coefficients for each third improves from 0.42, to 0.59, ending at 0.65. This increase in correlation between choosing the blue channel and seeing the blue object indicates that Rovio was progressively learning to track the movement of the small blue robot over time.

Based on the predictability of each object, we expected that CBIM would cause the Rovio to first focus on the red stationary robot and then on the moving blue robot. Figure 6 shows the same CBIM run from Figure 5 in which a shift in focus from the red object to the blue object can be seen. In the first 1500 steps, the Rovio was not very successful at finding either the red or blue object. Then, in the middle of the run, it was able to find and focus on both the red and blue objects in turn, with a peak in finding the red object at about 2800 steps. After about 4300 steps, there is a clear shift in focus away from the red object and toward the blue object. This provides evidence of a developmental trajectory.

In the five control experiments done, the random action selection led to a focus of 33% on each of the three color channels, as expected. In contrast, in the 10 CBIM experiments done, the intrinsically motivated action selection led to a more varied focus. In three of the experiments, blue was the primary focus 37% of the time on average. In three of the experiments, red was the primary focus 37% of the time on average. Finally, in the remaining four experiments, both blue and red were the primary focus 34% of the time on average for each. In a statistical analysis of

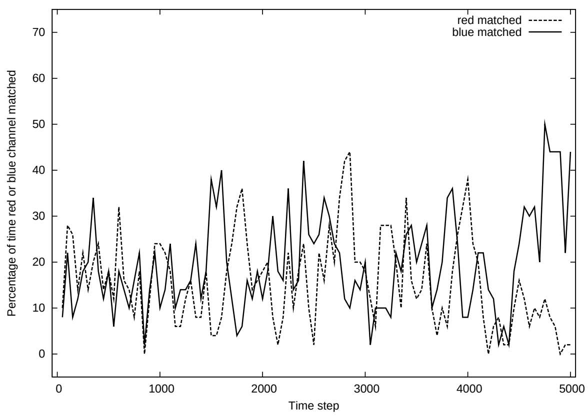


Figure 6: Evidence of a developmental shift from focusing on the more predictable red object to the harder to predict blue object.

the focus data, CBIM's green focus was significantly lower ($p < .01$) than either its red or blue focus across all of the experiments. Given that the green background was visible on nearly every time step, and was the easiest of the color channels to predict, the fact that CBIM's overall focus is primarily on the other two channels indicates that the intrinsic motivation is pushing the robot to explore the more challenging aspects of its environment.

## 4. Discussion

The results of our experiment suggest that the categorizational power of the GNG combined with the strength of IAC's measure of learning progress is effective at developing a useful set of categories that allow the robot to maximize its learning potential in the given environment. The set of model vectors developed by the GNG is a reflection of the particular characteristics of the sensorimotor stream experienced by the robot, which grows only as much as is necessary to capture the topological relationships between the data. This approach avoids the use of ad hoc mechanisms such as region-splitting and the addition of unnecessary model vectors at fixed time intervals that were present in earlier models. Recently a new variation of IAC has been developed to address some of the inefficiencies of the region-splitting approach (Baranes and Oudeyer, 2009). Yet even this improved version could benefit from the bottom-up categorization approach used in CBIM.

One remaining challenge for CBIM is its limited ability to handle time-dependent relationships in the robot's sensorimotor stream. Each expert bases its response only on the sensorimotor input at the current time step, without taking into account the recent past experiences of the robot. One possible direction of future work would be to incorporate recurrent neural networks into the model in order to take advantage of temporal information.

## References

Baranes, A. and Oudeyer, P.-Y. (2009). R-IAC: Robust intrinsically motivated active learning. In *Proceedings of the IEEE International Conference on Learning and Development*, pages 1–6.

Barto, A. G., Singh, S., and Chentanez, N. (2004). Intrinsically motivated learning of hierarchical collections of skills. In *The Third International Conference on Development and Learning*, pages 112–119.

Berlyne, D. E. (1966). Curiosity and exploration. *Science*, 153(3731):25–33.

Blank, D., Kumar, D., Meeden, L., and Yanco, H. (2006). The Pyro toolkit for AI and robotics. *AI Magazine*, 27(1):39–50.

Fritzke, B. (1995). A growing neural gas network learns topologies. In Tesauro, G., Touretzky, D. S., and Leen, T. K., (Eds.), *Advances in Neural Information Processing Systems 7*, pages 625–632. MIT Press.

Lungarella, M., Metta, G., Pfeifer, R., and Sandini, G. (2003). Developmental robotics: A survey. *Connection Science*, 15(4):151–190.

Marshall, J., Blank, D., and Meeden, L. (2004). An emergent framework for self-motivation in developmental robotics. In *The Third International Conference on Development and Learning*, pages 104–111.

Oudeyer, P.-Y., Kaplan, F., and Hafner, V. (2007). Intrinsic motivation systems for autonomous mental development. *IEEE Transactions on Evolutionary Computation*, 11(2):265–286.

Provost, J., Kuipers, B., and Miikkulainen, R. (2006). Developing navigation behavior through self-organizing distinctive state abstraction. *Connection Science*, 18(2):159–172.

Ryan, R. M. and Deci, E. L. (2000). Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being. *American Psychologist*, 55(1):68–78.

Schmidhuber, J. (2006). Developmental robotics, optimal artificial curiosity, creativity, music, and the fine arts. *Connection Science*, 18(2):173–187.

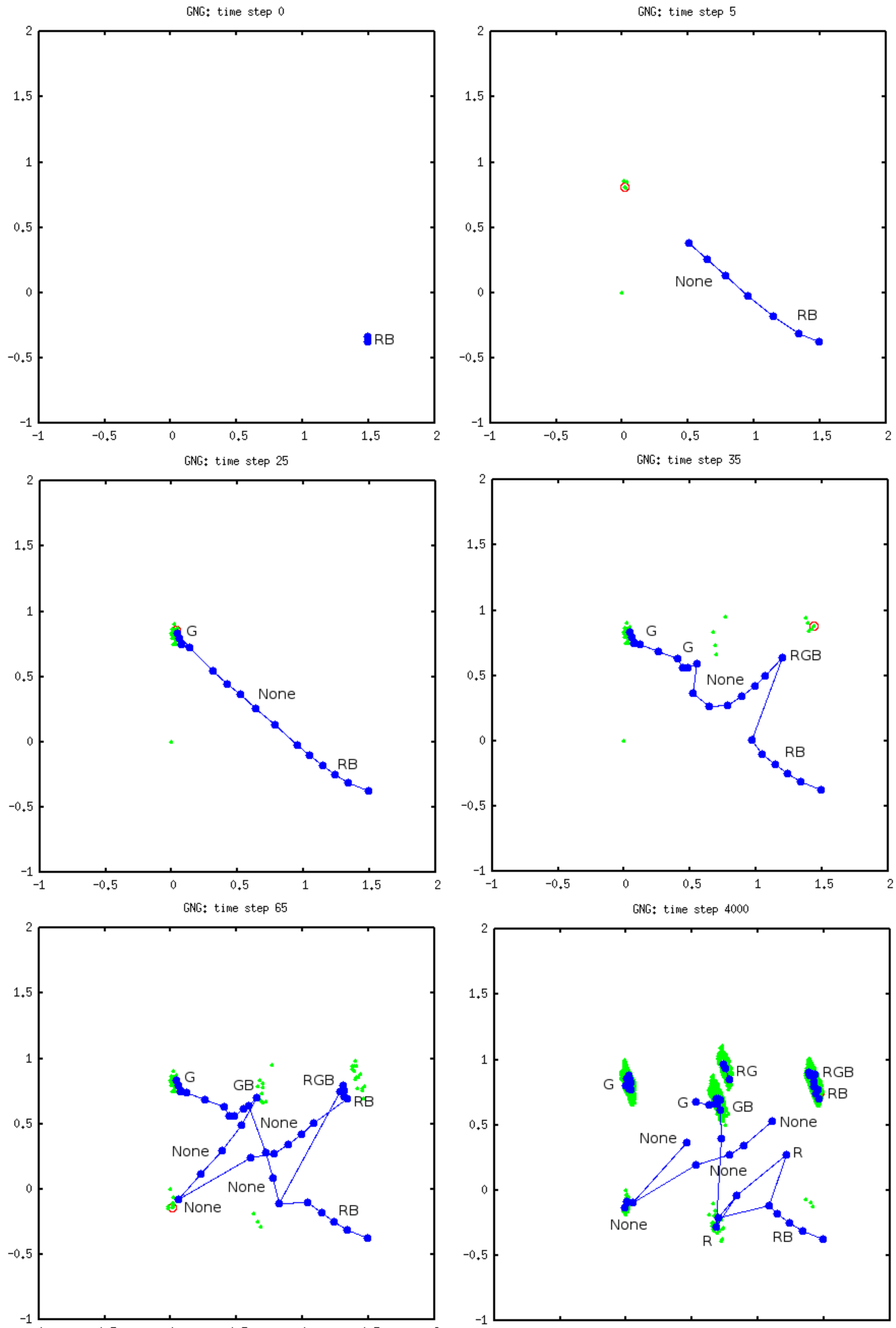


Figure 7: A 2-dimensional visualization of the 7-dimensional GNG model vectors and edges as they evolve over time in one CBIM run. The labels indicate which color channels are active for each sub-group of GNG units.