

# A Coarse-to-Fine Strategy for Multiclass Shape Detection

Yali Amit, Donald Geman, and Xiaodong Fan

**Abstract**—Multiclass shape detection, in the sense of recognizing and localizing instances from multiple shape classes, is formulated as a two-step process in which local indexing primes global interpretation. During indexing a list of instantiations (shape identities and poses) is compiled, constrained only by no missed detections at the expense of false positives. Global information, such as expected relationships among poses, is incorporated afterward to remove ambiguities. This division is motivated by computational efficiency. In addition, indexing itself is organized as a coarse-to-fine search simultaneously in class and pose. This search can be interpreted as successive approximations to likelihood ratio tests arising from a simple (“naive Bayes”) statistical model for the edge maps extracted from the original images. The key to constructing efficient “hypothesis tests” for multiple classes and poses is local ORing; in particular, *spread edges* provide imprecise but common and locally invariant features. Natural tradeoffs then emerge between discrimination and the pattern of spreading. These are analyzed mathematically within the model-based framework and the whole procedure is illustrated by experiments in reading license plates.

**Index Terms**—Shape detection, multiple classes, statistical model, spread edges, coarse-to-fine search, online competition.

## 1 INTRODUCTION

WE consider detecting and localizing shapes in cluttered gray-level images when the shapes may appear in many poses and there are many classes of interest. In many applications, a mere list of shape instantiations, where each item indicates the generic class and approximate pose, provides a useful global description of the image. (Richer descriptions, involving higher-level labels, occlusion patterns, etc., are sometimes desired.) The set of feasible lists may be restricted by global, structural constraints involving the joint configuration of poses; this is the situation in our application to reading license plates.

In this paper, *indexing* will refer to noncontextual detection in the sense of compiling a list of shape instantiations independently of any global constraints; *interpretation* will refer to incorporating any such constraints, i.e., relationships among instantiations. In our approach, indexing primes interpretation.

### 1.1 Indexing

Ideally, we expect to detect *all* instances from all the classes of interest under a wide range of geometric presentations and imaging conditions (resolution, lighting, background, etc.). This can be difficult even for one generic class without accepting false positives. For instance, all approaches to face detection (e.g., [8], [31], [36]) must confront the expected

variations in the position, scale, and tilt of a face, varying angles of illumination, and the presence of complex backgrounds; despite considerable activity and marked advances in speed and learning, no approach achieves a negligible false positive rate on complex scenes without missing faces. With multiple shape classes, an additional level of complexity is introduced and subtle confusions between classes must be resolved in addition to false positives due to background clutter.

*Invariant indexing*, or simply *invariance*, will mean a null false negative rate during indexing, i.e., the list of reported instantiations is certain to contain the actual ones. *Discrimination* will refer to false positive error—the extent to which we fantasize in our zeal to find everything. We regard invariance as a hard constraint. Generally, parameters of an algorithm can be adjusted to achieve near-invariance at the expense of discrimination. The important tradeoff is then *discrimination versus computation*.

Hypothetically, one could achieve invariance *and* high discrimination by looking separately for every class at every possible pose (“templates for everything”). Needless to say, with a large number of possible class/pose pairs, this would be extremely costly, and massive parallelism is not the answer. Somehow we need to look for many things at once, which seems at odds with achieving high discrimination.

Such observations lead naturally to organizing multiclass shape detection as a *coarse-to-fine (CTF) computational process*. Begin by efficiently eliminating entire subsets of class/pose pairs simultaneously (always maintaining invariance) at the expense of relatively low discrimination. From the point of view of computation, rejecting many explanations at once with a single, relatively inexpensive “test” is clearly efficient; after all, given an arbitrary subimage, the most likely hypothesis by far is “no shape of interest” or “background” and initially testing against this allows for early average termination of the search. If “background” is not declared, proceed to smaller class/pose subsets at higher levels of discrimination and, finally, entertain highly discriminating

- Y. Amit is with the Department of Statistics and the Department of Computer Science, University of Chicago, Chicago, IL, 60637. E-mail: amit@marx.uchicago.edu.
- D. Geman is with the Department of Applied Mathematics and Statistics, and the Whitaker Biomedical Engineering Institute, The Johns Hopkins University, Baltimore, MD 21218. E-mail: geman@jhu.edu.
- X. Fan is with the Department of Electrical and Computer Engineering, The Johns Hopkins University, Baltimore, MD 21218. E-mail: xdfan@cis.jhu.edu.

Manuscript received 18 Aug. 2003; revised 9 Apr. 2004; accepted 26 Apr. 2004. Recommended for acceptance by D. Forsyth.  
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-0234-0803.

procedures but dedicated to specific classes and poses. Accumulated false positives are eventually removed by more intense, but focused, processing. In this way, the issue of computation strongly influences the very development of the algorithms, rather than being an afterthought.

A natural control parameter for balancing discrimination and computation is the degree of invariance of *local* features, not in the sense of fine shape attributes, such as geometric singularities of curves and surfaces, but rather coarse, generic features which are common in a set of class/pose pairs. “Spread features” ([1], [3], [8]) provide a simple example: A local feature is said to be detected at a given location if the response of the feature detector is strong enough anywhere nearby. The larger the spreading (degree of local ORing), the higher the incidence on any given ensemble of classes and poses and checking for a certain number of distinguished spread features provides a simple, computationally efficient test for the ensemble. During the computational process, the amount of spreading is successively diminished.

## 1.2 Interpretation

The outcome of indexing is a collection of instantiations—class/pose pairs. No contextual information, such as structural or semantic constraints, has been employed. In particular, some instantiations may be inconsistent with prior information about the scene layout. Moreover, several classes will often be detected at roughly the same location due to the insistence on minimizing false negatives. In this paper, the passage from indexing to interpretation is largely based on taking into account prior knowledge about the number of shapes and the manner in which they are spatially arranged. Assuming shapes do not overlap, a key component of this analysis is a competition among shapes or sequences of shapes covering the same image region, for which we employ a likelihood ratio test motivated by our statistical model for local features. Since a relatively *small number* of candidate instantiations are ever involved, it is also computationally feasible to bring finer features into play, as well as template-matching, contextual disambiguation, and other intensive procedures.

## 1.3 New Directions

We explore three new directions:

- **Multiple Shape Classes.** Our previous work concerned coarse-to-fine (CTF) representations and search strategies for a single shape or object class and, hence, was based entirely on pose aggregation. We extend this to hierarchies based on recursively partitioning *both class and pose*.
- **Contextual Analysis.** With multiple classes, testing one specific (partial) interpretation against another is eventually unavoidable, which means we need efficient, *online* tests for competing hypotheses. In particular, we derive online tests based on local features for resolving one specific hypothesis (a character at a given pose) against another.
- **Model-Based Framework.** We introduce a statistical model for the local features which provides a unifying framework for the algorithms employed in all stages of the analysis and which allows us to mathematically analyze the role of “spread features” in balancing discrimination and computation during coarse-to-fine indexing.

These ideas are illustrated by attempting to read the characters appearing on license plates. Surprisingly, perhaps, there does not seem to be any published literature apart from patents. Several systems appear to be implemented in the US and Europe. For example, in London, cars entering the metropolitan area are identified in order to charge an entrance fee and, in France, the goal is to estimate the average driving speed between two points. We have no way to assess the performance of these implementations. Our work was motivated by the problem of identifying cars entering a parking garage for which current solutions still fall short of commercial viability, mainly due to the high level of clutter and variation in lighting. It is clear that, for any specific task, there are likely to be highly dedicated procedures for improving performance; for example, only reporting plates with identical matches on two different photos, taken at the same or different times. Our goal instead is a *generic solution* which could be easily adapted to other OCR scenarios and to other shape categories and, eventually, to three-dimensional objects. In particular, we do not use any form of traditional, bottom-up segmentation in order to identify candidate regions or jump start the recognition process. There are many well-developed techniques of this kind in the document analysis literature which are rather dedicated to specific applications; see, for example, the review [24] or the work in [19].

Related work on visual attention, CTF search, hierarchical template-matching, and local ORing is surveyed in the following section. Our formulation of multiclass shape detection is given in Section 3, followed in Section 4 by a brief overview of the computational strategy. The statistical model for the local features is described in Section 5, leading to a natural likelihood ratio test for an individual detection. Efficient indexing is the theme of Sections 6, 7, and 8. The spread local features are introduced in Section 6, including a comparison of spreading versus two natural alternatives—summing them and downsampling—and the discrimination/computation tradeoff is studied under a simple statistical model. How tests are learned from data and organized into a CTF search are discussed in Sections 7 and 8, respectively. In Section 9, we explain how an interpretation of the image is derived from the output of the indexing stage. The application to reading license plates, including the contextual analysis, is presented in Section 10 and we conclude with a discussion of our approach in Section 11.

## 2 RELATED WORK

**Focus-of-Attention.** The division of the system into indexing followed by global interpretation is motivated by computational efficiency. More generally, our indexing phase is a way of *focusing attention*, which is studied in both computer vision and in modeling biological vision. The purpose is to focus subsequent, detailed processing on a small portion of the data. Two frameworks are usually considered: task-independent, bottom-up control based on the “saliency” of visual stimuli (see, e.g., [16], [20], [28], [27]) and task-driven, top-down control (see, e.g., [3], [25], [35], [36]). Our approach is essentially top-down in that attention is determined by the shapes we search for, although the coarsest tests could be interpreted as generic saliency detectors.

**CTF Search.** CTF object recognition is scattered throughout the literature. For instance, translation-based versions can



Fig. 1. Two images of the back of a car from which the license plate is to be read.

be found in [31], [17] and work on distance matching ([32]). The version appearing in [12] prefigures our work. Related ideas on dealing with multiple objects can be found in [2]. In addition, CTF search motivated the face detection algorithm in [3] and was systematically explored in [8] based on a nested hierarchy of pose bins (and CTF in complexity within bins) and in [7] based on an abstract theoretical framework. Variations have also been proposed in [33] and [36]: Whereas most poses are explicitly visited, computational efficiency is achieved by processing which is CTF in the sense of progressively focusing on hard cases. Whatever the particular CTF mechanism, the end result is that intensive processing is restricted to a very small portion of the image data, namely, those regions containing actual objects or object-like clutter. Work on efficient “indexing” based on geometric hashing ([18]) and Hough transforms ([14], [30]) is also related.

**Context.** The issue of context is central to vision and several distinct approaches can be discerned in the literature. In ours, context refers to structural rather than semantic relationships; indexing is entirely noncontextual and is followed by global interpretation in conjunction with structural constraints. In contrast, all scene attributes are discovered simultaneously in the compositional approach ([13]), which provides a powerful method for dealing with context and occlusion, but involves formulating interpretation as global optimization, raising computational issues. Other work involves “contextual priming” ([34]) to overcome poor resolution by starting interpretation with an estimate of semantic context based on low-level features. Context can also be exploited ([6]) to provide local shape descriptors.

**Natural Vision.** There are strong connections between spreading local features and neural responses in the visual cortex. Responses to oriented edges are found primarily in V1, where so-called “simple” cells detect oriented edges at specific locations, whereas “complex” cells respond to an oriented edge anywhere in the receptive field; see [15]. In other words, local “ORing” is performed over the receptive field region and the response of a complex cell can thus be viewed as a “spread edge.” Because of the high density of edges in natural images, the extent of spreading must be limited; too much will produce responses everywhere. Neurons in higher level retinotopic layers V2 and V4 exhibit similar properties, inspiring the work in [9] and [10] about designing a neural-like architecture for recognizing patterns. In [1] and [4], the spreading of more complex features is incorporated into a neural architecture for invariant detection. An extension to continuous-valued variables can be achieved with a “MAX” operation, a generalization of ORing, as proposed in [29].

**Hierarchical Template Matching.** Recent work on hierarchical template matching using distance transforms, such as [11], is related to ours in several respects even though we are not doing template-matching per se. Local ORing as a device for gaining stability can be seen as a limiting, binary version of distance transforms such as the Chamfer distance ([5]). In addition, there is a version of CTF search in [11] (although only translation is considered based on multiple resolutions) which still has much in common with our approach, including edge features, detecting multiple objects using a class hierarchy, and imposing a running null false negative constraint. Another approach to edge-based, multiple object detection appears in [26].

**Local Features.** Finally, in connection with spreading local features, another mechanism has been proposed in [22] that allows for affine or 3D viewpoint changes or nonrigid deformations. The resulting “SIFT descriptor,” based on local histograms of gradient orientations, characterizes a neighborhood (in the Gaussian blurred image) around each individual detected key point, which is similar to “spreading” the gradients over a  $4 \times 4$  region. A detailed comparison of the performance of SIFT with other descriptors can be found in [23].

### 3 SHAPE DETECTION

Consider a single, gray-level image. In particular, there is no information from motion, depth, or color. We anticipate a large range of lighting conditions, as illustrated in Fig. 1 (see also Fig. 5), as well as a considerable range of poses at which each shape may be present. Moreover, we anticipate a complex background consisting partly of extended structures, such as clutter and nondistinguished shapes, which locally may appear indistinguishable from the shapes of interest.

Let  $I = \{I(z), z \in Z\}$  be the raw intensity data on the image lattice  $Z$ . Each shape of interest has a class  $c \in \mathcal{C}$  and each instantiation (presentation in  $I$ ) is characterized by a pose  $\theta \in \Theta$ . Broadly speaking, the pose  $\theta$  represents (“nuisance”) parameters which at least partially characterize the instantiation. For example, one component of the pose of a printed character might be the font. In some contexts, one might also consider parameters of illumination. For simplicity, however, we shall restrict our discussion to the *geometric* presentation and, specifically (in view of the experiments on license plates), to position, scale, and orientation. Much of what follows extends to affine and more general transformations; similarly, it would not be difficult to accommodate parameters such as the font of a character.

For a pose  $\theta$ , let  $z(\theta)$  be the translation,  $\sigma(\theta)$  the scale, and  $\rho(\theta)$  the rotation. Denote by  $\theta_0$  the identity pose, namely,

$z(\theta_0) = \rho(\theta_0) = 0$  and  $\sigma(\theta_0) = 1$ , and by  $R$  a reference sublattice of the full image lattice  $Z$  such that any shape at  $\theta_0$  fits inside  $R$ . For any subset  $F$  of  $Z$ , let

$$F(\theta) = \{z \in Z : \theta^{-1}z \in F\}.$$

In particular,  $R(\theta)$  is the “support” of the shape at pose  $\theta$ .

The set of possible *interpretations* for an image  $I$  is

$$\mathcal{Y} = \bigcup_{k=1}^K (\mathcal{C} \times \Theta)^k,$$

where, obviously,  $K$  represents the maximum number of shapes in any given layout. Thus, each interpretation has the form  $y = \{(c_1, \theta_1), \dots, (c_k, \theta_k)\}$ . The support of an interpretation is denoted

$$R(y) = \cup_{i=1}^k R(\theta_i).$$

We write  $y^*$  for the true interpretation and assume it is unambiguous, i.e.,  $y^* = y^*(I)$ .

Prior information will provide some constraints on the possible lists; for instance, in the case of the license plates, we know approximately how many characters there are and how they are laid out. In fact, it will be useful to consider the true interpretation to be a random variable,  $Y$ , and to suppose that knowledge about the layout is captured by a highly concentrated prior distribution on  $\mathcal{Y}$ . Most interpretations have mass zero under this distribution and many interpretations in its support, denoted by  $\mathcal{Y}_\pi = \{y \in \mathcal{Y} : \pi(y) > 0\}$ , have approximately the same mass. Indeed, for simplicity, we will assume that the prior is uniform on its support  $\mathcal{Y}_\pi$ .

## 4 OVERVIEW OF THE COMPUTATIONAL STRATEGY

What follows is a summary of the overall recognition strategy. All of the material from this point to the experiments pertains to one of four topics:

- **Statistical Modeling.** The gray-level image data  $I$  is transformed into an array of binary local features  $X(I)$  which are robust to photometric variations. For simplicity, we use eight oriented edge features (Section 5), but the entire construction can be applied to more complex features, for example, functions of the original edges (see Section 11). We introduce a likelihood model  $P(X|Y = y)$  for  $X(I)$  given an image interpretation  $Y = y$ . This model motivates the definition of an image-dependent set  $\mathcal{D}(X) \subset \mathcal{C} \times \Theta$  of detections, called an *index*, based on likelihood ratio tests. According to the invariance constraint, the tests are performed with no missed detections (i.e., null type I error), which in turn implies that  $Y \subset \mathcal{D}$  with probability one (at least in principle). However, direct computation of  $\mathcal{D}$  is highly intensive due to the loop over class/pose pairs.
- **Efficient Indexing.** The purpose of the CTF search is to accelerate the computation of  $\mathcal{D}$ . This depends on developing a “test”  $T_B$  for an entire subset  $B \subset \mathcal{C} \times \Theta$  whose complexity is of the order of the test for a single pair  $(c, \theta)$  but which nonetheless retains some discriminating power; see Section 6. The set  $\mathcal{D} \cap B$  is then found by performing  $T_B$  first and then exploring the individual hypotheses in  $B$  one-by-one only if  $T_B$  is

positive. This two-step procedure is then easily extended (in Section 8) to a full CTF search for the elements of  $\mathcal{D}$  and the computational gain provided by the CTF search can be estimated.

- **Spreading Features.** The key ingredient in the construction of  $T_B$  is the notion of a “spread feature” based on local ORing. Checking for a minimum number of spread features provides a test for the hypothesis  $Y \cap B \neq \emptyset$ . The same spread features are used for many different bins, thus precomputing them at the start yields an important computational gain. In the Appendix, the optimal domain of ORing, in terms of discrimination, is derived under the proposed statistical model and some simplifying assumptions.
- **Global Interpretation.** The final phase is choosing an estimate  $\hat{Y} \subset \mathcal{D}$ . A key step is a competition between any two interpretations  $y, y' \subset \mathcal{D}$  for which  $R(y) \sim R(y')$ , i.e., which cover the same image region. The subinterpretations must satisfy the prior constraints, namely,  $y, y' \in \mathcal{Y}_\pi$ ; see Section 9. A special case of this process is a competition between *single* detections with different classes but very similar poses. (We assume a minimum separation between shapes, in particular no occlusion.) The competitions once again involve likelihood ratio tests based on the local feature model.

## 5 DATA MODEL

We describe a statistical model for the possible appearances of a collection of shapes in an image as well as a crude model for “background,” i.e., those portions of the image which do not belong to shapes of interest.

### 5.1 Edges

The image data is transformed into arrays of binary edges, indicating the locations of a small number of *coarsely defined edge orientations*. We use the edge features defined in [3], which more or less take the local maxima of the gradient in one of four possible directions and two polarities. These edges have proven effective in our previous work on object recognition; see [2] and [8]. There is a very low threshold on the gradient; as a result, several edge orientations may be present at the same location. However, these edge features have three important advantages: They can be computed very quickly, they are robust with respect to photometric variations, and they provide the ingredients for a simple “background” model based on labeled point processes. More sophisticated edge extraction methods can be used [21], although at some computational cost. In addition, more complex features can be defined as functions of the basic edges, thus decreasing their background density and increasing their discriminatory power (see [2]) and in such a way that makes the assumed statistical models more credible. For transparency, we describe the algorithm and report experiments with the simple edge features.

Although the statistical models below are described in terms of the edges arrays, implicitly they determine a natural model for the original data, namely, uniform over intensity arrays giving rise to the same edges. Still, we shall not be further concerned with distributions directly on  $I$ .

Let  $X_\gamma(z)$  be a binary variable indicating whether or not an edge of type  $\gamma \in \Gamma$  is present at location  $z \in Z$ . The type  $\gamma$  represents the orientation and polarity. The resulting family

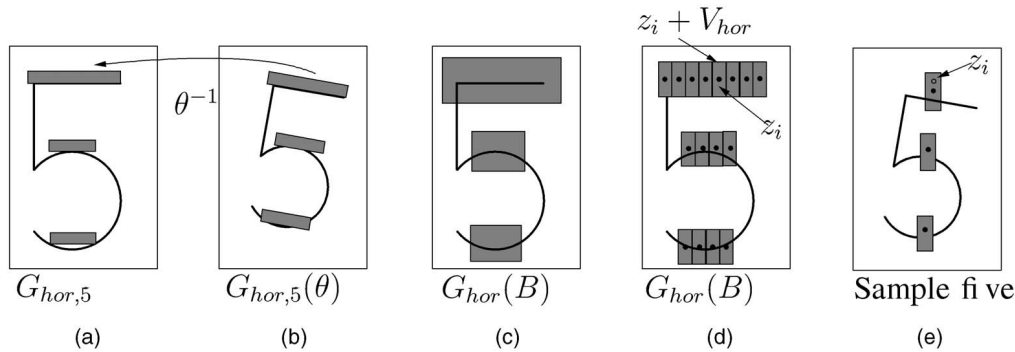


Fig. 2. (a) The horizontal edge locations in the reference grid,  $G_{hor,5}$ . (b) Edges in the image for one pose,  $G_{hor,5}(\theta)$ . (c) The model edges,  $G_{hor}(B)$ , for the entire pose bin  $B = \{5\} \times \Theta_0$ . (d) A partition of  $G_{hor}(B)$  into disjoint regions of the form  $z + V_{hor}$ . (e) The locations (black points) of actual edges and the domain of local ORing (gray strip), resulting in  $X^{spr}(z_i) = 1$ .

of binary maps—transformed intensity data—is denoted by  $X = X(I) = \{X_\gamma(z)\}_{\gamma,z}$ . We still assume that  $Y = Y(X)$ , i.e.,  $Y$  is uniquely determined by the edge data.

## 5.2 Probability Model

To begin with, we assume the random variables  $\{X_\gamma(z), \gamma \in \Gamma, z \in Z\}$  are *conditionally independent* given  $Y = y$ . We offer two principal “justifications” for this hypothesis as well as an important drawback:

1. **Conditioning.** In general, the degree of class-conditional independence among typical local features depends strongly on the amount of information carried in the “pose”  $\theta$ —the more detailed the description of the instantiation, the more decoupled the features. In the case of printed characters, most of the relevant information (other than the font) is captured by position, scale, and orientation.
2. **Simplicity.** In a Bayesian context, conditional independence leads to the “naive Bayes classifier,” a major simplification. When the dimensionality of the features is “large” relative to the amount of training data, favoring simple over complex models (and, hence, sacrificing modeling accuracy) may be ultimately advantageous in terms of *both* computation *and* discrimination.
3. **Drawback.** The resulting “background model” is not realistic. The background is a highly complex *mixture model* in which nearby edges are correlated due to clutter consisting of parts of occluded objects and other nondistinguished structures. In particular, the independence assumption renders the likelihood of actual “background” data (see (4)) far too small and this in turn leads to the traditional MAP estimator,  $\hat{Y}_{map}$ , being unreliable. It is for this reason that we will not attempt to compute  $\hat{Y}_{map}$ . Instead, we base the upcoming likelihood ratio tests on thresholds corresponding to a fixed missed detection rate learned from data, either by estimating background correlations or test statistics under shape hypotheses.

For any interpretation  $y = \{(c_i, \theta_i)\} \in \mathcal{Y}_\pi$ , we assume the shapes have *nonoverlapping supports*, i.e.,  $R(\theta_i) \cap R(\theta_j) = \emptyset, i \neq j$ . Decompose the image lattice into  $Z = R(y) \cup R(y)^c$ . The region  $R(y)^c$  represents “background.” Of course, the image data over  $R(y)^c$  may be quite complex due to clutter and

other regular structures, such as the small characters and designs which often appear on license plates. It follows that

$$P(X|Y = y) = P(X_{R(y)^c}|Y = y) \prod_{i=1}^k P(X_{R(\theta_i)}|Y = y), \quad (1)$$

where we have written  $X_U$  for  $\{X_\gamma(z), \gamma \in \Gamma, z \in U\}$  for a subset  $U \subset Z$ .

We assume that the conditional distribution of the data over each  $R(\theta_i)$  depends only on  $(c_i, \theta_i)$  and, hence, the distribution of  $X_{R(\theta_i)}$  is characterized by the product of the individual (marginal) edge probabilities

$$P(X_\gamma(z) = 1|c_i, \theta_i), \quad z \in R(\theta_i), \quad (2)$$

where we have written  $P(\dots|c_i, \theta_i)$  to indicate conditional probability given the event  $\{(c_i, \theta_i) \in Y\}$ . Notice that (2) is well-defined due to the assumption of nonoverlapping supports.

For ease of exposition, we choose a very simple model of constant edge probabilities on a distinguished, class and pose-dependent set of points. The ideas generalize easily to the case where the probabilities vary with type and location. Specifically, we make the following approximation: For each class  $c$  and for each edge type  $\gamma$ , there is a distinguished set  $G_{\gamma,c} \subset R$  of locations in the reference grid at which an edge of type  $\gamma$  has high relative likelihood when shape  $c$  is at the reference pose (see Fig. 2a). In other words,  $G_{\gamma,c}$  is a set of “model edges.” Furthermore, given shape  $c$  appears at pose  $\theta$ , the probabilities of the edges at locations  $z \in R(\theta)$  are given by:

$$P(X_\gamma(z) = 1|c, \theta) = \begin{cases} p & \text{if } z \in G_{\gamma,c}(\theta) \text{ (i.e., } \theta^{-1}z \in G_{\gamma,c}) \\ q & \text{otherwise,} \end{cases}$$

where  $p \gg q$ . Finally we assume the existence of a “background” edge frequency which is the same as  $q$ .

From (1) and (2), the full data model is then

$$P(X|Y = y) = \prod_{\gamma \in \Gamma} \prod_{z \in R(y)^c} q^{X_\gamma(z)} (1-q)^{1-X_\gamma(z)} \times \prod_{i=1}^k \left( \prod_{z \in G_{\gamma,c_i}(\theta_i)} p^{X_\gamma(z)} (1-p)^{1-X_\gamma(z)} \prod_{z \in R(\theta_i) \setminus G_{\gamma,c_i}(\theta_i)} q^{X_\gamma(z)} (1-q)^{1-X_\gamma(z)} \right). \quad (3)$$

Under this model, the probability of the data given no shapes in the image is

$$P(X|Y = \emptyset) = \prod_{\gamma \in \Gamma} \prod_{z \in \mathcal{Z}} q^{X_\gamma(z)} (1-q)^{1-X_\gamma(z)}. \quad (4)$$

## 6 INDEXING: SEARCHING FOR INDIVIDUAL SHAPES

Indexing refers to compiling a list  $\mathcal{D}(I)$  of class/pose candidates for an image  $I$  without considering global constraints. The model described in the previous section motivates a very simple procedure for defining  $\mathcal{D}$  based on likelihood ratio tests. The snag is computation—compiling the list by brute force computation is highly inefficient. This motivates the introduction of “spread edges” as a mechanism for accelerating the computation of  $\mathcal{D}$ .

### 6.1 Likelihood Ratio Tests

Consider a nonnull interpretation  $y = \{(c_1, \theta_1), \dots, (c_k, \theta_k)\} \in \mathcal{Y}_\pi$ . We are going to compare the likelihood of the edge data under  $Y = y$  to the likelihood of the same data under  $Y = \tilde{y}$ , where  $\tilde{y}$  is the same as  $y$  except that one of the elements is replaced by the background interpretation, say  $\tilde{y} = \{(c_2, \theta_2), \dots, (c_k, \theta_k)\}$ . Then, using (3) and cancellation outside  $G_{\gamma, c_1}(\theta_1)$ :

$$\frac{P(X|Y = y)}{P(X|Y = \tilde{y})} = \prod_{\gamma} \prod_{z \in G_{\gamma, c_1}(\theta_1)} \left(\frac{p}{q}\right)^{X_\gamma(z)} \left(\frac{1-p}{1-q}\right)^{1-X_\gamma(z)}. \quad (5)$$

This likelihood ratio simplifies to:

$$\log \frac{P(X|Y = y)}{P(X|Y = \tilde{y})} = \sum_{\gamma} \sum_{z \in G_{\gamma, c_1}(\theta_1)} \alpha X_\gamma(z) - \beta,$$

where

$$\alpha = \log \frac{p(1-q)}{(1-p)q} \quad \text{and} \quad \beta = \log \frac{1-q}{1-p}$$

and the resulting statistic only involves edge data relevant to the class pose pair  $(c_1, \theta_1)$ .

The log likelihood ratio test at zero type I error relative to the null hypothesis  $(c, \theta) \in Y$  (i.e., for class  $c$  at pose  $\theta$ ) reduces to a simple, linear test—evaluating

$$T_{c, \theta}(X) \doteq \mathbf{1}(J_{c, \theta}(X) > \tau_{c, \theta}) = \begin{cases} 1 & \text{if } J_{c, \theta}(X) > \tau_{c, \theta} \\ 0 & \text{otherwise,} \end{cases}$$

where

$$J_{c, \theta}(X) \doteq \sum_{\gamma} \sum_{z \in G_{\gamma, c}(\theta)} X_\gamma(z) \quad (6)$$

and the threshold  $\tau_{c, \theta}$  is chosen such that  $P(T_{c, \theta}(X) = 0 | c, \theta) = 0$ . Note that the sum is over a relatively small number of features, concentrated around the contours of the shape, i.e., on the set  $G_{\gamma, c}(\theta)$ . We therefore seek the set  $\mathcal{D}(I)$  of all pairs  $(c, \theta)$  for which  $T_{c, \theta}(X) = 1$ . Notice that  $P(Y \subset \mathcal{D}) = 1$ .

**Bayesian Inference.** Maintaining invariance (no missed detections) means that we want to perform the likelihood ratio test in (5) with no missed detections. Of course computing the actual (model-based) threshold which achieves this is intractable and, hence, it will be *estimated from training data*; see Section 7. Notice that the threshold of unity in (5) would correspond to a likelihood ratio test

designed to minimize total error; moreover,  $(c_1, \theta_1) \in \hat{Y}_{map}$  implies that the ratio in (5) must exceed unity. However, due to the severe underestimation of background likelihoods (due to the independence assumption), taking a unit threshold would result in a great many false positives. In other words, the thresholds that arise from a strict Bayesian analysis are far more conservative than necessary to achieve invariance. *It is for these reasons that the model motivates our computational strategy rather than serving as a foundation for Bayesian inference.*

### 6.2 Efficient Search

We begin with purely pose-based subsets of  $\mathcal{C} \times \Theta$ . Fix  $c$ , let  $\Theta_0$  be a neighborhood of the identity pose  $\theta_0$ , and put  $B = \{c\} \times \Theta_0$ . Suppose we want to find all  $\theta \in \Theta_0$  for which  $T_{c, \theta}(X) = 1$ . We could perform a brute force search over the set  $\Theta_0$  and evaluate  $J_{c, \theta}$  for each element. Generally, however, this procedure will fail for *all* elements in  $B$  since the background hypothesis is statistically dominant (relative to  $B$ ). Therefore, it would be preferable to have a computationally efficient binary test for the compound event  $B$ . If that test fails, there is no need to perform the search for individual poses. For simplicity, we assume that either the image contains only one instance from  $B$ —( $H_B$ ) or no shape at all—( $H_0$ ).

The test for  $H_B$  versus  $H_0$  will be based on a thresholded sum of a moderate number of binary features, approximately the same number as in (6). The test should be computationally efficient (hence, avoiding large loops and online optimization) and have a reasonable false positive rate at a very small false negative rate. Note that the brute force search through  $B$  can be viewed as a test for the above hypothesis of the form

$$T_B^{brute}(X) = \begin{cases} 1 & \text{if } \max_{\theta \in \Theta_0} T_{c, \theta}(X) = 1 \\ 0 & \text{otherwise.} \end{cases}$$

Let  $G_\gamma(B)$  denote the set of image locations  $z$  of all model edges of type  $\gamma$  for the poses in  $B$ :

$$G_\gamma(B) = \bigcup_{\theta \in \Theta_0} G_{\gamma, c}(\theta). \quad (7)$$

This is shown in Fig. 2c for the class  $c = 5$  for horizontal edges of one polarity and a set of poses  $\Theta_0$  consisting of small shifts and scale changes. Roughly speaking,  $G_\gamma(B)$  is merely a “thickening” of the  $\gamma$ -portion of the boundary of a template for class  $c$ .

### 6.3 Sum Test

One straightforward way to construct a bin test from the edge data is simply to sum all the detected model edges for all the poses in  $B$ , namely, to define

$$J_B^{sum} \doteq \sum_{\theta \in \Theta_0} J_{c, \theta}(X) = \sum_{\gamma} \sum_{\theta \in \Theta_0} \sum_{z \in G_{\gamma, c}(\theta)} X_\gamma(z) = \sum_{\gamma} \sum_{z \in G_\gamma(B)} X_\gamma(z).$$

The corresponding test is then

$$T_B^{sum} = \mathbf{1}(J_B^{sum} > \tau_B^{sum}), \quad (8)$$

meaning, of course, that we choose  $H_B$  if  $T_B^{sum} = 1$  and choose  $H_0$  if  $T_B^{sum} = 0$ . The threshold should satisfy

$$P(T_B^{sum}(X) = 0 | H_B) = 0.$$

The discrimination level of this test (i.e., false positive rate or type II error) is

$$\delta_B^{sum} = P(T_B^{sum}(X) = 1 | H_0).$$

We would not expect this test to be very discriminating. A simple computation shows that, under  $H_B$ , the probabilities of  $X_\gamma(z) = 1$  for  $z \in G_\gamma(B)$ , are all on the order of the background probabilities  $q$ . Consequently, the null type I error constraint can only be satisfied by choosing a relatively low threshold  $\tau_B^{sum}$ , in which case  $\delta_B^{sum}$  might be rather large. In other words, in order to capture all the shapes of interest, we would need to allow many configurations of clutter (not to mention other shapes) to pass the test. This observation will be examined more carefully later on.

#### 6.4 Spread Test

A more discriminating test for  $B$  can be constructed by replacing  $\sum_{G_\gamma(B)} X_\gamma(z)$  by a smaller sum of “spread edges” in order to take advantage of the fact that, under  $H_B$ , we know approximately how many on-shape edges of type  $\gamma$  to expect in a small subregion of  $G_\gamma(B)$ . To this end, let  $V_\gamma$  be a neighborhood of the origin whose shape may be adapted to the feature type  $\gamma$ . (For instance, for a vertical edge  $\gamma$ ,  $V_\gamma$  might be a horizontal strip.) Eventually, the size of  $V_\gamma$  will depend on the size of  $B$ , but for now let us consider it fixed. For each  $\gamma$  and  $z \in Z$ , define the *spread edge* of type  $\gamma$  at location  $z$  to be

$$X_\gamma^{spr}(z) = \max_{z' \in z + V_\gamma} X_\gamma(z').$$

Thus, if an edge of type  $\gamma$  is detected *anywhere* in the  $V_\gamma$ -shaped region “centered” at  $z$  it is recorded at  $z$ . (See Fig. 2e.) Obviously, this corresponds to a local disjunction of elementary features. The spread edges  $X_\gamma^{spr} = X_\gamma^{spr}(I) = \{X_\gamma^{spr}(z)\}_{\gamma,z}$  are precomputed and stored. Define also  $X_\gamma^{sum}(z) = \sum_{z' \in z + V_\gamma} X_\gamma(z')$ .

Let  $z_{\gamma,1}, \dots, z_{\gamma,n}$  be set of locations whose surrounding regions  $z_{\gamma,i} + V_\gamma$  “fill”  $G_\gamma(B)$  in the sense that the regions are *disjoint* and

$$\bigcup_{i=1}^n (z_{\gamma,i} + V_\gamma) \subset G_\gamma(B).$$

(See Fig. 2d.) To further simplify the argument, just suppose these sets coincide; this can always be arranged up to a few pixels. In that case, we can rewrite  $J_B^{sum}$  as

$$J_B^{sum} = \sum_{\gamma} \sum_{G_\gamma(B)} X_\gamma(z) = \sum_{\gamma} \sum_{i=1}^n X_\gamma^{sum}(z_{\gamma,i}).$$

Now, replace  $X_\gamma^{sum}(z_{\gamma,i})$  by  $X_\gamma^{spr}(z_{\gamma,i})$ . The corresponding bin test is then

$$T_B^{spr} = \mathbf{1}(J_B^{spr} > \tau_B^{spr}) \quad \text{where} \quad J_B^{spr} = \sum_{\gamma} \sum_{i=1}^n X_\gamma^{spr}(z_{\gamma,i}) \quad (9)$$

and  $\tau_B^{spr}$  satisfies

$$P(T_B^{spr}(X) = 0 | H_B) = 0.$$

The false positive rate is

$$\delta_B^{spr} = P(T_B^{spr}(X) = 1 | H_0).$$

#### 6.5 Comparison

Both  $T_B^{sum}$  and  $T_B^{spr}$  require an implicit loop over the locations in  $G_B$ . The exhaustive test  $T_B^{brute}$  requires a similar size loop (somewhat larger since the same location can be hit twice by two different poses). However, there is an important

difference: The features  $X_\gamma^{spr}$  and  $X_\gamma^{sum}$  can be computed *offline* and used for all subsequent tests. They are *reusable*. Thus, the tests  $T_B^{spr}, T_B^{sum}$  are significantly more efficient than  $T_B^{brute}$ . Since all tests are invariants for  $B$  (i.e., have null type I error for  $H_B$  vs  $H_0$ ), the key issue is really one of *discrimination-comparing*  $\delta_B^{sum}$  with  $\delta_B^{spr}$ . Notice that, as  $|V_\gamma|$  increases, the probability of occurrence of the features  $X_\gamma^{spr}(z)$  increases, both conditional on  $H_B$  and conditional on  $H_0$ . As a result, the effect of spreading on false positive rate is not entirely obvious.

Henceforth, we only consider rectangular sets  $V_\gamma^{s,k}$ , which are of length  $s$  in the direction orthogonal to the edge orientation and of length  $k$  in the parallel direction. (See Figs. 2d, 2e, 12b, and 12d.) Note that  $T_B^{sum} = T_B^{spr}$  if we take regions  $V_\gamma^{1,1}$ , i.e., regions of just one pixel. Assume now that the set  $G_\gamma(B)$  has more or less fixed width  $\ell$ .

In the Appendix, we show, under simplifying assumptions, that:

*The test  $T_B^{spr}$  with regions  $V_\gamma^{\ell,1}$  is the most discriminating over all possible combinations  $s, k$ . In other words, the smallest  $\delta_B^{spr}$  is achieved with  $s = \ell, k = 1$  and, hence, the optimal choice for  $V_\gamma$  is a single-pixel strip whose orientation is orthogonal to the direction of the edge type  $\gamma$  and whose length roughly matches the width of the extended boundary  $G_\gamma(B)$ . This result is very intuitive: Spreading—as opposed to summing—over a region  $z + V_\gamma^{\ell,1}$  that can contain at most one shape edge for any instantiation in  $B$  prevents off-shape edges from contributing excessively to the total sum. Note that, if  $q = 0$ , i.e., no off-shape edges appear, then the two tests are identical.*

For future use, for a general spread length  $s$ , let  $X_\gamma^s(z) = \max_{z' \in z + V_\gamma^{s,1}} X_\gamma(z')$ . Also,  $T_B^{spr}$  now refers to the optimal test using regions  $V_\gamma^{\ell,1}$ .

#### 6.6 Spreading versus Downsampling

A possible alternative for a bin test could be based on the same edge features, computed on blurred and downsampled versions of the original data. This approach is very common in many algorithms and architectures; see, for example, the successive downsampling in the feedforward network of [19] or the jets proposed in [37]. Indeed, low resolution edges do have higher incidence at model locations, but they are still less robust than spreading at the original resolution. The blurring operation smooths out low-contrast boundaries and the relevant information gets lost. This is especially true for real data such as that shown in Fig. 5 taken at highly varying contrasts and lighting conditions. As an illustration, we took a sample of the “A” and produced 100 random samples from a pose bin involving shifts of  $\pm 2$  pixels, rotations of  $\pm 10$  degrees, and scaling in each axis of  $\pm 20$  percent; see Fig. 3a. With spread 1 in the original resolution, plenty of locations were found with high probability. For example, in Fig. 3b, we show a probability map of a vertical edge type at all locations on the reference grid, darker represents higher probability. Alongside is a binary image indicating all locations where the probability was above 0.7. In Fig. 3c, the same information is shown for the same vertical edge type from the images blurred and downsampled by 2. The probability maps were magnified by a factor of 2 to compare to the original scale. Note that many fewer locations are found of probability over 0.7. The structures on the right leg of the “A” are unstable at low resolution. In general, the

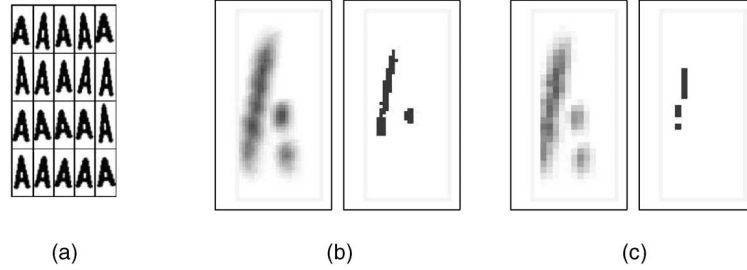


Fig. 3. (a) A sample of the population of “A”s. (b) Probability maps of a vertical edge type on the population of “A”s alongside locations above probability 0.7. (c) Probability maps of a vertical edge type on the population of “A”s blurred and downsampled by 2, alongside locations above probability 0.7.

probabilities at lower resolution without spread are lower than the probabilities at the original resolution with spread 1.

### 6.7 Computational Gain

We have proposed the following two-step procedure. Given data  $X$ , first compute the  $T_B^{spr}$ ; if the result is negative, stop and, otherwise, evaluate  $T_{c,\theta}$  for each  $c, \theta \in B$ . This yields a set  $\mathcal{D}_B$  which must contain  $Y \cap B$ ; moreover, either  $\mathcal{D}_B = \emptyset$  or  $\mathcal{D}_B = \mathcal{D} \cap B$ .

It is of interest to compare this “CTF” procedure to directly looping over  $B$ , which by definition results in finding  $\mathcal{D} \cap B$ . Obviously, the two-step procedure is more discriminating since  $\mathcal{D}_B \subset \mathcal{D} \cap B$ . Notice that the degree to which we overestimate  $Y \cap B$  will affect the amount of processing to follow, in particular, the number of pairwise comparison tests that must be performed for detections with poses too similar to coexist in  $Y$ .

As for computation, we make two reasonable assumptions: 1) Mean computation is calculated under the hypothesis  $H_0$ . (Recall that the background hypothesis is usually true.) 2) The test  $T_B^{spr}$  has approximately the same computational cost, say  $\beta$ , as  $T_{c,\theta}$ , i.e., checking for a single hypothesis  $(c, \theta)$ . As a result, the false positive rate of  $T_B^{spr}$  is then  $\delta_B^{spr}$ . Consequently, direct search has cost  $|B|\beta$ , whereas the two-step procedure has (expected) cost  $\beta + \delta_B^{spr} \times |B|\beta$ . Measuring the computational gain by the ratio gives

$$gain = \frac{|B|}{1 + \delta_B^{spr}|B|},$$

which can be very large for large bins. In fact, typically,  $\delta_B^{spr} \ll 0.5$ , so that we gain even if  $B$  has only two elements.

There is some extra cost in computing the features  $X^{spr}$  relative to simply detecting the original edges  $X$ . However, since these features are to be used in many tests for different bins, they are computed once and for all a priori and this extra cost can be ignored. This is an important advantage of *reusability*—the features that have been developed for the bin test can be reused in any other bin test.

## 7 LEARNING BIN TESTS

We describe the mechanism for determining a test for a general subset  $B$  of  $\mathcal{C} \times \Theta$ . Denote by  $\Theta_B$  and  $\mathcal{C}_B$ , respectively, the sets of all poses and classes found in the elements of  $B$ . From here on, all tests are based on spread edges. Consequently, we can drop the superscript *spr* and simply write  $T_B, \tau_B$ , etc.

For a general bin  $B$ , according to the definitions of  $G_\gamma(B)$  in (7) and  $T_B$  in (9), we need to identify  $G_{\gamma,c}(\theta)$  for each  $\gamma \in \Gamma, (c, \theta) \in B$ ; the locations  $z_i$  and the extent  $s$  of the spread edges appearing in  $J_B$ ; and the threshold  $\tau_B$ . In testing individual candidates  $B = \{c, \theta\}$  using (6), there is no spread and the points  $z_i$  are given by the locations in  $G_{\gamma,c}(\theta)$ . These, in turn, can be directly computed from the distinguished sets  $G_{\gamma,c}$ , which we assume are derived from shape models, e.g., shape templates. In some cases, the structure of  $B$  is simple enough that we can do everything directly from the distinguished “model” sets  $G_{\gamma,c}$ . This is the procedure adopted in the plate experiments (see Section 10.1).

In other cases, identifying all  $(c, \theta) \in B$ , and computing  $G_\gamma(B)$ , can be difficult. It may be more practical to directly learn the distinguished spread edges from a sample  $\mathcal{L}_B$  of subimages with instantiations from  $B$ . Fix a minimum probability  $\rho$ , say  $\rho = 0.5$ . Start with spread  $s = 1$ . Find all pairs  $(\gamma, z), \gamma \in \Gamma, z \in \cup_{\theta \in \Theta_B} R(\theta)$  such that  $\hat{P}(X_\gamma^s(z) = 1|H_B) > \rho$ , where  $\hat{P}$  denotes an estimate of the given probability based on the training data  $\mathcal{L}_B$ . If there are more than some minimum number  $N$  of these, we consider them a preliminary pool from which the  $z_i$ s will be chosen. Otherwise, take  $s = 2$  and repeat the search, and so forth, allowing the spread to increase up to some value  $s_{max}$ .

If fewer than  $N$  such features with frequency at least  $\rho$  are found at  $s_{max}$ , we declare the bin to be too heterogeneous to construct an informative test. In this case, we assign the bin  $B$  the trivial test  $T_B \equiv 1$ , which is passed by any data point. If, however,  $N$  features are found, we prune the collection so that the spreading regions of any two features are disjoint.

This procedure will yield a spread  $s = s(B)$  and a set of feature/location pairs, say  $(\gamma, z) \in F_B$ , such that the spread edge  $X_\gamma^s(z)$  has (estimated) probability at least  $\rho$  of being found on an instantiation from the bin population. The basic assumption is that, with a reasonable choice of  $\rho$  and  $N$ , the estimated spread  $s(B)$  will more or less correspond to the width of the set  $G_B$ . Our bin test is then

$$T_B = \mathbf{1}(J_B > \tau_B), \text{ where } J_B = \sum_{(\gamma,z) \in F_B} X_\gamma^s(z)$$

and  $\tau_B$  is the threshold which has yet to be determined.

Estimating  $\tau_B$  is delicate, especially in view of our “invariance constraint”  $P(T_B = 0|H_B) \approx 0$ , which is severe and somewhat unrealistic, at least without massive training sets. There are several ways to proceed. Perhaps the most straightforward is to estimate  $\tau_B$  based on  $\mathcal{L}_B$ :  $\tau_B$  is the minimum value observed over  $\mathcal{L}_B$  or some fraction thereof to insure good generalization. This is what is done in [8], for instance.



<b>For</b> $z \in Z_\eta$ $m = 0$ . <b>Check</b> $(B^{(m)}, m, z)$ . <b>Endfor</b>	Function <b>Check</b> $(B, m, z)$ . <b>If</b> $(m > M)$ Add $B$ to $\mathcal{D}^*$ , <b>Return</b> . <b>For</b> $B' \in \mathcal{B}^{(m+1)}$ such that $B' \subset B$ <b>If</b> $[T_{B'+z}(X) = 1]$ <b>Check</b> $(B', m + 1, z)$ .
---	--

Fig. 4. Pseudocode for multilevel search.

An alternative is to use a Gaussian approximation to the sum and determine  $\tau_B$  based on the distribution of  $J_B$  on background. Since the variables  $\{X_\gamma^s(z), (\gamma, z) \in F_B\}$  are actually correlated on background, we estimate a background covariance matrix  $C^s$  whose entries are the covariances between  $X_\gamma^s(z)$  and  $X_{\gamma'}^s(z + dz)$  under  $H_0$  for a range of displacements  $|dz| < 4s$ . The matrices  $C^s$  are then used to determine  $\tau_B$  for any  $B$  as follows: First, estimate the marginal probabilities  $P(X_\gamma^s(z) = 1 | H_0)$  based on background samples; call this estimate  $q_\gamma^s$ , which allows for  $\gamma$ -dependence but is, of course, translation-invariant. The mean and variance of  $J_B^s$  are then estimated by

$$\begin{aligned} \mu_{B,\emptyset} &= \sum_{(\gamma,z) \in F_B} q_\gamma^s, \quad \text{and} \\ \sigma_{B,\emptyset}^2 &= \sum_{(\gamma,z) \in F_B} \sum_{\substack{(\gamma',z') \in F_B \\ |z-z'| < 4s}} q_\gamma^s q_{\gamma'}^s C(\gamma, \gamma', z - z'). \end{aligned} \quad (10)$$

Finally, we take

$$\tau_B = \mu_{B,\emptyset} + m \cdot \sigma_{B,\emptyset},$$

where  $m$  is, as indicated, *independent* of  $B$ , i.e.,  $m$  is adjusted to obtain no false negatives for *every*  $B$  in the hierarchy. This is possible (at the loss of some discrimination) due to the inherent background normalization. Of course, since we are not directly controlling the false positive error, the resulting threshold might not be in the “tail” of the background distribution.

## 8 CTF SEARCH

The two step procedure described in Section 6.2 was dedicated to processing that portion of the image determined by the bin  $B = \{c\} \times \Theta_0$ . As a result of imposing translation invariance, this is easily extended to processing the entire image in a two-level search and even further to a multilevel search.

### 8.1 Two-Level Search

Fix a small integer  $\eta$  and let  $\Theta_{cent}$  be the set of poses  $\theta$  for which  $|z(\theta)| \leq \eta$ . For any  $B \subset \mathcal{C} \times \Theta_{cent}$  and any element  $z \in Z$ , denote by  $B + z$  the set of class/pose pairs

$$\{(c, \theta) : z(\theta) = z(\theta') + z \text{ for some } (c, \theta') \in B\},$$

namely, all poses appearing in  $B$  with positions shifted by  $z$ . Thus,

$$J_{B+z}^s = \sum_{(\gamma,z') \in F_B} X_\gamma^s(z + z').$$

Due to translation invariance, we need only develop models for subsets of  $\mathcal{C} \times \Theta_{cent}$ . Let  $\mathcal{B}$  be a partition of  $\mathcal{C} \times \Theta_{cent}$ ; it is not essential that the elements of  $\mathcal{B}$  be disjoint. In any case, assume that, for each  $B \in \mathcal{B}$ , a test  $T_B(X^s) = 1 (J_B(X^s) > \tau_B)$  has been learned as in Section 7 based on a set  $F_B$  of distinguished features.

Let  $Z_\eta$  be the sublattice of the full image lattice  $Z$  based on the spacing  $\eta$ :  $Z_\eta = \{(k_1\eta, k_2\eta)\}$ . Then, the full set of poses is covered by shifts of the elements of  $\mathcal{B}$  along the coarse sublattice:

$$\mathcal{C} \times \Theta = \bigcup_{B \in \mathcal{B}} \bigcup_{z \in Z_\eta} B + z.$$

In order to find the full index set  $\mathcal{D}$ , we first loop over all elements  $B \in \mathcal{B}$  and, for each  $B$ , we loop over all  $z \in Z_\eta$  and perform the test  $T_{B+z}$ , where  $z = (k_1\eta, k_2\eta)$ . For those subsets  $B + z$  for which  $T_{B+z} = 1$ , we loop over all individual explanations  $(c, \theta)$  and examine each one separately based on the likelihood ratio test  $T_{c,\theta}(X)$  described in Section 6.1.

### 8.2 Multilevel Search

The extension to multiple levels is straightforward. Let  $\mathcal{B}^{(0)}, \mathcal{B}^{(1)}, \dots, \mathcal{B}^{(M)}$  be a sequence of finer and finer partitions of  $\mathcal{B} \doteq \mathcal{B}^{(0)}$ . Each element  $B^{(m)} \in \mathcal{B}^{(m)}$  is the union of elements in  $\mathcal{B}^{(m+1)}$ . Perform the same loop over shifts described above for all elements  $\mathcal{B}^{(1)}$ . If  $T_{B+z}(X) = 1$  for some  $B \in \mathcal{B}^{(1)}$  and  $z \in Z_\eta$ , loop over all elements of  $B' \in \mathcal{B}^{(2)}$  such that  $B' \subset B$ , etc., until the finest level  $M$ . Elements of  $\mathcal{B}^{(M)}$  that are reached and pass their test are added to  $\mathcal{D}^*$ . Note that the loop over all shifts in the image is performed only on the coarse lattice at the top level of the hierarchy. This is summarized in Fig. 4.

### 8.3 Indexing

The result of such a CTF search is a set of detections (or “index”)  $\mathcal{D}^* \subset \mathcal{C} \times \Theta$  which, of course, depends on the image data. More precisely,  $(c, \theta) \in \mathcal{D}^*$  if and only if  $T_B = 1$  for every  $B$  appearing in the entire hierarchy (i.e., in any partition) which contains  $(c, \theta)$ . In other words, such a pair  $(c, \theta)$  has been “accepted” by every relevant hypothesis test. If indeed every test in the hierarchy had zero false negative error, then we would have  $Y \subset \mathcal{D}^*$ , i.e., the true interpretation would only involve elements of  $\mathcal{D}^*$ . In any case, we do confine future processing to  $\mathcal{D}^*$ .

In general,  $\mathcal{D}^*$  and  $\mathcal{D}$ , the set of class/pose pairs satisfying the individual hypothesis test (6), are different. However, if the hierarchy goes all the way down to individual pairs  $(c, \theta)$ , then  $\mathcal{D}^* \subset \mathcal{D}$ . Of course, constraints on learning and memory render this difficult when  $\mathcal{C} \times \Theta$  is very large. Hence, it may be necessary to allow the finest bins  $B$  to represent multiple explanations, although perhaps “pure” in class.

## 9 FROM INDEXING TO INTERPRETATION: RESOLVING AMBIGUITIES

We now seek the admissible interpretation  $y \subset \mathcal{D}^* \cap \mathcal{Y}_\pi$  with highest likelihood. In principle, we could perform a brute force loop over all subsets of  $\mathcal{D}^* \cap \mathcal{Y}_\pi$ . But, this can be significantly simplified.

Let  $y = (y_1, y_2)$ , where  $y_1, y_2$  are two admissible interpretations whose concatenation gives  $y$ , and similarly let

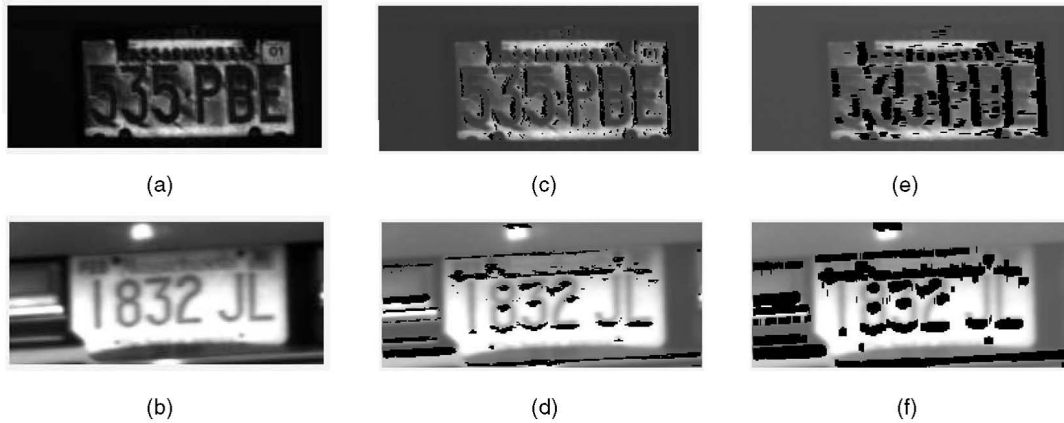


Fig. 5. (a) and (b) The subimages extracted from the images in Fig. 1 using a coarse detector for a sequence of characters. (c) Vertical edges from (a). (d) Horizontal edges from (b). (e) Spread vertical edges on (a). (f) Spread horizontal edges on (b).

$y' = (y'_1, y'_2)$ . Assume that  $y_1 = y'_1$  and that the supports of the two interpretations  $y, y'$  are the same, i.e.,  $R(y) = R(y')$ , which implies that  $R(y_2) = R(y'_2)$ . Then, due to cancellation over the background and over the data associated with  $y_1$ , it follows immediately from (3) that

$$\frac{P(X|Y = y)}{P(X|Y = y')} = \frac{P(X_{R(y_2)}|Y = y_2)}{P(X_{R(y'_2)}|Y = y'_2)}.$$

### 9.1 Individual Shape Competition

In the equation above, if the two interpretations  $y, y'$  differ by only one shape, i.e., if  $y_2 = (c_2, \theta_2)$  and  $y'_2 = (c'_2, \theta'_2)$ , then the assumptions imply that  $\theta_2 \sim \theta'_2$ . Thus, we need to compare the likelihoods on the two largely overlapping regions  $R(\theta_2)$  and  $R(\theta'_2)$ . This suggests that an efficient strategy for disambiguation is to begin the process by resolving competing detections in  $\mathcal{D}^*$  with very similar poses.

Different elements of  $\mathcal{D}^*$  may indeed have very similar poses; after all, the data in a region can pass the sequence of tests leading to more than one terminal node of the CTF hierarchy. In principle, one could simply evaluate the likelihood of the data given each hypothesis and take the largest. However, the estimated pose may not be sufficiently precise to warrant such a decision and such straightforward evaluations tend to be sensitive to background noise. Moreover, we are still dealing with individual detections and the data considered in the likelihood evaluation involves only the region  $R(\theta)$ , which may not coincide with  $R(\theta')$ .

A more robust approach is to perform likelihood ratio tests between pairs of hypotheses  $(c, \theta)$ , and  $(c', \theta')$  on the region  $R^* = R(\theta) \cup R(\theta')$  so that the data considered is the same for both hypotheses. The straightforward likelihood ratio based on (3) and taking into account cancellations is given by

$$\log \frac{P(X_{R^*}|c, \theta)}{P(X_{R^*}|c', \theta')} = \sum_{\gamma} \left[ \sum_{z \in G_{\gamma,c}(\theta) \setminus G_{\gamma,c'}(\theta')} X_{\gamma}(z) \log \frac{p}{q} + (1 - X_{\gamma}(z)) \log \frac{1-p}{1-q} \right. \\ \left. - \sum_{z \in G_{\gamma,c'}(\theta') \setminus G_{\gamma,c}(\theta)} X_{\gamma}(z) \log \frac{p}{q} + (1 - X_{\gamma}(z)) \log \frac{1-p}{1-q} \right]. \quad (11)$$

### 9.2 Spreading the Likelihood Ratio Test

Notice that, for each edge type  $\gamma$ , the sums range over the symmetric difference of the edge supports for the two shapes at their respective poses. In order to stabilize this log-ratio, we restrict the two sums to regions where the two sets  $G_{\gamma,c}(\theta)$  and  $G_{\gamma,c'}(\theta')$  are really different as opposed to being slight shifts of one another. This is achieved by limiting the sums to

$$G_{\gamma,c}(\theta) - [G_{\gamma,c'}(\theta')]^s \text{ and } G_{\gamma,c'}(\theta') - [G_{\gamma,c}(\theta)]^s, \quad (12)$$

respectively, where, for any set  $F \subset Z$ , we define the expanded version  $F^s = \{z : z \in z' + N_s \text{ for some } z' \in F\}$ , where  $N_s$  is a neighborhood of the origin. These regions are illustrated in Fig. 9.

### 9.3 Competition between Interpretations

This pairwise competition is performed only on detections with similar poses  $\theta, \theta'$ . It makes no sense to apply it to detections with overlapping regions where there are large nonoverlapping areas, in which case the two detections are really not "explaining" the same data. In the event of such an overlap, it is necessary, as indicated above, to perform a competition between admissible interpretations with the same support. The competition between two such sequences  $y = (c_1, \theta_1, \dots, c_k, \theta_k)$  and  $y' = (c'_1, \theta'_1, \dots, c'_m, \theta'_m)$  is performed using the same log likelihood ratio test as for two individual detections. For edge type  $\gamma$  and each interpretation, let

$$G_{\gamma,y} = \cup_{i=1}^k G_{\gamma,c_i}(\theta_i).$$

The two sums in (11) are now performed on  $G_{\gamma,y} - [G_{\gamma,y'}]^s$  and  $G_{\gamma,y'} - [G_{\gamma,y}]^s$ , respectively. These regions are illustrated in Fig. 10.

The number of such subinterpretation comparisons can grow very quickly if there are large chains of partially overlapping detections. In particular, this occurs when detections are found that straddle two real shapes. This does not occur very frequently in the experiments reported below and various simple pruning mechanisms can be employed to reduce such instances.

## 10 READING LICENSE PLATES

Starting from a photograph of the rear of a car, we seek to identify the characters in the license plate. Only one font is

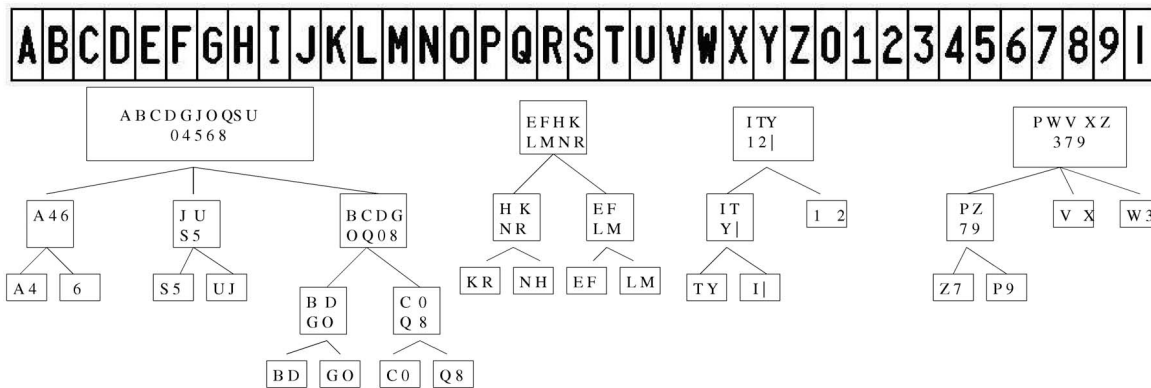


Fig. 6. Top: The 37 prototypes for characters in Massachusetts license plates. Bottom: The class hierarchy. Not shown is the root and the final split into pure classes.

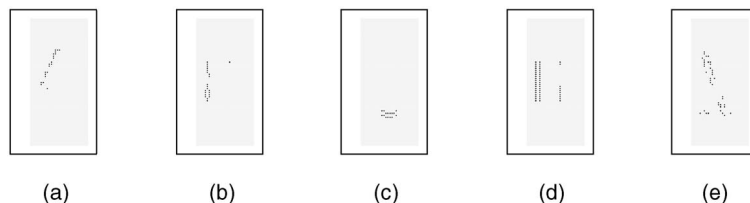


Fig. 7. The sparse subset  $F_B$  of edge/location pairs for some of a few bins  $B$  in the hierarchy. (a) 45 degree edges on the cluster  $\{A, 4, 6\}$ . (b) 90 degree edges on cluster  $\{B, C, D, G, O, Q, 0, 8\}$ . (c) 0 degree edges on cluster  $\{J, S, U, 5\}$ . (d) 45 degree edges on cluster  $\{G, O\}$ . (e) 135 degree edges on cluster  $\{X\}$ .

modeled—all license plates in the data set are from the state of Massachusetts—and all images are taken from more or less the same distance, although the location of the plate in the image can vary significantly. Two typical photographs are shown in Fig. 1, illustrating some of the challenges. Due to different illuminations, the characters in the two images have very different stroke widths despite having the same template. Also, different contrast settings and physical conditions of the license plates produce varying degrees of background clutter in the local neighborhood of the characters, as observed in the left panel. Other variations result from small rotations of the plates and small deformations due to perspective projection. For example, the plate on the right is somewhat warped at the middle and the size of the characters is about 25 percent smaller than the size of those on the left. For additional plate images, together with the resulting detections, see Fig. 11.

The plate in the original photograph is detected using a very coarse, edge-based model for a set of six generic characters arranged on a horizontal line and surrounded by a dark frame, at the expected scale, but at  $1/4$  of the original image resolution. A subimage is extracted around the highest scoring region and processed using the CTF algorithm. If no characters are detected in this subimage, the next highest scoring plate detection is processed, etc. In almost all images, the highest scoring region was the actual plate. In a few images, some other rectangular structure scored highest, but then no characters were actually detected so that the region was rejected and the next best detection was the actual plate. We omit further details because this is not the limiting factor for this application. Subimages extracted from the two images of Fig. 1 are shown in Figs. 5a and 5b.

The mean spatial density of edges in the subimage then serves as an estimate for  $q$ , the background edge probability, and we estimate  $q_s^*$  in (10) by  $sq$ . In this way, the thresholds  $\tau_B$  for the bin tests are adapted to the data, i.e., image-dependent.

The edges and spread edges on the extracted images in Figs. 5a and 5b are shown in Figs. 5c, 5d, 5e, and 5f.

### 10.1 The CTF Hierarchy

Since the scale is roughly known and the rotation is generally small, we can take  $\Theta_0 = \Theta_{cent}$ , defined as follows:  $0.8 \leq \sigma(\theta) \leq 1.2$ ;  $|\rho(\theta)| \leq 10$  degrees;  $|z(\theta)| \leq \eta = 2$  (i.e., confined to a  $5 \times 5$  window). There are 37 classes defined by the prototypes (bit maps), shown in Fig. 6. Bottom-up, binary clustering yields the pure-class hierarchy. Starting from the *edge maps* of the prototypes, at every level of the hierarchy each cluster is merged with the nearest one still available, where the distance between two clusters is measured as the average Hamming distance between any two of their elements. The hierarchy is shown in Fig. 6 without the root (all classes together) and the leaves (individual classes).

The class/pose hierarchy starts with the same structure—there is a bin  $B$  corresponding to each  $\mathcal{C}_B \times \Theta_{cent}$ , where  $\mathcal{C}_B$  is a set in the class hierarchy. Each bin in the last layer is then of the form  $B = \{c\} \times \Theta_{cent}$  and is split into  $2 \times 9 = 18$  subbins corresponding to two scale ranges ( $[.8, 1]$  and  $[1, 1.2]$ ) and to nine (overlapping)  $3 \times 3$  windows inside the  $5 \times 5$  determined by  $|z(\theta)| \leq 2$ .

The spreading is determined as in Section 7 and the sets  $G_{\gamma,c}$  are computed directly from the character templates. The tests for bins  $B = \mathcal{C}_B \times \Theta_{cent}$  are constructed by taking all edge/location pairs that belong to *all* classes in  $B$  at the reference pose. The spread is not allowed under  $s = 5$  because we can anticipate the “width” of  $G_B$  based on the range of poses in  $\Theta_{cent}$ . A subsample of all edge/location pairs is taken to ensure nonoverlapping spreading domains. This provides the set  $F_B$  described in Section 7. There is no test for the root; the search commences with the four tests corresponding to the four subnodes of the root because merging any of these four with spread  $s = 5$  produced very small sets  $F_B$ . Perhaps this could be done more gradually.

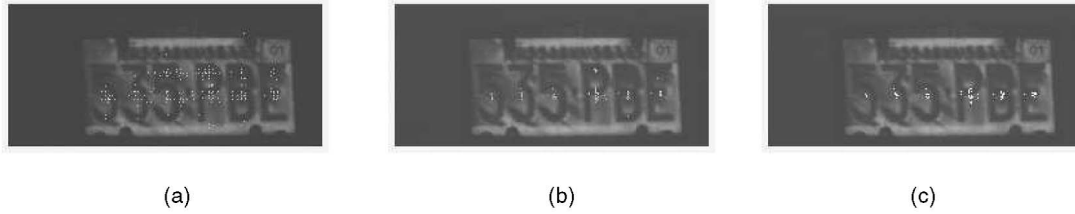


Fig. 8. (a) Coarse level detections. (b) Fine level detections. (c) Detections after pruning based on vertical alignment.

The subsets  $F_B$  for several bins are depicted in Fig. 7. For the subbins described above, which have a smaller range of poses, the spread is set to  $s = 3$ . Moreover, since this part of the hierarchy is purely pose-based and the class is unique, only the highest scoring detection is retained for  $\mathcal{D}^*$ .

## 10.2 The Indexing Stage

We have set  $\eta = 2$  (see Section 8.1) so that the image (i.e., subimage containing the plate) is scanned at the coarsest level every 5 pixels, totaling approximately  $4 \times 1,000$  tests for a plate subimage of size  $250 \times 110$ . The outcome for this stage is shown in Fig. 8a; each white dot represents a  $5 \times 5$  window for which one of the four coarsest tests (see Fig. 6) is positive at that shift. If the test for a coarse bin  $B$  passes at shift  $z$ , the tests at the children  $B$  are performed at shift  $z$  and so on at their children if the result is positive, until a leaf of the hierarchy is reached. Note that, due to our CTF strategy for evaluating the tests in the hierarchy, if the data  $X$  do reach a leaf  $B$ , then, necessarily,  $T_A(X) = 1$  for every ancestor  $A \supset B$  in the hierarchy; however, the condition  $T_B(X) = 1$  by itself does not imply that all ancestor tests are also positive. The set of all leaf bins reached (equivalently, the set of all complete chains) then constitutes  $\mathcal{D}^*$ . Each such detection has a unique class label  $c$  (since leaves are pure in class), but the pose has only been determined up to the resolution of the subbins of  $\Theta_{cent}$ . Also, there can be several detections corresponding to different classes at the same or nearby locations. The set of locations in  $\mathcal{D}^*$  is shown in Fig. 8b. The pose of each detection in  $\mathcal{D}^*$  is refined by looping over a small range of scales, rotations, and shifts and selecting the  $c, \theta$  with the highest likelihood, that is, the highest score under  $T_{c,\theta}$ .

## 10.3 Interpretation: Prior Information and Competition

The index set  $\mathcal{D}^*$  consists of several tens to several hundred detections depending on the complexity of the background and the type of clutter in the image. At this point, we can take advantage of the a priori knowledge that the characters

appear on a straight line by clustering the vertical coordinates of the detected locations and using the largest cluster to estimate this global pose parameter (see Fig. 8c.) This eliminates some false positives created by combining part of a real character with part of the background, for example part of the small characters in the word “Massachusetts” at the top of the plate; see Fig. 8b.

Among the remaining detections we perform, the pairwise competitions as described in Section 9. This is illustrated in Fig. 9 showing a region in a plate where both a “3” and a “5” were detected. For one type of edge—vertical—the regions  $G_{vert,c_1}(\theta_1)$  and  $G_{vert,c_2}(\theta_2)$  are shown in gray (Figs. 9a and 9b). The white areas illustrate a “spreading” of these regions as defined in Section 9. Figs. 9c and 9d show in white the locations in  $G_{vert,c_1}(\theta_1) \setminus [G_{vert,c_2}(\theta_2)]^s$  ( $G_{vert,c_2}(\theta_2) \setminus [G_{vert,c_1}(\theta_1)]^s$ ), where an edge is detected.

After the pairwise competitions, there are sometimes unresolved *chains of overlapping detections*. It is then necessary to perform competitions, as described in Section 9, between valid candidate *subsequences* of the chain. A valid subsequence is one which does not have overlapping characters, and is not a subsequence of a valid subsequence. This last criterion follows simply from (5). In Fig. 10a, we show a region in a plate where a chain of overlapping detections was found. The regions  $G_{\gamma,y}, G_{\gamma,y}^s$  for one competing subsequence (“RW”) are shown in Fig. 10b, for another (“|K|”) in Fig. 10c, and the resulting symmetric difference in Fig. 10d.

## 10.4 Performance Measures

### 10.4.1 Classification Rate

We have tested the algorithm on 520 plates. The correct character string is found on all but 17 plates. The classification rate per symbol is much higher—over 99 percent. Most of the errors involve confusions between  $I$  and  $|$  and between  $O$  and  $D$ . Some detections are shown in Fig. 11. However, there are also false positives, about 30 in all the plates combined, including a small number in the correctly labeled plates, usually due to detecting the symbol “|” near the borders of the plate. Other false positives are due to pairs of smaller characters as in the last row of Fig. 11. We have not attempted to deal separately with these in the sense of designing dedicated procedures for eliminating them.

### 10.4.2 Computation Time

The average classification time is 3.5 seconds per photograph on a Pentium 3 1Mghz laptop. Approximately 1.6 seconds is needed to obtain the set  $\mathcal{D}^*$  via the CTF search. The remaining 1.9 seconds is devoted to refining the pose and performing the competitions.

Of interest is the average number of detections per bin in the tree hierarchy as a function of the level, of which there

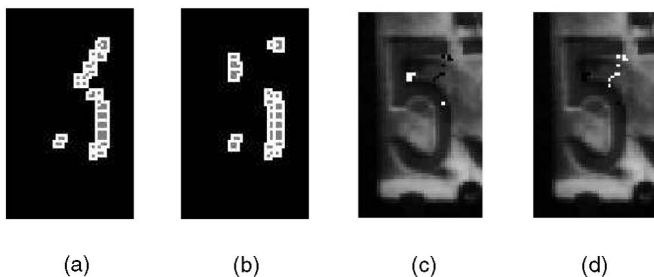


Fig. 9. Competition between  $c_1 = 3$  and  $c_2 = 5$  at a location on the plate. (a) In gray  $G_{vert,c_1}(\theta_1)$ , in white  $[G_{vert,c_1}(\theta_1)]^s$ . (b) Same for class  $c_2$ . (c) Locations in  $G_{vert,c_1}(\theta_1) \setminus [G_{vert,c_2}(\theta_2)]^s$ , where an edge is detected. (d) Locations in  $G_{vert,c_2}(\theta_2) \setminus [G_{vert,c_1}(\theta_1)]^s$ , where an edge is detected.

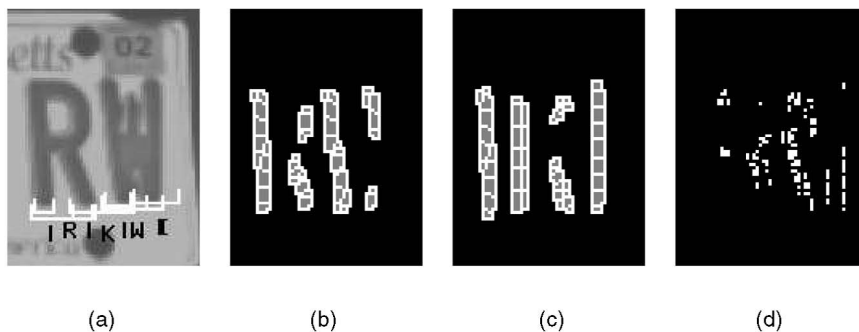


Fig. 10. Sequence competition. (a) Detected classes on a subimage—a chain with labels  $|, R, |, K, |, R, |, I$ . (b) The sets  $G_{vert,y}$  and  $G_{vert,y}^s$  for the subsequence “RW.” (c)  $G_{vert,y}, G_{vert,y}^s$  for the subsequence “|K|.” (d) The symmetric difference  $G_{vert,y} \setminus G_{vert,y}^s \cup G_{vert,y}^s \setminus G_{vert,y}$ .



Fig. 11. Examples of detections on a variety of plates. The last row illustrates false positives.

are five, not including the root. For the coarsest level (which has four bins), there are, on average, 183 detections per bin per plate, then 37, 29, and 18 for the next three levels, and, finally, four for the finest level. On average, the CTF search yields about 150 detections per plate.

If the CTF search is initiated with the leaves of the hierarchy in Fig. 6, i.e., with the pairwise clusters, the classification results are almost the same, but the computation time doubles and detection takes about 5 seconds. Therefore, approximately the same amount of time is devoted to the postdetection processing (since the resulting  $D^*$  is about the same). This clearly demonstrates the advantage of the CTF computation.

## 11 DISCUSSION

We have presented an approach to multiclass shape detection which focuses on the computational process, dividing it into two rather distinct phases: a search for instances of shapes from multiple classes which is CTF, context-independent,

and constrained by minimizing false negative error, followed by arranging subsets of detections into global interpretations using structural constraints and model-based competitions to resolve ambiguities. Spread edges are the key to producing efficient tests for subsets of classes and poses in the CTF hierarchy; they are reusable and, hence, efficient, common on shape instantiations, and yet sufficiently discriminating against background to limit the number of false detections. Spreading also serves as a means to stabilize likelihood ratio tests in the competition phase.

The experiments involve reading license plates. In this special scenario, there is exactly one prototype shape for each object class, but the problem is extremely challenging due to the multiplicity of poses, extensive background clutter, and large variations in illumination.

The CTF recognition strategy can be extended in various directions, for instance to multiple prototypes per class, (e.g., multiple fonts in OCR), to situations in which templates do not exist (e.g., faces) and the tests for class/pose bins are

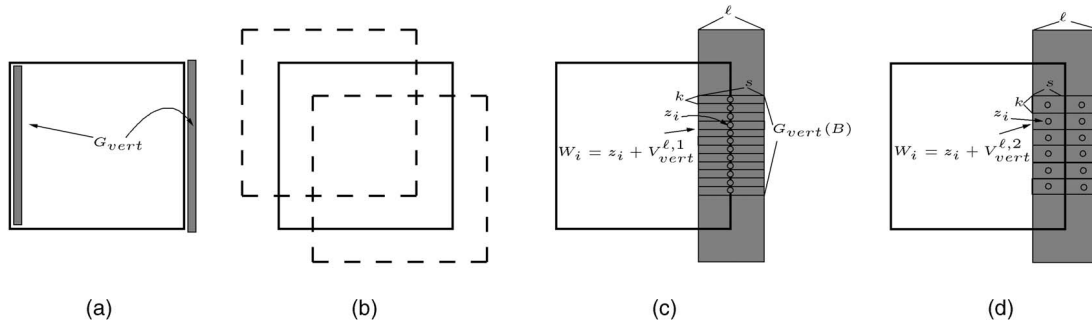


Fig. 12. (a) The model square with the region  $G_{vert}$ . (b) The range of translations of the square. (c) The model square with the region  $\hat{G}_{vert}(B)$  tiled by regions  $W_i$  with  $s = \ell$  and  $k = 1$ , centered around points  $z_i$ . (d) The same with  $s = \ell/2$  and  $k = 2$ .

learned directly from sample images, and perhaps to three-dimensional and deformable objects.

Furthermore, the framework can be extended from edges to more complex features having much lower background probabilities. Indeed it seems imperative to adopt more discriminating features in order to cope with more challenging clutter and a wider range of objects with more variability. Even in the present context, it is possible that the number of indexed instantiations could be significantly reduced using more complex features; some evidence for this with a single class can be found in [2]. This is a direction we are currently exploring, along with several others, including hypothesis tests against specific alternatives (rather than “background”), inducing CTF decompositions directly from data in order to generalize to cases where templates are not available, and sequential learning techniques such as incrementally updating CTF hierarchies, and refining the tests, as additional classes and samples are encountered.

## APPENDIX

Recall from Section 6.4 that our goal is to determine the optimal domain of ORing for a bin of the form  $B = \{c\} \times \Theta_0$  under our statistical edge model.

### A.1 Simplifying Assumptions

To simplify the analysis, suppose the class  $c$  is a square. In this case, there are two edge types  $\gamma$  of interest—horizontal and vertical—and a corresponding set of model edge locations  $G_\gamma(B)$  for each one. Suppose also that  $\Theta_0$  captures only translation in an  $\ell \times \ell$  neighborhood of the origin; scale and orientation are fixed. This is illustrated in Fig. 12. The regions  $W_i^{s,k} = z_i + V_\gamma^{s,k}$  are  $k \times s$  rectangles, for  $1 \leq s \leq \ell$  and  $k = 1, 2, \dots$ . When  $k = 1$ , a detected edge is spread to a strip oriented perpendicular to the direction of the edge; for instance, for a vertical edge, an edge detected at  $z$  is spread to a horizontal strip of width 1 and length  $s$  centered at  $z$ . See Figs. 12c and 12d for two different region shapes corresponding to  $s = \ell, k = 1$  and  $s = \ell/2, k = 2$ .

We restrict the analysis to a single  $\gamma$ , say vertical, and drop the dependence on  $\gamma$ ; the general result, combining edge types, is then straightforward. Define

$$J_B^{s,k} = \sum_{i=1}^n X_i^{s,k}, \quad \text{where } X_i^{s,k} = \max_{z' \in W_i^{s,k}} X_\gamma(z').$$

The thresholds are chosen to insure a null type I error and we wish to compare the type II errors,  $\delta^{s,k}$ , for different values of  $s$  and  $k$ . Note that  $J_B^{sum} = J_B^{1,1}$  and  $J_B^{spr} = J_B^{\ell,1}$ .

The  $W_i^{s,k}$  are taken to be disjoint and, for any choice of  $s$  and  $k$ , their union is a fixed set  $\hat{G}(B) \subset G(B)$ ; see Fig. 12. Thus, the smaller  $s$  or  $k$ , the larger the number of regions. We also assume that the image either contains no shape or it contains one instance of the shape  $c$  at some pose  $\theta \in \Theta_0$ .

Fix  $s$  and  $k$  and let  $n$  denote the number of regions  $W_i^{s,k}$  in  $\hat{G}(B)$ . Since  $\ell$  is the width of the region  $F_B$ , we have  $n \sim M/k \cdot \ell/s$ , where  $M$  is the number of regions used when  $s = \ell, k = 1$ . Let  $m = M/k$  and  $\alpha = \ell/s$ . Note that we assume each pose hits the same number,  $m$ , of regions.

Conditioning on  $\theta$ , we have

$$P(X_i^{s,k} = 1 | c, \theta) = \begin{cases} 1 - (1-p)^k (1-q)^{(s-1)k} \doteq P & \text{if } G(\theta) \cap W_i^{s,k} \neq \emptyset \\ 1 - (1-q)^{sk} \doteq Q & \text{if } G(\theta) \cap W_i^{s,k} = \emptyset, \end{cases}$$

and  $P(X_i^{s,k} = 1 | H_0) = Q$ . This implies that  $P(X_i^{s,k} = 1 | H_B) = \alpha P + (1-\alpha)Q$ , but the  $X_i^{s,k}$  variables are not independent given  $H_B$ . Furthermore,

$$E\left(\sum_{i=1}^n X_i^{s,k} | c, \theta\right) = mP + (n-m)Q,$$

$$\text{Var}\left(\sum_{i=1}^n X_i^{s,k} | c, \theta\right) = mP(1-P) + (n-m)Q(1-Q).$$

Since the conditional expectation does not depend on  $\theta$ , we have

$$E_{B,s,k} \doteq E\left(\sum_{i=1}^n X_i^{s,k} | H_B\right) = mP + (n-m)Q = n(\alpha P + (1-\alpha)Q).$$

The conditional variance is also independent of  $\theta$  and, since the variance of the conditional expectation is 0:

$$V_{B,s,k} \doteq \text{Var}\left(\sum_{i=1}^n X_i^{s,k} | H_B\right) = n(\alpha P(1-P) + (1-\alpha)Q(1-Q)).$$

On background, the test is binomial  $B(n, Q)$  and we have  $E_{\emptyset,s,k} = nQ$ , and  $V_{\emptyset,s,k} = nQ(1-Q)$ .

### A.2 The Case $p = 1$

This case, although unrealistic, is illuminating. Since  $P = 1$ ,  $J^{s,k}$  is a nonnegative random variable added to the constant  $m$ . Thus, the largest possible zero false negative threshold is  $\tau_{s,k} = m$ . For any fixed  $k$ , we have  $J^{\ell,k} \leq J^{s,k}$  for  $1 \leq s < \ell$  since we are simply replacing parts of the sum by

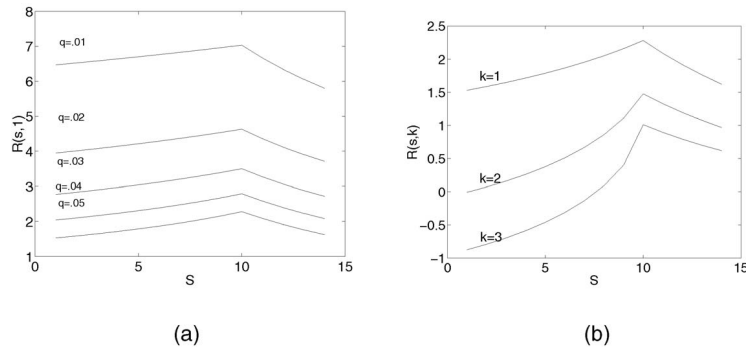


Fig. 13.  $R(s, k)$ . (a)  $m = 20, p = 0.8, \ell = 10, k = 1$  for  $0.01 \leq q \leq 0.05$ . (b)  $m = 20, p = 0.8, \ell = 10, q = 0.05$  for  $k = 1, 2, 3$ .

maxima. Since  $\tau_{s,k}$  is independent of  $s$ , it follows that  $\delta^{\ell,k} \leq \delta^{s,k}$ .

**Proposition.** For  $k \geq 2$ , assume 1)  $\ell q \leq \frac{1}{2}$  and 2)  $(1-q)^{k\ell} \leq 1 - \ell q$ . Then,  $\delta^{1,1} < \delta^{\ell,k}$ . As a result,

$$\delta^{\ell,1} \leq \delta^{1,1} < \delta^{\ell,k} \leq \delta^{1,k}.$$

In particular, the test  $J_B^{spr}(s = \ell, k = 1)$  is the most efficient.

**Note.** The assumptions are valid within reasonable ranges for the parameters  $q, \ell$ , say  $0.01 \leq q \leq 0.05$  and  $2 \leq \ell \leq 10$ .

**Proof.** When  $s = \ell$ , we have  $n = m$  and, under  $H_0$ , the statistic  $J^{\ell,k}$  is binomial  $B(m, 1 - (1-q)^{\ell k})$  and the statistic  $J^{1,1}$  is binomial  $B(M\ell, q)$ . Using the normal approximation to the binomial,

$$\delta^{1,1} \approx 1 - \Phi \left[ \frac{M - M\ell q}{(M\ell q(1-q))^{1/2}} \right] = 1 - \Phi \left[ M^{1/2} \frac{1 - \ell q}{(\ell q(1-q))^{1/2}} \right]$$

and, similarly,

$$\begin{aligned} \delta^{\ell,k} &\approx 1 - \Phi \left[ \frac{m - m(1 - (1-q)^{\ell k})}{\left( (1-q)^{\ell k} (1 - (1-q)^{\ell k}) \right)^{1/2}} \right] \\ &= 1 - \Phi \left[ m^{1/2} \left( \frac{(1-q)^{k\ell}}{(1 - (1-q)^{k\ell})} \right)^{1/2} \right]. \end{aligned}$$

Now,

$$m \frac{(1-q)^{k\ell}}{(1 - (1-q)^{k\ell})} \leq m \frac{1 - \ell q}{\ell q} \leq 2m \frac{(1 - \ell q)^2}{\ell q} \leq M \frac{(1 - \ell q)^2}{\ell q(1-q)},$$

where we have used 2) in the first inequality, 1) in the second, and  $k \geq 2$  in the third. The result follows directly from this inequality.  $\square$

### A.3 The Case $p < 1$

For  $p < 1$ , we cannot guarantee no false negatives. Instead, let  $\sigma_{B,s,k} = \sqrt{V_{B,s,k}}$  and choose  $\tau_{s,k} = E_{B,s,k} - 3\sigma_{B,s,k}$ , making the event  $J^{s,k} = 0$  very unlikely under  $H_B$ . Again, using the normal approximation, the error  $\delta^{s,k}$  is a decreasing function of

$$R(s, k) = \frac{E_{B,s,k} - 3\sigma_{B,s,k} - E_{\emptyset,s,k}}{\sigma_{\emptyset,s,k}}.$$

For general  $p$ , we do not attempt analytical bounds. Rather, we provide numerical results for the range of values of interest:  $0.01 < q < 0.05, 0.5 < p \leq 1, 10 < M < 50, 1 \leq k \leq 3$ , and  $\ell = 10$ . In Fig. 13, we show plots for the values  $M = 20, k = 1, p = 0.8, 0.01 \leq q \leq 0.05$ , and  $k = 1, 2, 3$ . The conclusions are the same as for  $p = 1$ :

- $\delta^{s,k}$  is decreasing in  $s$  in the range  $1 \leq s \leq \ell$  and increasing for  $s > \ell$ .
- $\delta^{1,1} < \delta^{\ell,k}$  for any  $k > 1$ .
- The optimal test is  $J^{sprB}$ , corresponding to  $s = \ell, k = 1$ .

### ACKNOWLEDGMENTS

Yali Amit was supported in part by US National Science Foundation ITR DMS-0219016. Donald Geman was supported in part by the US Office of Naval Research under contract N000120210053, the US Army Research Office under grant DAAD19-02-1-0337, and US National Science Foundation ITR DMS-0219016. Xiaodong Fan was supported in part by the US Office of Naval Research under contract N000120210053.

### REFERENCES

- [1] Y. Amit, "A Neural Network Architecture for Visual Selection," *Neural Computation*, vol. 12, pp. 1059-1082, 2000.
- [2] Y. Amit, *2D Object Detection and Recognition*. MIT Press, 2002.
- [3] Y. Amit and D. Geman, "A Computational Model for Visual Selection," *Neural Computation*, vol. 11, pp. 1691-1715, 1999.
- [4] Y. Amit and M. Mascaró, "An Integrated Network for Invariant Visual Detection and Recognition," *Vision Research*, 2003.
- [5] H. Barrow, J.M. Tenenbaum, R.C. Boles, and H.C. Wolf, "Parametric Correspondence and Chamfer Matching: Two New Techniques for Image Matching," *Proc. Int'l Joint Conf. Artificial Intelligence*, pp. 659-663, 1977.
- [6] S. Belongie and J. Malik, and S. Puzicha, "Shape Matching and Object Recognition Using Shape Context," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, pp. 509-523, 2002.
- [7] G. Blanchard and D. Geman, "Hierarchical Testing Designs for Pattern Recognition," *Annals of Statistics*, 2005, to appear.
- [8] F. Fleuret and D. Geman, "Coarse-to-Fine Face Detection," *Int'l J. Computer Vision*, vol. 41, pp. 85-107, 2001.
- [9] K. Fukushima and S. Miyake, "Neocognitron: A New Algorithm for Pattern Recognition Tolerant of Deformations and Shifts in Position," *Pattern Recognition*, vol. 15, pp. 455-469, 1982.
- [10] K. Fukushima and N. Wake, "Handwritten Alphanumeric Character Recognition by the Neocognitron," *IEEE Trans. Neural Networks*, vol. 2, pp. 355-365, 1991.
- [11] D.M. Gavrila, "Multi-Feature Hierarchical Template Matching Using Distance Transforms," *Proc. IEEE Int'l Conf. Pattern Recognition '98*, 2003.

- [12] S. Geman, K. Manbeck, and E. McClure, "Coarse-to-Fine Search and Rank-Sum Statistics in Object Recognition," technical report, Brown Univ., 1995.
- [13] S. Geman, D. Potter, and Z. Chi, "Composition Systems," *Quarterly J. Applied Math.*, vol. LX, pp. 707-737, 2002.
- [14] W.E.L. Grimson, *Object Recognition by Computer: The Role of Geometric Constraints*. Cambridge, Mass.: MIT Press, 1990.
- [15] H.D. Hubel, *Eye, Brain, and Vision*. New York: Scientific Am. Library, 1988.
- [16] L. Itti, E. Koch, and C. amd Niebur, "A Model of Saliency-Based Visual Attention for Rapid Scene Analysis," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 20, pp. 1254-1260, 1998.
- [17] T. Kanade and H. Schneiderman, "Probabilistic Modeling of Local Appearance and Spatial Relationships for Object Recognition," *Computer Vision and Pattern Recognition*, 1998.
- [18] Y. Lamdan, J.T. Schwartz, and H.J. Wolfson, "Object Recognition by Affine Invariant Matching," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 335-344, 1988.
- [19] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-Based Learning Applied to Document Recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278-2324, Nov. 1998.
- [20] T. Lindeberg, "Detecting Salient Blob-Like Image Structures and Their Scales with a Scale Space Primal Sketch: A Method for Focus-of-Attention," *Int'l J. Computer Vision*, vol. 11, pp. 283-318, 1993.
- [21] T. Lindeberg, "Edge Detection and Ridge Detection with Automatic Scale Selection," *Int'l J. Computer Vision*, vol. 30, pp. 117-156, 1998.
- [22] D.G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," technical report, Univ. of British Columbia, 2003.
- [23] K. Mikolajczyk and C. Schmid, "A Performance Evaluation of Local Descriptors," *Proc. IEEE Computer Vision and Pattern Recognition '03*, pp. 257-263, 2003.
- [24] G. Nagy, "Twenty Years of Document Image Analysis," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, pp. 38-62, 2000.
- [25] V. Navalpakkam and L. Itti, "Sharing Resources: Buy Attention, Get Recognition," *Proc. Intl Workshop Attention and Performance Computer Vision*, 2003.
- [26] C.F. Olson and D.P. Huttenlocher, "Automatic Target Recognition by Matching Oriented Edge Segments," *IEEE Trans. Image Processing*, vol. 6, no. 1, pp. 103-113, Jan. 1997.
- [27] C.M. Privitera and L.W. Stark, "Algorithms for Defining Visual Regions-of-Interest: Comparison with Eye Fixation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, pp. 970-982, vol. 22, 2000.
- [28] D. Reisfeld, H. Wolfson, and Y. Yeshurun, "Context-Free Attentional Operators: The Generalized Symmetry Transform," *Int'l J. Computer Vision*, vol. 14, pp. 119-130, 1995.
- [29] M. Riesenhuber and T. Poggio, "Hierarchical Models of Object Recognition in Cortex," *Nature Neuroscience*, vol. 2, pp. 1019-1025, 1999.
- [30] A.S. Rojer and E.L. Schwartz, "A Quotient Space Hough Transform for Scpaie-Variant Visual Attention," *Neural Networks for Vision and Image Processing* G.A. Carpenter and S. Grossberg, eds. MIT Press, 1992.
- [31] H.A. Rowley, S. Baluja, and T. Kanade, "Neural Network-Based Face Detection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 20, pp. 23-38, 1998.
- [32] W. Rucklidge, "Locating Objects Using the Hausdorff distance," *Proc. Int'l Conf. Computer Vision*, pp. 457-464, 1995.
- [33] D.A. Socolinsky, J.D. Neuheisel, C.E. Priebe, J. De Vinney, and D. Marchette, "Fast Face Detection with a Boosted CCD Classifier," technical report, The Johns Hopkins Univ., 2002.
- [34] A. Torralba, "Contextual Priming for Object Detection," *Int'l J. Computer Vision*, vol. 53, pp. 153-167, 2003.
- [35] S. Ullman, "Sequence Seeking and Couter Streams: A Computational Model for Bidirectional Information Flow in the Visual Cortex," *Cerebral Cortex*, vol. 5, pp. 1-11, 1995.
- [36] P. Viola and M.J. Jones, "Robust Real-Time Face Detection," *Proc. Int'l Conf. Computer Vision*, vol. II, p. 747, 2001.
- [37] L. Wiskott, J.-M. Fellous, N. Kruger, and C. von der Marlsburg, "Face Recognition by Elastic Bunch Graph Matching," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, pp. 775-779, 1997.



**Yali Amit** received the PhD degree in mathematics from the Weizmann Institute, Israel, in 1988. He spent three years as a visiting assistant professor in the Division of Applied Math at Brown University, where he started working on image analysis. In 1991, he joined the Department of Statistics at the University of Chicago. In 2000, he was appointed a full professor with a joint appointment in the Departments of Statistics and Computer Science. In recent years, his research interests have focused on object detection and recognition, speech recognition, machine learning, and computational models for the biological visual system. Much of this work is presented in a monograph published in 2002 by MIT Press.



**Donald Geman** received the BA degree in English literature from the University of Illinois in 1965 and the PhD degree in mathematics from Northwestern University in 1970. He joined the Department of Mathematics and Statistics at the University of Massachusetts-Amherst in 1970, where he became a "Distinguished University Professor." In 2001, he moved to The Johns Hopkins University, where he is currently a professor in the Department of Applied Mathematics and Statistics and a member of the Center for Imaging Science in the Whitaker Institute. Visiting appointments include those at the University of North Carolina (1976-1977), Brown University (1991-1992), Ecole Polytechnique (1997-1999), and Ecole Normale Supérieure-Cachan (2000-2003). His current research interests include computational vision, statistical and sequential learning, and bioinformatics.



**Xiaodong Fan** received the BS and MS degrees in electrical engineering from Shanghai Jiao Tong University, Shanghai, Peoples Republic of China, in 1997 and 2000, respectively. He joined the Department of Electrical and Computer Engineering at The Johns Hopkins University in 2000, where he is currently a PhD student. His research interests include computer vision, pattern recognition, machine learning, and image and video processing.

► **For more information on this or any other computing topic, please visit our Digital Library at [www.computer.org/publications/dlib](http://www.computer.org/publications/dlib).**