Efficient Focusing and Face Detection

Yali Amit, * Donald Geman † and Bruno Jedynak ‡

October 1997

Technical Report no. 459

Department of Statistics

University of Chicago

Abstract

We present an algorithm for shape detection and apply it to frontal views of faces in still grey level images with arbitrary backgrounds. Detection is done in two stages: (i) "focusing," during which a relatively small number of regions-of-interest are identified, minimizing computation and false negatives at the (temporary) expense of false positives; and (ii) "intensive classification," during which a selected region-of-interest is labeled face or background based on multiple decision trees and normalized data. In contrast to most detection algorithms, the processing is then very highly concentrated in the regions near faces and near false positives.

Focusing is based on spatial arrangements of edge fragments. We first define an enormous family of these, all invariant over a wide range of photometric and geometric transformations. Then, using only examples of faces, we select particular arrangements which are more common in faces than in general backgrounds. The second phase is texture-based; we recursively partition a training set consisting of registered and standardized regions-of-interest of both faces and non-faces.

The face training data consist of 30 individuals, 10 images per person, obtained from the Ollivetti data base. The processing time (on a Ultra Sparc 2) is under a second for a test image of size 100×100, and scales linearly with the size of the image. We achieve a false positive rate of about .2 per 10000 pixels; we estimate a false negative rate of 10%.

*Department of Statistics, University of Chicago, Chicago, IL, 60637; Email: amit@galton.uchicago.edu. Supported in part by the Army Research Office under grant DAAH04-96-1-0061 and the Department of Defense grant DAAH04-96-1-0445.

†Department of Mathematics and Statistics, University of Massachusetts, Amherst, MA 01003; Email:geman@math.umass.edu. Supported in part by the NSF under grant DMS-9217655, ONR under contract N00014-91-J-1021, and the Army Research Office (MURI) under grant DAAH04-96-1-0445.

‡Department of Statistics, University of Chicago, Chicago, IL, 60637; Email: bruno@galton.uchicago.edu. Supported in part by the Army Research Office under grant DAAH04-96-1-0061.

1 Introduction

We present an algorithm for detecting instances of isolated objects against general backgrounds based on still grey level images. The algorithm is applied to detecting and localizing faces from frontal views, which is currently an active research area. Complications arise from diverse lighting, complex backgrounds, facial expressions and extra facial features (e.g., beards and glasses). One of the main applications is face recognition; indeed, most recognition algorithms assume the face is already detected or that the background is very simple.

In most existing algorithms every candidate for a "bounding box" is directly classified as "face" or "background". The gray levels are preprocessed to account for variations due to lighting and image acquisition using techniques such as histogram equalization and plane-fitting. The classifier is induced from training data, normalized in the same manner, and consisting of both face and "non-face" images; examples include [7, 14, 15, 16, 17, 20]. Most of these authors report low false positive rates and a false negative rate of less than 10%.

However, applying these algorithms directly to every candidate subimage is very costly. As a result, the more intensive processing is sometimes preceded by a fast filter designed to identify plausible locations with very few false negatives (i.e., missed faces), but at the temporary expense of a considerable number of false positives. For example, in [16], two neural networks are developed, one for 30×30 regions which allows the face to be displaced from the center and is only applied every ten pixels, and another, more discriminating, one, trained on 20×20 subimages and only applied to the subimages filtered by the first one. The total running time is then 20 seconds (on a Sun Ultra-Sparc 2) for the 392×272 image in figure 2. Other methods are based on first extracting "interest points," especially distinguished facial features, such as an elliptical outline [12], the eyes and mouth [9, 18, 19], and local extrema [11, 10]. "Key features" are also prominent in [4, 6, 20].

Efficient focusing is our primary objective. However, in contrast to the work cited above, our approach to visual selection does not utilize complex features, which might be as difficult to detect as the face itself. Instead, we use shape information derived from local primitives, basically edge fragments, which are invariant to linear gray level transformations. In addition, they are independent at distances on the order of the scale of the object, both in the "generic background" and "object" image populations. Significant differences in the density of these features in the two populations renders *global arrangements* even more discriminating and leads to efficient focusing. Final disambiguation between object and background at the flagged locations is based on texture information: After registration and standardization, the greyscale values serve as queries for constructing multiple decision trees. Other proposals for combining edge and texture information appear in [8] and [17].

Another link with some prior work, notably [4, 6, 8, 19], as well as our own previous work on shape recognition [2, 3] and model registration ([1]), is the emphasis on *geometrical relationships* among selected points. In our framework, it is not the points themselves which are distinguished, but rather the global arrangements among their locations. Indeed, the localized features are generic (as in [13]) and too primitive and common to be informative about shape. All the discriminating power derives from spatial arrangements.

In fact, we initially consider a virtually infinite- dimensional family of arrangements, sufficiently rich that an appropriately chosen *subfamily* can separate nearly any generic shape class from general backgrounds. The selection of specific features for selective attention is based on training data. A similar procedure was investigated in [2] and [3], where shape features and tree classifiers were jointly induced during learning. The work here extends that program.

The paper is organized as follows. In section 2 we define the problem more precisely and give a compact summary of the algorithm together with some experimental results. The various components are fleshed out in the ensuing sections. In section 3 we introduce the family of features and

pinpoint some key assumptions regarding their joint distribution in faces and background. Training is explained in section 4. The focusing procedure is detailed in section 5 and section 6 is a brief conclusion.

2 Overview of the Algorithm

2.1 Problem Formulation

The problem of face detection can be viewed as a classification problem: Each image location must be classified as face or background. Since we will be dealing with approximately frontal views, we assume the location of the face is identified by a basis of three distinguished points or *landmarks*, taken to be the "centers" of the two eyes and the mouth. It should be emphasized that there is no explicit search for "eyes" and "mouth" in the algorithm; this is merely a way of defining the location of a face.

Let G denote a 48×48 *reference grid*; the points $l_1 = (20, 20), l_2 = (33, 20)$ and $l_3 = (27, 33)$ serve as reference points for the landmarks. Let $I = \{I(z) : z \in Z\}$ be a test image on a lattice Z. Every triple of points $\mathbf{b} = (b_1, b_2, b_3) \in Z^3$ represents a candidate *basis* for the location of a face. These points also determine a unique affine map $A_{\mathbf{b}}$ from G into the image, carrying (l_1, l_2, l_3) to (b_1, b_2, b_3). The reference grid G is mapped by $A_{\mathbf{b}}$ into a region $R_{\mathbf{b}} = A_{\mathbf{b}}(G) \subset Z$ which is called the *region-of-interest* (ROI) of the basis $\mathbf{b}$. *Note that a ROI is actually a reference frame or candidate pose rather than simply a subimage.* Each ROI $R_{\mathbf{b}}$ (equivalently, each basis $\mathbf{b}$) will be classified "F" or "B" based on the image data $\{I(z), z \in R_{\mathbf{b}}\}$.

Our goal is to detect faces at a range of distances of 10-20 pixels between the two eyes and at rotations of +/-10 degrees. Larger faces in the original image are detected by downsampling to .75, .5 and .25 the original resolution. Taking into account these variations in scale and rotation at a fixed resolution and the variability of the relative locations of the landmarks in the population of faces, we calculate on the order of 10^7 possible bases in a 100×100 image. Since errors of 2-3 pixels in the location of the eyes or mouth can be ignored, each face corresponds to approximately 15000 bases. Under the assumption of one face per 100×100 image, the prior probability that a basis corresponds to the landmarks on a face is on the order of 10^{-3}.

2.2 Training

For any basis $\mathbf{b}$, the *registered image* $RI_{\mathbf{b}}$ on the reference grid G is $RI_{\mathbf{b}}(z) = I(A_{\mathbf{b}}z)$. The three landmarks are marked on all training images of faces and each of these is registered to G; thus the three marked landmarks appear at l_1, l_2, l_3.

The purpose of training is to identify a collection of local binary features $X_i, i = 1, \ldots, k$, and corresponding locations $c_i \in G, i = 1, ..., k$, such that each X_i is present in a small neighborhood of c_i for approximately one-half the training images of faces. We presume these properties "generalize" to face images on arbitrary ROI's $R_{\mathbf{b}}$ provided, of course, that the locations c_i (and their neighborhoods) have the same *local coordinates* in $R_{\mathbf{b}}$. In addition, the "density" of these features in the "background population" is sufficiently low that if a randomly sampled region-of-interest is registered to G, the likelihood of finding X_i near c_i is considerably less than one-half for each $i = 1, ..., k$.

Such features are easy to identify because they are extracted from the pool of all local arrangements of edge fragments, which is enormous and which has certain invariance properties. Moreover, due to very weak statistical dependence among the features, one can then identify a collection $\{(i_1, i_2, i_3)\}$ of *triples* of local features with the property that each training image has at least one

triple present, i.e., feature X_{i_j} is found at c_{i_j} for $j = 1, 2, 3$. The feature triple $X_{i_1}, X_{i_2}, X_{i_3}$ is then associated with the triangle $(c_{i_1}, c_{i_2}, c_{i_3})$ in G.

2.3 Focusing

The locations of all the local features $X_i, i = 1, ..., k$, in a test image are precomputed. For each triangle $(c_{i_1}, c_{i_2}, c_{i_3})$, a search is then carried out for triples of pixels $\mathbf{z} = (z_1, z_2, z_3), z_i \in Z$, such that feature X_{i_j} is at $z_j, j = 1, 2, 3$, and such that the two triangles defined by $\mathbf{z}$ and $(c_{i_1}, c_{i_2}, c_{i_3})$ are similar to within scale changes of +/-25%, rotations on the order of 10 degrees, and other small deviations in shape. There is a unique affine map $A_{\mathbf{z}}$ taking the locations of c_{i_j} into $z_j, j = 1, 2, 3$. The triple $b_1 = A_{\mathbf{z}} l_1, b_2 = A_{\mathbf{z}} l_2, b_3 = A_{\mathbf{z}} l_3$ is then a hypothesis for the location of a face in the image.

There are on the order of 15000 bases around the hypothesized one which would yield locations consistent with each $\mathbf{z}$. The hypothesized basis serves as a representative of this cluster. This amounts to *a posteriori* clustering of the bases as opposed to *a priori* clustering utilizing some coarser grid. As observed in other algorithms, if the bases are clustered *a priori* in order to reduce computation time then the number of false negatives increases sharply. The collection of triangles is thus a mechanism for visual attention ("focusing") by identifying plausible clusters of bases. Due to the sparsity of the local features in the background, the number of ROI's identified by the collection of triangles is on the order of several hundreds as opposed to millions.

2.4 Intensive Classification

Each hypothesized ROI $R_{\mathbf{b}}$ is then registered to G and classified as face or background based on grey-level patterns (i.e., texture) in $RI^0_{\mathbf{b}} = \{RI_{\mathbf{b}}(z), z \in G^0\}$, where G^0 is a 20×20 subgrid G located around the landmarks (two eyes and center of mouth). The subimage $RI^0_{\mathbf{b}}$ is *standardized* by subtracting the mean of the gray level values and dividing by the standard deviation. This yields a *normalized* (i.e., registered and standardized) image $NI^0_{\mathbf{b}}$ for each basis $\mathbf{b}$ detected by a triangle. The vector $NI^0_{\mathbf{b}}$ is classified using a collection of randomized decision trees induced from a data set consisting of normalized training faces, and a collection of false positives identified on generic background images by the collection of triangles, which are also normalized. The splits in the trees are based on thresholds of the gray levels at individual pixels. These trees are aggregated to yield a classification of a detected basis.

2.5 Performance

The training data consist of 30 people, 10 images per person, obtained from the Ollivetti data base. In addition, a set of several hundred background images with no faces was downloaded from the net. We achieve a false positive rate of about .2 per 10000 pixels. A false negative rate of under 10% is estimated using an extra 100 test images from the Ollivetti data base tested at a variety of resolutions. The processing time (on a Ultra Sparc 2) is about one second for an image of size 100×100, and scales linearly with the size of the image.

In figure 1 we show the result for one image; in this case only one box is detected by the algorithm. Also shown is the corresponding basis triangle which represents an estimate on the locations of the eyes and mouth. Figures 2 and 3 show similar results for two other images. Figure 2 shows the detected boxes on an image alongside a gray-scale rendering of the *logarithm* of the number of times each pixel in the image is accessed for some form of calculation. The corresponding image for most other approaches to face detection, particularly those based on artificial neural networks, would be virtually flat.

Figure 1: The detected bounding box and the detected eyes and mouth.

3 A Class of Features

Our aim is to construct a very rich family of binary features. The construction is recursive, basically problem-independent, and leads to a hierarchy of shape descriptors with certain invariance properties. The selection of specific features for detecting faces is based on training data and the learning process is described in the following section.

Let $X(z)$ be a binary function denoting the presence or absence of a *local* image property in the vicinity of z. Thus $X(z)$ is a function of the image data in $I(z+N)$, where $\{I(z), z \in Z\}$ is the raw image data and N is small neighborhood of the origin. We seek invariance to gray scale transformations induced by changes in lighting (or other factors), and to spatial deformations in the range prescribed earlier, i.e. scaling of +/-25%, rotations of +/- 10 degrees and other small deformations. In particular, the local features should be largely invariant under the registration process itself.

3.1 Elementary Tests

All the features used for visual selection are based on comparisons of differences of gray levels, which we refer to as *elementary tests*. The neighborhood N is the immediate 3×3 neighborhood of the origin.

Comparison of differences: $E(z_2) = 1 \Leftrightarrow |I(z_1) - I(z_2)| < |I(z_2) - I(z_3)|$, where z_1, z_2, z_3 are adjacent pixels, either in a row, column or diagonal, or forming a right angle with z_2 at the vertex. It is clear that these tests are strictly invariant to linear grey-scale transformations and nearly invariant under the types of geometric deformations mentioned above.

Let $R_1, R_2, \ldots, R_k$ be a collection of $r \times r$ subregions of a ROI $R_{\mathbf{b}}$ with centers $z_i, i = 1, \ldots, k$, such that $|z_i - z_j| > r$. (Typically $5 < r < 10$). Let $\mathbf{E}_i$ be the collection of all elementary tests at all locations in the region R_i. We make the following two assumptions:

Assumption One: Conditional Independence. The random vectors $\mathbf{E}_i, i = 1, \ldots, k$, are independent conditional on the class of the data (here face or background). In other words, given that the image data in the ROI is a face, with the two eyes near b_1 and b_2 and the center of the mouth near b_3, the random vectors $\mathbf{E}_1, ..., \mathbf{E}_k$ are statistically independent; similarly given "background."

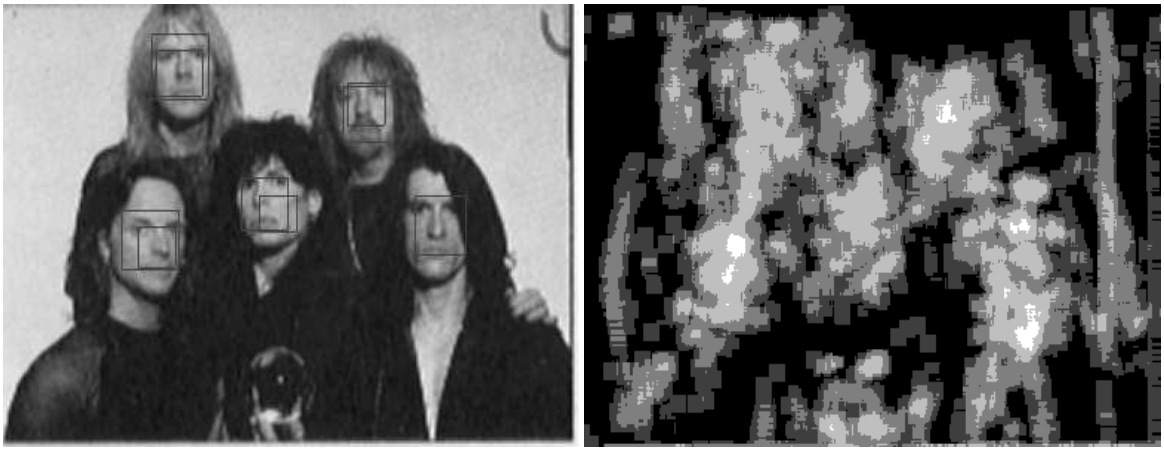

Figure 2: Top: The detected boxes. Bottom: A gray scale rendering, on a log-scale, of the number of times a pixel is accessed.

We do not believe this property holds for the actual gray level intensities $\{I(z), z \in R_i\}, i = 1 \ldots, k$, due to long range correlations induced, for example, by lighting effects.

Assumption Two: Stationarity. Realizations of each $\mathbf{E}_i$ in the "background" represent a stationary point process. That is, the probability of any particular value of $\mathbf{E}_i$ is independent of the location of R_i in the ROI. This is obviously not true if the image is a face.

Let X_R denote a binary feature which depends only on the elementary tests in an $r \times r$ support region R. There are several consequences of these two assumptions for any features $X_{R_i}, i = 1, \ldots, k$, relative to the regions R_i above.

1. $X_{R_1}, ..., X_{R_k}$ are conditionally independent given face and given background.

2. Meaningful estimates of the likelihood of the event $\{X_R = 1\}$ given background and given face can be obtained due to the local nature of the features and the invariance to linear gray level transformations. Due to translation invariance the statistics for background are expressed by *density per pixel*, and estimated by counting the number of pixels z in a large number of background images for which $X_R = 1$ where R is centered at z. The estimates for faces are obtained by counting the number of *registered* training images for which $X_R = 1$.

3. Any family $X_{R_i}, i = 1, \ldots, k$, which has different statistics on face and background, i.e., $P(X_{R_i} = 1|F) > P(X_{R_i} = 1|B)$, immediately yields a classical likelihood ratio test. Thresholds for rejection of the background hypothesis at various levels of false positive and false negative probabilities can be essentially calculated analytically. If the X_{R_i} were (conditionally) identically distributed, then a sufficient statistic for the likelihood ratio test would simply be $\xi = \sharp\{1 \leq i \leq k : X_{R_i} = 1\}$; the background hypothesis is rejected for $\xi > N$.

Every feature X we consider is a disjunction (ORing) of conjunctions of elementary tests. However, we do not entertain this entire family, but rather a subfamily, still extremely rich, which is constructed in a recursive fashion. For example, the elementary tests always appear in fixed local patterns which we call "tags." The end result is an enormous family of *generic* features. Then, during training (section 4), we single out certain ones with suitable discriminating power and computational cost for detecting faces.

Figure 3: Two additional detection examples.

3.2 Tags

Tags are conjunctions of *adjacent* elementary tests in small neighborhoods of at most 4×4. These are essentially oriented "edge detectors." Given a pair of adjacent pixels z, y let $v = z - y$ and w be the 90 degree rotation of v. Denote $z_1 = z + w, z_2 = z - w, z_3 = z + v$ and $y_1 = y + w, y_2 = y - w, y_3 = y - v$. An edge is present at z if

$$|I(z) - I(y)| > \max_{i=1,2,3} (\max(|I(z) - I(z_i)|, |I(y) - I(y_i)|)) .$$

The orientation of the edge is v if $I(z) > I(y)$ and $-v$ otherwise. Thus six intensity difference comparisons are involved in defining the edge. (We rule out any location where the magnitude of $|I(z) - I(y)|$ is less than 8 on a scale of 0 to 255.) There are eight tag types corresponding to four orientations matched with the sign of the center difference. Let $T_1, ..., T_8$ denote these features: $T_t(z) = 1$ if and only if there is a tag of type t at $z \in Z$.

We use "edge tags" mainly due to the importance of edges in image analysis and our familiarity with features of this nature. Furthermore, such features provide a small number of simple repeatable structures from which all higher level features can be composed. This has important implications for storage and computation. Finally since the tags involve conjunctions of adjacent elementary tests they exhibit a high degree of invariance to spatial deformations, in particular scaling. Still, it may well be that there are other, more effective functionals of the elementary tests.

3.3 Tag Arrangements

It is not efficient to terminate the "grouping process" with the tags themselves because their density in the background is high and their individual discriminating power is low. Moreover, any single, fixed spatial arrangement of tags, namely $\{T_{t_1}(z_1) = 1, ..., T_{t_m}(z_m) = 1\}$, is clearly lacking in invariance and much too rare for all but small values of m. Every feature we consider is a *disjunction* of such spatial arrangements which we continue to call a *tag arrangement* (TA), although a more appropriate description might be "flexible tag arrangement."

More formally, let $\mathbf{D} \subset Z^m$ be any subset of the m-dimensional lattice with the property that $z_1 \in W_1$ for any $(z_1, \ldots, z_m) \in \mathbf{D}$ where W_1 is a small neighborhood of the origin. Let $\mathbf{t} = (t_1, ..., t_m)$

$(1 \leq t_i \leq 8)$ be any sequence of tag types. The pair $Z = (\mathbf{D}, \mathbf{t})$ is called a tag arrangement. For $z \in Z$ let $\mathbf{D}_z = \{(z_1, \ldots, z_m) : (z_1 - z, \ldots, z_m - z) \in \mathbf{D}\}$. Define a binary feature $X(z)$ at $z \in Z$ by

$$X(z) = \max_{(z_1,\ldots,z_m)\in\mathbf{D}} \min_{1\leq i\leq m} T_{t_i}(z_i + z) = \max_{(z_1,\ldots,z_m)\in\mathbf{D}_z} \min_{1\leq i\leq m} T_{t_i}(z_i).$$

Thus, $X(z) = 1$ if and only if the arrangement $\{T_{t_1}(z_1) = 1, ..., T_{t_m}(z_m) = 1\}$ appears for some $(z_1, ..., z_m) \in \mathbf{D}_z$. Our family consists of all such binary features over choices of m, $\mathbf{t}$ and $\mathbf{D}$. We will assume that the arrangement is at the scale of the reference so that $|z_i - z_j| \leq 48$ for any $(z_1, ..., z_m) \in \mathbf{D}$.

Given an image I, each occurrence $X(z) = 1$ is referred to as an *instance* of X. *Every feature we use, from the tags to the triangles mentioned earlier, is a tag arrangement.* Notice that the types of tags in a TA are not necessarily distinct. On the contrary, some of the most invariant and discriminating TA's involve, naturally, several tags of the same type which "line up" in accordance with their orientation to form small curve-like structures.

A TA X will be called a *local tag arrangement* (LTA) if in fact *all* the spatial configurations $(z_1, ..., z_m) \in \mathbf{D}$ are confined to a fixed $r \times r$ neighborhood of the origin. In this way, the assumptions of section 3.1 are in force. The *density* of an LTA X in the background, is $P(X(z) = 1|B)$, which is independent of z due to stationarity.

Example1 (LTA). $\mathbf{D} = \{0\} \times \{W_2\} \cdots \times \{W_m\}$ where $W_2, ..., W_m$ are neighborhoods of the origin of size at most $r \times r$. In other words the tags "around" t_1 are allowed to "float" over a small subregion determined by the location of t_1. In this case $X(z) \leq T_{t_1}(z)$ so that the density of such a TA is necessarily lower than that of T_{t_1}. The LTA's we use are of this form and therefore provide a filter on the tags at which they are centered. For an example of the regions defining an LTA see figure 4. The TA's used in our previous work [2] were also of similar form.

Example 2 (Triangle). Consider three LTA's $X_i = (\mathbf{D}_i, \mathbf{t}_i), i = 1, 2, 3$ made respectively from m_1, m_2, m_3 tags. Given three regions U_1, U_2, U_3, the binary feature corresponding to the requirement that, for each $i = 1, 2, 3$, $X_i(z) = 1$ for some $z \in U_i$, is again a TA with $m = m_1 + m_2 + m_3$, $\mathbf{t} = (\mathbf{t}_1, \mathbf{t}_2, \mathbf{t}_3)$ and $\mathbf{D} = \cup_{z\in U_1}\mathbf{D}_{1,z} \times \cup_{z\in U_2}\mathbf{D}_{2,z} \times \cup_{z\in U_3}\mathbf{D}_{3,z}$.

For the definition of tags in terms of the elementary tests there was no need for ORing because the tests were adjacent and in the range of allowable spatial deformations this adjacency is preserved. This accounts for the high degree of invariance we observe in the tags themselves. On other hand, the LTA's involve relations among more complex structures which may not necessarily be adjacent. Spatial deformations may alter the distance between such structures so that some degree of flexibility is needed in the definition to preserve invariance. The degree of invariance can be controlled by the size of the regions $W_2, ..., W_m$ in the definition. Of course increasing the size of the W increases the density of the LTA's and hence the computational cost of the algorithm.

On a set of several hundred generic background images we have observed a tag density of .034 per pixel for each type. For LTA's constructed with regions W consisting of about ten pixels, the densities corresponding to $m = 2, 3, 4$ are, respectively, $.01, .006, .004$. The rather gradual decrease in density is due to the high degree of dependence among nearby edge fragments. Figure 5 shows all the instances of tag '0' on the image processed in figure 1 and all the instances of an LTA centered at tag '0'. The reduction in density is apparent. On the other hand this LTA appeared on over 50% of the registered training images in a small region in the upper left hand part of the face. See figure 6 below.

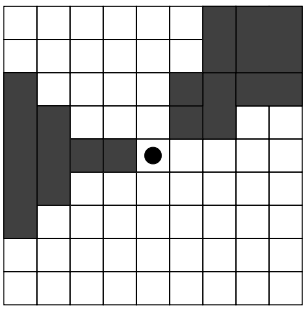

Figure 4: An example of regions W_2, W_3 for an LTA.

Figure 5: Top: Instances of tag type '0' on an image. Bottom: Instances of an LTA centered at tag type '0'.

4 Training

The features are selected from the family of TA's. The particular ones chosen are those with the highest discriminating power for a given classification problem and a given set of parameters controlling error rates and total computation. The classification problem at hand is separating ROI's which represent a face from all others. There are two distinct training phases corresponding to focusing, which is based on edge information, and final classification, which is based on grey level configurations in normalized data.

4.1 Selecting Dedicated TA's for Focusing

The training set consists of 300 *registered* images of faces based on 30 individuals. Our goal is identify LTA's which are relatively more common in faces *near a given location* in G than in arbitrary background images registered to G. Focusing is then based on TA's which are constructed from the selected LTA's by the procedure outlined in Example 2 above and discussed in more detail below.

More specifically, we seek LTA's X which appear in a small, particular, $n \times n$ region $C \subset G$ in at least a fraction $1 - \alpha$ of the faces and at most a fraction δ of background ROI's registered

Figure 6: Left: The locations of one LTA aggregated over the training set; the frequency at a pixel is proportional to darkness. The three landmarks are also shown. Right: Another LTA.

to G. We assess the prevalence in faces by the fraction of the 300 registered training faces for which $X(C) = \max_{z \in C} X(z) = 1$. The particular parameters chosen depend on target error rates and computational load; this will be discussed below. The "false positive" rate δ is estimated by multiplying the density of the LTA by n^2, the number of pixels in C; this is a crude upper bound since in effect we are replacing the probability of the union of events by the sum of the probabilities. Figure 6 shows the sum of 300 binary images in each of which all locations of one particular LTA is marked; the dark spots indicate high frequency locations.

The search procedure is recursive and very similar to the one described in [3]. The basic idea is to add the tags one at a time based on trying to keep the faces together. For each location $z \in G$ and tag type t_1 look for pairs (t_2, W_2) such that the LTA corresponding to $\mathbf{D} = \{0\} \times \{W_2\}$ maximizes $P(X(C) = 1|F)$, where C is the $n \times n$ neighborhood of z. (the probability is estimated from the 300 images). If more than $100(1-\alpha)\%$ of the faces stay together, add one more tag by maximizing the incidence in faces over LTA's with $m = 3$ and t_1, t_2 and W_2 fixed. If the maximizer still achieves the threshold $1 - \alpha$, then add a fourth tag, and so forth.

The whole procedure is easily automated and a set of LTA's $X_1, ..., X_k$, with regions $C_1, \ldots, C_k$, and with the desired α and δ, can be identified in minutes. This is possible (for virtually any shape class) due to the use of non-informative, ordinary edge fragments and the resulting extraordinary richness of the family of all local arrangements of these. In our view, a procedure based instead on *informative* and *distinguished* points, such as junctions, boundary singularities, concavities, etc., is not feasible. Special points are too scarce and too hard to identify, leading to a comparatively limited family of unstable arrangements

4.2 Parameters

The three principal "performance" parameters are of course the false negative rate, the false positive rate and the total amount of computation. Keeping the latter in check necessitates using repeatable substructures and limiting the number of ROI's which pass the "threshold test" or "triangle search," i.e., finding at least $N = 3$ of the X_i somewhere in designated regions C_i. It is only these ROI's which are considered for further processing, including normalization. The parameters α and δ defined above are the corresponding error rates for the individual LTA's $X_i(C_i)$. Since any face ROI which is "lost" during the triangle search is lost forever, we seek a very low false negative rate during focusing.

There are several "nuisance" parameters in the training algorithm, the main ones being the "depth" m of the LTA's and the number of triples, say M. We shall assume certain other parameters

are fixed throughout, such as the size of the regions W (set at around ten pixels) and the threshold in the likelihood ratio test (set at $N = 3$ since this is the minimum number necessary to determine a basis). Changing m and M has a direct bearing on performance. For example, increasing m leads to rarer events and smaller densities, but then M must be increased in order to "cover" the faces, and the net effect on total computation is not obvious. We have not systematically explored the tradeoffs. The choices given below were obtained mostly by trial and error. This was possible without excessive experimentation due to certain *analytic* calculations which exploit the two assumptions of conditional independence and background stationarity. For example, the error rates of a triangle, $(X_1(C_1), X_2(C_2), X_3(C_3))$, can be computed directly from the individual α_i and δ_i by exploiting conditional independence; and these rates in turn can be approximated, at least in the background, using the densities of $X_1(z), X_2(z), X_3(z)$.

We made $k = 9$ LTA's of depth $m = 4$. Using 5×5 regions C, this yielded $\alpha_i \leq .4$ and $\delta_i \leq .1 = 25 \times .004$ for each $X_i(C_i)$. Roughly speaking, then, each LTA covers at least one-half the faces and occurs in at most one-tenth the registered background ROI's. A simple calculation using the binomial distribution yields global error rates of under 10% false positives and under 10% false negatives for the threshold test. (Due to various approximations explained in section 5, the actual number of false negatives appears to be lower.) The number of triangles we actually use is somewhat less than $\binom{9}{3} = 84$ since we reject those with small angles (i.e., not sufficiently spread out) in order to obtain stable affine mappings (see section 5).

The ten percent false positive rate should be interpreted as follows. There are 10^7 candidate bases. We predict that 10^6 survive the threshold test and are not classified as background. But these bases are very highly correlated. Indeed we know that for each basis which passes this test, approximately 15000 others in its vicinity will pass as well, and all of these bases would flag the same face if it were present. Hence there are only on the order of 100 clusters of bases which survive this test. In section 5 we explain how one can locate these clusters without actually looping through all the bases.

4.3 Classification Based on Normalized Grey Levels

Consider the subset of ROI's which are detected by the triangles. These subimages form a more homogeneous population than randomly chosen ROI's. This is due simply to the presence of certain local features at certain locations. Each such ROI has an associated basis $\mathbf{b}$. A sample of such ROI's, $\{R_{\mathbf{b}}\}$, is obtained from a large collection of background images. Each of these is then registered and standardized to yield a "normalized" image $NI^0_{\mathbf{b}}$ (see section 2.4.).

For the face training data, the landmarks are provided manually and the corresponding normalized image $NI^0_{\mathbf{b}}$ is then determined. In order to enrich the training set of faces we randomly perturb the locations of the three landmarks. We regard the normalized ROI of a face as a robust source of information about characteristic grey level patterns in faces, i.e., about typical face textures. For example, the area around the mouth is usually darker than the area around the cheeks. Clearly the only reliable information of this nature resides in *relative* brightness values for *registered* data, which provides the justification for the normalization process.

Each normalized image is regarded as a 400 dimensional feature vector of real numbers typically lying between -2 and 2. The standard CART algorithm ([5]) is applied to grow a classification tree from this training sample. There are two classes, "F" and "B" and the splits (questions) are elementary tests which compare a normalized grey level to a threshold. Each terminal node of this tree can also be assigned an estimate of the posterior distribution on faces and background. Note that this posterior is also conditional on the ROI having been detected by one of the triangles.

5 Focusing

Recall that $I = \{I(z) : z \in Z\}$ denotes a test image on Z and there is a region-of-interest $R_{\mathbf{b}}$ corresponding to each basis $\mathbf{b} = (b_1, b_2, b_3)$, where the family $\mathcal{B}$ of bases is limited by the allowed range of scalings, rotations, etc. This yields on the order of 10^7 ROI's, each to be classified as "B" or "F" based on the image data $\{I(z), z \in R_{\mathbf{b}}\}$.

Let X_{ij} denote the j'th LTA in the i'th triangle, $i = 1, ..., M$, and let C_{ij} be its corresponding region in the reference grid. (Of course each X_{ij} is one of the LTA's $X_1, ..., X_9$ and similarly for the C_{ij}.) Recall that $RI_{\mathbf{b}}$ denotes the data in $R_{\mathbf{b}}$ registered to G. Exact implementation of the entire detection algorithm means performing the following four steps:

1. For each $\mathbf{b} \in \mathcal{B}$, register the data in $R_{\mathbf{b}}$ to G.
2. Check whether at least one of the triangles is present in $RI_{\mathbf{b}}$, i.e., calculate $\mathcal{F}(\mathbf{b}) = \max_i \min_{j=1,2,3} X_{ij}(C_{ij})$, where i runs over triangles.
3. If $\mathcal{F}(\mathbf{b}) = 0$ (i.e., no triangle is present), classify $R_{\mathbf{b}}$ as "B" and stop.
4. If $\mathcal{F}(\mathbf{b}) = 1$, normalize the data in $RI_{\mathbf{b}}$ (as described in Section 2) and send the normalized grey-levels down each classification tree. Add the resulting distributions and classify $R_{\mathbf{b}}$ as "F" or "B" according to the higher mass, or according to some other classification rule.

The algorithm we have implemented is a much faster variation based on an image-wide search for triangles in the original coordinates. First, notice that whereas the regions C_{ij} are specified in the reference frame, the LTA's X_{ij} themselves are defined in global coordinates in a basis-independent manner. Hence, due to the (near) invariance to the registration process (as discussed in section 3), we can search for individual LTA's *directly* in $R_{\mathbf{b}}$ rather than in the image $RI_{\mathbf{b}}$. Specifically, define $C_{ij}(\mathbf{b}) = A_{\mathbf{b}}(C_{ij})$, where again $A_{\mathbf{b}}$ is the affine map taking the registered landmark locations $(l_1, l_2, l_3) \in G^3$ to $\mathbf{b} \in R^3_{\mathbf{b}}$. Then, with high likelihood, $\min_{j=1,2,3} X_{ij}(C_{ij}) = 1$ relative to the registered data $RI_{\mathbf{b}}$ if and only if $\min_{j=1,2,3} X_{ij}(C_{ij}(\mathbf{b})) = 1$ relative to the raw data on $R_{\mathbf{b}}$.

However, the algorithm still requires a loop over bases, even though the registration process need no longer be applied to every ROI. We can eliminate this loop by defining a tag arrangement Δ_i for each $i = 1, ..., M$ in terms of *global coordinates*, and a small basis-dependent region $S_i(\mathbf{b})$ such that: $\mathcal{F}(\mathbf{b}) \leq \mathcal{F}^*(\mathbf{b}) \equiv \max_i \max_{z \in S_i(\mathbf{b})} \Delta_i(z)$. In other words, $\mathcal{F}^*$ is more conservative filter. Let $\mathcal{B}^*$ be the set of all bases for which $\mathcal{F}^*(\mathbf{b}) = 1$. Then clearly, $\mathcal{B}^* = \bigcup_{i,z:\Delta_i(z)=1} \mathcal{B}_{iz}$ where $\mathcal{B}_{iz} = \{\mathbf{b} \in \mathcal{B} : z \in S_i(\mathbf{b})\}$. It follows directly that we can replace the original algorithm by

1. Loop over $z \in Z$
2. Loop over $i = 1, \ldots, M$,
   calculate $\Delta_i(z)$.
   a. If $\Delta_i(z) = 0$, goto next i.
   b. If $\Delta_i(z) = 1$, normalize the data in $R_{\mathbf{b}}$ for each $\mathbf{b} \in \mathcal{B}_{iz}$, and send the normalized grey-levels down each classification tree. Add the resulting distributions and classify $R_{\mathbf{b}}$ for each $\mathbf{b} \in \mathcal{B}_{iz}$ as "F" or "B" according to the higher mass, or according to some other classification rule.
   Goto next i.
3. Classify all remaining ROI's as "B".

A major speed-up is obtained by representing the "cluster" of bases $\mathcal{B}_{iz}$ by a single element $\mathbf{b} \in \mathcal{B}_{iz}$. This basis is easily identified by a coordinate transformation based on the locations of three image locations in the detected triangle Δ_i.

We conclude this section by describing the TA's Δ_i and the resulting search process in the full image plane. The Δ_i's are defined in order to accommodate a range of scales of +/-25%, small rotations and other deformations. Let $c_1, ..., c_9$ denote the centers of the 5×5 regions $C_1, ..., C_9$.

1. Loop over the entire image and find the locations of each of the tag types.
2. Loop through the locations of the tag types and find those which are at the center of any of the LTA's $X_1, ..., X_9$. The locations of all nine LTA's are then identified.
3. For each triangle c_{i1}, c_{i2}, c_{i3}:
   (a) For each location z_1 of X_{i1} search for X_{i2} in the 9×9 region around $z_1 + (c_{i2} - c_{i1})$.
   (b) If an instance is found, say at z_2, calculate the predicted location z of X_{i3} by mapping c_{i3} according to the translation, scale and rotation which takes c_{i1} to z_1 and c_{i2} to z_2.
   (c) Search for X_{i3} in a 5×5 region around z. If it is found, say at z_3, a match $\mathbf{z} = (z_1, z_2, z_3)$ is identified for the triangle c_{i1}, c_{i2}, c_{i3}.
4. For each such triple $\mathbf{z} = (z_1, z_2, z_3)$ identify the map $A_{\mathbf{z}}$ as above and the corresponding basis $\mathbf{b} = (A_{\mathbf{z}} l_1, A_{\mathbf{z}} l_2, A_{\mathbf{z}} l_3)$. This basis is a representative of the cluster of bases mentioned above; each ROI in the cluster has the indicated triple of LTA's.

This loop is many times faster than a loop through all bases. This image-wide search also provides an *a posteriori* clustering of the bases of the image. Only representative bases are then processed by the CART trees and ultimately classified as face or background. Our attempts to accelerate the loop through the bases by clustering them in an *a priori* way, say using a coarser grid, resulted in significant increases in false negatives.

6 Conclusion

We have described a detection algorithm with two stages - focusing and intense classification. Focusing involves detecting at least one member of a family of global arrangements of local image features. This step is computationally very efficient and all but a relatively small number of potential regions-of-interest are discarded. Intensive classification involves normalizing the remaining regions-of-interest and implementing a tree-based classifier for object versus background.

The training algorithm has two corresponding steps: identifying discriminating and invariant local features based on examples of faces, and making classification trees by recursively partitioning a training set consisting of both positive and negative examples, the latter being registered and standardized false positive ROI's detected by the feature arrangements in sample background images. Although described in the specific context of face detection, the algorithm is easily ported to other problems as will be illustrated in a forthcoming paper in which we also investigate connections to selective attention in natural visual systems.

References

[1] Y. Amit. Graphical shape templates for automatic anatomy detection, application to mri brain scans. *IEEE Trans. Medical Imaging*, 16:28–40, 1997.

[2] Y. Amit and D. Geman. Shape quantization and recognition with randomized trees. *Neural Computation*, 9:1545–1588, 1997.

[3] Y. Amit, D. Geman, and K. Wilder. Joint induction of shape features and tree classifiers. *IEEE Trans. PAMI*, 19(11), 1997.

[4] M. Bichsel. Strategies of robust object recognition for the automatic identification of human faces. Technical report, ETH-Zurich, 1991.

[5] L. Breiman, J. Friedman, R. Olshen, and C. Stone. *Classification and Regression Trees.* Wadsworth, Belmont, CA., 1984.

[6] M.C. Burl, T.K. Leung, and P. Perona. Face localization via shape statistics. In *Proceedings, International Workshop on Automatic Face and Gesture Recognition*, pages 154–159, 1995.

[7] A. J. Colmenarez and T. S. Huang. Face detection with information-based maximum discrimination. In *Proceedings, CVPR*, pages 782–787. IEEE Computer Society Press, 1997.

[8] T. F. Cootes and C. J. Taylor. Locating faces using statistical feature detectors. In *Proceedings, Second International Conference on Automatic Face and Gesture Recognition*, pages 204–209. IEEE Computer Society Press, 1996.

[9] C. C. Han, H. Liao, L. Yu, and L. Hua. Fast face detection via morphology-based pre-processing. Technical report, 1996.

[10] R. Hoogenboom and M. Lew. Face detection using local maxima. In *Proceedings, Second International Conference on Automatic Face and Gesture Recognition*, pages 334–339. IEEE Computer Society Press, 1996.

[11] J. Huang, S. Gutta, and H. Wechsler. Detection of human faces using decision trees. In *Proceedings, Second International Conference on Automatic Face and Gesture Recognition*, pages 248–252. IEEE Computer Society Press, 1996.

[12] A. Jecquin and A. Eleftheriadis. Automatic location tracking of faces and facial features in video sequences. In *Proceedings, International Workshop on Automatic Face and Gesture Recognition, Zurich*, 1995.

[13] T. Maurer and C. von der Malsburg. Tracking and learning graphs and pose on image sequences of faces. In *Proceedings, Second International Conference on Automatic Face and Gesture Recognition*, pages 176–181. IEEE Computer Society Press, 1996.

[14] E. Osuna, R. Freund, and F. Girosi. Training support vector machines: an application to face detection. In *Proceedings, CVPR*, pages 130–136. IEEE Computer Society Press, 1997.

[15] A. Pentland, B. Moghaddam, and T. Starner. View-based and modular eigenspaces for face recognition. In *Proceedings, Computer Vision and Pattern Recognition 94*, pages 84–91, 1994.

[16] H. A. Rowley, S. Baluja, and T. Kanade. Human face detection in visual scenes. Technical Report CMU-CS-95-158R, School of Computer Science, Carnegie Mellon University, 1995.

[17] K. K. Sung and T. Poggio. Example-based learning for view-based human face detection. Technical Report A.I Memo 1521, Artificial Intelligence Laboratory, M.I.T, 1994.

[18] A. Tankus, H. Yeshurun, and N. Intrator. Face detection by direct convexity estimation. In *Proceedings of the First Intl. Conference on Audio- and Video-based Biometric Person Authentication, Springer, Crans-Montana, Switzerland*, 1997.

[19] K. C. Yow and R. Cipolla. Towards an automatic human face localization system. In *Proceedings, British Machine Vision Conference*, pages 701–710. BMVA Press, 1995.

[20] A. L. Yuille, D. S. Cohen, and P. Halliman. Feature extraction from faces using deformable templates. *Inter. J. Comp. Vision*, 8:104–109, 1992.