Using attentive focus to discover action ontologies from perception

Amitabha Mukerjee
Department of Computer Science & Engineering
Indian Institute of Technology Kanpur
amit@cse.iitk.ac.in

Abstract

The word "symbol", as it is used in logic and computational theory, is considerably different from its usage in cognitive linguistics and in everyday life. Formal approaches that define symbols in terms of other symbols ultimately need to be grounded in perceptual-motor terms. Based on cognitive evidence that the earliest action structures may be learned from perception alone, we propose to use attentive focus to identify the agents participating in an action, map the characteristics of their interaction, and ultimately discover actions as clusters in perceptuo-temporal space. We demonstrate its applicability by learning actions from simple 2D image sequences, and then demonstrate the learned predicate by recognizing 3D actions. This mapping, which also identifies the objects involved in the interaction, informs us on the argument structure of the verb, and may help guide syntax. Ontologies in such systems are learned as different granularities in the clustering space; action hierarchies emerge as membership relations between actions.

1 Introduction

Learning the concepts for concrete objects require the perceptual system to abstract across visual presentations of these objects. In contrast, modeling actions present a more complex challenge [Fleischman and Roy, 2005], [Sugiura and Iwahashi, 2007]. Yet actions are the central structure for organizing concepts; the corresponding language units (verbs) also acts as "heads" (predicates) in sentences, controlling how an utterance is to be interpreted. Typically the structure for an action/verb includes a set of possible constituents that participate in the action, and also some constraints on the type of action (e.g. the type of motion that may constitute "A chases B").

In this work, we consider the learning of the structure of actions, based on image sequences. Cognitively, there is evidence that some action schemas are acquired through perception in a pre-linguistic stage [Mandler, 2004]; later these are reinforced via participation, and may eventually seed linguistic aspects such as argument structure.

We postulate that a key aspect of this process is the role of perceptual attention [Regier, 2003],[Ballard and Yu, 2003]. Thus, an action involving two agents may involve attention shifts between them, which helps limit the set of agents participating in the action. The set of agents participating in an action eventually generalizes to the argument structure. In [Ballard and Yu, 2003], human gaze was directly tracked and matched with language fragments, and verbs such as "picking up" and "stapling" were associated with certain actions. However, the verbal concepts learned were specific to the context, and no attempt was made to generalize these into action schemas, applicable to new scenes or situations. Top down attention guided by linguistic inputs is used to identify objects in [Roy and Mukherjee, 2005]. More recently, in [Guha and Mukerjee, 2007] attentive focus is used to learn labels for simple motion trajectories, but this is also restricted to a particular visual domain.

1.1 From Percept to Concept to Symbol

The word "symbol", as it is used in logic and computational theory is considerably different from its usage in cognitive linguistics and in everyday life. The OED defines it as "Something that stands for, represents, or denotes something else". This meaning carries over to the cognitive usage, where it is viewed as a tight coupling of a set of mental associations (the *semantic pole*) with the psychological impression of the sound (the *phonological pole*) [**?**]. Formally, however, a symbol is detached from any meaning, it is just a token constructed from some finite alphabet, and is related only to other such tokens. A computer system dealing with such symbols can define many relations with other symbols, but finds it difficult to relate it to the world, and this makes it difficult also to keep the relations between symbols up to date. The objective of this work is to try to align a symbol to a perceptual stimulus, so as to provide grounding for the symbols used in language or in reasoning.

In other work, we have addressed the question of

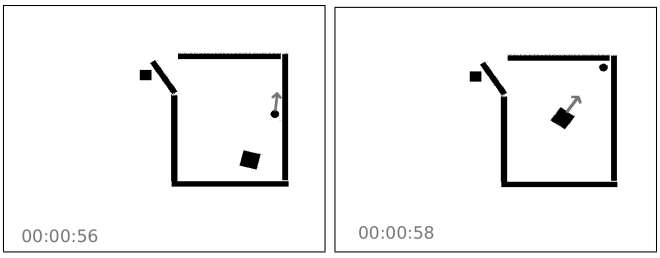

Figure 1: *Scenes from 2D video: "Chase"*: Three agents, "big square", 'small square" and "circle" play and chase each other. Velocities are shown as gray arrows.

learning the language label (or the phonological pole) of a symbol [Satish and Mukerjee, 2008]. Here we focus on modeling the semantic pole, especially with respect to action ontologies. Such models, called *Image Schema* in Cognitive Linguistics [Langacker, 1999] or *Perceptual Schema* in Experimental Psychology [Mandler, 2004], involve abstractions on low-level features extracted from sensorimotor modalities (positions and velocities), as well as the argument structure.

We ask here if, given a system that is observing a simple 2D scene (see fig. 1) with shapes like squares and circles chasing each other, is it possible for it to cluster all 2-agent interactions in some meaningful way into a set of action schemas? If so, do these action schemas relate reliably to any useful conceptual structures? Further, is there any possibility of learning any relationships between these action schemata, thus constructing a primitive ontology? Note that all this has to take place without any language, witout any human inputs in any form.

Constructing such action templates has a long history in Computer vision, but most gather statistics in view-specific ways with an emphasis on recognition [Xiang and Gong, 2006; **?**]. We restrict ourselves to two-object interactions, using no priors, and our feature vectors are combinations of relative position and velocity vectors of the objects (we use a simple inner product). We perform unsupervised clustering on the spatio-temporal feature space using the Merge Neural Gas algorithm [Strickert and Hammer, 2005]; the resulting clusters constitute our action schemas. By considering different levels of cluster granularity in the unsupervised learning process, we also learn subsets of coarse concepts as finer action concepts, resulting in an action hierarchy which may be thought of as a rudimentary ontology.

Having learned the action schema based on a given input, we apply it to recognize novel 2-body interactions in a 3D fixed camera video, in which the depth of a foreground object is indicated by it's image y-coordinate. We show that the motion features of humans can be labelled using the action schemas learned.

2 Analysis: Role of Attentive Focus

One of the key issues we explore in this work is the relevance of perceptual attention. It turns out that restricting computation to attended events somehow results in a better correlation with motions that are named in language. This may reflect a bias in conceptualization towards actions that attract attention. Like other models that use attention to associate agents or actions to language [Ballard and Yu, 2003; Guha and Mukerjee, 2007], we use attentive focus to constrain the region of visual salience, and thereby the constituents participating in an action. We use a computational model of dynamic visual attention [Singh *et al.*, 2006] to identify agents possibly in focus.

In order to analyze the different types of motion possible in the scene, we first perform a qualitative analysis of the motions. We assume that all objects have an intrinsic frame with a privileged "front" direction defined either by its present direction of motion, or by the last such observed direction. Let the reference object be A, then the pose of located object B w.r.t. the frame of A can be described as a 2-dimensional qualitative vector [Forbus *et al.*, 1987], where each axis is represented as $\{-, 0, +\}$ instead of quantitative values. This results in eight possible non-colliding states for the pose of B. In each pose, the velocity of B is similarly encoded, resulting in 9 possible velocities (including non-moving).

This results in 72 possible relations, and distinguishing the situation when the reference object A is moving, from that when it is stationary, results in a total of 144 possible states. Linguistic labels(Come-Close(CC), Move-Away(MA), Chase(CH), Go-Around(GoA), Move-Together(MT), Move-Opposite(MO)) are manually assigned to each of these qualitative relative motion states. The motion in nearly half the states do not appear to have clear linguistic terms associated with them, and these undenominated interactions are left empty. The remaining classes assigned are shown in Figure 2. Qualitative classification for the frames in Fig.1 is shown in Fig. 3.

Next, we analyze the frequency of these cases observed on the Chase video. Fig. 2 compares the frequency of the qualitative states with non-stationary first object, in the situation where all possible object pairs are considered (no attentive focus), versus that where using attentive cues pairs of agents attended to within a temporal window of 20 frames become candidates for mutual interaction; all other agent pairings are ignored. The frequency of indeterminate qualitative cases are 58% in the first situation and 24% in the second. Thus, attentive focus biases the learning towards relations that we have names for in language.

3 Visual Attention

We consider a bottom-up model of visual attention (not dependent on task at hand) [Itti, 2000]. Here we consider a model designed to capture bottom-up attention in dynamic scenes based on motion saliency [Singh *et al.*,

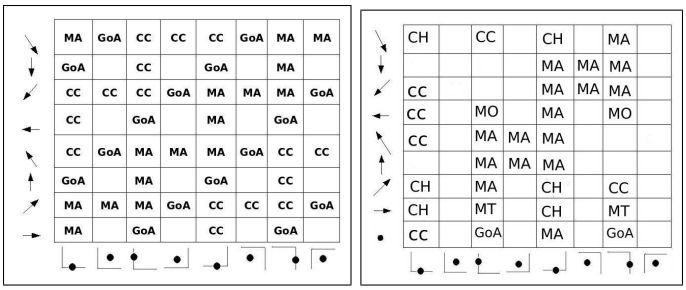

Figure 2: Qualitative analysis of two object interaction: Single frame qualitative classification for *(a)* stationary first object and *(b)* when the first object is moving horizontally to the right. X-axis gives the different positions of the second object and Y-axis gives the different velocity directions(including zero velocity) of second object w.r.t. the first object at origin. Cases when motion does not have a simple English label are blank. Others labels are: Come Closer (CC), Move {Away,Opposite, Together} (MA,MO,MT), Chase (CH) and Go Around (GoA)

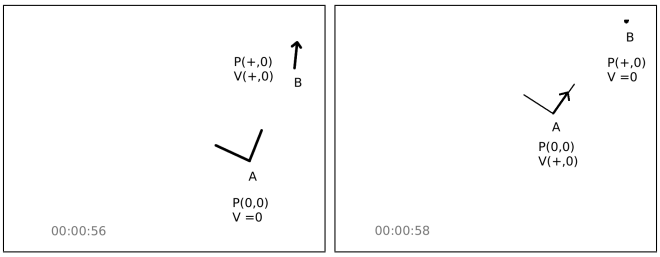

Figure 3: Single frame qualitative classification for the frames in Fig.1. The big square is taken as the first object. The labels assigned are MA(*left*) and CC(*right*). P and V refer to the position and velocity in the reference frame with origin at the first object and x-axis along its velocity.

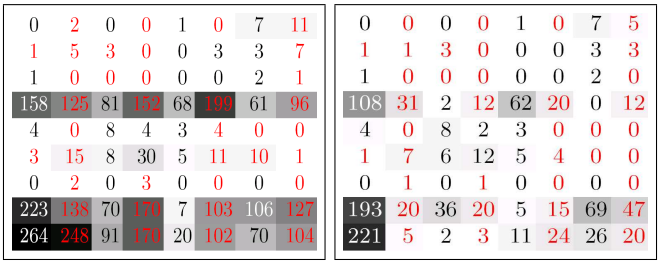

Figure 4: Without and with Attention: Frequency map for Single frame qualitative classification for the case of non-stationary first object. % of the feature vectors that can not be labelled (given in Red) is 58% without attention, and 24% with attentive focus.

2006]. Objects are taken as the attentive foci instead of pixels. Motion saliency map is computed from optical flow, a confidence map is introduced to assign higher salience to objects not visited for a long time. A small foveal bias is introduced to mediate in favour of proximal fixations against large saccadic motions. Winner-Take-All network on the combined saliency map gives the most salient object for fixation.

Name	Formula
pos·velDiff	$(\vec{x}_B - \vec{x}_A) \cdot (\vec{v}_B - \vec{v}_A)$
pos·velSum	$(\vec{x}_B - \vec{x}_A) \cdot (\vec{v}_B + \vec{v}_A)$

Table 1: Dyadic Features Formulae. A and B refer to the two objects; A is said to be the Reference Object (The more salient, usually the larger of the two objects, is taken as Reference Object) and B the Located Object in the feature computation; $\vec{v_A}$ refers to velocity vector of A; $\vec{x_A}$ refers to position vector of A; and '$\cdot$' refers to the inner product of the vectors.

4 Unsupervised Perceptual Clustering

Perceptual systems return certain abstractions of the raw sensory data - "features" - which are used for recognition, motor control, categorization, etc. In this work we use two features that capture the interaction of two agents. All learning takes place in the space of these two features, (Table 1); the first feature captures the combination of relative position and velocity, the second the relative position and magnitude.

These feature vectors are then clustered into categories in an unsupervised manner based on a notion of distance between individuals. We use the Merge Neural Gas(MNG) algorithm[Strickert and Hammer, 2005] for unsupervised learning which has been shown to be well-suited for processing complex dynamic sequences as compared to the other existing models for temporal data processing like Temporal Kohonen map, Recursive SOM etc. This class of temporal learning algorithms are more flexible with respect to the state specifications and time history compared to HMMs or VLMMs. MNG algorithm performs better than other unsupervised clustering algorithms like K-Windows [Vrahatis *et al.*, 2002], DBSCAN [Ester *et al.*, 1996] because of the utilization of the temporal information present in the frame sequences unlike the other algorithms.

4.1 Merge Neural Gas algorithm

The Neural Gas algorithm [Martinetz and Schulten, 1994] learns important topological relations in a given set of input vectors (signals) in an unsupervised manner by means of a simple Hebb-like learning rule. It takes a distribution of high-dimensional data, P(ξ) and returns a densely connected network resembling the topology of the input.

For input feature vectors arriving from temporally connected data, the basic neural gas algorithm can be generalized by including explicit context representation which utilizes the temporal ordering present in the feature vectors of the frames, resulting in the Merge Neural Gas algorithm [Strickert and Hammer, 2005]. Here, a *Context* vector is adjusted based on the present winning

Table 2: *Clustering Accuracy*: The i^{th} row, j^{th} column gives the number of i^{th} action labels in the j^{th} NG Cluster. % is the fraction of vectors of an action correctly classified to the total vectors of that type. Total Classification Accuracy(TCA) is the % of total vectors correctly classified .

Action	C1	C2	C3	C4	Tot	%	TCA
CC	399	6	10	29	444	90	
MA	16	311	5	48	380	82	84
Chase	21	59	149	154	383	79	

neuron data. Cluster labels for the frames are obtained in the final iteration of the algorithm based on the winner neuron.

5 Concept Acquisition: Chase video

Unsupervised clustering using the Merge Neural Gas algorithm is used on the feature vectors from the video, corresponding to object pairs that were in attentive focus around the same time. Salient objects in a scene are ordered by a computational model of bottom-up dynamic attention[Singh *et al.*, 2006]. The most salient object is determined for each frame, and other objects that were salient within k frames before and after (we use $k = 10$) are considered as attended simultaneously. Dyadic feature vectors are computed for all object pairs in these $2k$ frames.

Owing to the randomized nature of the algorithm, the number of clusters varies from run to run. Clusters with less than ten frames are dropped. With the *aging* parameter set to 30, the number of clusters came out to be four in 90% of the runs; the set of four clusters with highest total classification accuracy (refer Table 2) are considered below.

In order to validate these clusters with human concepts, we asked three subjects (Male, Hindi-English/ Telugu-English bilinguals, Age-22, 20 and 30) to label the scenes in the video. They were shown the video twice and in the third viewing they were asked to speak out one of three action labels (CC, MA, Chase) which was recorded. Given the label and the frame when this was uttered, the actual event boundaries and participating objects for the groundtruth data were assigned by inspection. In case of disagreement, we took the majority view.

The percentage accuracies shown in table 2 do not reflect the degree of match, since although an event may last over 15 frames, even if 10 frames have been detected, it is usually quite helpful. This can be seen in 6 which present results along a time line for *Chase*; each row reflects a different combination of agents (small square, big square, circle). At first glance, figures like 6 would seem to reflect a higher accuracy than 84% in table 2.

A surprising result was found when by experimenting with the *edge aging* parameter in the Merge Neural Gas

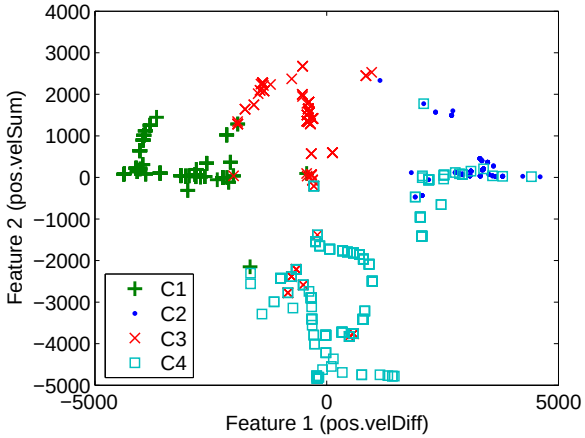

Figure 5: *Feature Vectors of the Four Clusters from the MNG Algorithm*: CC - C_1, MA - C_2, Chase(Reference Object is the Chaser) - C_3, Chase(Reference Object is the Leader) - C_4; The clusters reflect the spatio-temporal proximity of the vectors.

Figure 6: Comparison of Human and Algorithm Labelling of "chase" over first 1500 frames. Because of our choice of reference object, frames in first row are in C_4 and second row are in C_3.

Table 3: *Hierarchical clustering*: Using a larger number of clusters reveals a sub-classification; e.g. frames classified as CC in Table 2, are now in $C_1, C_5, or C_6$, reflecting two cases of $CC_{one-object-static}$, or one case of $CC_{both-moving}$.

	C1	C2	C3	C4	C5	C6	C7	C8
CC	201	3	9	20	189	21	1	0
MA	8	126	4	45	9	1	181	6
Chase	1	9	142	151	13	9	32	26

Table 4: *Relevance of Argument Order*: Value at i^{th} row, j^{th} column gives number of vectors that were originally in Cluster i and now assigned to Cluster j when object order was switched in dyadic feature vectors. Note that C3 and C4, the clusters corresponding to *Chase*, are flipped.

	C1	C2	C4	C3
Cluster 1	390	20	11	15
Cluster 2	9	323	15	29
Cluster 3	6	12	1	145
Cluster 4	22	48	152	9

algorithm. The number of clusters increase as aging parameter is decreased, and at one stage eight clusters were formed (edge aging parameter=16). The Total Classification Accuracy (TCA) was about 51 and we would have discarded the result, but inspecting the frames revealed that the clusters may be reflecting what appeared to be hierarchy of action types. Thus cluster C_1 from the earlier classification (majority correlation=CC) was broken up into C_1, C_5, C_6. C_1 was found to contain frames where both objects are moving towards each other whereas C_5 contains frames where the smaller object is stationary and the other moves closer. Thus *Come-Closer* and *Move-Away* appear to be sub-classified into 3 classes (two *one object static* cases, and one *both moving* case). This 'finer' classification is given in Table 3.

5.1 Argument order in Action Schemas

In another experiment, we investigated the importance of argument ordering by re-classifying the same frames, but reversing the order of the objects used in the dyadic vector computation. Earlier, if the larger object was $arg1$ or reference object, now it became $arg2$ or non-reference object. If the corresponding concept changed, especially if it flipped, this would reflect a semantic necessity to preserve the argument order; otherwise the arguments were commutative. Using the coarser clusters, we observe that the argument order is immaterial since the majority relation is unchanged (black) for C1 and C2 (CC,MA respectively). On the other hand, both C3 and C4 (correlations with Chase) are flipped (Table 4). Thus, the fact that argument order is important for *Chase* is learned implicitly within the action schema itself. The non-commutativity of $CC_{one-object-static}$ and $MA_{one-object-static}$ could not be established because of the skewed distribution of frames in the input video amongst the two sub-classes for the action verbs.

5.2 Comparison with K-Windows Clustering

We compare the clustering accuracy obtained by the unsupervised Merge Neural Gas algorithm with K-windows algorithm [Vrahatis *et al.*, 2002]. K-Windows is an improvement of K-Means clustering algorithm with a better time complexity and clustering accuracy. We set the value of k in this algorithm to 4 and run it on the input feature vectors obtained after attentive pruning. The initial cluster points for the algorithm are set randomly. Table 5 gives the clustering results obtained.

Table 5: *Clustering Accuracy by K-Windows*: The value of *'k'* is set to 4. The i^{th} row, j^{th} column gives the number of i^{th} action labels in j^{th} Cluster. % is the fraction of vectors of an action correctly classified to the total vectors of that type. Total Classification Accuracy(TCA) is the % of total vectors correctly classified .

Action	C1	C2	C3	C4	Total	%	TCA
CC	277	19	51	97	444	62	
MA	29	234	61	56	380	61	59
Chase	95	83	91	114	383	54	

The lower accuracy (as compared to results in Table 2) is expected because K-windows treats each feature vector as a separate entity without utilizing the information present in the temporal ordering of the frames.

6 Recognizing actions in 3D

In order to test the effectiveness of the clusters learned, we test the recognition of motions from a 3D video of three persons running around in a field (Fig.7). In human classification of the action categories (into one of *CC, MA, Chase*), the dominant predicate in the video, (777 out of 991 frames), is Chase.

In the image processing stage, the system learns the background over the initial frames based on which it segments out the foreground blobs. It is then able to track all the three agents using the Meanshift algorithm. Assuming camera height near eye level, the bottom-most point in each blob corresponds to that agent's contact with the ground, from which its depth can be determined within some scaling error (157 frames with extensive occlusion between agents were omitted). Given this depth, one can solve for the lateral position - thus, we are able to obtain, from a single view video, the (x, y) coordinates for each agent in each frame, within a constant scale. Based on this, the relative pose and motion parameters are computed for each agent pair, and therefrom the features as outlined earlier. Now these feature vectors are classified using the action schemas (coarse clusters) al-

Figure 7: *Test Video :* Scenes from the 3D video

Table 6: Distribution of Chase frames(ground truth) from the 3D video across the Neural gas clusters

	C1	C2	C3	C4	Chase total	%
Chase	13	15	496	253	777	96

ready obtained from the Chase video (2D) (Table 6).

7 Discussion and Conclusion

We have outlined how our unsupervised approach learns action schemas of two-agent interactions resulting in an action ontology. The image schematic nature of the clusters are validated by producing a description for a 3D video. The approach provided here underlines the role of concept argument structures in aligning with linguistic expressions, and that of bottom-up dynamic attention in pruning the visual input and in aligning linguistic focus.

Once a few *basic* concepts are learned, other concepts can be learned without direct grounding, by using conceptual blending mechanisms on the concept itself. These operations are often triggered by linguistic cues, resulting in new concepts, as well as their labels being learned together, in a later stage. Indeed, the vast majority of our vocabularies are learned later purely from the linguistic input [Bloom, 2000]. But this is only possible because of the grounded nature of the first few concepts, without which these later concepts cannot be grounded. Thus the perceptually grounded nature of the very first concepts are crucial to subsequent compositions.

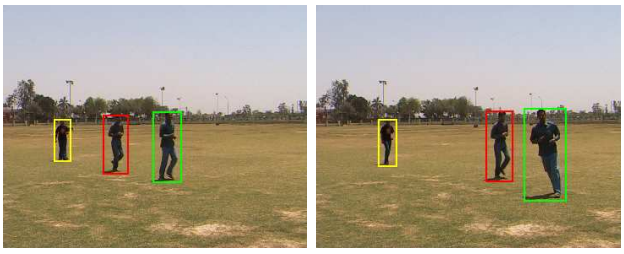

Figure 8: *Image Schemas identified for actions*: "Red *Chase* Green", "*Move Away*(Red,Yellow)", "*Move Away*(Green,Yellow)"

References

[Ballard and Yu, 2003] Dana H. Ballard and Chen Yu. A multimodal learning interface for word acquisition. In *International Conference on Acoustics,Speech and Signal Processing(ICASSP03)*, volume 5, pages 784–7, April 2003.

[Bloom, 2000] Paul Bloom. *How Children Learn the Meanings of Words*. MIT Press, Cambridge, MA, 2000.

[Ester *et al.*, 1996] Martin Ester, Hans-Peter Kriegel, Jorg Sander, and Xiaowei Xug. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining*, 1996.

[Fleischman and Roy, 2005] Michael Fleischman and Deb Roy. Why verbs are harder to learn than nouns: Initial insights from a computational model of intention recognition in situated word learning. In *Proceedings of the 27th Annual Meeting of the Cognitive Science Society*, 2005.

[Forbus *et al.*, 1987] Kenneth D Forbus, Paul Nielsen, and Boi Faltings. Qualitative kinematics: A framework. In *IJCAI*, pages 430–436, 1987.

[Guha and Mukerjee, 2007] Prithwijit Guha and Amitabha Mukerjee. Language label learning for visual concepts discovered from video sequences. In Lucas Paletta, editor, *Attention in Cognitive Systems. Theories and Systems from an Interdisciplinary Viewpoint*, volume 4840, pages 91–105. Springer LNCS, Berlin / Heidelberg, 2007.

[Itti, 2000] L. Itti. *Models of Bottom-Up and Top-Down Visual Attention*. PhD thesis, Pasadena, California, Jan 2000.

[Langacker, 1999] Ronald Wayne Langacker. *Grammar and Conceptualization*. Berlin/New York: Mouton de Gruyer, 1999.

[Mandler, 2004] J M Mandler. *Foundations of Mind*. Oxford University Press, 2004.

[Martinetz and Schulten, 1994] T. Martinetz and K. Schulten. Topology representing networks. *Neural Networks*, 7(3):507–522, 1994.

[Regier, 2003] Terry Regier. Emergent constraints on word-learning: A computational review. *Trends in Cognitive Sciences*, 7:263–268, 2003.

[Roy and Mukherjee, 2005] Deb Roy and Niloy Mukherjee. Towards situated speech understanding: visual context priming of language models. *Computer Speech and Language*, 19(2):227–248, 2005.

[Satish and Mukerjee, 2008] G. Satish and A. Mukerjee. Acquiring linguistic argument structure from multimodal input using attentive focus. In *7th IEEE International Conference on Development and Learning, 2008. ICDL 2008*, pages 43–48, 2008.

[Singh *et al.*, 2006] Vivek Kumar Singh, Subranshu Maji, and Amitabha Mukerjee. Confidence based updation of motion conspicuity in dynamic scenes. In *Third Canadian Conference on Computer and Robot Vision*, 2006.

[Strickert and Hammer, 2005] Marc Strickert and Barbara Hammer. Merge som for temporal data. *Neurocomputing*, 64:39–71, 2005.

[Sugiura and Iwahashi, 2007] Komei Sugiura and Naoto Iwahashi. Learning object-manipulation verbs for human-robot communication. In *WMISI '07: Proceedings of the*

2007 workshop on Multimodal interfaces in semantic interaction, pages 32–38, New York, NY, USA, 2007. ACM.

[Vrahatis *et al.*, 2002] Michael N Vrahatis, Basilis Boutsinas, Panagiotis Alevizos, and Georgios Pavlides. The new k-windows algorithm for improving thek -means clustering algorithm. *Journal of Complexity*, 18(1):375–391, March 2002.

[Xiang and Gong, 2006] T. Xiang and S. Gong. Beyond tracking: Modelling activity and understanding behaviour. *International Journal of Computer Vision*, 67(1):21–51, 2006.