A classifier system for learning spatial representations, based on a morphological wave propagation algorithm.

Michael M. Skolnick
Department of Computer Science
R.P.I.

Abstract

A system is proposed which learns spatial representations of planar feature point sets under supervised learning. A key (preprocessing) aspect of the learning system is the transformation of the each "static" point set instance into a "dynamic" set of measures of spatial relationships spread out over time. A morphologically based wave propagation algorithm [1, 2, 3] performs this transformation of spatial structure into temporal structure. The learning system [4] is based upon classifiers using bucket brigade and genetic algorithms [5] to respectively modify strengths and create new classifier rules. Such learning systems are designed to exploit temporal regularities in learning environments and, thus, fit well with the wave propagation preprocessing. An initial test environment is proposed that attempts to re-label arbitrary feature labels into structurally meaningful labels.

1. Introduction

In the field of computer vision it is the general practice to decompose visual objects into features and the spatial relations between features. The advantages and justifications for this practice are compelling. In computational terms, there is a natural mapping of features onto graph nodes and spatial relations onto graph arcs. In terms of natural systems, it is clear that a similar decomposition is performed by the primate visual system (with the locational system being associated with the more primitive pathways through the superior colliculus onto the parietal cortex and the feature system being associated with the primary and secondary projection areas of the striate cortex). Unfortunately, there are fundamental issues that prevent this decomposition from giving a fully satisfactory account.

In artificial vision systems it is clear that there remains a combinatorial explosion in matching object graph models to full blown scenes. Objects appear in highly contextual scenes, with occlusion, distortion and large amounts of pose variation, resulting in an explosion in the complexity of the graph models and in the complexity of the search through such models. In natural vision systems, there has been no systematic account of how the lower levels of feature analysis and feature location are combined in the distributed representations that must underlie perception. The only theoretical account that comes close is that of the Hebbian cell assembly and it falls short due to our ignorance of the mechanisms underlying self-reverberating distributed neural networks.

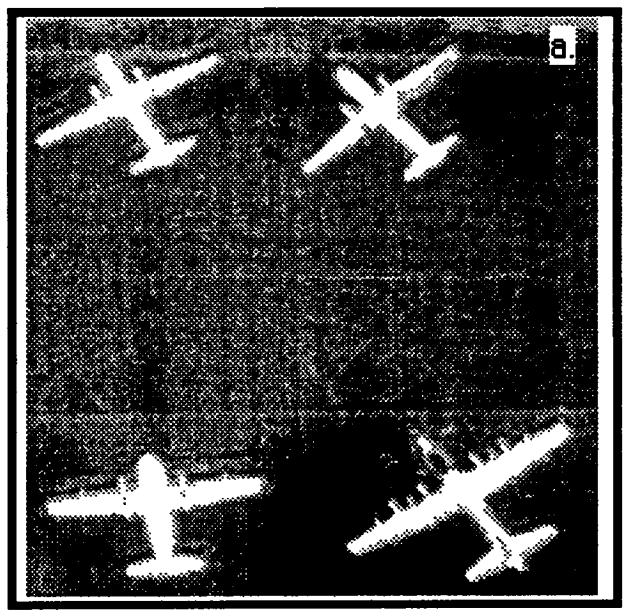

Figure 1. Image of planes on tarmac. The goal of system is to classify planes according to shape.

Recent advances in various kinds of learning systems hold out some hope in dealing with these problems. In contrast to model based machine vision systems, where all the complex non-linear contextual relations must be mapped out a priori, learning algorithms suggest ways of evolving such a knowledge base. Further, most learning systems are based upon massively parallel systems of primitives that are involved in some "tacit" form of competition and/or cooperation. This provides the kind of search performance (after learning has occurred) that is required. On the other hand, our understanding of what happens within these systems seems to be circumscribed in fundamental ways, e.g., via "hidden nodes" in neural networks and "implicit parallelism" in genetic algorithms. In the context of these general issues, I have been considering the form of a learning system that evolves representations of spatial information.

The two major issues in the design of such a system involve the choice of preprocessed spatial information to be given to the learning system and the type of learning system to receive this information. The learning task will be that of supervised pattern classification; the learning system attempts to predict as to class and is punished or rewarded if the classification is wrong or right. Section 2 describes the morphological preprocessing which maps the input spatial data into the temporal data exploited by the learning system.

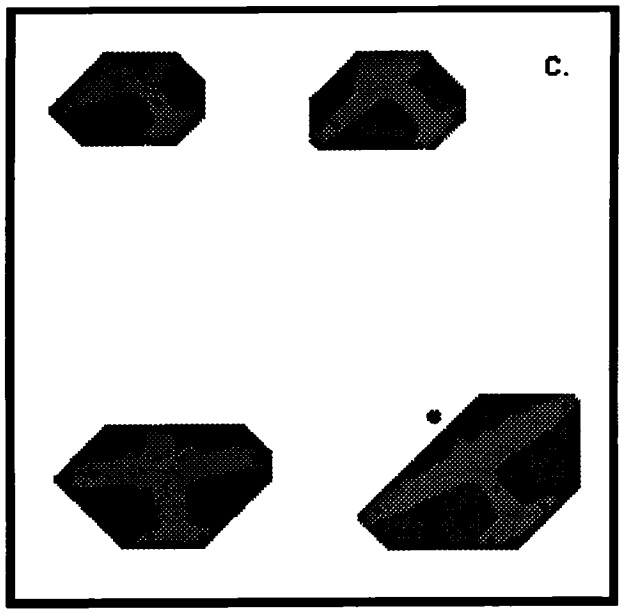

Figure 2. Thresholded silhouettes of planes superimposed on pseudo-convex hulls (which function as local "puddles" for wave propagations). Locations of features of high curvature are marked in black and will function as the "pebble" sources for the wave propagation.

Section 3 gives an intuitive description of the proposed learning system ([4] provides a more detailed description).

2. Pebble_Pond Wave Propagation

In terms of preprocessing, assume that each image is processed so that the locations of key features are detected. For example, consider the image of airplanes in Figure 1. Now consider Figure 2. After thresholding the grey-scale data, morphological openings and closings are applied to detect silhouette features of high internal and external curvature respectively. In this case, the input to the learning system preprocessing is the set of all points of "high" curvature detected in the binary image, e.g., on a plane the nose, tail and wing tips (for internal curvature) and points where wing and fuselage meet (for external curvature). In addition, pseudo convex hulls are computed to provide a preliminary segmentation and, in turn, a local region within which to perform wave propagation.

The morphological preprocessing algorithm, Pebble_Pond [1, 2, 3], sends out wavefronts from each of the feature locations. One visualization of the this process in shown in Figure 3. The wavefronts contain information as to the identity of the feature source, where the identity is determined by a line scan labeling of the features. The basic intuition of Pebble_Pond is that if a general wave propagation process can be sustained morphologically, then the resulting cellular state space should provide a rich set of spatial measures. Morphological filters have been applied to the Pebble_Pond state space to extract specific spatial measures from a diverse group of such measures, e.g., all k nearest neighbors, k-th

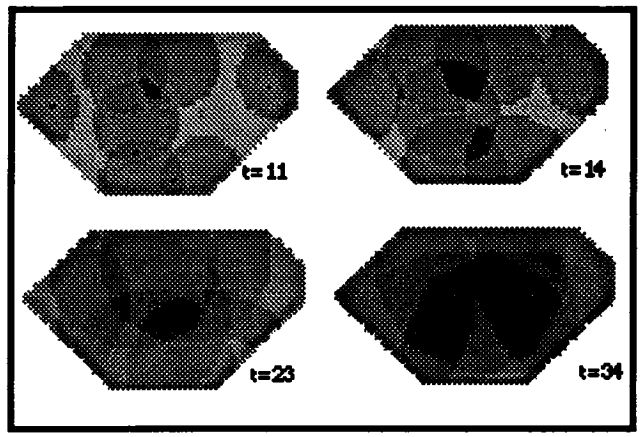

Figure 3. Illustration of wave propagation process emanating out of the Pebble_Puddle shown in lower left of Figure 2. The measures on the wave state space in this case involve the cardinality of the intersections of the waves. The black pixels mark detected local maxima over the intersection grey-scale surface. The maxima turn out to be useful [2,3] in finding scale and rotation invariant shape signatures. The initial focus of the learning system will be on another set of measures involving wave front meetings.

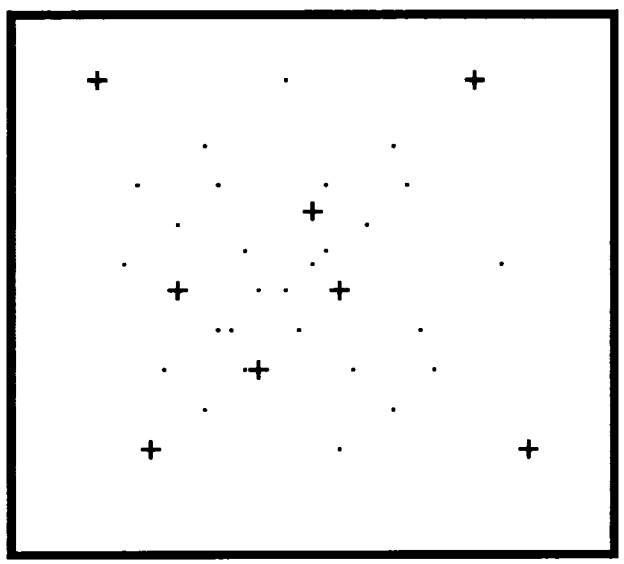

Figure 4. The midpoints of edges of the complete graph of 7 "pebbles" (derived from detected wave front meetings), which will be input to the learning system in the order of their detection

order Voronoi tessellations, and k-th order Gabriel graphs, morphological co-variance and various rotation/scale invariant spatial signatures. All of these measures are based upon the detection of more "primitive" events in the wave state space: wave front meetings, wave front crossings and wave state intersections. The reader is referred to previous work [1, 2, 3] for details on the underlying wave state space simulation (especially basic time/space trade-offs) and the suite of morphological filters required to extract the primitive events, from which are derived the diverse class of spatial measures.

This paper will focus on the exploitation of wave front meetings as the basic input to the classifier learning system. The set of wave front meetings is used to detect all midpoints of the complete graph that connects the features (see Figure 4). We believe that this restricted class of events in Pebble_Pond state space provides a sufficiently robust starting point for the exploration of learning algorithms for spatial structure. The crucial idea behind Pebble_Pond is that it takes spatial structure and transforms it into temporal structure. That is, at each iteration in the wave propagation, measures on the state space reflect spatial structure at the spatial scale corresponding to the current iteration. In terms of the midpoints of the complete feature graph, Pebble_Pond will output the midpoint locations (along with the point source identities) in order according to the length of the edge. The task of the learning system is to exploit this temporal translation of the spatial structure by learning to predict over time certain regularities in the "messages" produced by Pebble_Pond.

3. Classifier Learning system

Classifier systems [5] are especially useful in environments where pattern recognition actions at given time steps need to be linked with related pattern recognition actions occurring at latter time steps. Pebble_Pond produces early sets of messages that report on spatial structure at near spatial scales, while latter sets of messages report on more distant spatial scale. This separation of spatial scales over time provides robustness in the face of spurious or missing data, i.e., with respect to the partial matching problem. In other words, spurious data has a localized affect at early states in wave space. Further, the wave propagation throughout the space allows all points to interact with all other points and - assuming the learning system can search out such invariant interactions - there is added resistance to spurious and missing data. Classifiers can look to sensory input messages coming out of Pebble_Pond and then place internal messages that function as predictions that certain other messages should arrive at latter time steps. Eventually, classifiers should develop which link these internal and sensory input messages together into predictions as to pattern class. The predictions are then subject to reward or punishment, with the learning system using this information to change its internal structure.

The input to the classifier system at each time step is the set of midpoints currently detected and the identifying labels of the meeting wavefronts. The classifier system attempts to predict when and where these same wavefronts will participate in meetings with other wavefronts. One can view the learning system's linked predictions as essentially performing a geometrically meaningful rebelling of the initial arbitrary labels assigned to the feature locations. This brings up a basic problem in computer vision: internal models being matched to data have features identified in some semantically meaningful way (e.g., this feature is part of the arm, etc.) but - because input data is "raw" - all initial labels given to input data will be arbitrary. Thus, we need the labels to keep track of the input features, but such labels must be remapped into something meaningful. The proposed learning system focuses on this problem and provides mechanisms which take the arbitrary labels and attempts to have "emerge" a meaningful remapping.

Given the restriction on the size of this paper, we will not summarize the mechanisms of classifier systems (e.g., bucket-brigade algorithm) nor describe the detailed formal specifications of our particular classifier system and Pebble_Pond. A full explanation can be found in [4]. The remainder will attempt to provide an intuitive feel for the system.

There are two basic issues that must be confronted in such learning systems: 1. how can new classifiers be formed to improve performance, and 2. how can classifiers be made to cooperate with each other so that extended sequences of messages will be passed between "associated" classifiers. The genetic algorithm is typically used in classifier systems to generate better classifiers out of the current "population" of classifiers. We will not consider the genetic algorithm in this paper since the proposed test system can be designed [4] as to remove the need for the "discovery" of new classifiers. It is noted simply that each classifiers strength is a measure of the "fitness" of the classifier and, under the genetic algorithm, the most fit classifiers are allowed to "mate" and "reproduce" in such a way - via operators such as mutation and crossover - that new "generations" of classifiers tend to be improved. Thus, the current discussion will focus on how classifiers might begin to link together into cooperating/competing assemblies.

The classifiers and messages are separated into <u>three</u> disjoint sets: 1. Sensory classifiers will take the information in the sensory messages generated by Pebble_Pond and post internal messages that will function as predictions about future sensory messages. 2. The (sensory, internal) classifiers will attempt to verify these predictions with the current sensory input. When these classifiers fire they will post new internal messages, which now represent predictions about future sensory input based upon some integration of information from previous internal messages and the current sensory input. The internal messages are intended to represent "linkages" between related midpoint events from Pebble_Pond, i.e., they are intended to search out invariant "paths" through the midpoints of the complete graph that are useful in final classifications. 3. The final set of (internal, internal) classifiers are intended to combine internal classifiers into a categorization or classification "action," with the result that the system gets payoff, which is then intended to trickle back via the bucket-brigade algorithm.

Thus, as sensory messages from Pebble_Pond come into the system, they are either treated as "new" events by sensory classifiers or are integrated with previous sensory events via previously posted internal messages. The internal messages are intended to represent stable spatial structures that might

be directly relevant to the final categorization of a specific class of inputs or might be shared by different classes (e.g., two different classes might share a rectangular substructure). As internal messages are successively generated from a sequence of previous internal and sensory messages, successively larger spatial structures are constructed. The (internal, internal) messages attempt to combine these spatial structures (hopefully at the right time!) into categorizations.

Consider the configuration of 5 points shown in Figure 5, which represent a prototypical pattern that might be learned. The five point sources are indicated by crosses and labeled i - m. The small diamonds represent some edge midpoint events (messages) generated by Pebble_Pond. Since Pebble_Pond will generate messages in spatial scale order (small to large), the two midpoint messages, m(j,k) and m(i,k), will be generated before other midpoint messages. These messages might be picked up by sensory classifiers and incorporated into a prediction about future midpoint messages. For example, if m(i,k) is matched by a sensory classifier, a prediction might be made that at some future point in time another match will occur involving a midpoint event between either point i or k and some other point; the relationship between the events in the prediction is given by the vector drawn between m(j,k) and m(m,l), where the length of the vector corresponds to the predicted delay in time between m(j,k) and m(m,l) and the angle prediction of θ_2 will be part of the internal message. (Note that there is not complete consistency in the origin used to define the angles, so as not to clutter up Figure 5 any more than necessary.) The information on the sequence of vectors - time delays and angles - is passed on from one active classifier to the next. It is in this sense that the arbitrary (line-scan order) labels of the features are associated or remapped onto semantically meaningful labels. In other words, learning will result in sequences of cooperating classifiers which represent structurally meaningful information in terms of the sequence of vectors. As data comes in from Pebble_Pond, the arbitrary feature labels become "bound" to the vector "slots" that have been built into the classifiers via learning.

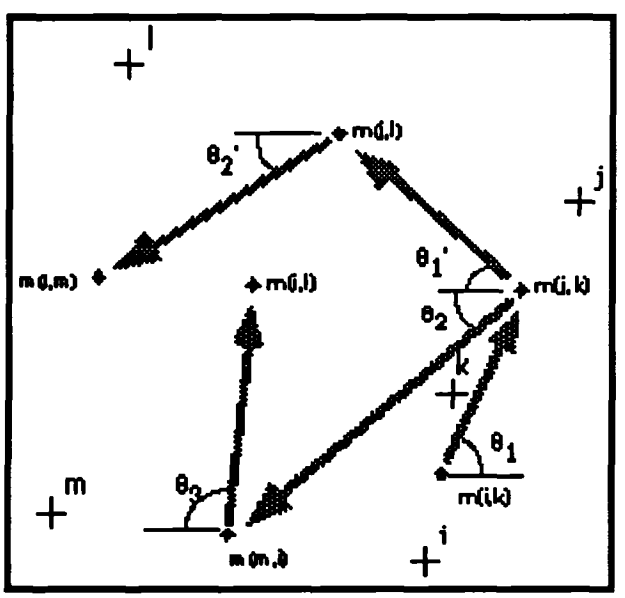

Figure 5. Two possible paths that might be learned in a prototypical pattern of 5 points.

Two possible paths that might be learned in a prototypical pattern of 5 points are shown in Figure 5. The first path starts at the Pebble_Pond event corresponding to the meeting midpoint of waves from point i and k, while the second starts at the midpoint event from points j and k. The classifier system needs to learn to predict the orientation and time delay (distance) to the next meeting event. Because all events produced by Pebble_Pond are ordered over time according to spatial scale, predictions can only be made to next midpoint meetings that are greater than the time scale of the current prediction point in the path being learned. Note that paths can share meeting midpoints or sub-paths. If either of these paths are relevant to a classification then they can be used by (internal, internal) classifiers to participate in external payoff.

The basic idea is that a sequence of activated classifiers will correspond to a path through the set of edge midpoints. The paths will be constrained to involve midpoints at increasing spatial scales, with multiple paths potentially being relevant to a pattern classification. Presumably, at some point there will be a pair of messages (representing some paths) that capture some structure that is relevant to a classification action and, thus, some (internal, internal) classifier might hazard a guess. If it is correct it gets payoff, which via the bucket-brigade should trickle back to increase the strengths of the (sensory, internal) and sensory classifiers that set it up.

Note that (internal, internal) classifiers are restricted to just function as a way to hook internal messages to actions. They function as binary operators for combining pairs of paths into classification decisions. This leads to a concern about how the system can learn pattern classes that require more general n-ary path relations. In order to simplify the learning system - i.e., to avoid issues concerning the generation of "higher order" messages - the system will have different learning versus performance modes. In learning mode, any pairwise combination of internal messages that can get a vote out for a given categorization will be allowed to participate in the competition. This way salient pairwise combinations of paths can be strengthened. In order to allow more than pairwise relationships to have an effect, in performance mode the action of the entire system will be the majority vote of all classifiers making predictions. This approach is intended to be a restricted but still robust first cut at the problem.

Thus, sensory messages will be coming into the system constantly and sensory classifiers will attempt to start predicting their participation in a path composed of related sensory messages arriving at latter scales, with (sensory, internal) classifiers attempting to continue the predictive path. Finally, (internal, internal) classifiers will be attempting to combine pairs of paths into classification actions. During learning

phase, correct classification actions will get payoff; during performance phase, the majority vote will win.

4. CONCLUSION

The basic ideas behind a proposed system for learning spatial structure have been presented. For preprocessing, a wave propagation algorithm (Pebble_Pond) is used to transform the static spatial structure in planar point sets in a temporal sequence of measures, which can be thought of as messages emanating from the evolving wave state space. A classifier system might then be used to search for temporal regularities in the sequence of these measures, where the temporal regularities reflect regularities in spatial structure.

5. REFERENCES

1. Skolnick, M. M., Kim, S. and O'Bara, R., "Morphological algorithms for computing non-planar point neighborhoods on cellular automata," Second International Conference on Computer Vision, 1988, pop. 106-111.

2. Marineau, P. and Skolnick, M., "Pebble_Pond: a morphological algorithm for representing diverse spatial structure," Technical report 91-3, Dept. of Computer Science, R.P.I., Troy, N.Y., 1991 (to appear in Mathematical Morphology: Theory and Hardware, R. Haralick, Ed., Oxford U. Press, in press.)

3. Skolnick, M. and Marineau, P., "Pebble_Pond: a morphological wave propagation algorithm for representing spatial point patterns," SPIE San Diego Conference on Image Algebra and Mathematical Morphology, 1992.

4. Skolnick, M., "A classifier system for learning spatial representations, based on a Pebble_Pond morphological wave propagation algorithm," SPIE Conference on Visual Communications and Image Processing, 1992, SPIE Vol. 1818, p. 919-930.

5. Holland, J. H., Holyoak, K.J., Nisbet, R.E. and Thagard, P.R. Induction, Processes of Inference, Learning and Discovery, MIT Press, 1986.