The Optimal Distance Measure for Object Detection

Shyjan Mahamud Martial Hebert
Dept. of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

We develop a multi-class object detection framework whose core component is a nearest neighbor search over object part classes. The performance of the overall system is critically dependent on the distance measure used in the nearest neighbor search. A distance measure that minimizes the mis-classification risk for the 1*-nearest neighbor search can be shown to be the probability that a pair of input image measurements belong to different classes. In practice, we model the optimal distance measure using a linear logistic model that combines the discriminative powers of more elementary distance measures associated with a collection of simple to construct feature spaces like color, texture and local shape properties. Furthermore, in order to perform search over large training sets efficiently, the same framework was extended to find hamming distance measures associated with simple discriminators. By combining this discrete distance model with the continuous model, we obtain a hierarchical distance model that is both fast and accurate. Finally, the nearest neighbor search over object part classes was integrated into a whole object detection system and evaluated against an indoor detection task yielding good results.*

1 Introduction

The reliable detection of an object of interest in an input image with arbitrary background clutter and occlusion has to a large extent remained an elusive goal in computer vision since the beginning. In a multi-class object detection task, we would like to detect the presence or absence of an object of interest in an input image, given a prior training set (2D or 3D data) for the objects of interest. The factors that confound reliable detection include background clutter, occlusion of the objects of interest and the variability in viewing conditions. See Figures 1 and 2 for the objects of interest and sample test images for an indoor detection task.

Previous approaches to object detection can be grouped under two main categories: (a) exemplar based and (b) non-exemplar based approaches. Broadly speaking, the latter

Figure 1: The 15 objects of interest for the indoor detection task.

set of approaches can be characterized by the assumptions they make about the objects being detected. For example, in model-based or generative approaches [7], a model for each object of interest is assumed, while for invariant-based approaches [15], geometric or texture based features are assumed to exist for each object that are invariant to lighting or viewpoint changes. The main difficulty in such non-exemplar based approaches is the development of good models or invariants and their estimation from training data. This is especially a problem for a general object detection task in which we are interested in detecting an arbitrary set of objects. Each object of interest might in general require different modeling assumptions.

Exemplar-based approaches [10, 11, 12, 16] on the other hand avoid making assumptions about the objects of interest and instead represent them by a training set of images of the objects under various viewing conditions and scene illumination. At run-time, a nearest neighbor (NN) search is performed over the training set and the object class label of the exemplar that best matches the input image is reported. The classification performance of the NN rule is crucially dependent on the distance measure used for finding the nearest neighbor. Consequently, it is natural to ask the following questions: (a) What is the optimal distance measure for NN search ? and (b) How do we model the optimal distance measure in practice ? Furthermore, for run-time performance we will also be interested in the following: (c) How do we perform efficient NN search ?

The rest of the paper addresses these questions as follows: In § 2, we derive the optimal distance measure that

Figure 2: Sample test images for the indoor detection task. White empty squares indicate correct detections by our system described in § 5.3, while squares with a cross indicate false positives.

minimizes the NN mis-classification risk. We then present a simple linear logistic model in § 3 that directly model the optimal distance measure in terms of more elementary distance measures defined over simple feature spaces like histograms of color, shape and texture. We extend this model in § 4 to learn weighted hamming distance measures associated with a set of discriminators. This is used in a heirarchical distance measure for object detection that is both fast and accurate in practice. Section 5 describes the details of a practical system for object detection under occlusion and clutter whose core component is the nearest neighbor search over object part classes. Finally, we evaluate our detection scheme in an indoor detection task in § 6.

2 Optimal NN Distance Measure

We assume that we have a training set $S_n = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$ where each tuple (x_i, y_i) is chosen i.i.d. from some unknown distribution $p(x, y)$ over $X \times Y$ where X is the space of image measurements and Y is some discrete set of object class labels. A measurement is the representation of the image in terms of a set of features like color, shape or texture. We are also given a distance measure $d : X \times X \rightarrow \mathbb{R}$ between any two image measurements.

On input measurement $x \in X$, the 1-*nearest neighbor rule* reports the class label y' associated with the training image $x' \in S_n$ that is closest to x according to the distance measure d. The n-sample NN mis-classification risk $R(n)$ is defined as:

$$R(n) \equiv \mathrm{E}_{(x,y),S_n}[L(y, y')] \quad (1)$$

where L is the 0-1 loss given by $L(y, y') = 1$ if $y \neq y'$ and 0 otherwise. Note that the risk is averaged over all inputs x *as well as* all training sets of size n. The large sample or asymptotic risk is then defined as $R \equiv \lim_{n\rightarrow\infty} R(n)$.

Conditioning on input x, the risk can be re-written as:

$$\begin{aligned} R(n) &\equiv \mathrm{E}_{x,X_n}[r(x, x')] \\ r(x, x') &\equiv \mathrm{E}_{y,y'}[L(y, y')|x, x'] \\ &= p(y \neq y'|x, x') \end{aligned} \quad (2)$$

where $r(x, x')$ is the conditional risk on an input x and X_n is the set of training measurements x_i from S_n. For any given training set size of n, the risk $R(n)$ depends only on the distance measure d used for the nearest neighbor search. Thus, it is natural to ask for the distance measure that minimizes the risk.

Since the conditional risk $r(x_i, x_j) = p(y_i \neq y_j|x_i, x_j)$ is itself a measure defined over any two input measurements $x_i, x_j \in X$, we can consider using it as a candidate distance measure. Under this distance measure, two images are "closer" to each other if they are both likely to come from the same class. Thus intuitively at least, the conditional risk seems like a good distance measure to use. We can in fact easily show that this distance measure minimizes the NN risk.

For a given input x and training set S_n, using $r(\cdot, \cdot)$ as the distance measure gives the training example x' that minimizes the conditional risk over the training set S_n since by construction the distance measure used is also the conditional risk and thus finding $x' \in S_n$ that minimizes the distance measure also minimizes the conditional risk. Since the conditional risk $r(x, x')$ is minimized for any input x by the chosen distance measure, the unconditional risk $R(n)$ is also minimized. We have thus shown the following :

Theorem 1 *The distance measure* $d(x_i, x_j) \equiv p(y_i \neq y_j|x_i, x_j)$ *minimizes the risk* $R(n)$ *for any* n.

We now list a few important properties of the optimal distance measure without proof, for details see [1]:

The optimal distance measure is not a metric distance. In particular it does not satisfy the axiom of self-similarity: $d(x_i, x_j) \geq 0$ with equality iff $x_i = x_j$. Lack of self-similarity is a direct consequence of the lack of complete certainty for the class membership for any given measurement as will be the case for most real tasks. Somewhat surprisingly however, the optimal distance measure does satisfy the triangle inequality which is useful for some applications like efficient image retrieval [3]. Most prior work [8] on the other hand have studied the use of optimal *metric* distance measures primarily due to strong asymptotic results for classification performance for any metric distance.

Classification performance. It can be shown that the misclassification risk R (in the limit as training set size $n \to \infty$) for the 1-NN rule when using the optimal distance measure is no worse than the risk R^M when using any metric distance measure and in general can be better. In fact, depending on the task, the risk can approach even the bayes optimal risk R^B:

$$R^B \leq R \leq R^M \tag{3}$$

where the lower bound is tight.

3 Modeling the Optimal Distance

Under the i.i.d. assumption the optimal distance measure $p(y_i \neq y_j | x_i, x_j)$ can be expressed in terms of generative models $p(x|y)$ for each class as follows: [1]

$$p(y_i \neq y_j \mid x_i, x_j) = \sum_y p(y|x_i)(1 - p(y|x_j)) \tag{4}$$

Thus one approach [6] is to first estimate a generative model $p(x|y)$ for each class from training data and then construct the optimal distance measure using the expression above. The Achilles' heel of such an approach is the need to reliably estimate generative models from data. We argue that such an approach is flawed on two counts especially for a multi-class object detection task. First, if we can estimate generative models reliably from data, then we should get better classification performance using the Bayes' decision rule directly (see the inequality 3). More likely, estimating generative models from data may not be reliable since a good model may require the estimation of many parameters, even though most of which may be irrelevant to the task of discriminating one object from another. Secondly, in the context of a multi-class object detection system, coming up with a generative model is likely to be difficult in practice since it entails making modeling assumptions which are not obvious for an arbitrary collection of objects of interest. In fact, the reason for adopting the nearest neighbor framework is to avoid making any assumptions about the objects of interest.

[1] The posteriors $p(y|x)$ are obtained from $p(x|y)$ and the priors $p(y)$ using Bayes rule

Our approach instead will be to model the optimal distance *directly* in terms of more elementary distance measures defined on simple to construct feature spaces like color, texture and local shape properties. Several discriminative simple features for objects have been well-studied in the literature. Examples of feature spaces include normalized pixel intensities [11], edge maps [9] and shape contexts [5]. Each of these feature spaces are associated with elementary distance measures for comparing two measurements, examples include Euclidean distance, the χ^2 or L_1 distance for histograms and the Hausdorff distance measure [9]. Our motivation for using such simple features are because they are easy and efficient to implement. Thus from a practical point of view, we seek to model the optimal distance measure by combining such elementary distance measures defined over simple feature spaces.

For simplicitly of implementation, we consider a linear logistic model for combining the elementary distance measures for approximating the optimal distance measure. Formally, let $\mathcal{C} = \{d_1, d_2, \ldots, d_N\}$ be a possibly large collection of elementary distance measures, each of which is associated with some simple feature space. We wish to select K elementary distance measures $d_k \in \mathcal{C}$ from this collection that best approximate the optimal distance measure using the following linear logistic model:

$$\log \frac{p(y_i \neq y_j | x_i, x_j)}{p(y_i = y_j | x_i, x_j)} \approx \alpha_0 + \sum_k^K \alpha_k d_k(x_i, x_j) \tag{5}$$

where $\boldsymbol{\alpha} = \{\alpha_0, \ldots, \alpha_K\}$ is a set of linear combining coefficients.

Let y_{ij} be a binary variable taking the value -1 if $y_i = y_j$ and $+1$ otherwise. Then it can be seen that by inverting the transform above we get:

$$p(y_{ij} | x_i, x_j) \approx \sigma \left(\sum_{k=0}^K \alpha_k y_{ij} d_k(x_i, x_j) \right) \tag{6}$$

where $\sigma(u) = 1/(1 + e^{-u})$ is the sigmoid function and where for compactness of notation we have assumed the inclusion of a constant distance measure $d_0 \equiv 1$ corresponding to α_0.

In practice, we need to estimate the best model for the optimal distance measure from training data. We can use the maximum likelihood framework for the estimation as follows. As before, let $S = \{(x_1, y_1), \ldots, (x_N, y_N)\}$ be the training set of image measurements and corresponding class labels. Let $\mathbf{d} = \{d_0, \ldots, d_K\}$ be a particular selection of elementary distance measures from $\mathcal{C}$. The log-likelihood

$l(\boldsymbol{\alpha}, \mathbf{d}|S)$ for a particular model for the optimal distance measure that is parametrized by $\boldsymbol{\alpha}$ and $\mathbf{d}$ given the training data S is defined as:

$$l(\boldsymbol{\alpha}, \mathbf{d}|S) \equiv \sum_{i,j}^{N} \log p(y_{ij}|x_i, x_j) \quad (7)$$

For a given choice for $\mathbf{d}$, the optimal value for the combining coefficients $\boldsymbol{\alpha}$ under the maximum likelihood framework is that which maximizes the likelihood. Substituting the model (6), maximizing the likelihood above amounts to minimizing the following cost function:

$$J_{\mathbf{d}}(\boldsymbol{\alpha}) \equiv \sum_{i,j}^{N} \log\left(1 + e^{-\sum_k \alpha_k y_{ij} d_k(x_i, x_j)}\right) \quad (8)$$

This cost function is convex [1] and can be optimized using standard iterative techniques like Newton's method [14].

Finally, the best choice for $\mathbf{d}$ is the one that maximizes the likelihood or equivalently minimizes $J_{\mathbf{d}}$ over all choices of K distance measures from the collection $\mathcal{C}$. The brute-force search over all choices is clearly infeasible when K is large. Instead we adopt a simple greedy strategy in which at each iteration k we choose the best $d_k \in \mathcal{C}$ that along with the distance measures $\{d_1, \ldots, d_{k-1}\}$ chosen in the previous iterations minimizes the cost function.

4 Efficient NN Search

In practice, given an input measurement x we need to search the training set S efficiently for the nearest neighbor x'. The basic idea behind most previous attempts [4, 13] to make NN search efficient is to (possibly recursively) *partition* the measurement space X. For example, in Kd-trees [4], each node of the tree recursively partitions X based on the component of the measurement with maximum variance. However, Kd-trees are not appropriate in our case since the image measurement will be composed of measurements from different feature types like color, texture and shape. It does not make sense to compare variances of measurements from different feature spaces as required for the construction of Kd-trees.

In [13], the space of measurements is partitioned by a collection of random hash functions. Our strategy is similar in spirit, but instead uses a collection of discriminators each of which is constructed in some simple feature space. Furthermore, the choice of discriminators is not random. As we shall see, our motivation for using such a scheme is so that we can re-use the framework presented above for finding a continuous model for the optimal distance measure to also construct a distance measure for performing efficient NN search.

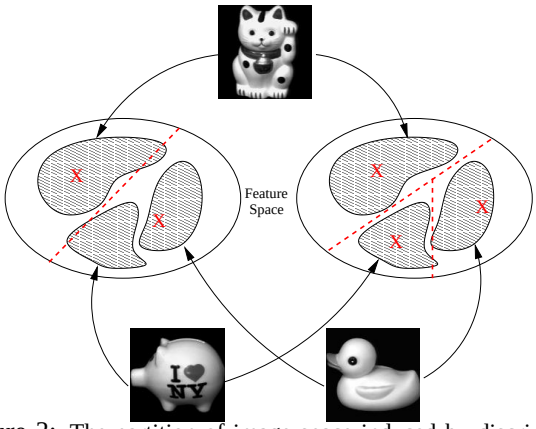

Figure 3: The partition of image space induced by discriminators. Three classes of objects are shown within the image space (depicted as an ellipse). The particular type of discriminators illustrated here are the nearest prototype discriminators described in § 4.2 constructed in some feature space. The discriminator on the left is a nearest 2-prototype discriminator, the prototypes are marked by ×'s. The discriminator on the right has 3 prototypes. The partition boundaries in each case is given by the voronoi diagram induced by the prototypes which are at the center of each cell.

Any discriminator can be characterized by the partition in measurement space X that it induces. For example, a simple discriminator might test whether the average intensity or some other simple statistic of the input image crosses a threshold, in which case the the measurement space X is split into two parts. A decision tree on the other hand partitions the measurement space into many parts, where each part corresponds to a leaf node of the decision tree. Another type of discriminator which we use in our work due to its ease of implementation and wide applicability is the nearest prototype discriminator that is described later in § 4.2. Figure 3 illustrates the partition induced by discriminators.

Formally, let the discriminator h induce the partition $X = X_1 \cup X_2 \cup \ldots \cup X_n, X_i \cap X_j = \emptyset, i \neq j$. On input x, let $h(x)$ denote the partition X_i that x falls under. Given a set of such discriminators $\{h_1, h_2, \ldots, h_K\}$, an input measurement x has a "code" $\{h_1(x), h_2(x), \ldots, h_K(x)\}$ in terms of the partitions that x falls under for each discriminator h_k. Thus a set of discriminators partitions the measurement space X, where each partition corresponds to a unique code.

Given two measurements x_i and x_j, a distance measure between the corresponding codes is given by the hamming distance. More generally, we consider a weighted hamming distance:

$$H(x_i, x_j) = \alpha_0 + \sum_k \alpha_k [h_k(x_i) = h_k(x_j)]$$

where $[h_k(x_i) = h_k(x_j)]$ is the one-dimensional hamming distance for each discriminator defined by:

$$[h_k(x_i) = h_k(x_j)] \equiv \begin{cases} -1 & \text{if } h_k(x_i) = h_k(x_j) \\ +1 & \text{otherwise} \end{cases}$$

A good hamming distance measure can be used to efficiently search over the training set as follows. As noted before, most approaches for performing efficient NN search works by effectively partitioning the image measurement space X. In our scheme, the set of chosen discriminators $\{h_1, \ldots, h_K\}$ partitions X where each partition corresponds to a unique code in terms of the set of discriminator outputs. The ideal code is that for which separates measurements from different classes into different partitions. To access these partitions efficiently, we use a hash-table where on input x, the hash function accesses the bucket corresponding to the code $\{h_1(x), h_2(x), \ldots, h_K(x)\}$. At training time, in each bucket we store all the training measurements that maps to the bucket, which are returned at run-time.

We now describe how to construct optimal hamming distance. The optimal distance measure is the one that minimizes the mis-classification risk. Thus we can use the maximum likelihood framework presented in § 3 for finding the hamming distance measure that best approximates the optimal NN distance measure. Formally, let $\mathcal{H} = \{h_1, h_2, \ldots\}$ be a (possibly large) collection of discriminators. For ease of implementation, each of these discriminators are constructed in some simple feature space like color, shape or texture as described in more detail later in § 4.2. Corresponding to $\mathcal{H}$, we have the collection of elementary one-dimensional hamming distance measures (4) $\mathcal{C} = \{[h(x_i) = h(x_j)] \mid h \in \mathcal{H}\}$. Similar to the case for estimating the best continuous model for the optimal distance measure in § 3, we select the best K best discriminators $h_k \in \mathcal{H}$ in a greedy manner that gives the hamming distance that best approximate the optimal distance measure. The corresponding cost function to be minimized is then given by:

$$J \equiv \sum_{i,j}^{N} \log\left(1 + e^{-\sum_k \alpha_k y_{ij}[h_k(x_i)=h_k(x_j)]}\right) \quad (9)$$

(compare with (8)).

4.1 Hierarchical Distance Measure

Although the hamming distance measure can be used for efficient NN search, it cannot be expected to be accurate in terms of returning the true nearest neighbor since it is a discretized distance measure. To overcome this shortcoming, we propose the use of a simple heirarchical distance measure that combines the search efficiency when using the discrete hamming distance with the accuracy of the continuous distance measure described in § 3. The scheme is explained in Figure 4.

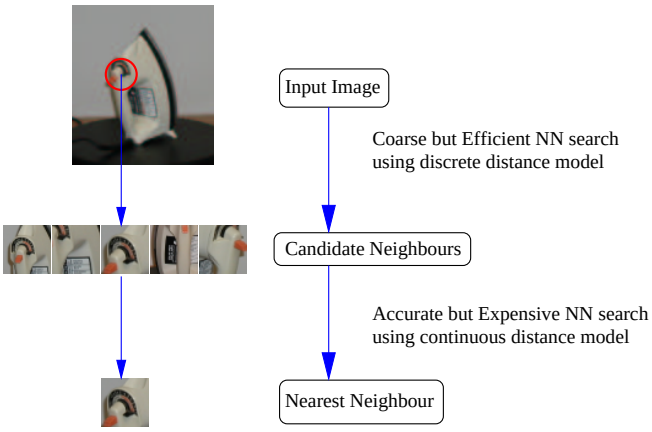

Figure 4: Our scheme for efficient and accurate nearest neighbor search. An input measurement is matched against the training set using the coarse but efficient hamming distance measure discussed in § 4, yielding a small list of candidate nearest neighbors, rather than just the nearest neighbor. These candidate neighbors are then searched for the closest neighbor using the more accurate continuous model for the optimal distance measure discussed in § 3. On the left is shown an actual example from our experiments reported in § 6). The nearest neighbors shown are for the patch from the input that is circled.

4.2 Constructing Candidate Discriminators

We conclude this section by specifying how we generate the collection of discriminators $\mathcal{H}$ from which the best K discriminators h_k are chosen. For ease of implementation and wide applicability, the type of discriminators we choose are what we call the nearest prototype discriminators constructed in simple feature spaces like color, texture and shape. Such a discriminator is completely specified by a set of prototypes in some feature space. Figure 3 illustrates such discriminators. The image space is partitioned by the set of prototypes where each partition corresponds to the subset of the image space that is closest to one of the prototypes. The distance used for the construction is any elementary distance measure associated with the feature space. Such discriminators are similar in spirit to vector quantization in signal processing and have been used earlier for object detection in [2].

In our work we use discriminators with at most 3 prototypes. Ideally, we should consider all possible nearest prototype discriminators that we can construct. However this is infeasible in practice. Instead we sample the location of the prototypes from actual training data. Such a sampling scheme is sufficient since we do not require great accuracy for the resulting hamming distance measure due to the use of the hierarchical distance measure discussed above.

5 Implementation

We have thus far only discussed the issue of using the optimal distance measure for nearest neighbor search for object detection. In practice, there are several other issues that need to be addressed when using a nearest neighbor search framework in the context of an overall scheme for object detection. Since the main focus of this paper is on developing and using an optimal distance measure for object detection, for the rest of the object detection system, we will seek the simplest implementation that we can get away with, but yet which is sufficient and realistic enough for evaluating the distance measures that we develop.

The rest of the section describes (a) representing objects in terms of a few discriminative parts, (b) the feature spaces we use to represent image measurements and (c) the whole object detection scheme.

5.1 Representation in terms of Parts

In practice, the objects that we are interested in detecting can be of varying sizes and shapes. The naive approach of performing a nearest neighbor search at each location over a training set with whole object views will result in poor performance since no single choice for the size of the support window can be expected to be optimal for all objects. Instead we represent each object training view in terms of a few discriminative parts, each of which has a support window of 32×32 pixels in our work. The nearest neighbor search is then performed over *part classes* rather than whole object views. Conceptually, a part class corresponds to image measurements of some surface patch of an object of interest, taken under differing viewpoints and lighting conditions, just as in the case for whole object classes.

For run-time considerations, we represent a training object view using the most discriminative parts (10 in our experiments). The discriminative power of a part is defined as follows: let z be a candidate part patch, i.e. a 32×32 patch from some training view of an object and let Z be a random sample of part patches that do not belong to the same object class as z, as well as random patches from background clutter which for the current purpose is considered a pseudo-class. Then a natural measure for the discriminative power for part z is the log-likelihood $l(z, Z)$ that z and any part $z' \in Z$ belong to different classes:

$$l(z, Z) \equiv \sum_{z' \in Z} \log p(y \neq y' | z, z') \qquad (10)$$

where y and y' are corresponding part class labels. Here $p(y \neq y'|z, z')$ is the optimal distance measure for part classes. It is estimated using the maximum likelihood framework presented in § 3 using a random training sample of part patches from all whole object training views.

Figure 5: The top discriminative part patches selected for sample training images.

Each selected part patch from an object view is a representative of some part class, that models the image views of the underlying surface patch of the object. We collect additional training images for each part class as follows. Training images under variations in translation, in-plane rotation and scale can be collected from the original whole object training view from which the part patch was chosen. For our experiments we sample translations of ± 4 pixels along both axis, rotations of $\pm 10^\circ$ and scale variations of 0.9 and 1.1. Ideally, we would also like to sample rotations in depth. However, this requires finding part correspondences in other whole object training views. Solving such a correspondence problem is error-prone in practice. For simplicity of implementation, we model parts in different whole object views independently. This implies that the same underlying surface patch of an object might be represented redundantly by parts in multiple whole object views.

Figure 5 shows the discriminative part patches selected for sample training images. Once the parts are selected for all training views and additional training images for the corresponding part classes are sampled as described above, a new optimal distance measure for NN search is estimated from these part classes.

5.2 Feature Spaces

As discussed in § 3, we approximate the optimal distance measure by a linear combination of elementary distance measures in simple feature spaces based on color, shape or texture. We now describe the details of the types of feature spaces that we use in our experiments.

The histogram of various image feature types is a widely used feature space in computer vision [16, 18, 17]. Histograms are popular in the computer vision literature since they are efficient to create from an input image by making one sweep across the image from top to bottom and left to right, as well as being robust to a fair amount of geometric transformations [16, 18].

For additional discriminative power, we also coarsely discretize the spatial location of the feature. This is similar in spirit to the work on shape context [5]. In our work, we discretize each coordinate axis into two levels within the 32×32 pixel window of support (the size of a part) centered around the point of interest in the input image.

We conclude by listing all of the specific feature types that we use in our work:

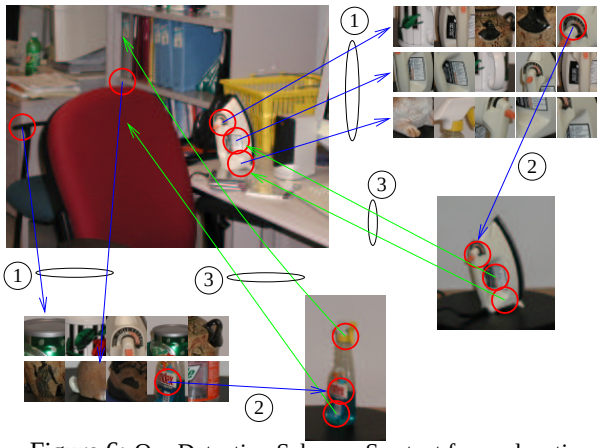

Figure 6: Our Detection Scheme. See text for explanation.

Color Three single dimensional feature spaces are considered corresponding to the red, green and blue bands, each of which is normalized illumination invariance.

Texture Characterized by Gaussian derivative filter responses [16] along the two coordinate axis with the width of the Gaussian set to $\sigma = 2.0$ pixels. Additionally, we also use the magnitude of the derivative. Again, the each of these bands are normalized for illumination invariance.

Local Shape Two local shape properties are constructed from the contours detected by the Canny edge detector: (a) the orientation of the edges that fall within the support window and (b) the local curvature along the contours that fall within the support window.

Thus we have a total of 8 one-dimensional feature spaces (3 for color, 3 for texture, and 2 for shape) which are combined to approximate the optimal distance measure. The elementary distance measure that we use for comparing histograms in each feature space is the L_1 norm.

5.3 Detection System

The detection system that we have built has a nearest neighbor search over part classes as its core component. Figure 6 walks through the following steps in our detection pipeline using an actual test input (the step number here and in the figure correspond): (1) After pre-processing the image to extract histograms of various features, the NN parts from the training set are determined at each sampled location across the image and at two scales using the hierarchical distance measure (§ 4.1). Shown here are the top 5 parts for a few locations. (2) Each NN part detected forms an object view hypothesis corresponding to the training view that the part came from. (3) The locations of the other parts in the training view for each hypothesis is determined and the corresponding parts are searched around the expected location in the input image. The hypothesis is scored by accumulating the NN scores (distance measure between training part and input patch) of these parts along with the NN score for the part that generated the hypothesis. Shown here are 2 object view hypotheses formed from parts detected at two locations. In the actual system, each part detected at each location forms a hypothesis, each of which is scored. Finally, object detections are reported after thresholding the score with a value θ for each hypotheses and performing local non-maximal suppression.

6 Experiments

The detection scheme was tested on a collection of everyday objects of interest in an indoor environment under clutter and occlusion. Figure 1 shows a collection of 15 objects of interest. Training images for each object were taken at two elevations that were 10° apart and which were close to the height of a person at a distance of approximately 7 ft from the object. At each elevation, training images were taken over a 180° sweep horizontally around the object at intervals of 20°. As described in § 5.1, up to 10 discriminative part patches are selected in each training image, for each of which training views for the corresponding part class are sampled synthetically from the whole object training image at different scales and rotations (see § 5.1). The training images were taken under illumination conditions that were natural and kept constant for an indoor setting. Rather than collecting more training images under varying illumination conditions, we chose to rely on normalizing the various feature spaces as described in § 5.2. This was found to be sufficient in compensating for the moderate amount of illumination variation encountered in typical indoor settings.

Testing images were collected under a large number of backgrounds with varying viewpoint and scale changes for the objects of interest along with some occlusion and variations in lighting. 25 images for each object of interest were taken for a total of 375 test images. Figure 2 shows sample test images.

For the indoor discrimination task, a hamming distance measure used in the hierarchical distance measure was constructed with 80 discriminators as detailed in § 4. The number of nearest neighbors returned by the first stage of the hierarchical distance measure that uses this hamming distance was set to 10.

See § 5.3 to review the details of our detection scheme. Figure 7 shows various ROC curves for the detection scheme plotting the detection performance as the threshold θ used for pruning each hypothesis score is varied. The main ROC curve corresponds to the case when we use the hierarchical distance measure for the NN part search. As a representative point, we get a detection rate of 78% corresponding to a false positive rate of 0.5 per test image.

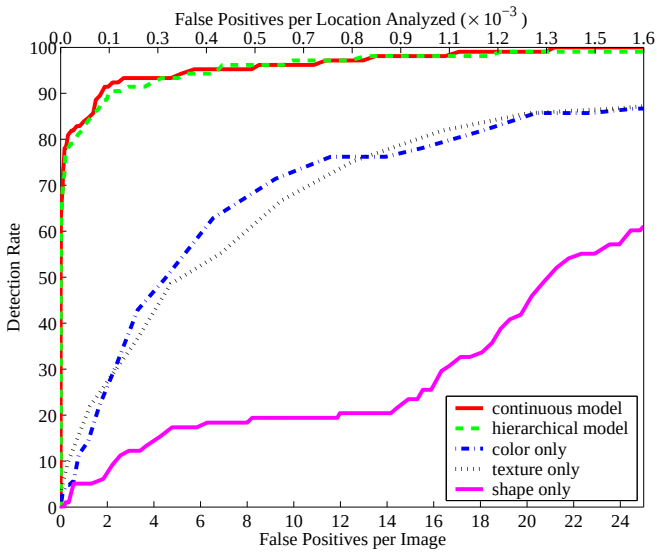

Figure 7: ROC plots for the indoor detection task. See text for explanation.

For comparison, we also show the performance when using just the more accurate continuous optimal distance model (which forms the second stage of the hierarchical distance measure) in a naive brute-force NN search over the training set. As a representative point, we get a detection rate of 82% corresponding to a false positive rate of 0.5 per test image (compare with the representative point above). As can be seen, there is little degradation in detection performance when using the hierarchical distance measure. On the hand, there is an order of magnitude difference in runtime performance. On a 1.5 GHz CPU x86 machine, it took ≈ 40 seconds when using the hierarchical distance measure compared with more than 13 minutes when using just the continuous distance measure, giving a speed-up of around 20. The implementation was done in OCAML, a high-level functional language.

Also shown in Figure 7 are the relative performance of the various feature types when used in isolation. Note that each feature type is comprised of more than one feature space (3 for color, 3 for texture and 2 for local shape, see § 5.2). All of the feature spaces comprising a given feature type are used when that feature type is tested in isolation. For our implementation of these feature types, both color and texture are quite discriminative on their own, while local shape is the least discriminative. However, all of these detection rates are far lower than the rate obtained when using all the feature types together. For a false positive rate of 0.5 per test image, each of the features in isolation gives a detection rate $< 15\%$ Thus we see that the various feature types complement each other to a substantial degree when used together, especially at operating points with low false positive rates, which is precisely the region that is most useful in practice.

7 Conclusion

In this thesis, we derived and modeled the optimal distance measure for use in a nearest neighbor framework for object detection. The optimal distance measure was modeled directly by a linear logistic model that combined more elementary distance measures associated with simple feature spaces.

In this paper, the distance models that we considered were all global models, that is the distance score output by these models did not depend on where in measurement space they were used. One promising avenue for future work is to investigate adapting distance models locally, say one for each part class, for better performance.

References

[1] Same Author. PhD thesis, 2002.

[2] Same Author. In *Conference paper*, 2002.

[3] Julio E. Barros, James C. French, Worthy N. Martin, Patrick M. Kelly, and T. Michael Cannon. Using the triangle inequality to reduce the number of comparisons required for similarity-based retrieval. In *Storage and Retrieval for Image and Video Databases (SPIE)*, pages 392–403, 1996.

[4] J. Beis and D. Lowe. Shape indexing using approximate nearest-neighbor search in highdimensional spaces. In *CVPR*, pages 1000–1006, 1997.

[5] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *PAMI*, 24(4):509–522, April 2002.

[6] Enrico Blanzieri and Francesco Ricci. A minimum risk metric for nearest neighbor classification. In *Proc. 16th International Conf. on Machine Learning*, pages 22–31. Morgan Kaufmann, San Francisco, CA, 1999.

[7] R.T. Chin and C.R. Dyer. Model-based recognition in robot vision. *Surveys*, 18(1):67–108, March 1986.

[8] B.V. Dasarathy. *Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques*. Computer Society Press, 1991.

[9] D.P. Huttenlocher, G.A. Klanderman, and W.J. Rucklidge. Comparing images using the hausdorff distance. *PAMI*, 15(9):850–863, September 1993.

[10] B.W. Mel. Seemore: Combining color, shape, and texture histogramming in a neurally inspired approach to visual object recognition. *NeurComp*, 9(4):777–804, May 1997.

[11] S.K. Nayar, S.A. Nene, and H. Murase. Real-time 100 object recognition system. In *ARPA96*, pages 1223–1228, 1996.

[12] R.C. Nelson and A. Selinger. A cubist approach to object recognition. In *ICCV98*, pages 614–621, 1998.

[13] R. Motwani P. Indyk. Approximate nearest neighbors: Towards removing the curse of dimensionality. In *STOC*, pages 604–613, 1998.

[14] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in C*. Cambridge University Press, 1992.

[15] M.A. Rodrigues. Special issue: Invariants for pattern recognition and classification. *PRAI*, 13(8):1103, December 1999.

[16] B. Schiele. *Object Recognition using Multidimensional Receptive Field Histograms*. PhD thesis, I.N.P. Grenoble, 1997.

[17] H. Schneiderman. *A Statistical Approach to 3D Object Detection Applied to Faces and Cars*. PhD thesis, Robotics Institute, Carnegie Mellon University, 2000.

[18] M.J. Swain and D.H. Ballard. Color indexing. *IJCV*, 7(1):11–32, November 1991.