Toward Learning Mixture-of-Parts Pictorial Structures

Robin Hess HESS@EECS.OREGONSTATE.EDU
Alan Fern AFERN@EECS.OREGONSTATE.EDU
School of Electrical Engineering and Computer Science, Oregon State University, Corvallis, OR 97331

Abstract

For many multi-part visual object classes, the set of parts can vary not only in location but also in type. For example, player formations in American football involve various subsets of player types, and the spatial constraints between players depend largely upon which subset of player types constitutes the formation. In this paper, we consider the problem of learning to jointly localize and classify the parts of such objects, driven by our application focus in the domain of American football. Standard models from computer vision and structured machine learning do not appear adequate for our problem class, and we have in turn developed the mixture-of-parts pictorial structure (MoPPS) model which allows for joint constraints on the types and locations of object parts. Here we review the MoPPS model and its application in the football domain, and we discuss opportunities for learning suggested by our experience, including opportunities for structure and parameter learning, speed-up learning, active learning, and transfer learning.

1. Introduction

The problem motivating the work in this paper is the recognition of player formations in American football. Specifically, given an image of the player formation from the beginning of a football play, we want to compute the specific players (e.g. left tight end, tailback, right flanker) comprising that formation as well as the locations of those players. An important aspect of this problem, and one that distinguishes it from other object recognition problems is the fact that different formations involve different subsets of players, and the spatial constraints between players depend largely on the particular subset of players in the formation, as is illustrated in Figure 1. In addition, the rules of football dictate certain hard constraints on formations that restrict the number of certain types of players in the formation as well as their spatial configurations. For instance, there may be only a single quarterback in a formation, and every formation must contain exactly eleven players—specifically, seven linemen and four backs. Because players' appearances are nearly identical in our imagery, these are uninformative from a classification standpoint. For this reason, the relative spatial locations of players are the most important piece of evidence available for formation recognition.

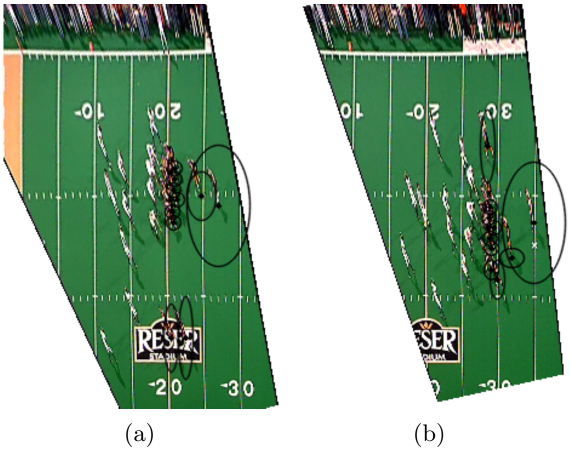


Figure 1. The configuration of players in an American football formation can vary drastically depending on the subset of players in the formation. Above are depicted, mapped to an overhead view, two very different formations containing different subsets of players. Player locations are marked along with confidence ellipses at two standard deviations based on distributions of the relative locations of players.

In general, pictorial structure models (Felzenszwalb & Huttenlocher, 2005; Crandall et al., 2006) provide a means by which to leverage the spatial configuration of a set of constituent object parts for object recognition and localization. Pictorial structures are

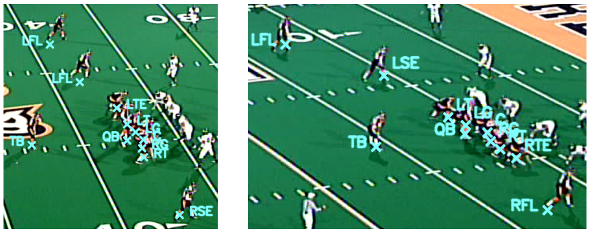
Appearing in the *ICML 2007 Workshop on Constrained Optimization and Structured Output Spaces*, Corvallis, OR, 2007. Copyright 2007 by the author(s)/owner(s).

graphical models which represent an object as a set of parts with local appearance models for each part and deformable connections between parts that describe their ideal relative locations. By jointly reasoning about part appearances and relative positions, pictorial structures can provide more robust inference than approaches that reason about object parts in isolation, as has been demonstrated for a number of multi-part object recognition problems (Crandall et al., 2006; Lan & Huttenlocher, 2005; Fergus et al., 2005). Moreover, for restricted—but useful—classes of pictorial structures, efficient algorithms for solving this joint inference problem have been developed that make recognition quite reasonable in practice (Felzenszwalb & Huttenlocher, 2005).

A fundamental assumption underlying pictorial structure models is that each object instance contains the same set of parts with the same set of deformation constraints between those parts. Unfortunately, when individual players are considered as the parts constituting an American football formation, this assumption does not hold. This factor, combined with the hard constraints on formations imposed by the rules of football makes it very difficult to formulate a single pictorial structure to recognize all possible football formations. Furthermore, because there are thousands of legal formations, formulating a pictorial structure model for each one is practically infeasible and would ignore the significant degree of common structure between similar formations.

We have also found it difficult to effectively formulate our problem using standard models from machine learning for structured outputs, e.g. low tree-width linear discriminant functions such as conditional random fields. Complicating features of our problem include: 1) the use of attribute-valued and numeric output variables to model player types and locations respectively, 2) the necessity to capture hard constraints on player types, and 3) the fact that the underlying graphical structure over location variables depends on the joint assignment to player type variables. One of the goals of this paper is to describe this problem to other machine-learning researchers in order to better assess the applicability of existing models.

The apparent difficulty of applying existing models led us to develop a generalization of the classical pictorial structures model which we call the mixture-of-parts pictorial structure (MoPPS) model. Intuitively, a MoPPS model can be viewed as an implicit representation of a very large collection of pictorial structures that captures the possible variations of objects whose constituent parts vary in both type and location. Under a generative view of this model, a subset of parts is first drawn from the corresponding prior distribution, with each subset defining a potentially different pictorial structure which can be used to generate positions and appearances for each of the parts. Inference on a MoPPS model corresponds to jointly computing the most likely, or least cost, subset of parts along with their positions.



(a) (b)

Figure 2. Some formations in American football differ only very subtly. The offensive formations (the orange and black players) in (a) and (b) are two such ones with inferred player positions overlaid on the images. These formations differ by three players, but the difference between the spatial configuration of the players in each formation is slight and may be difficult for an untrained human eye to detect. Still, as shown, we are able to correctly locate and classify all of the players in both formations using a MoPPS model.

Using a hand-encoded MoPPS model, we have been able to very successfully recognize offensive American football formations, despite the fact, illustrated in Figure 2, that the differences between formations are sometimes extremely subtle. Unfortunately, designing this model was a labor-intensive process for which a learning-based alternative would be desirable. Indeed, for classical pictorial structures, model learning procedures do already exist, as we discuss in Section 2.

In what follows, we first summarize the classical pictorial structure. Next we introduce the MoPPS model and overview its initial application in American football. Finally, we describe some of the prime opportunities for learning suggested by our initial experience.

2. Pictorial Structures

Under the classical pictorial structure model, a class of objects is represented as a graph with n vertices $V = \{v_1, \ldots, v_n\}$ representing the parts of the object and a set of edges $E = \{(v_i, v_j) \mid v_i \text{ and } v_j \text{ are connected}\}$ representing the connections between parts. Associated with each object class is also a set of model parameters Θ which includes part appearance parameters $A = \{a_1, \ldots, a_n\}$ and connection parameters $\Delta = \{\delta_{ij} \mid (v_i, v_j) \in E\}$ describing the ideal relative locations of connected parts. A particular instance of an object is represented as a set of locations of its parts $L = \{l_1, \ldots, l_n\}$.

Given an image I and a set of object model parameters Θ, the posterior distribution over the set of part locations is

$$p(L \mid I, \Theta) \;=\; \alpha \, p(I \mid L, \Theta) \; p(L \mid \Theta), \qquad (1)$$

where α is a normalizing term, $p(I|L, \Theta)$ measures the likelihood of the image data given a particular configuration of the object, and $p(L|\Theta)$ is the prior distribution over object configurations.

Locating a single object in an image corresponds to maximizing (1). Felzenszwalb and Huttenlocher have shown that if E, $p(I|L, \Theta)$, and $p(L|\Theta)$ satisfy certain, reasonable conditions, then efficient algorithms exist to perform this maximization exactly (Felzenszwalb & Huttenlocher, 2005). Specifically, if the edges in E form a tree and $p(I \mid L, \Theta)$ can be factored as a product of individual part appearance models, then the posterior distribution takes the form

$$p(L \mid I, \Theta) \;=\; \alpha \prod_{i=1}^{n} p(I \mid l_i, a_i) \frac{\prod_{(v_i,v_j) \in E} p(l_i, l_j \mid \delta_{ij})}{\prod_{i=1}^{n} p(l_i \mid \Theta)^{\deg(v_i)-1}},$$

where the $p(I \mid l_i, a_i)$ are individual part appearance models, $p(l_i, l_j|\delta_{ij})$ are the priors over relative locations of connected parts, $p(l_i|\Theta)$ are the priors over individual part locations, and $\deg(v_i)$ is the degree of vertex v_i.

Under this factorization, finding the optimal configuration L^* of an object corresponds to the following well known cost minimization problem:

$$L^* = \arg\min_L \left(\sum_{i=1}^{n} m_i(l_i) + \sum_{(v_i,v_j) \in E} d_{ij}(l_i, l_j) \right), \quad (2)$$

where $m_i(l_i) = -\log p(I|l_i, a_i) + (\deg(v_i) - 1) \log p(l_i|\Theta)$ is the local match cost for each part and $d_{ij} = -\log p(l_i, l_j|\delta_{ij})$ is the deformation cost between each pair of connected parts. If $p(l_i, l_j \mid \delta_{ij})$ is Gaussian, then (2) can be computed exactly in $O(hn)$ via a combination of distance transforms and belief propagation, where h is the number of possible part locations (Felzenszwalb & Huttenlocher, 2005; Felzenszwalb & Huttenlocher, 2004). Note that standard belief propagation would scale as $O(h^2 n)$ which is impractical due to the size of h, which is typically the number of pixels in the image I.

2.1. Learning Pictorial Structures

Given a training set of images with labeled part locations, the parameters Θ of a classical pictorial structure model can be learned via a generative ML-based approach. Because $p(L \mid I, \Theta)$ is factored as a product of the appearance model $p(I \mid L, \Theta)$ and the structure model $p(L \mid \Theta)$, the appearance parameters A and structure parameters $\{E, \Delta\}$ can be learned independently. If we are given the edge set E of the model, finding the ML parameters A and Δ depends on the specific representations of the appearance and connection distributions, but this can be as simple as computing a sample mean and covariance from the training data or solving a linear least squares problem.

Heuristic approaches have also be developed for learning the edge set E. Felzenszwalb and Huttenlocher (2005) considered inducing a tree by first building a fully connected, undirected, weighted graph over V, where the weight of the edge between any two vertices v_i and v_j is a measure of the correlation between the relative positions of parts v_i and v_j. The edge set is then taken to be the minimum spanning tree of that graph. In cases where a non-tree edge set is desired, more expensive methods, such as exhaustive search, have been considered (Crandall et al., 2006).

Other parts-and-structure-type models for object recognition, such as those described in (Fergus et al., 2005; Fergus et al., 2003) and (Quattoni et al., 2004), also use ML-based model learning. However, these models are designed for classification, not localization, and they rely heavily on relatively accurate part detectors/localizers. Hence, the problem solved by these models, chiefly that of labeling parts, is fundamentally different than the problem we are concerned with here, which involves simultaneously determining part types and localizing parts.

3. Mixture-of-Parts Pictorial Structures

The classical pictorial structure model's assumption of a static part set undermines its ability to recognize some multi-object classes, like American football formations, whose parts can vary not only in location but also in type. Here we introduce the mixture-of-parts pictorial structure (MoPPS) model to help overcome this limitation.

3.1. General MoPPS Model

The MoPPS model is a triple $M = \langle \mathcal{V}, p_v, f \rangle$ where $\mathcal{V}$ is a finite set of parts, p_v is a probability distribution over $2^{\mathcal{V}}$ (i.e. subsets of $\mathcal{V}$), and f is a function that assigns a pictorial structure model to each subset $V \in 2^{\mathcal{V}}$ with $p_v(V) > 0$. We use Θ_V to denote the parameters of the pictorial structure assigned to part set V and take the vertices and edges of the structure to be implicit in Θ_V. Later, we discuss a particular representation for p_v and f.

In the case of American football, the set of parts $\mathcal{V}$ corresponds to all possible players, each of which has a specific role (e.g. left tight-end, full-back, left flanker, shotgun quarterback etc). The probability distribution p_v assigns non-zero probability only to those formations that contain exactly 11 parts, the number of players required in a formation, and that obey the formation constraints dictated by the rules of football (e.g. there must be 7 players on the line). Given a legal subset of players V, the corresponding pictorial structure Θ_V encodes the spatial constraints among the players in V along with local observation models for each player. Note that in this domain, the observation models for each player/part are identical since most players have very similar appearances.

Given an image I and a MoPPS model $M = \langle \mathcal{V}, p_v, f \rangle$, we are interested in inferring the most likely part set V and the locations L of those parts. The joint posterior distribution over V and L is given by

$$p(V, L \mid I, M) \ = \ \alpha \, p(I \mid L, \Theta_V) \, p(L \mid \Theta_V) \, p_v(V), \tag{3}$$

where α is a normalizing term, $p(I|L, \Theta_V)$ measures the image data likelihood of the pictorial structure model for V, and $p(L|\Theta_V)$ is the corresponding prior distribution over joint object locations. Note that under this model the marginal probability of the image data can be viewed as a mixture distribution of pictorial structure components, with one component per legal subset of parts, hence the name MoPPS.

Let $C(L \mid I, V) = -\log\left(p(I \mid L, \Theta_V) \, p(L \mid \Theta_V)\right)$ denote the cost assigned to locations L for parts V by pictorial structure Θ_V. We can then write our objective of finding the most likely set of parts and their locations as computing

$$(V^*, L^*) = \arg\min_{(V,L)} \left(C(L \mid I, V) - \log p_v(V)\right). \tag{4}$$

Assuming all pictorial structures Θ_V allow for efficient minimization of $C(L \mid I, V)$, e.g. using tree structures and Gaussian edge potentials, then the primary complexity in the above minimization problem is the potentially exponentially large set of part subsets that must be considered. To achieve practically efficient inference, we developed the MoPPS tree representation for a restricted class of MoPPS models. We present this representation in the next subsection. To simplify the discussion, we will assume for the remainder of the paper that p_v is a uniform distribution over all legal sets of parts.

3.2. The MoPPS Tree Representation

A MoPPS tree representation is a triple $\langle \mathcal{V}, \Theta, T \rangle$, where $\mathcal{V}$ is again a finite set of available parts, Θ is a tree-structured pictorial structure (the *global pictorial structure*) over the entire set of parts, and T is a boolean function that maps each part subset V to either **true** or **false** depending, respectively, on whether or not it is a legal part subset. We will denote by $\Theta|_V$ the projection of Θ onto V, which is just the subgraph of Θ induced by the part set V. Given a MoPPS tree representation the corresponding MoPPS model is given by $\langle \mathcal{V}, p_v, \Theta|_V \rangle$, where p_v is uniform over subsets V such that $T(V) = \textbf{true}$.

This representation can be viewed as compactly specifying $f(V) = \Theta|_V$ using a single global pictorial structure, returning the projection of part set V onto this structure for any legal V. An important property of this representation that is utilized in our inference procedures is monotonicity of the pictorial structure cost function $C^*(I, V) = \min_L C(L \mid I, V)$, which is the minimum pictorial structure cost for part set V. Specifically, we have that for any part subsets (legal or illegal) V and V', if $V \subseteq V'$ then $C^*(I, V) \leq C^*(I, V')$.

Clearly MoPPS trees cover only a subclass of possible MoPPS models. Intuitively, MoPPS trees are unable to represent object classes for which the spatial relationships between parts are not pairwise independent or for which parts' observation models can depend on other parts in V. Also, MoPPS trees cannot represent models in which a legal part set is a subset of another legal part set because, due to the monotonicity property of MoPPS trees, the larger part set will always achieve a higher cost and hence will never be selected as the best solution. Extending to allow for richer subclasses while maintaining practical inference is an interesting direction for future work. Nonetheless, MoPPS trees are rich enough for our current application and we believe for many others.

3.3. MoPPS Inference

Given a MoPPS model M represented as a MoPPS tree $\langle \mathcal{V}, \Theta, T \rangle$ we wish to solve the minimization problem defined in (4). Note that that if we know V^*, then we can efficiently compute L^* via the pictorial structure $\Theta|_{V^*}$. Thus, the fundamental problem here is to compute V^*. Under our assumption of a uniform p_v we can formulate the optimization problem as

$$V^* = \arg\min_{\{V : T(V)\}} C^*(I, V). \tag{5}$$

In our inference approach, we cast the problem as branch-and-bound search (BBS), which is a classical approach to combinatorial optimization that searches through a tree structure in which every node represents a subset of the space of combinatorial objects. Leaves of the BBS tree typically represent singleton sets or single combinatorial structures. As BBS proceeds, it continually expands new tree nodes and prunes any node from consideration whenever it can be proven that all structures it represents are suboptimal. Finding these nodes is done by computing both an upper and a lower bound on the cost of the combinatorial structures represented by each expanded node. Specifically, whenever a node's lower bound is greater than any other node's upper bound, we can prune the node from consideration without sacrificing optimality.

In the case of MoPPS inference (i.e. finding the minimum cost part set V^*), the combinatorial objects of interest are legal part sets, and, hence, each node of the search tree represents collections of part sets. Each node is labeled by a set of parts V, indicating that the node represents all legal part sets V' that contain the parts in V. More formally, we assume that a search space $\langle V_0, s\rangle$ is available for a given MoPPS optimization problem, where V_0 represents the initial search node (V_0 is just a set of parts or possibly the empty set), and s is a successor function that for any node of the tree V returns $s(V) = \{V'_1, \ldots, V'_k\}$ where the V'_i form a successor part set of V and it is assumed that the space satisfies $V \subseteq V'_i$ for all successors.

The other basic elements that must be specified to cast MoPPS inference as BBS are methods for computing informative upper and lower bounds on the cost of the set of part sets represented by a particular search node V. As we show in the coming paragraphs, we can leverage the special structure of the MoPPS tree representation to compute these bounds efficiently.

Given a search space, we consider using a best-first search strategy for BBS. This strategy requires an ordering relation on search nodes to maintain a priority queue of encountered nodes. Each search step removes the first search node from the priority queue, expands that node according to s, and adds its children to queue. Search stops when all search nodes have been eliminated except for a single leaf node, indicating that it must be optimal. In our experiments, we consider two ordering relations, one that orders nodes by their lower bound and one that orders them by their upper bound. In what follows, we denote these two search strategies as LB BBS and UB BBS, respectively.

Lower Bound Computation. An important property resulting from the subset relationship maintained by the successor function s is that any descendant V' of a search node V is a superset of V and hence, due to the monotonicity of the MoPPS tree representation, the cost of a node V will never be greater than that of any leaf node (i.e. legal part set) under V. This means that to compute a lower bound on the cost of any complete part set represented by V, i.e. the any of the leaf nodes under V, we need only to compute $C^*(I, V)$, which can be done efficiently using the pictorial structure $\Theta|_V$.

This lower bound can be easily improved in cases where one can find out the minimum number of parts in any leaf node under V. If the minimum size leaf node has k additional parts beyond V and $C^*_v = \min_{v \notin V} C^*(I, \{v\})$ is the minimum cost of any pictorial structure $\Theta|_{\{v\}}$, where v is a part that is not in V (note that C^*_v is based only on the corresponding part's observation model), it is straightforward to verify that $c_l(V) = C^*(I, V) + k\ C^*_v$ is still a lower bound.

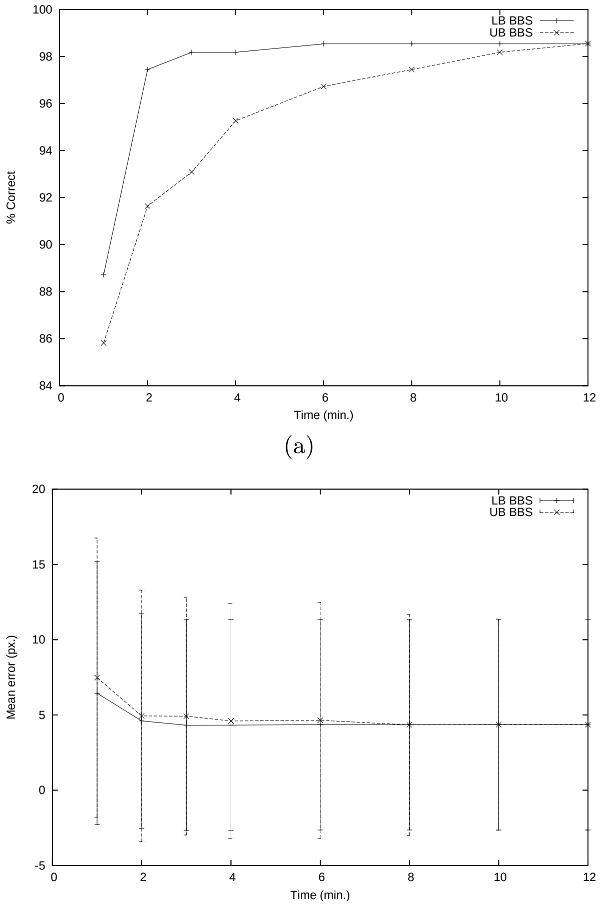
Upper Bound Computation. The main idea of our upper bound calculation is to quickly find a legal set of parts V_u that is a superset of the current node V and that we expect will have low (though perhaps not optimal) cost. If we can find such a set of parts, then we can use $C^*(I, V_u)$ as the upper bound on the cost of V. The key then is to quickly compute V_u, which we can do by leveraging the MoPPS tree representation.

In particular, prior to search, we use the global pictorial structure Θ to compute locations $\mathcal{L}$ for the entire set of parts $\mathcal{V}$. Then, to compute an upper bound on the cost of a node V using BBS, we select V_u as the minimum cost legal subset of $\mathcal{V}$ containing V with the location of each part in V_u fixed at the one specified in $\mathcal{L}$. That is, we select the V_u that minimizes $C(\mathcal{L}[V_u] \mid I, V_u)$ such that $V \subseteq V_u \subseteq \mathcal{V}$, $T(V_u) = \textbf{true}$, and where $\mathcal{L}[V_u]$ is the set of locations in $\mathcal{L}$ for parts in V_u. We can then use $C(\mathcal{L}[V_u] \mid I, V_u)$ as an upper bound on the cost of V. This upper bound may be tightened at the expense of an extra pictorial structure optimization by computing $C^*(I, V_u)$.

The search for the optimal V_u can be done via another branch-and-bound search, exhaustively (if computationally feasible), or even via a greedy, approximate hill-climbing search which at every step selects from the parts remaining in $\mathcal{V}$ the minimum cost part that does not make V_u an illegal part set.

3.4. Experiments in American Football.

We tested the MoPPS tree model by applying it to the American football formation recognition problem. As stated before, our goals in this domain are to classify



(a)

(b)

Figure 3. The graphs above depict the anytime behavior of two variants of BBS search over our entire dataset of 25 formations in terms of (a) the percent of players assigned the correct type and (b) the mean pixel error of the location estimates for correctly classified players. In our model, six pixels is equal to one yard on the football field, so the error of location estimates is generally less than a yard.

the players that constitute each formation as well as to determine their locations. This is an interesting problem, considering that all professional and most major college football teams employ crews of video scouts who spend many man hours each week using specialized software to label opponent video by formation and other factors to allow for content-based queries by coaches. Thus, a semi-automated system for this task would have commercial impact potential. Interestingly, the imagery we use in our experiments comes directly from the video used by the Oregon State University football team.

The raw observations provided to our model include responses from a background segmentation-based likelihood model and a color histogram-based likelihood model. Both of these observation models are quite noisy (e.g. the color-based model often gets confused with the several logos on the field) and using this information in the absence of joint inference would produce extremely poor results.

For our experiments, we used a MoPPS tree model with a total of 34 available parts corresponding to 16 basic player types as well as several subtypes that capture different attributes of certain players (such as whether the quarterback is in shotgun formation or under the center). These parts, subject to a set of hard constraints based on the rules of football, combine to form over 3200 legal formations.

Each image in our dataset can be automatically registered to an overhead view of the football field, as depicted in Figure 1, using the technique described in (Hess & Fern, 2007), which allows us to model the relative locations of players in 2D football field coordinates. Specifically, the connection parameters Δ are the mean and diagonal covariance of a Gaussian distribution over each player's ideal location in field coordinates relative to a "parent" player in the MoPPS tree. All parameters of the MoPPS tree were hand-coded using a small set of training images.

Figure 3 depicts the anytime behavior of the two variants of BBS described in the previous subsection over a dataset of 25 images of the initial offensive formation at the beginning of an American football play. On average the BBS procedure obtains optimal results in four minutes per play, which significantly improves over exhaustive search over part sets which requires close to an hour of processing. The model labels players by the correct type 98% of the time and the mean-squared error of location prediction is less than a yard.

We plan to release soon an extended version of our fully labeled dataset with close to 100 football formations from different games[1].

4. Directions for MoPPS Learning

Our initial experience using hand-coded MoPPS models in the American football domain has pointed out a number of important opportunities for learning. Below we outline each of these in more detail and for some provide initial thoughts on how to proceed. Each of these areas raises fundamental issues in machine learning for structured outputs and we believe that our football formation dataset will help drive research in these directions.

[1] This dataset will be made available on our project's website at `http://eecs.oregonstate.edu/football`.

4.1. Structure and Parameter Learning.

While we were eventually able to hand-code a reasonably accurate MoPPS model for our domain, the process of doing so was quite labor intensive requiring many iterations of debugging and analysis. Thus, one of our first objectives is to develop methods for learning the structure and parameters of a MoPPS model from labeled training data. In particular, we have so far considered only offensive formations and expect defensive formations to pose as great and likely a greater challenge, as there are very few hard constraints imposed on these by the rules of football.

Our goal is to provide the system with the set of hard constraints on part sets and a set of labeled training images and to produce a MoPPS tree model. As an initial approach we note that it is straightforward in concept to learn a MoPPS tree via a generative approach similar that used to learn tree-structured pictorial structures as described earlier. Importantly, this approach will not rely on performing the relatively expensive inference process during training.

While we expect that the generative approach will provide non-trivial performance, our hand-coding experience indicated that the purely generative model was problematic in certain cases. In particular, the observation likelihood we used has a tendency to over-count evidence. This is a problem which has been observed previously for pictorial structures. Some authors attempt to mitigate the effects of over-counting by applying a smoothing factor to the observation likelihood (Felzenszwalb & Huttenlocher, 2005; Torralba et al., 2003). However, we have found that this approach accentuates false peaks in the observation likelihood that are due to slight errors during registration with an overhead view. Instead, we added a reward term to the cost function of players types whose ideal locations make over-counting the evidence associated with them unlikely and tuned these rewards by hand. This suggests that it will likely be useful to perform some amount of discriminative training utilizing features on top of the initial generatively trained model.

Unfortunately, most discriminative methods for training structured output models require performing inference during each parameter update. Currently, our inference process requires an average of four minutes per example, which would make such training highly impractical. We plan to deal with this issue by organizing the discriminative training in rounds, where in each round we first gather during a BBS search a set of formations with cost better than or close to the target. We will then restrict discriminative learning to this set of formations. As an initial algorithm we plan to use averaged Perceptron updates (Collins, 2002).

4.2. Speed-up Learning.

Our current system still requires an average of 4 minutes to process each play. While this is much improved over the use of exhaustive search (close to an hour per play), it is reasonable to expect the system load to be up to 1000-2000 plays per week. This load would currently require substantial computing power to perform formation labeling in a time-frame that would be useful to coaches for analysis during the week. We plan to study various approaches to speed-up learning in the context of MoPPS models, in particular in the context of speeding-up BBS search. Given a learned or hand-coded MoPPS model, the goal of speed-up learning is to learn some form of knowledge that facilitates substantially faster inference, with little or no impact on accuracy. At this point several directions appear promising.

First, given a MoPPS model along with upper and lower bound functions, the specific structure of the branch and bound search space can have a substantial impact on the amount of pruning achieved and hence the speed of search. It is thus interesting to consider learning search space operators that maximize the effectiveness of branch and bound. Second, given a MoPPS model and a search space, it is interesting to consider learning more accurate upper and lower bound functions that result in more effective pruning. This for example could be done by learning to augment the current functions with learned correction terms. Such an approach has been successful for example in the area of learning for AI planning (Yoon et al., 2006). Finally, given a MoPPS model and a search space we would like to consider learning a priority queue ranking function that dictates the order in which nodes are expanded. The quality of the ranking function or heuristic can have a huge impact on the amount of pruning and the anytime performance of the algorithm.

4.3. Transfer and Active Learning

A deployed system in the football domain would be expected to work well across a wide range of game footage sources, and we believe some amount of parameter adjustment to the MoPPS model will be required for each particular source. This will often be due to differences in the observation models between sources and to varying registration accuracy. Obviously, for the system to be usable, this parameter adjustment should be mostly automated, requiring only minimal user interaction. For this purpose, a combination of transfer and active learning mechanisms appear to be warranted.

In particular, we would like to develop learning mechanisms that can utilize prior experience in order to reduce the amount of labeled data required for a new video source. In addition, we would like this mechanism to actively query the user, with the aim of minimizing the required amount of labeling. The best mechanisms for accomplishing this is left as an open problem. However, a rough idea for such an approach is to consider learning a prior model on MoPPS tree parameters across many video sources and then to use that prior to guide an active calibration mechanism given a new dataset.

Acknowledgments

This work was supported by NSF grant IIS-0307592. In addition to NSF, we would also like to thank the coaches and staff of the Oregon State University football team for providing us with the football video from which our image data was derived.

References

Collins, M. (2002). Discriminative training methods for hidden Markov models: Theory and experiments with the perceptron algorithm. *Conf. on Empirical Methods in NLP*.

Crandall, D., Felzenszwalb, P., & Huttenlocher, D. (2006). *Object recognition by combining appearance and geometry*, vol. 4170/2006 of *LNCS*, 462–482. Springer.

Felzenszwalb, P., & Huttenlocher, D. (2004). *Distance transforms of sampled functions* (Technical Report TR2004-1963). Cornell Computing and Information Science.

Felzenszwalb, P., & Huttenlocher, D. (2005). Pictorial structures for object recognition. *International Journal of Computer Vision*, *61*.

Fergus, R., Perona, P., & Zisserman, A. (2003). Object class recognition by unsupervised scale-invariant learning. *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*.

Fergus, R., Perona, P., & Zisserman, A. (2005). A sparse object category for efficient learning and exhaustive recognition. *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*.

Hess, R., & Fern, A. (2007). Improved video registration using non-distinctive local image features. *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. To appear.

Lan, X., & Huttenlocher, D. (2005). Beyond trees: common-factor models for 2D human pose recovery. *Proc. IEEE Int'l Conf. on Computer Vision*.

Quattoni, A., Collins, M., & Darrell, T. (2004). Conditional random fields for object recognition. *Proc. Advances in Neural Information Processing Systems*.

Torralba, A., Murphy, K., Freeman, W., & Rubin, M. (2003). Context-based vision system for place and object recognition. *International Conference on Computer Vision*.

Yoon, S., Fern, A., & Givan, R. (2006). Learning heuristic functions from relaxed plans. *International Conference on Automated Planning and Scheduling (ICAPS)*.