# Grounding via Scanning: Cooking up Roles from Scratch

Douglas Blank and Michael Gasser
Department of Computer Science
Indiana University
Bloomington, Indiana 47405

March 11, 1992

## 1 Introduction

At the heart of nearly every model in cognitive science and artificial intelligence is the notion of roles and their fillers. Roles appear as explicit slots in frame systems (Minsky, 1975), as particular positions in predicate calculus representations, as role-specific groups in some connectionist models (McClelland & Kawamoto, 1986), and as distributed patterns in role space in other connectionist models (Smolensky, 1990).

But where do roles come from? In most systems, this is not treated as an issue; objects are simply assigned fully formed structured representations.[1] But for some problems the decomposition of an object into a set of roles and the assigning of fillers to these roles appears to be half of the battle. For instance, systems attempting to make analogies between two domains are already half finished before they start if they are given sets of roles and associated fillers for the items being compared (Holyoak & Thagard, 1989).

We believe that problems such as analogy can be addressed adequately only when the structure of representations is "grounded" in perception and action (Harnad, 1991). Recent efforts to ground semantics in visual input (Bartell & Cottrell, 1991; Regier, 1990) have focused on what distinguishes predicates such as FAST from SLOW or ABOVE from BELOW. These approaches have not looked at questions of how objects are assigned to different roles or how a system would cope with predicates taking different numbers of arguments.

Consider a system which is given a visual display containing a number of objects and the task of describing what it sees. It must do several things. It must find one or more objects, assign labels of some sort to them, recognize that a relation exists between two or more of the objects (if it does), assign a label to the relation, and assign the objects to the roles of the relation. The assignment of objects to roles depends on the relative *salience* of the objects; in each relation the most salient object becomes the *trajector* (Langacker, 1987), realized linguistically as the subject; the other objects are *landmarks*, realized as the direct object or some other argument. This view recognizes the non-equivalence of pairs such as *the square is above the circle* and *the circle is below the square*; they differ with respect to the perceived relative salience of the two objects.

One possibility for how this might be done is as follows. The system scans the visual input for objects, identifying them as it locates them. It keeps track of the sequence of scanning actions it makes, and when two or more objects are perceived in succession, the memory for the recent

[1] Though there may be roles that are not yet filled, the particular role decomposition itself is normally given.

scanning actions as each object is recognized represents the role that object plays in the relation among the set of objects. There is also a short-term memory for the objects themselves so that they can be compared for salience. The key idea, then, is that roles (and relations) are grounded in the basic scanning operations of the visual system.

We have implemented a preliminary version of the system just described.

## 2 Task

In these initial experiments, we have limited the simulated spatial environment to a one-dimensional world. That is, objects in the environment sit next to each other and are not seen above or below other objects. Therefore, we are only considering relationships such as *between* and *to_the_right_of*. The objects remain stationary as the system views them, one at a time, in a left-to-right or right-to-left scanning motion.

The system is a set of three connectionist networks (discussed in the following section). The only input to the networks coming directly from the environment is a simple simulated image presented to a "retina". During the simulation, time is divided into discrete steps. At each time step, one image of an object is placed on the retina. There are also two other inputs given to the network per time step which do not come from the environment. One is the direction that the retinal eye has just scanned to get to its current position. The second additional input is a value indicating the relative salience of the current object being viewed (as determined, possibly, by a module which compares the objects as it sees them).

In this implementation each object is assigned random activations on the ten units which make up the retina. Since we have constrained this model to only left and right scans, the directions of the eye movements are given by localist representations for *left* and *right* on two units. The salience, or attention, gauge is given as a real-valued activation between 0 and 1.

For example, consider the scene in Figure 1. Imagine that the square has the highest relative salience, and is therefore considered to be the trajector of the scene. Scanning from left to right will give the square as input first, followed by the triangle, and finally the circle. As the square is presented, the system should respond with the description of the scene: *a square exists*. As the triangle is viewed, the system would respond with *the square is to the left of the triangle*. Note that this would be reversed if the triangle were the more salient. And finally, as the circle is presented, the system would respond with *the square is to the left of the triangle and circle*. If the same scene from Figure 1 were to be seen by scanning from right to left, the responses would be: *a circle exists*, *the circle is to the right of the triangle*, and *the square is to the left of the circle and triangle* respectfully. Since the system has no indication of the relative salience of the circle and triangle, it simply places the first object encountered in the subject slot.

The experiments described here test up to three-place relations (e.g., *the triangle is between the square and the circle*), but relations with more arguments are also theoretically possible.

## 3 Network Architecture

The full system consists of the three relatively simple networks shown in Figure 2. Network #1 encodes the scanning motion of the eye. Network #2 associates the sequence of scans from Network #1 with the current image on its "retina." Network #3 takes sequences of images encoded with scans from Network #2 and produces a localist description of what it has seen so far.

Network #1 is a small, sequential recursive auto-associative memory (RAAM) (Pollack, 1988), that is, a network which takes items as input, one a time, and following each forward pass, copies

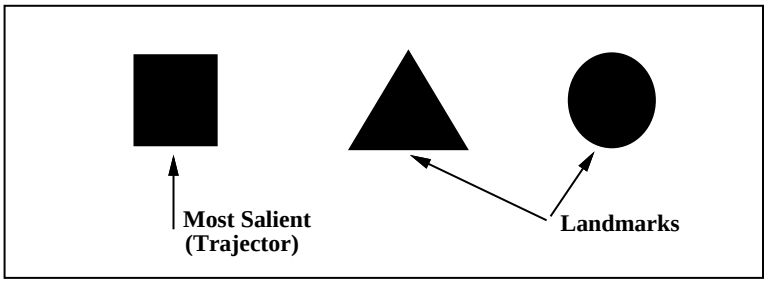


Figure 1: An example sequence. This is viewed either from left to right or from right to left, one object at a time. The square is (arbitrarily) considered to be the trajector.

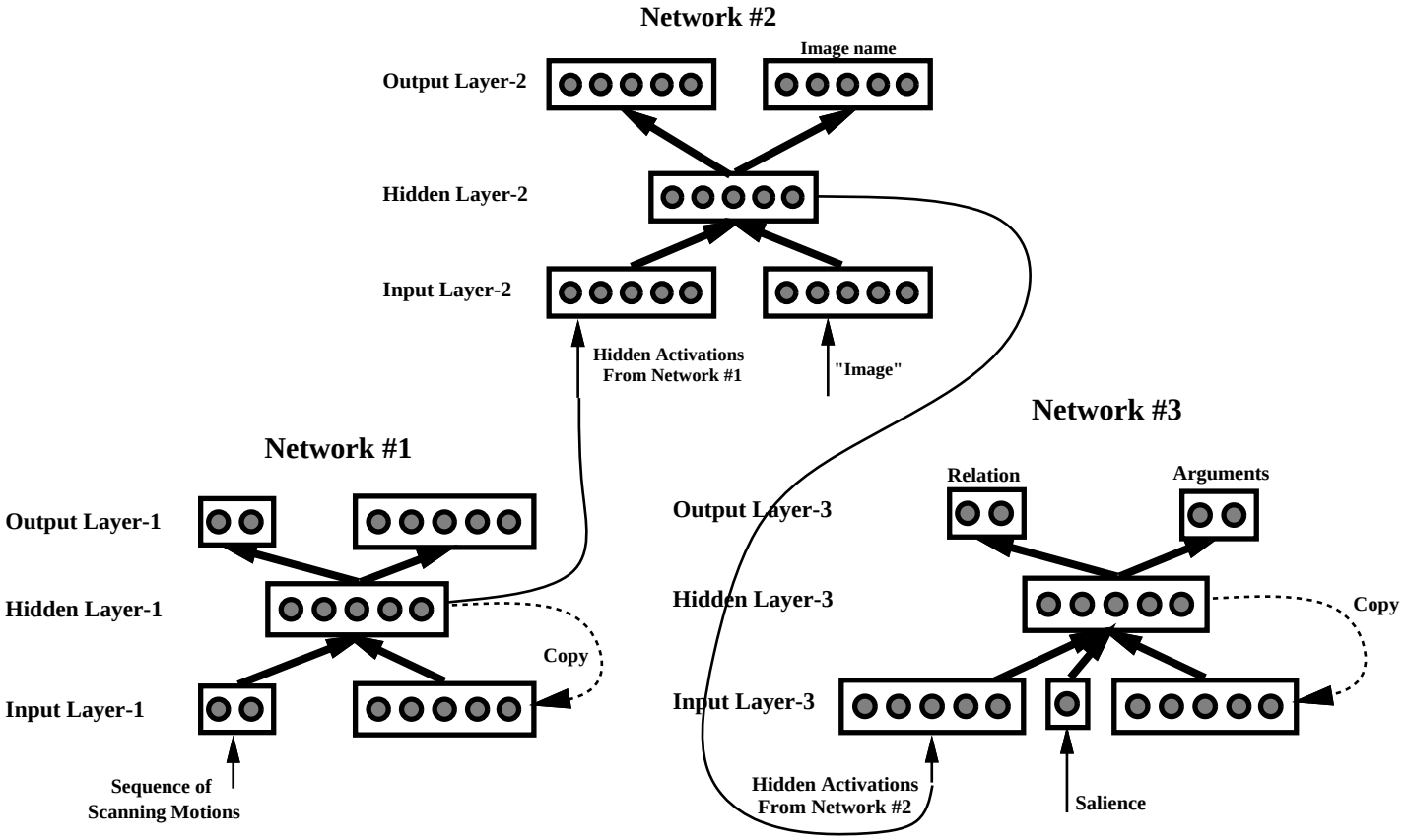


Figure 2: Network architecture. The thick, solid arrows indicate connections that are trained using back-propagation. The dotted arrows represent activations copied during training. The thin, sollid lines indicate activations copied after training of the source network was complete.

the hidden layer pattern onto a "stack" input layer. It is trained simply to reproduce both input layers on its output layers. Network #1 takes *left* and *right* scanning motions sequentially as input, encoding the sequence into a compact, distributed pattern of activation as it goes. The main purpose of this network is to provide an abstraction of the scan movements that have led to the current position of the retina.

Network #2, a simple feedforward network, has two functions. It is trained to encode objects and their relative positions and to identify objects. This net is given the hidden layer from Network #1 (the encoded scan movements) and the activations of the retina. The target for a given image is a localist representation which stands for that object. Thus this network is learning labels for each known object.

Network #3 is a simple recurrent network (Elman, 1990), that is, a sequential network which takes a copy of its previous hidden layer as input on each time step. Its input also includes the hidden layer from Network #2 and the relative salience rating for each image in a scene. It produces as output the relation seen so far and each object involved in the relation. The objects are produced in order of their salience on the argument output units.

We trained the system on ten different input objects and four relations, *exists*, *left*, *right*, and *between*.

## 4 Results

Each network was trained individually, beginning with Network #1. All three were trained on a subset of possible inputs and then tested for generalization on a portion of the those remaining. After Network #1 was trained, the hidden unit activations were used as inputs to Network #2. Not surprisingly, considering the tasks they had, Networks #1 and #2 performed nearly perfectly. Following Network #2's training and testing, its hidden layer activations were used as inputs to Network #3 as described above.

The final network (#3) was trained on about half of all possible combinations of salience, object order, and scanning direction (2000 of 4320 possible patterns.) After training for 180 epochs, Network #3 was tested to see how well it generalized on the task of accurately describing a novel scene. The scene could be novel in one of three ways: a variation in the object sequence (e.g., it had never seen a circle between a triangle and a square before), salience (i.e., it had never attended to a scene exactly in this way before), or scanning direction (i.e., it had never seen a scene scanned from this direction).

Presented with 100 novel scenes, it classified the relation and the landmark (subject) of the scene accurately in every case. However, it had problems correctly identifying the other objects in a scene. About 60% of the time, it gave an incorrect response for at least one of the other two objects. We believe this can be remedied by training the network on a greater variety of objects and relations.

## 5 Conclusions and Future work

The simulations described here are only a very tentative first step towards a model of the grounding of roles. We need to test the model on larger problems, including in particular two-dimensional ones, and to include other sorts of spatial relations such as *near/far* and motion relations. Also, it is clear that the idea of grounding roles in terms of scanning primitives can only go so far. We believe, however, that it may be possible to view many perceptual roles and relations which are

not spatial in terms of other simple operations of the system, for example, selectively attending to particular dimensions, zooming in, and comparing.

## References


Bartell, B. & Cottrell, G. W. (1991). A model of symbol grounding in a temporal environment. In *Working Notes: Connectionism and Natural Language Processing*, pp. 142–147.

Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, *14*, 179–212.

Harnad, S. (1991). The symbol grounding problem. In Forrest, S. (Ed.), *Emergent Computation: Self-organizing, collective, and cooperative phenomena in natural and artificial computing networks*, pp. 335–346. MIT Press, Cambridge, MA. Originally appeared in *Physica D* (1990), 42:335-346.

Holyoak, K. J. & Thagard, P. (1989). Analogical mapping by constraint satisfaction. *Cognitive Science*, *13*, 295–355.

Langacker, R. (1987). *Foundations of Cognitive Grammar*. Stanford University Press, Stanford.

McClelland, J. & Kawamoto, A. (1986). Mechanisms of sentence prcoessing: assigning roles to constituents. In Rumelhart, D. E. & McClelland, J. L. (Eds.), *Parallel Distributed Processing*, Vol. 2, pp. 272–325. MIT Press, Cambridge, MA.

Minsky, M. (1975). A framework for representing knowledge. In Winston, P. (Ed.), *The Psychology of Computer Vision*. McGraw-Hill, New York, NY.

Pollack, J. B. (1988). Recursive auto-associative memory: devising compositional distributed representations. In *Proceedings of the Tenth Annual Conference of the Cognitive Science Society*, pp. 33–39. Lawrence Erlbaum Associates.

Regier, T. (1990). Learning spatial terms without explicit negative evidence. Tech. rep. 57, International Computer Science Institute, Berkeley, California.

Smolensky, P. (1990). Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artificial Intelligence*, *46*, 159–216.