MASSACHUSETTS INSTITUTE OF TECHNOLOGY
ARTIFICIAL INTELLIGENCE LABORATORY

A.I. Memo No. 1461 December, 1993

C.B.C.L. Paper No. 89

Face Recognition Under Varying Pose

David J. Beymer

email: beymer@ai.mit.edu

Abstract

Researchers in computer vision and pattern recognition have worked on automatic techniques for recognizing human faces for the last 20 years. While some systems, especially template-based ones, have been quite successful on expressionless, frontal views of faces with controlled lighting, not much work has taken face recognizers beyond these narrow imaging conditions. Our goal is to build a face recognizer that works under varying pose, the difficult part of which is to handle face rotations in depth. Building on successful template-based systems (especially Brunelli and Poggio[7]), our basic approach is to represent faces with templates from multiple model views that cover different poses from the viewing sphere. To recognize a novel view, the recognizer locates the eyes and nose features, uses these locations to geometrically register the input with model views, and then uses correlation on model templates to find the best match in the data base of people. Our system has achieved a recognition rate of 98% on a data base of 62 people containing 10 testing and 15 modelling views per person.

This report describes research done at the Artificial Intelligence Laboratory and within the Center for Biological and Computational Learning. This research is sponsored by grants from the Office of Naval Research under contracts N00014-91-J-1270 and N00014-92-J-1879; by a grant from the National Science Foundation under contract ASC-9217041. Support for the A.I. Laboratory's artificial intelligence research is provided by ONR contract N00014-91-J-4038. The author is supported by a Howard Hughes Doctoral Fellowship from the Hughes Aircraft Company.

1 Introduction

Researchers in computer vision and pattern recognition have worked on automatic techniques for recognizing human faces for the last 20 years. While there have been successful systems, especially those systems using a pictorial representation for faces, most face recognition systems operate under relatively rigid imaging conditions: lighting is controlled, people are not allowed to make facial expressions, and facial pose is fixed at a full frontal view. We have developed a face recognition system that works under varying pose, with the ultimate goal of making the conditions under which face recognizers operate less rigid.

1.1 The problem

What is the problem of automatic face recognition? Given as input the visual image of a face, which might be a digitized signal from a video camera or a digitized photograph, compare the input face against models of faces that are stored in a library of known faces and report a match if one is found. A related problem is face verification: given an input face image and a proposed identity, verify that the face indeed belongs to the claimed person. The problem of face segmentation, distinguishing faces from a cluttered background, is usually avoided by imaging faces against a uniform background.

The problem of face recognition has attracted researchers not only because faces represent a challenging class of naturally textured 3D objects, but because of the many applications of automatic face recognition. In building security, a face recognizer could be used at the front entrance for automatic access control. They could be used to enhance the security of user authentication in ATMs by recognizing faces as well as requiring passwords. In the human/computer interface arena, workstations with cameras would be able to recognize users, perhaps automatically loading the user's environment when he sits down in front of the machine.

Face recognition is difficult for two major reasons. First, faces form a class of fairly similar objects – all faces consist of the same facial features in roughly the same geometrical configuration. Thus, the task of face recognition is a fine discrimination task which may require the use of subtle differences in facial appearance or the configuration of features. Second, face recognition is also made difficult because of the wide variation in the appearance of a particular face due to imaging conditions such as lighting and pose, as in the more generic task of 3D object recognition. Because of the detailed 3D structure of the face, the 2D image of a face changes as it undergoes rotations "in depth" or as the light source changes direction. The non-rigidity of faces, caused by changes in facial expressions, adds to the variability of facial appearance.

In this paper we describe a view-based approach to recognizing faces under varying pose. In our system, faces will be modelled with multiple views that cover the viewing sphere. To recognize a novel view, the recognizer locates the eyes and nose features, uses these locations to geometrically register the input with model views, and then uses correlation on model templates to find the best match in the data base of people.

1.2 Existing work

Since our face recognizer finds facial features in order to register the input image with the model views, our discussion of existing work, in addition to face recognition, will include facial feature detection.

1.2.1 Facial feature detection

Facial feature detection, for the most part, is the problem of locating the major facial features such as the eyes, nose, mouth, and face outline. Some researchers have also addressed the issue of characterizing facial features, usually with the parameters of a model fit to the feature. While most feature detection efforts are motivated by the need to geometrically normalize a face image prior to recognition, other applications of facial features include face tracking and attentional mechanisms for locating faces in cluttered images.

Most research to date has taken one of three major approaches, a parameterized model approach, a pictorial approach, and the use of grey level interest operators. In one parameterized model approach, deformable template models of individual facial features are fit to the image by minimizing an energy functional (Yuille, Hallinan, and Cohen[43], Hallinan[18], Shackleton and Welsh[34], Huang and Chen[20]). These deformable models are hand constructed from parameterized curves that outline subfeatures such as the iris or a lip. An energy functional is defined that attracts portions of the model to preprocessed versions of the image – peaks, valleys, edges – and model fitting is performed by minimizing this functional. A related model-based approach fits a global head model constructed from tens of feature locations (Bennett and Craw[4], Craw, Tock, and Bennett[14], Cootes, *et al.*[12]) to the image by varying individual feature locations. Terzopoulos and Waters [37] have used the active contour model of snakes to track facial features in image sequences.

In the pictorial approach, a pixel-based representation of facial features is matched against the image. This representation may be templates of the major facial features (Bichsel[6], Baron[3], Burt[8], Poggio and Brunelli[7]) or the weights of hidden layer nodes in neural networks (Vincent, Waite and Myers[39]). For the template-based systems, correlation on preprocessed versions of the image is the typical matching metric. The neural network approaches construct a network where implicit feature templates are "learned" from positive and negative examples.

Another major approach to facial feature finding is the use of low level intensity-based interest operators. As opposed to the model-based and template-based approaches, this approach does not find features with semantic content as, say, an eye, nose or mouth detector does. Instead, the features are defined by the local grey level structure of the image, such as corners (Azarbayejani, *et al.* [2]), symmetry (Reisfeld and Yeshurun[32]), or the "end-inhibition" features of Manjunath, Shekhar, Chellappa, and von der Malsburg[28], which are extracted from a wavelet decomposition of the image.

1.2.2 Face recognition

While the earliest work in automatic face recognition dates back two decades (Kanade[21]), the topic has seen renewed interest in the last few years. Most face recognition systems operate on intensity images of frontal or nearly frontal views of a face, and practically all of them follow the same basic recognition technique. The recognizer makes a linear scan through a library of known faces, comparing the input to each model face. This comparison is performed using a distance metric, such as a weighted Euclidean distance or correlation, in the space used for representing faces. The model yielding the smallest distance is reported as the identified person. In addition, some systems include the notion of rejecting the input if the best match is not good enough.

Our discussion of existing work will be divided into sections on input representation, invariance to imaging conditions, and experimental issues such as recognition rate.

Input representation Comparing model and input faces boils down to performing distance measurements in the space used to represent faces. As current face recognition systems use fairly standard distance metrics like weighted norms and correlation, the main factor that distinguishes different approaches is input representation. There are two main approaches to input representation, a geometrical approach that uses the spatial configuration of facial features, and a more pictorial approach that uses an image-based representation.

There have been several feature geometry approaches, beginning with the seminal work of Kanade[21], and including Kaya and Kobayashi[22], Craw and Cameron[13], Wong, Law, and Tsang[41], Brunelli and Poggio[7], and Chen and Huang[10]. These feature-based systems begin by locating a set of facial features, including such features as the corners of the eyes and mouth, sides of the face and nose, nostrils, the contour along the chin, etc. The spatial configuration of facial features is captured by a feature vector whose dimensions typically include measurements like distances, angles, and curvatures. Once faces are represented by feature vectors, the similarity of faces is measured simply by the Euclidean distance or a weighted norm, where dimensions are usually weighted by some measure of variance.

The second major type of input representation is pictorial in nature, representing faces by using filtered images of model faces. In template-based systems, the simplest pictorial representation, faces are represented either by images of the whole face or by subimages of the major facial features such as the eyes, nose, and mouth (Baron[3], Brunelli and Poggio[7], Yang and Gilbert[42], Burt[8], Bichsel[6]). Template images need not be taken from the original grey levels; some systems use the gradient magnitude or gradient vector field in order to get invariance to lighting. An input face is then recognized by comparing it to all of the model templates, typically using correlation as an image distance metric.

Principal components analysis has been explored as a means for both recognizing and reconstructing face images (Kirby and Sirovich[23], Turk and Pentland[38], Akamatsu, *et al.*[1], Craw and Cameron[13], Dalla Serra and Brunelli[33]). It can be read as an suboptimal pictorial approach, reducing the dimensionality of the input space from the number of pixels in the templates to the number of eigenpictures, or "eigenfaces", used in the representation. In this approach, one first applies principal components analysis to an ensemble of faces to construct "face space". This forms the representation onto which all faces are projected and distance measurements are performed.

Besides principal components analysis, other analysis techniques have been applied to images of faces, generating a new, more compact representation than the original image space. Examples include autocorrelation (Kurita, Otsu and Sato[25]), Singular Value Decomposition (Cheng, *et al.*[11] and Hong[19]), and vector quantization (Ramsay, *et al.*[31])

Connectionist approaches to face recognition also use pictorial representations for faces (Kohonen[24], Fleming and Cottrell [16], Edelman, Reisfeld, and Yeshurun[15], Weng, Ahuja, and Huang[40], Fuchs and Haken[17], Stonham[36]). Since the networks used in connectionist approaches are just classifiers, these approaches are similar to the ones described above. Different pixel-based representations have been used, with [24], [16], [17] using the original grey level images. [40] uses directional edge maps, [36] uses a thresholded binary image, and [15] uses Gaussian units applied to the grey level image.

Hybrid representations that combine the geometrical and pictorial approaches have been explored, such as Cannon *et al.*[9], whose feature vector face representation includes geometrical and template-based information. In another hybrid approach, Lades *et al.*[26] and Manjunath, Chellappa, and von der Malsburg[27] represent faces as elastic graphs of local textural features.

Invariance to imaging conditions The wide variation in face appearance under changes in pose, lighting, and expression makes face recognition a difficult task. While existing systems do not allow much flexibility in pose, lighting, and expression, most systems do provide some flexibility by using invariant representations or performing an explicit geometrical normalization step.

Representations invariant to changes in lighting and pose have been used to increase the robustness of face recognizers. For instance, filtering the face image with a bandpass filter like the Laplacian provides some invariance to lighting. Assuming that the image content due to lighting is lowpass, bandpass filtering should remove the lighting effects while still preserving the higher frequency texture information in the face. To provide shift invariance, some systems preprocess images using the Fourier transform magnitude (Akamatsu, *et al.*[1]) or autocorrelation (Kurita, Otsu and Sato[25]).

By finding at least two facial features – usually the eyes in existing systems – the face can be normalized for translation, scale, and image-plane rotation. In feature geometry approaches, distances in the feature vector are normalized for scale by dividing by a given distance such as the interocular distance. In template-based systems, faces are often geometrically normalized by rotating and scaling the input image to place the eyes at fixed locations. This normalization step reduces pose space from its original 6D formulation to a 2D space of rotations out of the image plane. In a recognizer that allows general pose, rotations on the viewing sphere still need to be handled.

Most face recognition systems are not designed to handle changes in facial expression or rotations out of the image plane. By tackling changes in pose and lighting with the invariant representations and normalization techniques described above, current systems treat face recognition mostly as a rigid, 2D problem. There are exceptions, however, as some systems have employed multiple views ([1], [25]) and flexible matching strategies ([27], [26]) to deal with some degree of expression and out-of-plane rotations. What distinguishes our approach from these techniques, which will be explained in section 1.3, will be a wider allowed variation in viewpoint.

Experimental issues The evaluation of face recognition systems is largely empirical, requiring experimental study on a set of test images. Probably the two most important statistics are the recognition rate and model library size. Systems that include rejection also report the false access rate, usually defined as the fraction of false accepts on test images of faces not in the library.

Some recent systems have been quite successful, achieving high recognition rates and using relatively large data bases of people. For example, Baron[3] reached an impressive 100% recognition rate on a library of 42 people and a false access rate of 0% on 108 images. Brunelli and Poggio's system[7] achieved a recognition rate of 100% on frontal views of 47 people. Cannon, *et al.*[9] report a 96% recognition rate on a library of 50, and Turk and Pentland[38] report a 96% recognition rate when their system, which uses a library of only 16 people, is tested under varying lighting conditions.

Needless to say, these recognition statistics are meaningful only if the library of model faces is sufficiently large. While there is no consensus on the sufficient size of the model database, some of the more recent approaches ([25], [6], [27]) have used libraries on the order of 70 people or more.

1.3 Our view-based recognizer

As discussed in the previous section, not much work has taken face recognizers beyond the narrow imaging conditions of expressionless, frontal views of faces with controlled lighting. More research is needed to enable automatic face recognizers to run under less stringent imaging conditions. Our goal is to build a face recognizer that works under varying pose, the difficult part of which is to handle face rotations in depth. Building on successful template-based systems, our basic approach is to represent faces with templates from multiple model views that cover different poses from the viewing sphere.

Our face recognizer deals with the problem of arbitrary pose by applying a feature finder and pose estimation module before recognition. As mentioned for existing work, one can normalize the input image for translation, scale, and image-plane rotation by detecting the eyes and then applying a similarity transform to place the eyes at known locations. The remaining pose parameters, rotations in depth, can be estimated by a pose module and then used to select model views similar in pose to the input.

Our feature finder/pose estimation module finds the two eyes and a nose lobe feature and estimates the pose rotation parameters out of the image plane. The method is template-based, with tens of facial feature templates covering different poses and different people. Organizing the search over pose space in a hierarchical coarse-to-fine manner helps keep the computation time under control. To geometrically align the input face with a model view, the recognizer applies an affine transform to the input to bring the three feature points into correspondence with the same points on the model.

The template-based recognizer uses templates of the eyes, nose, and mouth to represent faces. These templates, as well as the input image, are preprocessed with a differential operator such as the gradient or Laplacian in order to provide some invariance to lighting. After the geometrical alignment step, the templates are matched against a model view using normalized correlation as a metric.

This paper is divided into three main sections. The first describes the experimental setup for taking face images under varying pose and the data base of modelling and testing faces we have acquired. Next, we discuss the feature finder/pose estimator and its performance on the entire data base. Finally, we present the template-based recognizer and the results of recognition experiments for different types of preprocessing and different scales.

2 Experimental setup

In our view-based approach for face recognition under varying pose, faces are represented using many images that cover the viewing sphere. Currently we use 15 views per person, sampling 5 left/right rotations and 3 up/down rotations, as shown in figure 1. When a subject is added to the library of faces, modelling and test image data is taken with a camera perched on top of a workstation monitor. To help collect the modelling views, we fit a large piece of posterboard around the monitor with dots indicating the viewing sphere locations being sampled. When taking the modelling views, the subject is asked to rotate his head to point his nose at each of the 15 dots. No mechanisms are used to make the subjects poses accurate relative to the ideal "dot" poses other than our oral instructions fine tuning the subject's pose. This field of dots sample the 5 left/right rotations at approximately -30, -15, 0, 15, and 30 degrees and the 3 up/down rotations at approximately -20, 0, and 20 degrees. The two rotation parameters are restricted so that the two eyes are always visible; this is why the left/right rotation parameter is not sampled beyond 30 degrees.

In addition to the 15 modelling views, 10 test views are taken per person. For these test views, the subject is instructed to choose 10 points at random within the rectangle defined by the outer border of dots. The test poses can fall close to model poses or in between them. The 10 views are divided into two groups of 5. The first group is similar to the modelling views in that only the left/right and up/down rotational parameters are allowed to vary.

For the second group of 5, the subject is allowed to introduce image-plane rotation. See figure 2 for example test views.

We currently have 62 people in the data base for a total of 930 modelling and 620 testing views. The collection of people is fairly varied, including 44 males and 18 females, people from different races, and an age range from the 20s to the 40s. We have plans in the future to expand the data base to around 100 people.

For both the modelling and testing views, the lighting conditions are fixed and consist of a 60 watt lamp near the camera supplemented by background lighting from windows and overhead lights. Facial expressions are also fixed at a neutral expression.

After taking the modelling and testing images, we manually specify the locations of the two irises, nose lobes, and corners of the mouth. These manual feature locations are used for four purposes. During batch evaluations of the feature finder, they serve as ground truth data for validating the locations returned by the feature finder. Also in the feature finder, the manual locations actually define the "interest points" – irises, lobes of the nose – within the templates used by the feature finder. For the recognizer itself, feature locations are used to automatically define the bounding boxes of facial feature templates in the model images, as will be discussed in section 4. Lastly, the recognizer also uses manual locations in the model views during the geometrical alignment step between input and model images.

3 Feature detection and pose estimation

The first stage of processing in the proposed face recognition architecture is a person-independent feature finding and pose estimation module. As mentioned in the introduction, the kind of facial features sought by the feature finder are the two eyes and at least one nose feature. The locations of these features are used to bring input faces into rough geometrical alignment with model faces. Pose estimation is used as a filter on the library models, selecting only those models whose pose is similar to the input's pose. By pose estimation we really mean an estimate of the rotation angles out of the image plane since feature locations have already been used to normalize for position, scale, and image-plane rotation. Pose estimation is really an optimization step, for even in the absence of a robust pose estimator, the system could still test the input against all model poses of all people.

3.1 Overview

While techniques already exist for finding facial features, no current system can deal with large face rotations out of the image plane, so we needed to build a system that addresses this issue. As mentioned in the introduction, existing methods for finding facial features with semantic content (i.e. the eyes or nose, as opposed to, say, a grey level interest operator) tend to fall into one of two categories, a pictorial approach and a model-based approach. In the model-based approach, however, the models and fitting procedures are ad hoc and require experimentation to fine-tune the models. The amount of work is manageable for one view but might become tedious as models and fitting rules for different views on the viewing sphere are developed. Thus, we chose to explore a template-based approach for our feature finder, primarily for its simplicity.

To serve as the front end of a pose independent face recognizer, the feature finder must, of course, handle varying pose and be person independent. The current system addresses these requirements by using a large number of templates taken from multiple poses and from different people. To handle rotations out of the image plane, templates from different views on the viewing sphere are used. Templates from different scales and image-plane rotations can be generated by using standard 2D rotation and scaling operations. To make the feature finder person independent, the templates must cover identity-related variability in feature appearance (e.g. tip of nose slanted up versus down, feature types specific to certain races). I use templates from a variety of exemplar faces that sample these basic feature appearances. The choice of exemplars was guided by a simple clustering algorithm that measures face similarity though correlation.

Our feature finder, then, entails correlation with a large number of templates sampling different poses and exemplars. To keep this search under control, we use a hierarchical coarse-to-fine strategy on a 5 level pyramid representation of the image. In what follows level 0 refers to the original image resolution while level 4 refers to the coarsest level. The search begins by generating face location hypotheses at level 4, where the pose parameters are very coarsely sampled and only one exemplar is used. Exploring a level 4 hypothesis is organized as a tree search through the finer pyramid levels. As processing proceeds to finer levels, the pose parameters are sampled at a higher resolution and the different exemplars are used. A branch at any level in the search tree is pruned if the template correlation values are not above a level-dependent threshold.

The tree searching strategy starts out as a breadth first search at the coarser levels where the correlation scores are not entirely reliable. As processing reaches lower levels in the pyramid, correlation scores become more reliable and the search strategy switches to depth first. Search at levels 4 and 3 is breadth first: all possible level 3 hypotheses are generated from all level 4 hypotheses and then sorted by correlation score. Then the search strategy switches to a depth first search of level 3 hypotheses. If any leaves in the search tree (at level 0) pass the template correlation threshold tests, then the search is terminated – no more level 3 hypotheses are explored – and the leaf with the highest correlation scores is reported.

3.2 Hierarchical processing

Search over different poses and exemplars through the 5 levels of the pyramid is organized as follows. At the coarsest level, level 4, the system is trying to get an estimate of the overall position of the face, so a bank of 30 different whole-face templates are correlated over the entire image. Because the resolution at this pyramid level is very coarse – the interocular distance is only around 4 pixels – the pose parameters can be sampled very coarsely, and only one exemplar is used. Currently, the system uses 5 left/right rotations (-30, -15, 0 15, 30), three image-plane rotations (-30, 0, 30), and two scales (interocular distances of 3 and 3.75). Local maxima above a certain threshold in the correlation scores generate face location hypotheses, which are explored by refining the search over pose parameters at the mid levels resolutions, levels 3 and 2.

When a pose hypothesis is being refined at level 3 or 2, pose space is explored at a higher resolution in a small neighborhood around the coarser pose estimate of the previous level. At level 3, for instance, the 5 left/right viewing sphere angles are expanded to include 3 up/down rotations (-20, 0, 20), bringing up to 15 the number of viewing sphere angles explored. Also at level 3 the image-plane rotation parameter is sampled at twice the resolution of level 4, now including 7 different rotations at 15 degree increments. The different exemplars are also tested. As mentioned before, pose space is explored in a small neighborhood around the coarse estimate of the previous level, so a level 4 hypothesis is examined at level 3 by searching over 3 up/down rotations, 3 image-plane rotations, and the different exemplars (currently 6) in a neighborhood around the level 4 correlation maxima. Pose hypotheses from levels 3 through 0 keep track of how all exemplars match the image at that pose.

For each of these level 3 hypotheses, search at level 2 occurs only if the template correlation is above a certain threshold. At level 2, the resolution of image-plane rotations is doubled again to every 7.5 degrees (for a total of 15 rotations from -52.5 to 52.5) and the search over the 3 up/down rotations is repeated. For level 2 hypotheses surviving the threshold test on the correlation values, the resolution of the image is high enough to allow estimating the locations of features, in this case the two irises and a nose lobe.

The repetition of the up/down rotation search on level 2 is done to increase the flexibility of the search – it is not always possible to make a choice on the up/down rotation at level 3, but including the extra up/down rotation templates at that level assures that true positives are not rejected by the thresholding step. In general, the level for which the decision for a pose parameter is made may either be hard to estimate or person-dependent, so while repeating a search at two adjacent levels may increase running time, it also increases system flexibility.

Processing at the finest levels of the pyramid, levels 1 and 0, are essentially verification steps. Level 2 hypotheses provide relatively good estimates of feature locations, and the finer levels use the eye locations to geometrically align the templates and image before correlating with templates. No further search over pose space or exemplars is performed. The correlation tests at these levels serve to weed out any remaining false positives; hypotheses surviving level 0, which is at the resolution of the original image, are assumed to be correct and cause termination of the depth first search.

3.3 Template matching

Templates are manually chosen from 15 modelling images of the exemplars covering the viewing sphere. A special mask-defining program is utilized to draw template boundaries over the example modelling images. As templates are defined by these binary masks, templates can be tailored to tightly encircle certain features, not being limited to square regions. Actual templates used by the feature finder vary according to the level of processing. At level 4, the system is trying to get a general estimate of the face position, so full face templates are used, templates that run from above the eyebrows to below the chin. At finer resolutions the feature finder uses multiple templates that cover smaller areas. At level 3, two templates that cover the eyes and nose region are employed, as shown in figure 3. The template in the middle handles faces where bangs come down to the eyebrows and obscure the skin above the eyebrows. At level 2, the same eye/nose masks at level 3 are used, but the template is broken up into two eye and one nose subtemplates. At level 1, the same eye/nose masks are again used, but each eye and the nose are themselves vertically divided into two subtemplates, which yields 6 subtemplates total. Level 0 uses the subtemplate set of level 1 augmented by a circular subtemplate centered around the iris center or nose lobe feature.

The correlation thresholding test is based on eye and nose features, their subtemplates, and the fact that a pose hypothesis keeps track of the different exemplars. For a particular exemplar eye or nose feature, the correlation thresholding test requires that all subtemplates of the eyes and nose features exceed the threshold. For a pose hypothesis to pass the thresholding test, there must be some combination of passing eye and nose templates; the passing templates need not come from the same exemplar. This mixing of eye and nose templates across exemplars increases the flexibility of the system, as a face whose eyes match only exemplar A and whose nose matches only exemplar B will still be allowed.

Template matching is performed by using normalized correlation on processed versions of the image and templates. Normalized correlation follows the form

$$r = \frac{<TI> - <T><I>}{\sigma(T)\sigma(I)}$$

where T is the template, I is the subportion of image being matched against, $<>$ is the mean operator, and $\sigma()$ measures standard deviation. We hope that normalized correlation will give the system some invariance to lighting conditions and the dynamic range of the camera, as the image mean and standard deviation are factored out. Correlation is normally carried out on preprocessed versions of the image and templates, again to provide for some invariance to lighting. While we have explored the x and y components of the gradient, the Laplacian, and the original grey levels, no preprocessing type has stood out as the best. Performing correlation using these different preprocessings and then summing the result, however, empirically yields more robust performance than any single type of preprocessing. Thus, the current system performs separate correlations using the grey levels, x and y components of the gradient, and Laplacian, and then sums the results.

At higher resolutions in the pyramid, the details of individual features emerge. This might foil the matching process because the features in the input will not precisely match the templates due to differences in identity and pose. For instance, the features in the input may not sufficiently close to *any* of the exemplar features, or the input features may be from a novel pose that is in between the template modelling views. In order to bring the input features into a better correspondence with the templates, we apply an image warping algorithm based on optical flow to "warp" the input features to make them look like the templates. First, the optical flow is measured between the input features and the template using the hierarchical gradient-based scheme of Bergen and Hingorani[5]. This finds a flow field between the input feature and template, which can be interpreted as a dense set of correspondences. The input feature, as shown in figure 4, is then graphically warped using the flow field to make the input feature mimic the appearance of the template. This helps to compensate for small rotational and identity-related differences between the input features and templates. Correlation is performed after the image warping step.

Final feature locations are determined from a successful level 0 match returned by the depth first search. Feature points at the center of the irises and the nose lobes, which are manually located in the templates, are mapped to the corresponding points in the input image using the correspondences from optical flow. Figure 5 shows the features located in some example test images. It is interesting to note that because correspondence from optical flow is dense, we could actually detect more than three feature points once we have brought our eye and nose templates into correspondence with the image; all we have to do is manually specify more points in the exemplar templates. We stop at three points because that is all that is needed to specify the affine transform used by the geometrical alignment stage in the recognizer.

To evaluate these feature finder locations, the system was run on all 1550 images in the data base, the 15 modelling and 10 testing images of each of the 62 people. For a particular test run, let d_{max} be the maximum distance between a detected feature and its manually chosen location. Four different feature finder outcomes were recorded: good ($d_{max} < t_{good}$), marginal ($t_{good} \leq d_{max} < t_{marginal}$), bad ($d_{max} \geq t_{marginal}$), and null (no features found; all hypotheses rejected). We chose t_{good} to be about 15% of the interocular distance d and $t_{marginal}$ to be 20% of d. In our exhaustive test of the data base, the system achieved a good outcome in 99.3% of the images, a marginal outcome in 0.3% of the images, and a bad outcome in 0.4%. No null cases were reported. The feature locations in either the good or marginal outcomes are sufficient for the geometrical alignment stage of the recognizer, so the recognizer can be run on the vast majority of the test images.

In most of the error cases, the far eye in a rotated face is misplaced, perhaps being located in a nearby dark region such as an eyebrow or a sliver of hair. Even in these cases, however, the nearer eye and the nose are correctly located. In all 1550 data base images except one, the feature finder returned at least two good features.

The pose estimated by the system is simply given by the model pose that the level 0 templates are taken from. In the present system this estimate is not always correct, primarily because the image warping based on optical flow makes matching a little *too* flexible. Sometimes the warping actually changes the pose of the input to match templates from a different pose. Since it is difficult for the warping operation to transform between leftward-looking poses and rightward-looking ones, the pose estimate can reliably distinguish between these two cases. Thus, the pose estimate passed on to the recognizer is currently "looking left" or "looking right". Even though this is a very coarse estimate, since pose estimation is only used to index the model library, we can compensate by simply letting more poses get through the indexing stage. Also, it should be possible to place a more refined pose estimation stage after feature extraction, an estimation stage that would use fixed templates and no warping operations.

Because of the large number of templates, the computation takes around 10-15 minutes on a Sun Sparc 2. Using fewer exemplars decreases the running time but also reduces system flexibility and recognition performance.

4 Face recognition using multiple views

As mentioned in the introduction, template-based face recognizers have been quite successful on frontal views of the face (Baron[3], Turk and Pentland[38], Brunelli and Poggio[7]). Our goal is to extend template-based systems to handle varying pose, notably facial rotations in depth. Our approach is view-based, representing faces with templates from many images that cover the viewing sphere. As discussed in section 2, our view-based face recognizer uses 15 views per person, sampling 5 left/right rotations and 3 up/down rotations. In this section we describe the view-based recognizer and experimental results on our data base of face images.

4.1 Input representation: templates

In order to build face models for the recognizer, templates from the eyes, nose, and mouth are extracted from the modelling images, as shown in figure 6. Before extracting the templates, scale and image-plane rotation are normalized in the model images to fix the interocular distance and eliminate any head tilt. This is done by placing the eyes, as located manually, at fixed locations in the image. Next, after the bounding boxes of the templates are automatically computed using the manually specified feature locations, the templates are extracted and stored to disk.

We have done experiments to explore two aspects of template design, model image preprocessing and template scale. As discussed previously in the introduction, it is common in face recognition to preprocess the templates to introduce some invariance to lighting conditions. So far we have tested preprocessing with the gradient magnitude, Laplacian, and x and y components of the gradient, as well as the original grey levels. The overall scale of the templates, as measured by the interocular distance, is another design parameter we examined. These experiments on preprocessing and scale will be described in the experimental results section.

4.2 Recognition algorithm

Our template-based recognizer takes as input a view of an unidentified person, compares it against all the people in the library, and returns the best match. Naturally, since we are exploring techniques for modelling varying pose, the face in the input image can be rotated away from the camera. The main constraint on input pose is that both eyes are visible.

Pseudocode sketching the steps of our recognizer is given in figure 7. First, in step (1), the pose calculated by the feature finder/pose estimation module acts as a filter on the model poses: only those model poses that are similar to the input pose will be selected. Since our current implementation of the pose estimator can only distinguish between looking left and looking right, the poses selected by the recognizer for comparison are either the left three columns or right three columns of figure 1. In the future a more refined pose estimate will allow the recognizer to further winnow down the number of model poses it needs to test for each person.

Next, in steps (2) and (3) the recognizer loops over the selected poses of all model people, recording template correlation scores from each of these model views in the **cor** array. The main part of the recognizer, steps (4)-(6), compares the input image against a particular model view. This comparison consists of a geometrical alignment step (step (4)) followed by correlation (steps (5)-(6)). The geometrical alignment step brings the input and model images into close spatial correspondence in preparation for the correlation step. To geometrically align the input image against the model image, first an affine transform is applied to the input to align three feature points, currently the two eyes and a nose lobe feature. In the input image these features are automatically located using the feature finder described in the previous section. For the models, manual feature locations are used. Figure 8 shows an example input image and the result of affine transforming the image to align its features with those of the model in figure 6.

The second part of the geometrical alignment step attempts to compensate for any small remaining geometrical differences due to rotation, scale, or expression. A dense set of pixelwise correspondence between the affine transformed input and the model is computed using optical flow [5]. Given this dense set of correspondences, the affine transformed input can be brought into pixel-level correspondence with the model by applying a 2D warp operation driven by the optical flow (also see Shashua[35]). Basically, pixels in the affine transformed input are "pushed" along the flow vectors to their corresponding pixels in the model. In figure 9, we first compute optical flow between the affine transformed input (left, from figure 8) and the model image (middle, from figure 6). Then a 2D warp driven by the optical flow is applied to the affine transformed input, which produces the result on the right. When the input and model are the same person, optical flow succeeds in finding correspondence and can compensate for small rotation, scale, and expression differences between the affine transformed input and model. When the input and model are different, optical flow can fail to find correct correspondence, in which case the 2D warp distorts the image and the template match will be poor. This failure case, however, does not matter since we want to reject the match anyway.

Now that the input and model image have been geometrically registered, in steps (5) and (6) the eye, nose, and mouth model templates are correlated against the input. Each model template is correlated over a small region (e.g. 5x5) centered around its expected location in the input. Normalized correlation is the matching metric, and it is of the same form described in section 3 on feature detection. We use normalized correlation because it factors out differences in template mean and standard deviation, which might be caused by differences in lighting.

When scoring a person in step (7), the system takes the sum of correlations from the best matching eye, nose, and mouth templates. Note that we maximize over the poses separately for each template, so the best matching left eye could be from pose 1 and the best matching nose from pose 2, and so on. We found that switching the order of the sum and max operations – first summing template scores and then maximizing over poses – gives slightly worse performance, probably because the original sum/max ordering is more flexible.

After comparing the input against all people in the library, the recognizer returns the person with the highest correlation score – we have not yet developed a criterion on how good a match has to be to be believable. Considering a task like face verification, however, having the ability to reject inputs is important and is something we plan under future work.

4.3 Experimental results

As mentioned previously in section 4.1 on template design, we have tested our face recognizer under different template resolutions and methods of preprocessing. For each recognition experiment, we ran the recognizer on our data base of 620 test images, 10 images each of 62 people. The recognition experiments use the eyes and nose features found by our feature finder to drive the geometrical alignment stage. Of the 620 test images in our data base, the feature finder returns a bad result for two images. As we run the recognizer on those test images for which the feature finder produces a good or marginal result, these two test images are excluded from the recognition tests. These excluded images are listed in the rightmost column of tables 1 and 2.

Table 1 summarizes our recognition results for the preprocessing experiments. The types of preprocessing we tested include the gradient magnitude (mag), Laplacian (lap), sum of separate correlations on x and y components of the gradient (dx+dy), and the original grey levels (grey). For these preprocessing experiments we used an intermediate template scale, an interocular distance of 30. In table 1, we list the number of correct recognitions and the number of times the correct person came in second, third, or past third place. Best performance was had from dx+dy, mag, and lap, with dx+dy yielding the best recognition rate at 98.7%. Preprocessing with the gradient magnitude performs nearly as well, a result in agreement with the preprocessing experiments of Brunelli and Pogggio[7]. Given that using the original grey levels produces the lower rate of 94.5%, our results indicate that preprocessing the image with a differential operator gives the system a performance advantage. We think the performance differences between dx+dy, mag, and lap are too small to say that one preprocessing type stands out over the others.

Table 2 summarizes our recognition results for the template scale experiments, where scale is measured by the interocular distance of a frontal view. The preprocessing was fixed at dx+dy. The intermediate and fine scales perform the best, indicating that at least for our input representation, the coarsest scale may be losing detail needed to distinguish between people. Since the intermediate scale has a computational advantage over the finer scale, we would recommend operating a face recognizer at the intermediate scale.

Consider the errors made for the best combination of preprocessing and scale, dx+dy at an intermediate scale. Of the 8 errors, 2 were due to the feature finder and 6 were recognition errors. In the one recognition error where the correct person was not even among the top three, the correspondences from optical flow were poor. For the other errors, the correct person came in either second or third place. For these false positive matches, using optical flow to warp the input to the model may be contributing to the problem. If two people are similar enough, the optical flow can effectively "morph" one person into the other, making the matcher a bit *too* flexible at times.

The problem with optical flow sometimes making the matcher too flexible suggests some extensions to the recognizer. Since we only want to compensate for rotational, scale, or expression changes and not allow "identity-changing" transforms, perhaps the optical flow can be interpreted and the match discarded if the optical flow is not from the allowed class of transformations. Another approach would be to penalize a match using some smoothness measure of optical flow. The new matching metric would have a regularized flavor, being the sum of correlation and smoothness terms

$$||I(x + \Delta x) - T||^2 + \lambda\phi(\Delta x),$$

where $I(x + \Delta x)$ is the input warped by the flow Δx, T is the template, ϕ is a smoothness functional including derivatives, and λ is a parameter controlling the trade off between correlation and smoothness. This functional has an interpretation as the combination of a noise model on the intensity image and priors on the flow.

Besides adding constraints on the flow-based correspondences, another technique for increasing the overall discrimination power of the face representation would be to add information about face geometry. A geometrical feature vector of distances and angles that is similar to current feature geometry approaches could be tried, but the representation would have to be extended to deal with varying pose.

In terms of execution time, our current system takes about 1 second to do each input/model comparison on a Sun Sparc 1. The computation time is dominated by resampling the image during the affine transform, optical flow, and correlation. On our unoptimized CM-5 implementation, it takes about 10 seconds for the template-based recognizer to run since we can distribute the data base so that each processor compares the input against one person. Specialized hardware, for example correlation chips[42], can be used to further speed up the computation.

5 Conclusion

In this paper we presented a view-based approach for recognizing faces under varying pose. Motivated by the success of recent template-based approaches for frontal views, our approach models faces with templates from 15 views that sample different poses from the viewing sphere. The recognizer consists of two main stages, a geometrical alignment stage where the input is registered with the model views and a correlation stage for matching. Our recognizer has achieved a recognition rate of 98% on a data base 62 people. The data base consists of 930 modelling views and 620 testing views covering a variety of poses, including rotations in depth and rotations in the image plane.

We have also developed a facial feature finder to provide feature locations for the geometrical alignment stage in the recognizer. Like the recognizer, our feature finder is template-based, employing templates of the eyes and nose regions to locate the two irises and one nose lobe feature. Since the feature finder runs before the recognizer, the feature finder must be pose independent and work for a variety of people. We satisfy this requirement by using a large set of templates from many views and across many people. While the features are currently used to register input and model views, the feature finder has other applications. For instance, it could be used to initialize a facial feature tracker, finding the feature locations in the first frame. This would be useful for virtual reality, HCI, and low bandwidth teleconferencing.

In the future, we plan on adding more people to the data base and adding a rejection criterion to the recognizer. We would also like to improve the estimate of pose returned by the feature finder. A better pose estimate will enable the recognizer to search over a smaller set of model poses.

In a related line of research, we plan on addressing the problem of recognizing a person's face under varying pose when only *one* view of the person is available. This will be useful in situations where you do not have the luxury of taking many modelling images. The key to making this work will be an example-based learning system that uses multiple images of prototype faces undergoing changes in pose to "learn" what it means to rotate a face (see Poggio[29], Poggio and Vetter[30]). The system will apply this knowledge to synthesize new "virtual" views of the person's face.

Overall, we have demonstrated in this paper that template-based face recognition systems can be extended in a straightforward way to deal with the problem of varying pose. However, to make a truly general face recognition system, more work needs to be done, especially to handle variability in expression and lighting conditions.

Acknowledgments

I would like to thank my advisor, Tomaso Poggio, for his support and encouragement to use the template-based approach. Thanks also to Amnon Shashua for our many discussions and for his suggestion that I use optical flow in the geometrical alignment step.

References

[1] Shigeru Akamatsu, Tsutomu Sasaki, Hideo Fukamachi, Nobuhiko Masui, and Yasuhito Suenaga. An accurate and robust face identification scheme. In *Proceedings Int. Conf. on Pattern Recognition*, volume 2, pages 217–220, The Hague, The Netherlands, 1992.

[2] A. Azarbayejani, T. Starner, B. Horowitz, and A. Pentland. Visually controlled graphics. Technical Report No. 180, MIT Media Lab, Vision and Modeling Group, 1992.

[3] Robert J. Baron. Mechanisms of human facial recognition. *International Journal of Man Machine Studies*, 15:137–178, 1981.

[4] Alan Bennett and Ian Craw. Finding image features using deformable templates and detailed prior statistical knowledge. In *Proc. British Machine Vision Conference*, pages 233–239, 1991.

[5] J.R. Bergen and R. Hingorani. Hierarchical motion-based frame rate conversion. Technical report, David Sarnoff Research Center, Princeton, New Jersey, April 1990.

[6] Martin Bichsel. *Strategies of Robust Object Recognition for the Automatic Identification of Human Faces.* PhD thesis, ETH, Zurich, 1991.

[7] Roberto Brunelli and Tomaso Poggio. Face recognition: Features versus templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(10):1042–1052, 1993.

[8] Peter J. Burt. Multiresolution techniques for image representation, analysis, and 'smart' transmission. In *SPIE Vol. 1199, Visual Communications and Image Processing IV*, pages 2–15, 1989.

[9] Scott R. Cannon, Gregory W. Jones, Robert Campbell, and Neil W. Morgan. A computer vision system for identification of individuals. In *Proc. IECON*, pages 347–351, Milwaukee, WI, 1986.

[10] Chin-Wen Chen and Chung-Lin Huang. Human face recognition from a single front view. *International Journal of Pattern Recognition and Artificial Intelligence*, 6(4):571–593, 1992.

[11] Yong-Qing Cheng, Ke Liu, Jing-Yu Yang, and Hua-Feng Wang. A robust algebraic method for human face recognition. In *Proceedings Int. Conf. on Pattern Recognition*, volume 2, pages 221–224, The Hague, The Netherlands, 1992.

[12] T.F. Cootes, C.J. Taylor, A. Lanitis, D.H. Cooper, and J. Graham. Building and using flexible models incorporating grey-level information. In *Proceedings of the International Conference on Computer Vision*, pages 242–246, Berlin, May 1993.

[13] Ian Craw and Peter Cameron. Face recognition by computer. In David Hogg and Roger Boyle, editors, *Proc. British Machine Vision Conference*, pages 498–507. Springer Verlag, 1992.

[14] Ian Craw, David Tock, and Alan Bennett. Finding face features. In *Proceedings of the European Conference on Computer Vision*, pages 92–96, 1992.

[15] Shimon Edelman, Daniel Reisfeld, and Yechezkel Yeshurun. Learning to recognize faces from examples. In *Proceedings of the European Conference on Computer Vision*, pages 787–791, 1992.

[16] Michael K. Fleming and Garrison W. Cottrell. Categorization of faces using unsupervised feature extraction. In *Proceedings of the International Joint Conference on Neural Networks*, volume 2, pages 65–70, 1990.

[17] A. Fuchs and H. Haken. Pattern recognition and associative memory as dynamical processes in a synergetic system; I. translational invariance, selective attention, and decomposition of scenes. *Biological Cybernetics*, 60:17–22, 1988.

[18] Peter W. Hallinan. Recognizing human eyes. In *SPIE Vol. 1570, Geometric Methods in Computer Vision*, pages 214–226, 1991.

[19] Zi-Quan Hong. Algebraic feature extraction of image for recognition. *Pattern Recognition*, 24(3):211–219, 1991.

[20] Chung-Lin Huang and Chin-Wen Chen. Human facial feature extraction for face interpretation and recognition. *Pattern Recognition*, 25(12):1435–1444, 1992.

[21] Takeo Kanade. Picture processing by computer complex and recognition of human faces. Technical report, Kyoto University, Dept. of Information Science, 1973.

[22] Y. Kaya and K. Kobayashi. A basic study on human face recognition. In Satosi Watanabe, editor, *Frontiers of Pattern Recognition*, pages 265–289. Academic Press, New York, NY, 1972.

[23] M. Kirby and L. Sirovich. Application of the Karhunen-Loeve procedure for the characterization of human faces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(1):103–108, 1990.

[24] T. Kohonen. *Self-organization and Associative Memory.* Springer-Verlag, Berlin, 1989.

[25] T. Kurita, N. Otsu, and T. Sato. A face recognition method using higher order local autocorrelation and multivariate analysis. In *Proceedings Int. Conf. on Pattern Recognition*, volume 2, pages 213–216, The Hague, The Netherlands, 1992.

[26] Martin Lades, Jan C. Vorbruggen, Joachim Buhmann, Jorg Lange, Christoph v.d. Malsburg, Rolf P. Wurtz, and Wolfgang Konen. Distortion invariant object recognition in the dynamic link architecture. preprint, August 1991.

[27] B.S. Manjunath, R. Chellappa, and C. von der Malsburg. A feature based approach to face recognition. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition*, pages 373–378, 1992.

[28] B.S. Manjunath, Chandra Shekhar, R. Chellappa, and C. von der Malsburg. A robust method for detecting image features with application to face recognition and motion correspondence. In *Proceedings Int. Conf. on Pattern Recognition*, volume 2, pages 208–212, The Hague, The Netherlands, 1992.

[29] T. Poggio. 3D object recognition and prototypes: one 2D view may be sufficient. Technical Report 9107–02, I.R.S.T., Povo, Italy, July 1991.

[30] Tomaso Poggio and Thomas Vetter. Recognition and structure from one 2D model view: Observations on prototypes, object classes, and symmetries. A.I. Memo No. 1347, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 1992.

[31] C.S. Ramsay, K. Sutherland, D. Renshaw, and P.B. Denyer. A comparison of vector quantization codebook generation algorithms applied to automatic face recognition. In David Hogg and Roger Boyle, editors, *Proc. British Machine Vision Conference*, pages 508–517. Springer Verlag, 1992.

[32] Daniel Reisfeld and Yehezkel Yeshurun. Robust detection of facial features by generalized symmetry. In *Proceedings Int. Conf. on Pattern Recognition*, volume 1, pages 117–120, The Hague, The Netherlands, 1992.

[33] M. Dalla Serra and R. Brunelli. On the use of the Karhunen-Loeve expansion for face recognition. Technical Report 9206-04, I.R.S.T., 1992.

[34] M.A. Shackleton and W.J. Welsh. Classification of facial features for recognition. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition*, pages 573–579, Lahaina, Maui, Hawaii, 1991.

[35] A. Shashua. Correspondence and affine shape from two orthographic views: Motion and Recognition. A.I. Memo No. 1327, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, December 1991.

[36] T.J. Stonham. Practical face recognition and verification with WISARD. In M. Jeeves, F. Newcombe, and A. Young, editors, *Aspects of Face Processing*, pages 426–441. Martinus Nijhoff Publishers, Dordrecht, 1986.

[37] Demetri Terzopoulos and Keith Waters. Analysis of facial images using physical and anatomical models. In *Proceedings of the International Conference on Computer Vision*, pages 727–732, Osaka, Japan, December 1990.

[38] Matthew Turk and Alex Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

[39] J.M. Vincent, J.B. Waite, and D.J. Myers. Location of feature points in images using neural networks. *BT Technology Journal*, 10(3):7–15, July 1992.

[40] John J. Weng, N. Ahuja, and T.S. Huang. Learning recognition and segmentation of 3-D objects from 2-D images. In *Proceedings of the International Conference on Computer Vision*, pages 121–128, Berlin, May 1993.

[41] K.H. Wong, Hudson H.M. Law, and P.W.M. Tsang. A system for recognizing human faces. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, pages 1638–1642, 1989.

[42] Woody Yang and Jeff Gilbert. A real-time face recognition system using custom VLSI hardware. In *IEEE Computer Architectures for Machine Vision Workshop*, December 1993.

[43] Alan L. Yuille, Peter W. Hallinan, and David S. Cohen. Feature extraction from faces using deformable templates. *International Journal of Computer Vision*, 8(2):99–111, 1992.

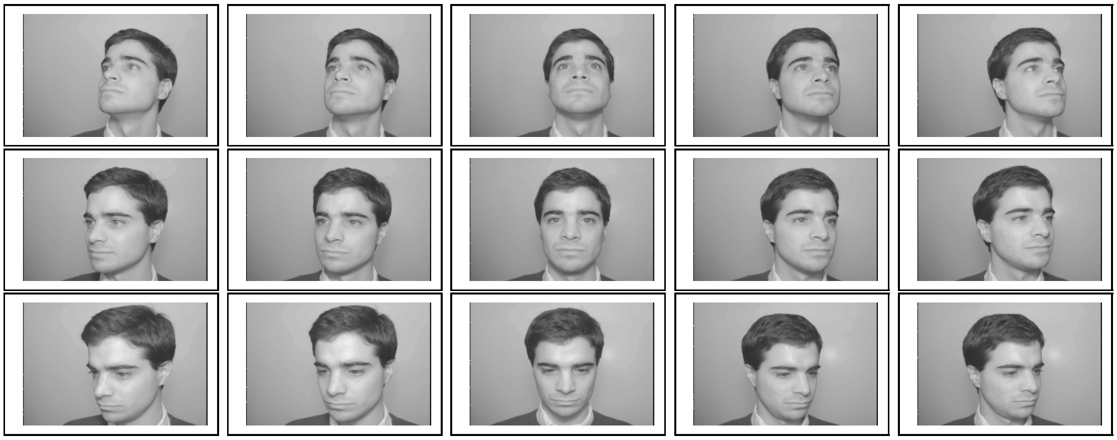

Figure 1: The view-based face recognizer uses 15 views to model a person's face.

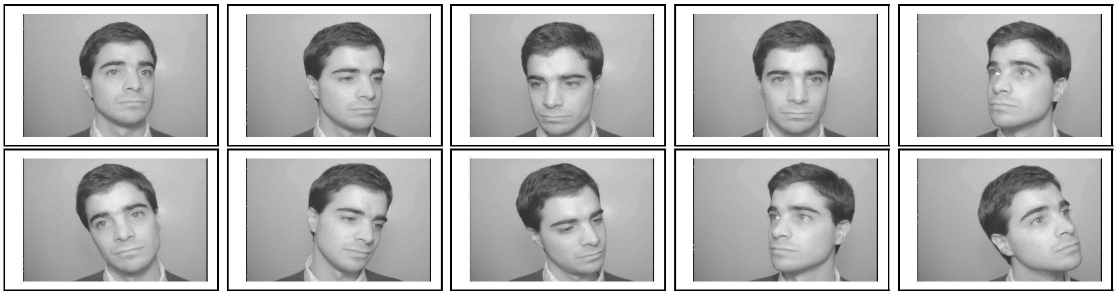

Figure 2: For each person, 10 test images are taken that sample random poses from the viewing sphere.

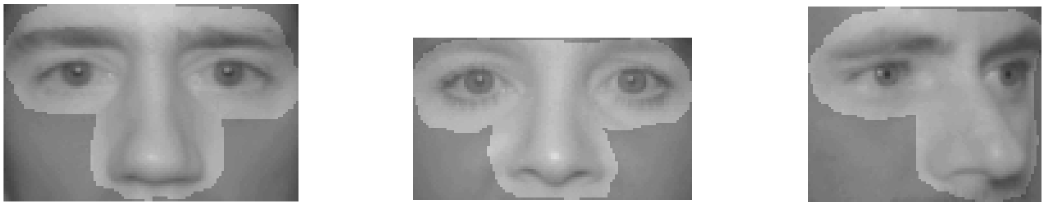

Figure 3: Example templates of the eyes and nose used by the feature finder.

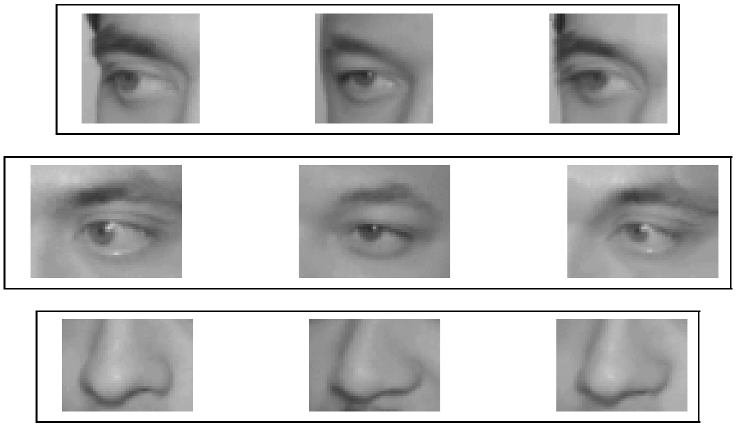

Figure 4: In the feature finding process, an extracted portion of the input (left) is brought into pixel level correspondence with a template (middle) using an optical flow algorithm. The input is then warped to make it mimic the geometry of the template (right).

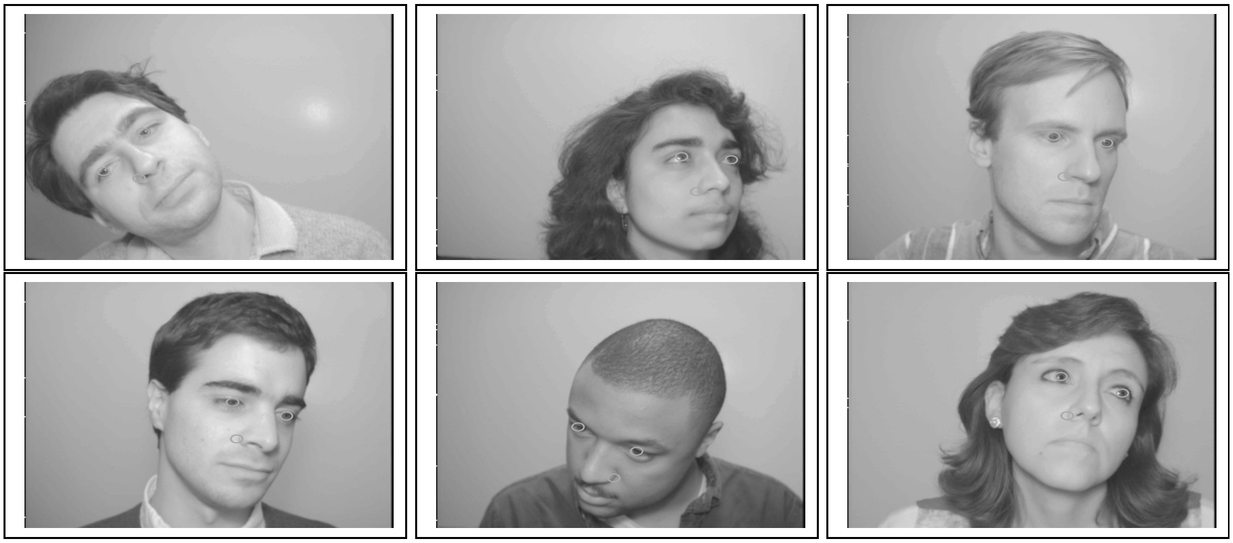

Figure 5: Iris and nose lobe features located by the feature finder in some example test images.

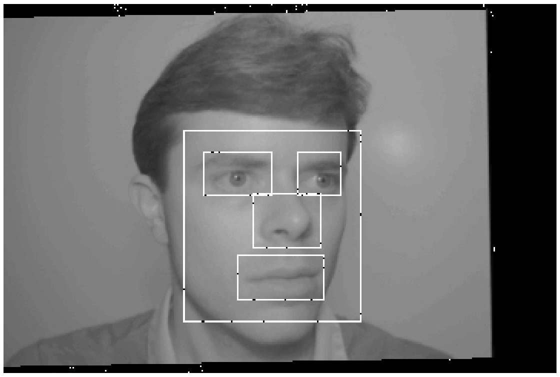

Figure 6: Templates of the eyes, nose, and mouth are used to represent faces.

Template-based recognizer

```
selected poses ← left or right group of poses, from pose estimator
for person ← 1 to NUM_PEOPLE                          /* for all people in data base */
    forall pose ∈ selected poses                      /* for all poses to search */
        align input to model pose: affine transform & optical flow
        for template ← 1 to NUM_TEMPLATES             /* loop over eyes, nose, mouth */
            cor[person][pose][template] ← correlation value
    score[person] ← $\sum_{\mathbf{template}=1}^{NUM\_TEMPLATES} ( \max_{\mathbf{pose} \in \mathbf{selected\ poses}} (\mathbf{cor}[\mathbf{person}][\mathbf{pose}][\mathbf{template}]))$
```

Figure 7: Pseudocode for our template-based recognizer.

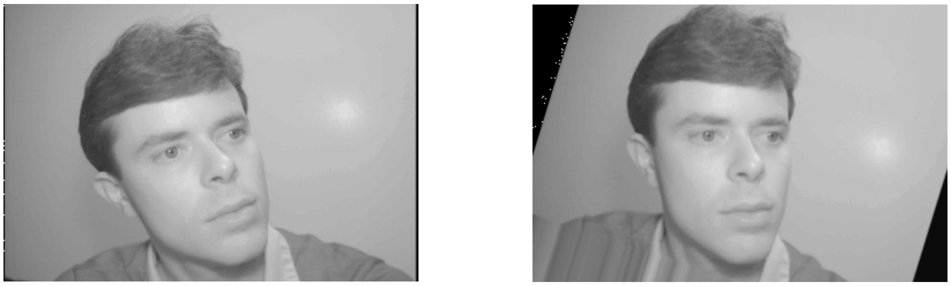

Figure 8: An example input image and the result of applying an affine transform to bring into correspondence the two eyes and a nose feature with the model face in figure 6.

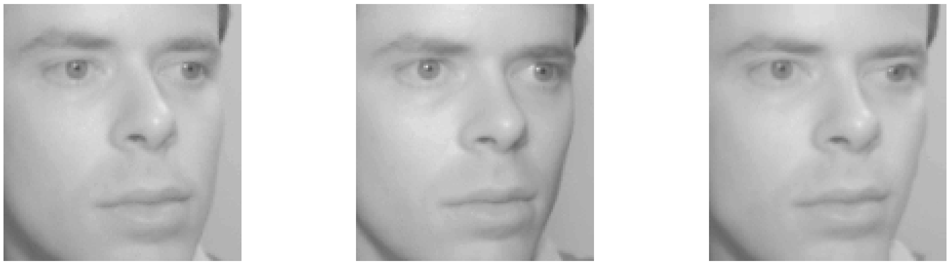

Figure 9: Using a 2D warp driven by the optical flow between the affine transformed input (left, from figure 8) and the model image (middle, from figure 6), the system warps the affine transformed input to produce the image on the right.

	performance – 620 test images				
preprocessing	correct	2nd place	3rd place	>3rd place	bad features
dx+dy	98.71% (612)	0.32% (2)	0.48% (3)	0.16% (1)	0.32% (2)
mag	98.23% (609)	0.81% (5)	0.32% (2)	0.32% (2)	0.32% (2)
lap	98.07% (608)	0.81% (5)	0.32% (2)	0.48% (3)	0.32% (2)
grey	94.52% (586)	1.94% (12)	0.48% (3)	2.74% (17)	0.32% (2)

Table 1: Face recognition performance versus preprocessing. Best performance is from using the gradient magnitude (mag), Laplacian (lap), or the sum of separate correlations on the x and y gradient components (dx+dy). An intermediate scale was used, with an interocular distance of 30.

	performance – 620 test images				
interocular distance	correct	2nd place	3rd place	>3rd place	bad features
15	96.13% (596)	2.26% (14)	0.32% (2)	0.97% (6)	0.32% (2)
30	98.71% (612)	0.32% (2)	0.48% (3)	0.16% (1)	0.32% (2)
60	98.39% (610)	0.81% (5)	0.16% (1)	0.32% (2)	0.32% (2)

Table 2: Face recognition performance versus scale, as measured by interocular distance (in pixels). The intermediate scale performs the best, a result in agreement with Brunelli and Poggio[7]. For preprocessing, separate correlations on the x and y components of the gradient were computed and then summed (dx+dy).