

Talking Robots: grounding a shared lexicon in an unconstrained environment

Matthieu Nottale Jean-Christophe Baillie
ENSTA-UEI cognitive robotics lab.
{matthieu.nottale,jean-christophe.baillie}@ensta.fr

Abstract

The symbol grounding problem has been identified as one of the challenges an embodied intelligent robotic system must face. The Talking Heads experiments introduced by Luc Steels in 1998 have shown interesting results addressing this issue. In this experiment, robotic agents build a shared grounded lexicon of words referring to properties of objects in their environment. This paper describes our attempts to extend its mechanisms to more autonomous robots, in the unconstrained environment of our laboratory. We show that all the concepts of the Talking Heads cannot be directly reused, as no a-priori notion of symbol referent is available, and propose a simple object model that can be used incrementally and without supervision, able to learn and recognize regions of the environment. Preliminary results for the guessing game between two aibos in our lab show success in the creation of shared grounded symbols between the two agents.

1. Introduction

The symbol grounding problem has been identified as one of the challenges an embodied intelligent robotic system must face (Searle, 1980). The Talking Heads experiments (Steels, 1998) have shown interesting results addressing this issue. In this experiment, robotic agents build a shared grounded lexicon of words referring to properties of objects in their environment. This initial experiment has been extended to cover scenes containing moving objects (Steels and Baillie, 2003), to more complex communications involving grammatical constructs, and to different learning models (Vogt and Coumans, 2003). But the relative simplicity of the environments used in these experiments limits their capability to scale up to more complex interactions, or to develop a more complex language. This paper describes our attempts to extend the Talking Heads mechanisms to more autonomous robots using Sony aibos, in the unconstrained environment of our laboratory. The next section quickly

describes the Talking Head experiments. Section three deals with the problem of attention sharing and related issues raised by the transition from cameras on tripod (the Talking Head's agents) to mobile robots. Section 4 exposes why image segmentation algorithms fail to provide referent similar to the Talking Head's geometric shapes and directly usable by the agents. Section 5 describes our object model and the algorithms used to create and update objects, and section 6 gives preliminary experimental results.

2. The Talking Heads

As the concepts and mechanisms introduced by the Talking Heads are used in the Talking Robots, we will briefly describe its main components. The environment is made of coloured geometric shapes on a white board. Each agent is equipped with binary tree classifiers for a set of visual channels such as width, redness, elongation... and a lexicon containing weighted associations between a classifier bin, called a meaning, and a word (see fig. 1). During an initial phase, each agent bootstraps its classifiers by performing the *Discrimination Game*: its classifiers are grown until the agent is able to consistently find classifier bins discriminating an object from a context of around 5 other objects.

The grounded shared lexicon is obtained by performing iterations of the *Guessing Game*: two agents are randomly selected from a population, and confronted with the same visual context of around 5 objects. A game leader called the *Speaker* is chosen randomly. The other agent is called the *Hearer*. The speaker first measures the saliency of all the objects, defined as its capacity to discriminate each object from the others, and picks one from the most salient ones. This object is called the *topic* of this game iteration. The speaker then selects a discriminating channel and the associated classifier bin for the object: no other object in the context fits in this classifier bin. It then utters the word from its lexicon with the strongest association to this meaning, randomly creating a word if necessary. The hearer then searches his lexicon for the strongest meaning associated with this word, and then for an unique object referring to this meaning. He points the object to the

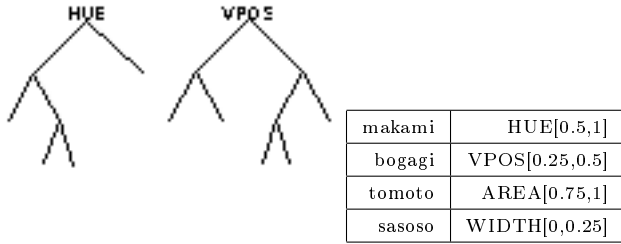


Figure 1: Sample discrimination tree and lexicon entry for the Talking Head experiment

Speaker, who notifies the Hearer if the pointed object is the topic (game success), or points the topic if the Hearer is mistaken (game failure), in which case the Hearer learns a new association between the word and one of the most salient possible meanings. The saliency-based selection process performed by both speaker and hearer is used to maximize the probability that the hearer will learn the correct meaning among all the possible ones. It of course requires saliency values perceived by both agents to be very similar.

The guessing game is designed in such a way that a high success rate in a sufficiently varied environment can only be obtained if the two agents share a common lexicon.

3. From cameras to autonomous robots

The initial Talking Heads experiment was using cameras on tripods. The task of pointing an object was implemented by exchanging the 2D image coordinates of its center. Extending to mobile robots and giving them as much autonomy as possible introduces new issues that needs to be addressed.

Pointing Since the environment is not flat, there is no precise relation to match the images perceived by the two robots. We need a pointing mechanisms that modifies the perception of both robots around the pointed zone. We achieve this using a 1mW laser pointer attached to the leg of each aibo. A robot can blink the laser by blinking one of the LEDs on its back. We use a simple blinking point detector algorithm: It sums the absolute difference between successive images over a period of a few seconds, and returns the region with the highest value, after checking its size and the contrast with the rest of the image. This algorithm can successfully detect a laser spot at up to three meters, even on black surfaces, using the 208x160 at 30fps aibo camera, provided that the scene does not contain any moving object.

The robots use this algorithm to point to objects by positioning the laser spot at less than 5 pixels from the target using a simple perception-action loop. They also use it to detect the region pointed

at by the other agent.

Moving around The capability for our agents to move around is important, as it gives them a virtually limitless environment. As the Talking Robots will require numerous hours of experimentation, we are working on a completely autonomous experiment that does not require any human intervention. Currently our aibos are able to return to their base when their battery power is low, and to restart the experiments when their battery is fully charged. The base is detected by learning simple models from different positions, made of all the visible SIFT features(scale and rotation invariant stable interest points, see(Lowe, 2004)) from the location. To walk toward the base, the aibo iteratively captures an image, finds the best-matching model (the one with the highest number of features in common with the image) and moves toward the base according to its location.

The robots can locate each other and move to a side-by-side position: the speaker sits on its back legs, and waves its forelegs. The hearer detects this movement using the same algorithm that detects the laser and walks toward the speaker. When the hearer is close enough, it signals the speaker, and both robots turn around and use their distance sensor to face each other. They then turn 90 clockwise/counterclockwise respectively and end-up next to each other, looking in the same direction.

When performing guessing games, the two agents make 10 iterations from the same position, then the speaker moves randomly to an other location, the hearer moves to its side as described above, and the process is repeated.

However, the imprecision of the walk for small amplitude movements is making the aibo miss its base approximately 20% of the times, and the hearer often knocks down the speaker when walking toward him due to false readings of the distance sensor. Further work is thus required to make this experiment more robust.

4. First attempt

Our initial attempt aimed at staying as close as possible from the Talking Heads in its principles and data structures.

The two cameras are replaced by two Sony aibos sitting next to each other in our laboratory, looking in the same direction.

The white board and its geometric shapes are replaced by the scene in front of the robots.

To select a context, the Speaker turns its head randomly and points to the center of the image. The Hearer detects the laser, and centers it in its field of view. Everything in their field of view (50 degrees) is part of the context.

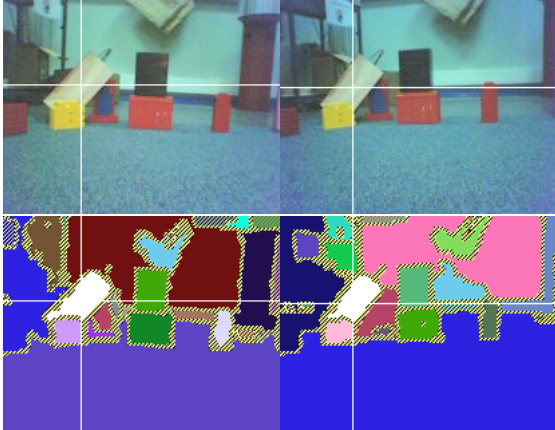


Figure 2: Sample images and CSC segmentation for a guessing game.

To obtain objects similar to the Talking Heads geometric shapes, each robot applies a segmentation algorithm to the image, tuning its parameters to obtain between 5 and 10 regions. We found that CSC was the most appropriate segmentation algorithm for this task (Baillie and Nottale, 2005).

The remaining steps remain unchanged: each object is associated with a value for each perceptual channel (width, height, position, elongation, RGB or HSV), which in turn is associated with a meaning from the discrimination trees. The speaker selects the topic among the salient objects, then finds and utters a word from its lexicon discriminating it from the other objects. The hearer searches an unique referent for this word and points it with its laser. The speaker detects the laser and signals success if it points the correct object, or signals failure and points the topic.

However, experiments using this setup have been unsuccessful in obtaining a shared lexicon between two agents: small variations in the perceived scene of each agent due to camera noise and slightly different perspective introduce differences in the segmentations which in turn provoke significant differences in the perceived channel values for all objects. Consequently the saliency values are also very different from one agent to the other. Since the world perceived by the agents are too different from each other, they are unable to exchange word-meaning relationship using this setup.

5. Talking robot object model

As we have shown in the previous section, the fact that we cannot rely on an a-priori 2D segmentation means we do not have an a-priori notion of object to act as symbol referent as the Talking Heads did: we have to create it. Of course, this concept will be less abstract than the common meaning of the word “object”: aibos are barely more than mobile cameras,

with very limited capabilities to modify their environment, so this object notion will be based purely on perception. The concept closest to our “objects” could in theory be reached by performing some kind of complete 3D mapping of the environment, but no such technique has been devised yet, and the limited (2D) movement capabilities and the low camera resolution would make it unlikely to succeed.

Instead, we chose to base our objects on low-level visual primitives, using the bag-of-words model (Dance et al., 2004): an object is defined and recognized as the co-occurrence of multiple local region descriptors (features) in the same region of an image, without taking into account their relative positions.

As a consequence, words learned by the agent will no longer corresponds to abstract properties of objects, but to objects themselves, and thus will not be usable directly in completely novel scenes. We argue that words for specific objects is a necessary step, as a shared concept of object is not available for the Talking Robots. Further experiments and interactions between the agents, using these shared objects will then be able to extract more abstract properties from them, and link them to words in the same shared and grounded way.

The rest of this section describes in details our object model, and the object creation and recognition steps. Object categorization and recognition is a very active research field, but few work has been made with the set of constraints needed for the Talking Robots: near-realtime learning and recognition, unsupervised object creation, and incremental operations (i.e. new objects can be learned at any time).

From features to elementary words. Many feature detectors have been developed by researchers, with robustness to perspective changes as their design goal, which makes them good building blocks for the Talking Robot objects. We designed our system to be able to use as many descriptors as possible, as their performance depends a lot on the type of environment.

Since feature comparison might be a costly operation, we use the classical dictionary approach: features from each algorithm are classified in a discrimination tree that assigns one or more words to each feature. The words are then used in a later step as building blocks for more abstract objects construction.

We now need to answer the question of how to build a dictionary of features fed incrementally. Since we have no a-priori knowledge of what an appropriate cluster radius will be for each feature detection algorithm, we have to perform hierarchical clustering.

There are two possible approaches for this prob-

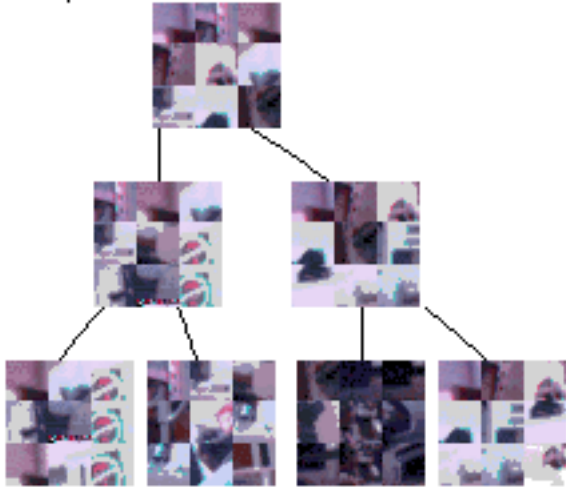


Figure 3: A sample sub-tree of a discrimination tree using SIFT.

lem: using a fixed cluster radius, or using a fixed number of children nodes per parent node.

We experimented with the first approach, but the high-dimensionality of features such as SIFT (Lowe, 2004) (128) makes it hard to obtain interesting (i.e. not degenerated) hierarchical structures.

The second approach we experimented with is very similar to (Nistér and Stewénius, 2006). Our structure is a binary tree. Each leaf node maintains a randomly selected sample of all the features associated with it (we used 100 features). When a node is selected for growing according to the Talking Head rule (the more used a node is, the more likely growing will occur), it uses its sample to split the feature space in two and associate each subspace to one of its children nodes, in one of two possible ways:

- by finding the hyperplane orthogonal to one axis that minimizes the sum of intraclass variance when used as a frontier, as in (Oudeyer et al., 2007).
- by randomly choosing two features f_1, f_2 : a feature f is in the left subspace if $d(f, f_1) < d(f, f_2)$. We use this method if the feature distance function is not L1 or L2, since computation of the previous method can't be optimized in that case.

The node's sampled features are then split among the two created children nodes.

The resulting tree is then used as a dictionary: when a feature is presented, it traverses the tree from the root node to one of the leaf nodes. Each node along the path is returned as a dictionary word matching this feature.

One potential problem is that nodes closer to the root node will match more features than nodes further down in the tree. The higher object-creation level does not directly have access to this information, but can measure node occurrence frequency and exploit it to use only the relevant nodes, i.e. the ones neither too generic nor too specific.

We have integrated the following algorithms in our framework: histograms and correlograms (Huang et al., 1997) with L1, L2, EMD (Rubner et al., 1998) or diffusion distance (Ling and Okada, 2006), SIFT and k-adjacent segments (Ferrari et al., 2006).

From elementary words to objects. At this point we have reduced an image to a set of dictionary entries, with their locations. Our next step is to detect frequently occurring subsets of those entries close to each other.

An object model is stored as a vector that records the probability of occurrence of each dictionary entry in the presence of the object. Global statistics on the a-priori occurrence probability of each entry are also stored. The object recognition and creation algorithm proceeds as follows: the image is divided in an overlapping multiscale grid, and each image window is treated independently. Feature detector algorithms are applied, and dictionary entries corresponding to each detected feature are extracted, yielding a word occurrence vector for each window. This vector is compared against all the object models using a comparison algorithm. We implemented so far:

- L1 and L2 norm directly on the occurrence vectors.
- L1, L2 and normalized scalar product (angle) between the term-frequency inverse-document frequency transformed vectors as described in (Squire et al., 1999) (Sivic and Zisserman, 2003). The TF-IDF, initially used in the textual information classification field associates to each component a weight proportional to its probability of occurrence in the document, and inversely proportional to its probability of occurrence in the whole corpus. The rationale is that the most frequent the term is, the less useful it is to discriminate between two document models. The exact formula used is $w_i = p_{di} \log \left(\frac{1}{p_i} \right)$ where w_i is the TF-IDF weight for component i , p_{di} is the probability that a feature in document d is i and p_i is the probability that a feature in the whole corpus is i .
- A probabilistic model using Bayes rule and the (obviously false) hypothesis that all the features are independent.

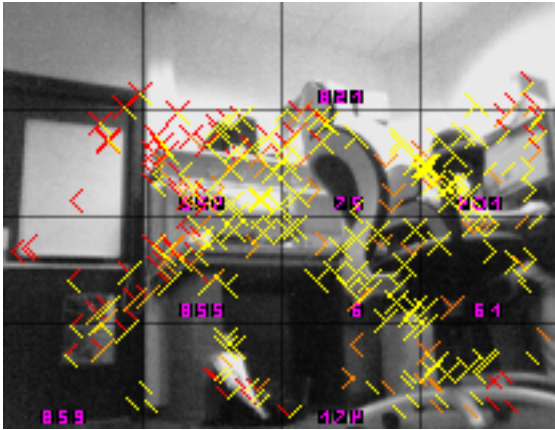


Figure 4: Image with features and detected objects. The color of each arrow branch indicates the relative weight of the corresponding feature in the bucket pointed at by the branch.

All the results presented in the next section use the TF-IDF L2 distance, determined to be the most efficient in preliminary experiments.

We obtain for each window a matching score against each object model.

The system then classifies a window as an instance of the model with the best score if this score is F times greater than the average of the N best scores. Models are then updated according to the content of the windows containing them.

If no object matches a window, a new object model is created and initialized with its content. This creation only occurs with a fixed probability to limit the rate at which objects are added.

The total number of objects is kept below a fixed value. When this value is exceeded, the least seen object older than a fixed amount of frames is deleted. This grace period gives each object some time to be seen again before deletion.

Comparing the best match with the N next matches (we use $N =$ between 5 and 10) instead of only the second match prevents the system from degenerating when two models are very close from each other.

We expect this system to constantly create and delete objects, most of them being noise never detected after being created, but to also have a few stable objects, whose model gets refined each time they are detected.

6. Experimental results

6.1 Unsupervised object creation in the Talking Robots setup

Since the complete Talking Robots experiment associates words with objects, we need our system to be able to maintain a set of object models over time,

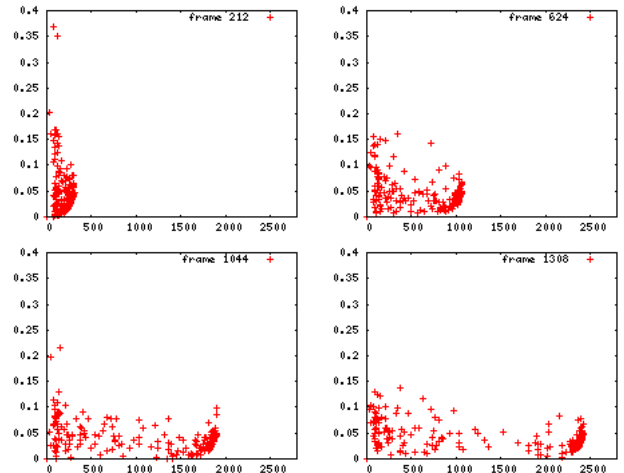


Figure 5: Object score against their creation rank at 4 different frames(see section 6.1).



Figure 6: Sample images from the aibo in our lab

and not to constantly delete and recreate them. To evaluate this capability, we placed a Sony aibo robot in the center of our lab, and had it move around and turn its head randomly, feeding pictures to the system described in the previous section. Figure 5 shows the internal state of the system in the form of object weights against their IDs at various frame numbers. The ID of an object is its creation rank. An object's weight gets decreased every frame by a fixed factor, and increased each time the object is seen. In this run the maximum number of objects is fixed to 150, and the grid contains 4x4 windows at the lowest level. The group of points on the right of each graph corresponds to all the recently created object models. Since most of them won't be detected again, their weights steadily decrease until they exit the grace period and are dropped. We observe that many of the objects created in the first frames are still present at frame 1300, which is encouraging.

6.2 Offline tests on the GRAZ-02 database

To validate the clustering capabilities of our discrimination trees and object models, we tested them on the GRAZ02 bike database(Opelt et al., 2006). This database contains images of bikes in various poses, sizes and backgrounds, with a lot of clutter and represents well the complexity of the environment in our

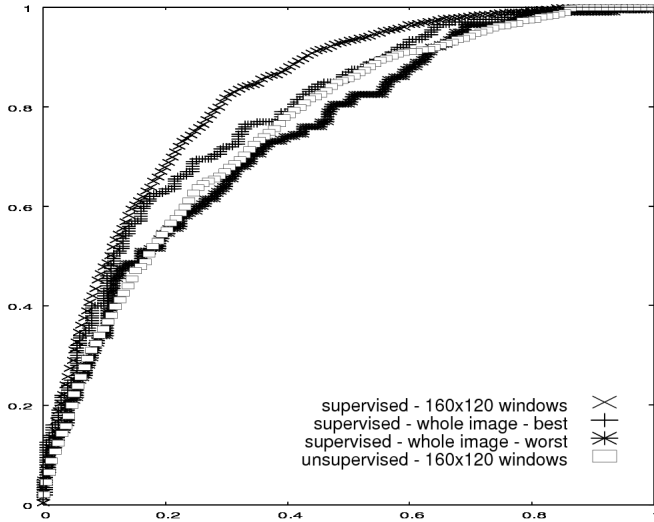


Figure 7: ROC curves for various setups on GRAZ02 bikes

laboratory.

We first tested a supervised learning setup on whole images: 150 random images were used to train a unique object model. This model is then tested on 150 other images from the bike set, and 150 images from the background set.

Our second setup is similar, but uses 160x120 windows instead of the whole image: each image is cut into non-overlapping 160x120 windows, and ground truth masks provided with the database are used to decide which class a window belongs to: bike if more than 10% of the area is within the mask, background otherwise. As in the previous setup, an unique object model is learnt from all the “bike” windows extracted from 150 random images, and tested with all the windows from 150 bike and 150 background images.

Our last setup is almost unsupervised: 160x120 pixels windows from 50 images are fed to the system in its unsupervised object creation mode, with a limit of 100 objects and an acceptance threshold of $F = 1.6$. This value was experimentally determined to obtain clusters made mostly of only background or only bike images. Then the models are labeled bike or background based on the class that contributed the most to each model creation. For testing, only the bike models are kept, and used to score windows from images in the bike and background sets. The best score from all the models is used for evaluation. Figure 8 shows a sample of the images matching 5 different models.

To evaluate the results, ROC curves are produced for each setup: they represent for each possible threshold the true positive detection rate against the false positive detection rate. Figure 7 summarizes the results.

Using whole images in supervised mode there is a lot of variance between the tests: ROC equal er-

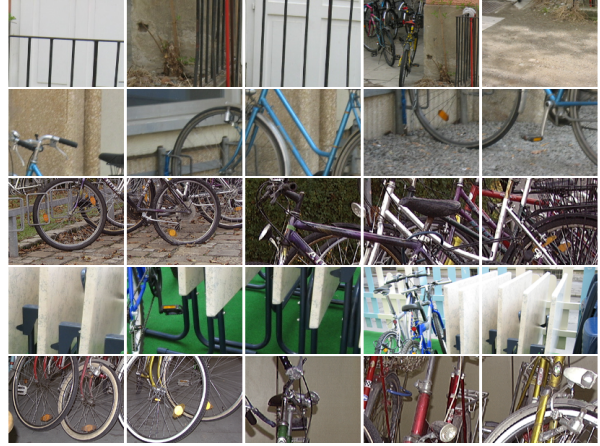


Figure 8: Instances of 5 object models generated in an unsupervised way on GRAZ02 bikes

ror rate (when true positive+false positive equals 1) is between 0.65 and 0.7. On average, bikes occupy only a small proportion of the image. Using 160x120 windows, the ROC-EER reaches 0.75 with much less variance. The unsupervised mode tests achieve a ROC-EER of 0.69 on average.

These results show that our classifier and object system can be used for non-trivial clustering tasks. Our results are below state of the art on the GRAZ02 database (see(Moosmann et al., 2006) for instance), but our model is built with more constraints: it is completely incremental (the feature dictionary can be expended, classes can be modified, and new classes added at any time) and unsupervised.

6.3 Talking Robot guessing game

At the time of this writing, we did not have time to run enough guessing game iterations using the complete setup to obtain results. We report here results obtained by having two aibos sit on their bases next to each other, which permits to run experiments 24h/24. The environment is thus limited to a fixed point of view 180 degrees wide.

Each robot bootstraps its discrimination tree for the SIFT detector on its own by turning its head randomly and feeding all features to the system, until 3000 nodes are reached. Each robot then feeds its object models for 1000 iterations using a 4x4 grid and a maximum of 120 objects.

Then object models are frozen and the guessing game is played by the two robots, swapping roles every 1000 iterations to speed up the learning process.

All the behavior and movement related functions are implemented in URBI. This code is interfaced with the C++ code handling vision primitive and object classification using the URBI UObject API.

Due to time constraints we were only able to perform 1500 iterations using this setup. The average

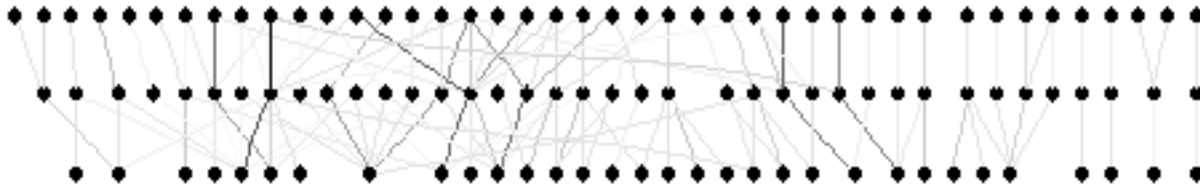


Figure 9: Lexicons after 1000 guessing game iterations. top row is the speaker objects, middle row is the lexicon words, and bottom row is the hearer objects. The intensity of the edge indicates the association's weight in the lexicon.

success rate on the last 100 iterations is 25%. This may seem low, but the main cause of failures are pointing failures: the geometry of the environment is such that one robot often points to a location masked from the view of the second robot by an object in front of him.

If we only consider games during which the speaker successfully detects the hearer's laser, the success rate reaches 50%.

Contrarily to the Talking Heads, we cannot directly define a notion of "correctness" between the agent's lexicons, as the words are referring to objects meaningful only to the agent that created them. We can however observe the ratio of homonyms and synonyms by plotting the associations as in figure 9. The dots on the top row are the object models for one of the agent, the third row are the object models for the second agent, and the middle row are the shared words. The intensity of a link corresponds to the weight of the association.

We observe a few strong one-to-one associations between the objects of both agents, and a few one-to-two associations. Note that strong one-to-two objects associations are likely not homonyms, but are indicating that two object models fit well the same region of the environment for one of the agents.

7. Conclusion

Scaling grounding symbol experiments deriving from the Talking Heads to more complex interactions and languages will require rich and complex environments. In this paper, we described experiments aiming to achieve symbol grounding between autonomous robots in such a richer environment. We addressed the issue of pointing raised by the extra complexity, and showed why the Talking Head model could not be directly reused. We replaced the Talking Head's geometrically shaped referent with bag-of-feature based models of regions of the visual environment, and introduced algorithms to create and recognize such models incrementally and unsupervised. Preliminary results shows successful symbol grounding and communication taking place between the two agents.

Further work will involve finalizing and integrating the fully autonomous version of our guessing

game, allowing us to run thousands of iterations. We then plan to integrate curiosity measures inspired from (Oudeyer et al., 2007) to have the aibos autonomously move and select the environment based on its "interest" for the interaction they are performing.

References

- Baillie, J.-C. and Nottale, M. (2005). Segmentation stability: A key component for joint attention. In *5th International Conference on Epigenetic Robotics*.
- Dance, C., Willamowski, J., Fan, L., Bray, C., and Csurka, G. (2004). Visual categorization with bags of keypoints. In *ECCV International Workshop on Statistical Learning in Computer Vision, Prague*.
- Ferrari, V., Fevrier, L., Jurie, F., and Schmid, C. (2006). Groups of adjacent contour segments for object detection. INRIA technical report.
- Huang, J., Kumar, S., Mitra, M., Zhu, W., and Zabih, R. (1997). Image indexing using color correlograms. In *IEEE Computer Vision and Pattern Recognition*, pages 762–768.
- Ling, H. and Okada, K. (2006). Diffusion distance for histogram comparison. In *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 246–253, Washington, DC, USA. IEEE Computer Society.
- Lowe, D. (2004). Distinctive image features from scale-invariant keypoints. *International Journal on Computer Vision*, 60(2):91–110.
- Moosmann, F., Larlus, D., and Jurie, F. (2006). Learning saliency maps for object categorization. In *ECCV International Workshop on The Representation and Use of Prior Knowledge in Vision*. Springer.
- Nistér, D. and Stewénius, H. (2006). Scalable recognition with a vocabulary tree. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 2161–2168.

- Opelt, A., Pinz, A., Fussenegger, M., and Auer, P. (2006). Generic object recognition with boosting. *IEEE Transactions on Pattern Recognition and Machine Intelligence (PAMI)*, 28(3):416–431.
- Oudeyer, P.-Y., Kaplan, F., and Hafner, V. V. (2007). Intrinsic motivation systems for autonomous mental development. *IEEE transactions on evolutionary computation*, 11(2):265–286.
- Rubner, Y., Tomasi, C., and Guibas., L. J. (1998). A metric for distributions with applications to image databases. *Proceedings of the 1998 IEEE International Conference on Computer Vision*, pages 59–66.
- Searle, J. (1980). Minds, brains and programs. *Behavioral and brain sciences*,, 3:417–421.
- Sivic, J. and Zisserman, A. (2003). Video google: A text retrieval approach to object matching in videos. In *Ninth IEEE International Conference on Computer Vision*, volume 2, pages 1470–1477.
- Squire, D., Muller, W., Muller, H., and Raki, J. (1999). Content-based query of image databases, inspirations from text retrieval: inverted files, frequency-based weights and relevance feedback. The 10th Scandinavian Conference on Image Analysis (SCIA'99).
- Steels, L. (1998). The origin of syntax in visually grounded robotic agents. *Artificial Intelligence*, 103:133–156.
- Steels, L. and Baillie, J.-C. (2003). Shared grounding of event descriptions by autonomous robots. *Robotics and autonomous systems*, 43:163–173.
- Vogt, P. and Coumans, H. (2003). Investigating social interaction strategies for bootstrapping lexicon development. *Journal of Artificial Societies and Social Simulation*, 6(1).