A New Bayesian Framework for Object Recognition

Yuri Boykov, Daniel Huttenlocher
Computer Science Department
Cornell University
Ithaca, NY 14853, USA
{yura,dph}@cs.cornell.edu

Abstract

We describe a new approach to feature-based object recognition, using maximum a posteriori (MAP) estimation under a Markov random field (MRF) model. The main advantage of this approach is that it allows explicit modeling of dependencies between individual features of an object model. For instance, it can capture the fact that unmatched features due to partial occlusion are generally spatially coherent rather than independent. Efficient computation of the MAP estimate in our framework can be accomplished by finding a minimum cut on an appropriately defined graph. An even more efficient approximation, that does not use graph cuts, is possible. We call this technique spatially coherent matching. Our framework can also be seen as providing a probabilistic understanding of Hausdorff matching. We present ROC curves from Monte Carlo experiments that illustrate the improvement of the new spatially coherent matching technique over Hausdorff matching.

1 Introduction

In this paper we present a new Bayesian approach to object recognition using Markov random fields (MRF's). As with many approaches to recognition we assume that an object is modeled as a set of features. The recognition task is then to determine whether there is a match between some subset of these object features and features extracted from an observed image. The central idea underlying our approach is to explicitly capture dependencies between individual features of the object model. Markov random fields provide a good theoretical framework for representing dependencies between features. Moreover, recent algorithmic developments make it quite practical to compute the maximum a posteriori (MAP) estimate for the MRF model that we employ (e.g., [Boykov *et al.*, 1998], [Greig *et al.*, 1989]).

Our approach contrasts with most feature-based object recognition techniques, as they do not explicitly account for dependencies between features of the object. It is desirable to be able to account for such dependencies, because they occur in real imaging situations. For example, a common case occurs with partial occlusion of objects, where features that are near one another in the image are likely to be occluded together. In our model, we assume that the process of matching individual object features is described *a priori* by a Gibbs distribution associated with a certain Markov random field. This model captures pairwise dependencies between features of the object. We then use *maximum a posteriori* (MAP) estimation to find the match between the object and the scene or to show that there is no such match. While a number of probabilistic approaches to recognition have been reported in the literature (e.g., [Pope and Lowe, 1996], [Olson, 1998],[Subrahmonia *et al.*, 1996]) these methods do not provide an explicit model of dependencies between features.

We show that finding the best match using the Hausdorff fraction [Huttenlocher *et al.*, 1993], [Rucklidge, 1996] is a special case of our technique, where the dependencies between all pairs of features in the object model are equally strong. Therefore, our Bayesian framework can be seen as providing a probabilistic understanding of Hausdorff matching. With this view of Hausdorff matching, it becomes apparent that one of the main limitations of the Hausdorff approach is its failure to take into account the continuity of matches between neighboring features. That is, the Hausdorff approach does not account for the fact that in a local neighborhood there tends to be a higher correlation between features. We suggest a modification to Hausdorff approach which we call *spatially coherent matching*. This method requires matching features to be coherent in a given neighborhood system of the model. We present some Monte Carlo experiments demonstrating that this spatially coherent matching measure is a substantial improvement over Hausdorff matching in the case that images are cluttered with many irrelevant features and have substantial occlusion of the object to be recognized.

2 The MAP-MRF Recognition Framework

In this section we describe our object matching framework in more detail. We represent an object by a set of features, indexed by integers in the set $M = \{1, 2, \ldots, m\}$. Each feature corresponds to some vector M_i in a feature space of the model. Commonly the vectors M_i will simply specify a feature location (x, y) in a fixed coordinate system of the model, although more complex feature spaces fit within the framework.

A given image I is a set of observed features from some underlying true scene. Each feature $i \in I$ corresponds to a vector I_i in a feature space of the image. The true scene can be thought of as some unknown set of features I^T in the same feature space. Similarly, I_i^T is a vector describing the feature $i \in I^T$ in the feature space of the image. We are interested in finding a match between the model M and the true scene I^T, using the observed features I.

A match of the model M to the true scene I^T is described by a pair $\{S, L\}$ where $S = \{S_1, S_2, \ldots, S_m\}$ is a collection of boolean variables and L is a location parameter. If $S_i = 1$ then the ith feature of the model has a matching feature in I^T and if $S_i = 0$ then it does not. In this case we say it is mismatched. For example, the event $\{S_1 = \ldots = S_k = 1,\ S_{k+1} = \ldots = S_m = 0,\ L = l\}$ implies that for $1 \le i \le k$, feature i of M has a matching feature $j \in I^T$, such that $I_j^T = M_i \oplus L$. Moreover, the last $(m - k)$ features are mismatched, meaning they have no such matching features. The operation $\oplus$ depends on the type of mapping from the model to the image feature space, which varies for the particular recognition task. In this paper we will use translation (vector summation), but other transformations are possible.

To determine the values of $\{S, L\}$ we use the maximum a posteriori (MAP) estimate

$$\{S^*, L^*\} = \arg\max_{S,L} \Pr(S, L|I).$$

Bayes rule then implies

$$\{S^*, L^*\} = \arg\max_{S,L} \Pr(I|S, L)\Pr(S)\Pr(L) \quad (1)$$

assuming that S and L are a priori independent. The prior distributions $\Pr(S)$ and $\Pr(L)$ are discussed in section 2.1. We assume that the prior distribution of S is described by a certain Markov random field, thus allowing for spatial dependencies among the S_i. The likelihood function $\Pr(I|S, L)$ is discussed in section 2.2.

Let $\mathcal{L}$ denote a set of possible locations of the model in the true scene. Then the range of the location parameter L is $\mathcal{L} \cup \emptyset$ where the extra value $\emptyset$ implies that the model is not in the scene. The basic idea of our recognition framework is to report a match between the model and the observed scene if and only if

$$S^* \neq \bar{0} \quad \text{and} \quad L^* \neq \emptyset. \quad (2)$$

In section 2.3 we develop the test in (2) for the model specified in 2.1 and 2.2.

2.1 Prior Knowledge

We assume that the prior distribution of the location parameter L can be described as

$$\Pr(L) \;=\; (1-\rho)\cdot f(L) \;+\; \rho\cdot\delta(L=\emptyset) \quad (3)$$

where $f(L) = \Pr(L|L \in \mathcal{L})$, the parameter ρ is the prior probability that the model is not present in the scene, and $\delta(\cdot)$ equals 1 or 0 depending on whether condition "$\cdot$" is true or false. Generally the distribution function $f(L)$ is uniform over $\mathcal{L}$. However in some applications $f(L)$ can reflect additional information about the model's location. For example, such information might be available in object tracking since the current location of the model can be estimated from previous iterations. The value of the constant ρ may be anywhere in the range $[0,1)$. In section 2.3 we will see that ρ appears in our recognition technique only as a threshold for deciding whether or not the model is present given the image.

We assume that the collection of boolean variables, S, indicating the presence or absence of each feature, forms a Markov random field independent of L. More specifically, the prior distribution of S is described by the Gibbs[1] distribution

$$\Pr\{S\} \;\propto\; \exp\left\{-\sum_{i\in M}\alpha\cdot(1-S_i) \right. \\ \left. -\sum_{\{i,j\}}\beta_{\{i,j\}}\cdot\delta(S_i\neq S_j)\right\} \quad (4)$$

where the second summation is over all distinct unordered pairs of model features.

The motivation for this model is that $\Pr(S)$ captures the probability that features will not be matched even though they are present in the true scene, given some fixed location, L. Such non-matches could be due to occlusion, feature extraction error, or other causes. The parameter $\alpha \geq 0$ is a penalty for such non-matching features. The coefficient $\beta_{\{i,j\}} \geq 0$ specifies a strength of interaction between model features i and j. For tractability, we consider only pairwise interaction between features. Nevertheless, the pairwise interaction model provided by this form of Gibbs distribution is rich enough to capture one important intuitive property: a priori it is less likely that a feature will be un-matched if other features of the model have a match. Note that if all $\beta_{\{i,j\}} = 0$ then there is no interaction between the features and the S_i's become independent Bernoulli variables with probability of success $\Pr(S_i = 1) = e^{\alpha}/(1+e^{\alpha}) \;\geq\; 0.5$.

[1] See [Li, 1995] for more details on Gibbs distribution.

2.2 Likelihood Function

The features of the observed image I may appear differently from the features of the unknown true scene I^T due to a number of factors. This includes sensor noise, errors of feature extraction algorithms (e.g. edge detection), and others. It is the purpose of the likelihood function to describe these differences in probabilistic terms.

We assume that the likelihood function is given by

$$\Pr(I|S,L) \;\propto\; \prod_{i\in M} g_i(I|S_i,L) \quad (5)$$

where $g_i(\cdot)$ is a likelihood function corresponding to the ith feature of the model. If $S_i = 0$ or $L = \emptyset$ then $g_i(I|S_i,L)$ is the likelihood of I given that the true scene does not contain the ith feature of the model. We assume that all cases of mismatching feature have the same likelihood. That is, for any $i \in M$ and $L \in \mathcal{L}$

$$g_i(I|1,\emptyset) = g_i(I|0,\emptyset) = g_i(I|0,L) = C_0 \quad (6)$$

where C_0 is a positive constant.

If $L \in \mathcal{L}$ then $g_i(I|1,L)$ is the likelihood of observing image I given that the i-th feature of the model is at location $(L \oplus M_i)$ in the feature space of the true scene I^T. The choice of $g_i(I|1,L)$ for $L \in \mathcal{L}$ will depend on the particular application.

Example 1. (Recognition based on edges) Consider an edge-based object matching problem, where all features of the model are edge pixels. We observe a set of image features I obtained by an intensity edge detection algorithm. One reasonable choice of $g_i(I|1,L)$ for $L \in \mathcal{L}$ is

$$g_i(I|1,L) \;=\; C_1\cdot g(d_I(L\oplus M_i)) \quad (7)$$

where $d_I(\cdot)$ is a distance transform of the image features I. That is, the value of $d_I(p)$ is the distance from p to the nearest feature in I. The function $g(\cdot)$ is some probability distribution that is a function of the distance to the nearest feature. Normally, g is a distribution concentrated around zero. The underlying intuition is that if if the true scene I^T has an edge feature located at $(L \oplus M_i)$ then the observed image I should contain an edge nearby. Thus the distance transform $d_I(L \oplus M_i)$ will be small with large probability. A number of existing recognition schemes use functions of this form, including Hausdorff matching [Huttenlocher *et al.*, 1993].

2.3 MAP Estimation

By substituting (3), (4), (5) into (1) and then taking the negative logarithm of the obtained equation we can show that MAP estimates $\{S^*, L^*\}$ minimize the value of the posterior energy function

$$E(S,L) = $$
$$= \begin{cases} H_L(S) - \ln f(L) & - \quad \ln(1-\rho) \quad \text{if } L \in \mathcal{L} \\ H_L(S) & - \quad \ln \rho \quad \text{if } L = \emptyset \end{cases}$$

where

$$\begin{aligned} H_L(S) \quad = & \sum_{\{i,j\}} \beta_{\{i,j\}} \cdot \delta(S_i \neq S_j) \qquad (8) \\ & + \sum_{i \in M} (\alpha \cdot (1 - S_i) - \ln g_i(I|S_i, L)) \, . \end{aligned}$$

Our goal is to find $\{S^*, L^*\}$. The main technical difficulty is to determine $\{\hat{S}, \hat{L}\}$ that minimize $H_L(S) - \ln f(L)$ for $L \in \mathcal{L}$. In general this can be done by computing a minimum cut of an appropriate graph (see in [Greig *et al.*, 1989] and [Boykov *et al.*, 1998]). In section 3 we consider some special cases where no sophisticated algorithmic scheme is needed to obtain $\{\hat{S}, \hat{L}\}$. For the moment assume that $\{\hat{S}, \hat{L}\}$ are given.

Consider $H_L(S)$ for $L = \emptyset$. Equation (6) implies that $H_\emptyset(S)$ is minimized by the configuration $S = \bar{1}$ where all $S_i = 1$. If $E(\hat{S}, \hat{L}) > E(\bar{1}, \emptyset)$ then $\{S^*, L^*\} = \{\bar{1}, \emptyset\}$. According to (2), in this case we report that the model is not recognized in the scene. If $E(\hat{S}, \hat{L}) \leq E(\bar{1}, \emptyset)$ then $\{S^*, L^*\} = \{\hat{S}, \hat{L}\}$. In this case $L^* \in \mathcal{L}$. Nevertheless, if $\hat{S} = \bar{0}$ we would still report the absence of the model in the scene.

Finally, our recognition framework can be summarized as follows. The match between the model and the observed scene is reported if and only if $\hat{S} \neq \bar{0}$ and

$$H_{\hat{L}}(\hat{S}) - \ln f(\hat{L}) \;\leq\; m \cdot \ln \frac{1}{C_0} + \ln \frac{1-\rho}{\rho} \qquad (9)$$

where (9) is derived from the inequality $E(\hat{S}, \hat{L}) \leq E(\bar{1}, \emptyset)$. The right hand side in (9) is a constant that represents a certain decision threshold. Note that this decision threshold depends on two things: first, the prior probability of occlusion, ρ; and second, the product of the number of model features, m, with the log-likelihood of a mismatch, C_0.

3 Discussion of special cases

In this section we identify some interesting properties of our recognition framework by considering several special cases. We concentrate on the problem of finding an optimal match configuration S_L that minimizes $H_L(S)$ at a fixed location $L \in \mathcal{L}$.

In section 3.1 we show that Hausdorff matching is a special case of our framework. In section 3.2 we discuss models where some neighborhood system is imposed over the features. For such models our framework yields a simple technique which we call *spatially coherent matching.* Spatially coherent matching is a natural generalization of the Hausdorff matching.

3.1 Hausdorff matching

In this section we show that Hausdorff matching is a special case of our framework where the strength of interaction between features of the model is uniform, that is, $\beta_{\{i,j\}} = \beta$ for all $\{i,j\}$ where β is a non-negative constant. The classical Hausdorff distance is a max-min measure for comparing two sets for which there is some underlying distance function on pairs of elements, one from each set. The application of Hausdorff matching in computer vision has used a generalization of this classical measure [Huttenlocher *et* *al.*, 1993], based on computing a quantile rather than maximum of distances.

One form of the generalized Hausdorff measure counts the number of model features that are within some distance r of the nearest image feature. Let $M_L = \{i \in M : d_I(L \oplus M_i) \leq r\}$ denote the subset of model features lying within distance r of image features, when the model is positioned at L. Then the model is matched if and only if the number of elements in this subset, $|M_L|$, is larger than some critical fraction of the total number of model features, m.

Thus, as in Example 1 we assume that $g_i(I|1, L) = C_1 \cdot g(d_I(L \oplus M_i))$, and moreover we use the particular function,

$$g(d) \;=\; \begin{cases} \frac{1}{r} & \text{if} \quad d \leq r \\ 0 & \text{if} \quad d > r \end{cases}$$

where r is the distance to the nearest model feature used in Hausdorff matching.

We will need the following notation. Any configuration S is uniquely defined by a collection of integers $1_S = \{i \in M : S_i = 1\}$ which is the subset of model features assigned a match by S. Consider also $0_S = \{i \in M : S_i = 0\}$. Note that for any configuration S we have $1_S \cup 0_S = M$ and $1_S \cap 0_S = \emptyset$. Therefore, $m = |1_S| + |0_S|$.

Our approach is based on minimizing the function $H_L(S)$ in (8) for a fixed location $L \in \mathcal{L}$. Note that if $d_I(L \oplus M_i) > r$ then $g_i(I|1, L) = 0$. This means that the likelihood of a match for a feature $i \in M$ is zero if the image I does not contain any features near $L \oplus M_i$. Thus, it does not make any sense to assign $S_i = 1$ if the ith feature of the model is such that $d_I(L \oplus M_i) > r$, and we must have $1_S \subset M_L$. Formally speaking, it is easy to check that $1_S \not\subset M_L$ implies $H_L(S) = \infty$. If $1_S \subset M_L$ then the second summation in (8) can be rewritten as $|0_S| \cdot (\alpha - \ln C_0) - |1_S| \cdot \ln \frac{C_1}{r}$.

The assumption that $\beta_{\{i,j\}} = \beta$ for all $\{i, j\}$ simplifies the first term of $H_L(S)$ in (8) to $\beta \cdot |1_S| \cdot |0_S|$. Since $|0_S| = m - |1_S|$ then $H_L(S)$ can be rewritten as a function of a single scalar

$$H_L(S) \;=\; \begin{cases} h(|1_S|) & \text{if} \;\; 1_S \subset M_L \\ \infty & \text{if} \;\; 1_S \not\subset M_L \end{cases} \qquad (10)$$

where

$$h(x) \;=\; \beta \cdot x \cdot (m - x) \; - x \cdot (\alpha + \ln \frac{C_1}{rC_0}) \; + \; m \cdot (\alpha - \ln C_0)$$

is a concave down parabola.

Now we can show how to find a configuration S_L that minimizes $H_L(S)$ for a fixed L. Equation (10) implies that $0 \leq |1_S| \leq |M_L|$. Thus $h(|1_S|)$ is minimized by either $|1_S| = 0$ or $|1_S| = |M_L|$. It is straightforward to check that $h(|M_L|) < h(0)$ if and only if $|M_L| > K$ where

$$K \;=\; m - \left(\frac{\alpha + \ln \frac{C_1}{rC_0}}{\beta} \right). \qquad (11)$$

Consequently, $S_L \neq \bar{0}$ if and only if $|M_L| > K$ which is the Hausdorff test described above.

3.2 Spatially Coherent Matching

In this section we consider models where certain pairs of features can be viewed as local neighbors. One simple kind of model with a natural local neighborhood system is successive points in an edge chain, as illustrated in figure 1. Let $\mathcal{N}_M$ denote a set of all pairs of neighboring features in a given model M. We assume that $\beta_{\{i,j\}} = \beta + \beta_N$ if the features $\{i, j\} \in \mathcal{N}_M$ are neighbors and $\beta_{\{i,j\}} = \beta$ if the features $\{i, j\} \notin \mathcal{N}_M$ are not neighbors. The coefficients β and β_N are some nonnegative constants. It is reasonable to expect that two neighboring features are more likely to have the same label than a pair of features isolated from each other. This type of interaction between model features generalizes the example in section 3.1.

We assume that the likelihood g_i is defined the same way as in section 3.1. Then equation (8) can be written as

$$H_L(S) = \begin{cases} \beta_N \cdot b(S) + h(|1_S|) & \text{if} \;\; 1_S \subset M_L \\ \infty & \text{if} \;\; 1_S \not\subset M_L \end{cases}$$

where $b(S) = |\{i, j\} \in \mathcal{N}_M : S_i \neq S_j|$ denotes the number of pairs of neighboring features assigned opposite labels by the configuration S. The rest of notation is borrowed from section 3.1.

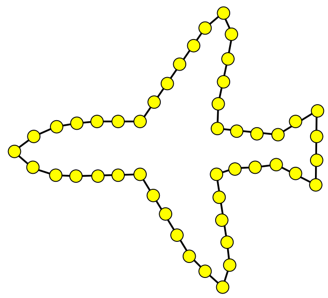


Figure 1: Example of a model with the chain neighborhood system. The neighborhood system $\mathcal{N}_M$ is represented by edges in the feature space of the model.

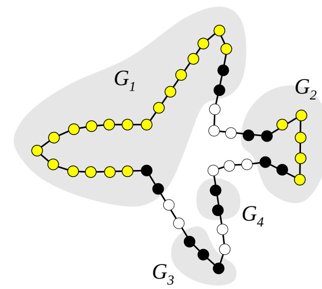


Figure 2: The features of M_L are highlighted by shading. The unmatchable features U_L are white. The boundary features B_L for $R = 2$ are shown black. In this case $1_{\tilde{S}} = G_1 \cup G_2$.

We now introduce a technique that approximately obtains the configuration S_L minimizing $H_L(S)$ for a fixed location L. We call this technique *spatially coherent matching*. It can be seen as a generalization of Hausdorff matching that requires matching features to be coherent in the neighborhood system of the model.

We use some of the notation from Section 3.1. Recall that $M_L = \{i \in M : d_I(L \oplus M_i) \leq r\}$ is the subset of model features lying within distance r of image features, when the model is positioned at L. The formula for $H_L(S)$ above implies that $1_S \subset M_L$. Thus, we can think of M_L as a set of *matchable* model features for a given location L. In addition, we define a subset of *unmatchable* model features $U_L = \{i \in M \mid d_I(L \oplus M_i) > r\}$ that also corresponds to a fixed location L. The set U_L consists of model features that are greater than distance r from any image features. Note that $U_L = M - M_L$. Since $1_S \subset M_L$ then the features in U_L must be mismatched (i.e., $U_L \subset 0_S$).

The main idea of the spatially coherent matching scheme is to require that matching features should form large connected groups. There should be no isolated matches. Assume that $\gamma(i,j)$ denotes the number of chains in the shortest sequence $\{i, i_1\}$, $\{i_1, i_2\}$, $\ldots, \{i_{k-1}, j\}$ in $\mathcal{N}_M$ connecting two feature i and j in M. Let $B_L \subset M_L$ denote the subset of features in M_L that are "near" features of U_L. That is, $B_L = \{i \in M_L \mid \exists\, j \in U_L, \gamma(i,j) \leq R\}$, where R is a fixed integer parameter. We will refer to B_L as a *boundary* of the set of matchable features M_L. Consider an example in figure 2 where the matchable features M_L are highlighted by shading and the unmatchable features U_L are white. The black features in figure 2 show the boundary B_L for $R = 2$. The locally coherent matching technique works as follows. The main test is

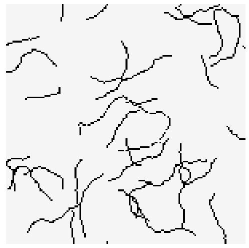
$$|M_L| - |B_L| \quad > \quad K \tag{12}$$

where K is the same as in (11). Note that $|M_L| - |B_L|$ is the number of non-boundary features in M_L. If (12) is false then $S_L = \bar{0}$ and there is no match. If the number of non-boundary features is sufficiently large so that (12) holds then the matching configuration is $S_L = \tilde{S}$ where

$$1_{\tilde{S}} = \bigcup_{G \in \mathcal{G}_L \,:\, G \not\subset B_L} G \tag{13}$$

and $\mathcal{G}_L = \{G_1, G_2, \ldots, G_{n_L}\}$ is the set of all connected components (or groups) in a given M_L. Each group $G \in \mathcal{G}_L$ is a subset of matchable features connected under the neighborhood system $\mathcal{N}_M$. For example, in figure 2 the features of M_L form four groups $\mathcal{G}_L = \{G_1, G_2, G_3, G_4\}$. Note that in (13) we include groups G which have some non-boundary features. Therefore, the match could be assigned only to those features which belong to sufficiently large connected components in M_L. In the example of figure 2 we have $1_{\tilde{S}} = G_1 \cup G_2$ since G_1 and G_2 are the only groups in M_L that contain some

a) A model.



b) An image. The model is in the center.

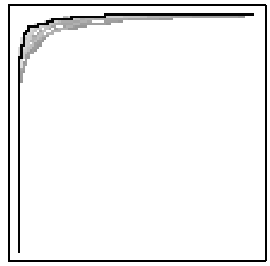
Figure 3: Monte Carlo experiments.

non-boundary features. The groups G_3 and G_4 are discarded because they are too small and lie completely inside the boundary B_L.

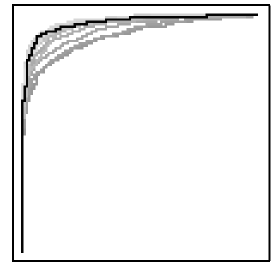
The spatially coherent matching technique is easy to implement. Note that in practice the boundary B_L can be often approximated by morphological dilation of the unmatchable features U_L in the model's feature space by the radius R and then collecting the matchable features in M_L that lie in the dilated area.

The spatially coherent matching method is a simple generalization of the Hausdorff matching technique explained in Section 3.1. Note that the size of the boundary $|B_L|$ is small if the features in M_L are grouped in large connected blobs and $|B_L|$ is large if the matchable features are isolated from each other. Therefore, spatially coherent matching technique is reluctant to match if the features in M_L are scattered in small groups even if the size of M_L is large. In contrast, the Hausdorff matching cares only about the size of M_L and ignores connectedness. Note that the spatially coherent matching technique is equivalent to the Hausdorff matching when $R = 0$.

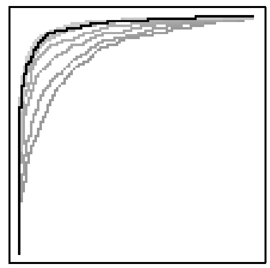
This spatially coherent matching technique can be related to the graph-based methods, and in fact provide a solution for a model with a chain neighborhood system introduced above given that $\beta_N = R \cdot \beta \cdot m$. Due to space limitations we do not discuss this further here.



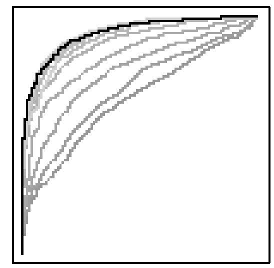
clut=3% occl=20%



clut=5% occl=20%



clut=3% occl=40%



clut=5% occl=40%

Figure 4: ROC curves.

4 Experimental results

In order to evaluate the new recognition measure developed in this paper, we have run a series of experiments using Monte Carlo techniques to estimate Receiver Operating Characteristic (ROC) curves for each measure. A ROC curve plots the probability of detection along the y-axis and the probability of false alarm along the x-axis. Thus, the ideal recognition algorithms would produce results near the top left of the graph (low false alarm and high detection probabilities).

We have estimated ROC curves for the measures described above by performing matching in synthetic images and using the matches found in these images to estimate the curve over a range of possible parameter settings. 1000 test images were used in the experiments, and were generated according to the following procedure. Random chains of edge pixels with a uniform distribution of lengths between 20 and 60 pixels were generated in a 150×150 image until a predetermined fraction of the image was covered with such chains. Curved chains were generated by changing the orientation of the chain at each pixel by a value selected from a uniform distribution between $-\frac{\pi}{8}$ and $+\frac{\pi}{8}$. An instance of a model image was then placed in the image, after rotating, scaling, and translating the model image by random values. The scale change was limited to $\pm 10\%$ and the rotation change was limited to $\pm\frac{\pi}{18}$. Occlusion was simulated by erasing the pixels corresponding to a connected chain of the model image pixels. Gaussian noise was added to the locations of the model image pixels ($\sigma = 0.25$). The pixel coordinates were finally rounded to the closest integer.

For the experiments reported here, we performed recognition using the 56×34 model image shown in Figure 3(a). An example of a synthetic image generated using this model image and the procedure described above is shown in Figure 3(b). In each trial, a given matching measure with a given parameter value was used to find all the matches of the model to the image. A trial was said to find the correct object if the position (considering only translation here) of one of the matches was within three pixels of the correct location of the object in the image. A trial was said to find a false positive if any match was found outside of this range (and that match was not contiguous with a correct match position).

Figure 4 shows the ROC curves corresponding to experiments with different levels of model occlusion and image clutter. The black curve shows the best results we could obtain from the general graph approach. The gray curves correspond to the spatially coherent matching technique for various values of $R \in [0, 25]$. As R gets larger, up to 20 or 21, the results improve, so the curves closer to the top left are for larger values of R. For even larger values of R, which we do not show, the ROC curves rapidly deteriorate. It is interesting to note that given this particular model, a distance of $R = 25$ corresponds approximately to the radius of the model. Thus the performance does not deteriorate until the coherence region begins connecting together very distant pieces of the model.

Note that the case of $R = 0$ corresponds to Hausdorff matching. Thus the spatial coherence approach plays a large role in improving the quality of the match, because $R = 0$ has the worst matching performance. The value of R does not make a big difference for lower clutter or occlusion cases (top row of the figure), but makes a very large difference when these are larger (bottom row of the figure). Note that in [Olson and Huttenlocher, 1997], using the same Monte Carlo framework, it was shown that Hausdorff matching works better than a number of other methods including binary correlation and Chamfer matching. Thus these results indicate that spatially coherent matching is a substantial improvement over other commonly used binary image matching techniques.

References

[Boykov *et al.*, 1998] Y. Boykov, O. Veksler, and R. Zabih. Markov random fields with efficient approximations. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 648–655, 1998.

[Greig *et al.*, 1989] D. Greig, B. Porteous, and A. Seheult. Exact maximum a posteriori estimation for binary images. *Journal of the Royal Statistical Society, Series B*, 51(2):271–279, 1989.

[Huttenlocher *et al.*, 1993] D. P. Huttenlocher, G. A. Klanderman, and W. J. Rucklidge. Comparing images using the Hausdorf distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(9):850–863, September 1993.

[Li, 1995] S. Z. Li. *Markov Random Field Modeling in Computer Vision.* Springer-Verlag, 1995.

[Olson and Huttenlocher, 1997] Clark F. Olson and Daniel P. Huttenlocher. An object recognition system for complex imagery that models the probability of a false positive. *IEEE transactions on Image processing*, (6):103–113, 1997.

[Olson, 1998] Clark F. Olson. A probabilistic formulation for Hausdorff matching. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 150–156, 1998.

[Pope and Lowe, 1996] Arthur Pope and David G. Lowe. Learning probabilistic appearance models for object recognition. In Shree K. Nayar and Tomaso Poggio, editors, *Early Visual Learning*, pages 67–98. Oxford University Press, 1996.

[Rucklidge, 1996] William Rucklidge. *Efficient Visual Recognition Using the Hausdorff Distance.* Number 1173 in Lecture Notes in Computer Vision. Springer-Verlag, 1996.

[Subrahmonia *et al.*, 1996] Jayashree Subrahmonia, David B. Cooper, and Daniel Keren. Practical reliable bayesian recognition of 2D and 3D objects using implicit polynomials and algebraic invariants. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(5):505–519, May 1996.