[In Proceedings of Workshop on Models versus Exemplars in Computer Vision, CVPR 2001]

Combining Models and Exemplars for Face Recognition: An Illuminating Example

Terence Sim Takeo Kanade
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

We propose a model- and exemplar-based approach for face recognition. This problem has been previously tackled using either models or exemplars, with limited success. Our idea uses models to synthesize many more exemplars, which are then used in the learning stage of a face recognition system. To demonstrate this, we develop a statistical shape-from-shading model to recover face shape from a single image, and to synthesize the same face under new illumination. We then use this to build a simple and fast classifier that was not possible before because of a lack of training data.

Keywords: model-based, exemplar-based, shape-from-shading, face recognition, eigenspace

1. Introduction

The task of automatic face recognition has been actively researched in recent years, with researchers employing both model-based and exemplar-based techniques. Although great strides have been made after almost three decades, the task remains unsolved in general. Current systems work very well whenever the task image to be recognized is captured under conditions similar to those of the training images. However, they are not robust enough if there is variation between task and training images [14]. Changes in incident illumination, head pose, facial expression, hairstyle (including facial hair), cosmetics (including eyewear) and age, all confound the best systems today.

As a general rule, we may categorize approaches used to cope with variation in appearance into three kinds: invariant features, canonical forms, and variation-modelling. The first approach seeks to utilize features that are invariant to the changes being studied. For instance, the Quotient Image [15] is (by construction) invariant to illumination and may be used to recognize faces (assumed to be Lambertian) when lighting conditions change. The second approach attempts to "normalize" away the variation, either by clever image transformations or by synthesizing a new image (from the given task image) in some "canonical" or "prototypical" form. Recognition is then performed using this canonical form. Examples of this approach include [23, 24]. In [23], for instance, the task image under arbitrary illumination is re-rendered under frontal illumination, and then compared against other frontally-illuminated prototypes. The third approach of variation-modelling is self-explanatory: the idea is to learn, in some suitable subspace, the extent of the variation in that space. This usually leads to some parameterization of the subspace(s). Recognition is then performed by choosing the subspace closest to the task image, after the latter has been appropriately mapped. In effect, the recognition step recovers the variation (e.g. pose estimation) as well as the identity of the person. For examples of this technique, please see [6, 8, 9, 21].

Despite the plethora of techniques, and the valiant effort of many researchers, face recognition remains a difficult, unsolved problem in general. While each of the above approaches works well for the specific variation being studied, performance degrades rapidly when other variations are present. For instance, a feature invariant to illumination works well as long as pose or facial expression remains constant, but fails to be invariant when pose or expression is changed. This is not a problem for some applications, such as controlling access to a secured room, since both the training and task images may be captured under similar conditions. However, for general, unconstrained recognition, none of these techniques are robust enough. Moreover, it is not clear that different techniques can be combined to overcome each other's limitations. Some techniques, by their very nature, exclude others. For example, the Symmetric Shape-from-Shading method of [23] relies on the approximate symmetry of a frontal face. It is unclear how this may be combined with a technique that depends on side profiles, where the symmetry is absent.

We can make two important observations after surveying the research literature: (1) There does not appear to be any

feature, set of features, or subspace, that is simultaneously invariant to all the variations that a face image may exhibit. (2) Given more training images, almost any technique will perform better. These two factors are the major reasons why face recognition is not widely used in real-world applications. The fact is that for many applications, it is usual to require the ability to recognize faces under different variations, even when training images are severely limited.

Another way to categorize face recognition techniques is to consider whether they are based on models or exemplars. Models are used in [15] to compute the Quotient Image, and in [6] to derive their Active Appearance Model. These models capture prior class information (the class of faces), and provide strong constraints when dealing with appearance variation. At the other extreme, exemplars may also be used for recognition. The ARENA method in [18] simply stores all training images and matches each one against the task image. Obviously, scalability is an issue. However, the visitor identification system reported in [19] that uses ARENA appears to naturally handle illumination changes, as exemplars under new lighting conditions are added to the system. Some subspace-methods are also based on exemplars. For example, the eigenface approach in [20] and the Fisherface method in [3] embed exemplars in their respective subspaces, and recognition is performed via nearest-neighbor classification in these subspaces.

As far as we can tell, current methods that employ models do not use exemplars, and vice versa. This is surprising, since these two approaches are by no means mutually exclusive, and we can leverage the strengths of each to better solve the problem. In this paper, we propose a way of combining models and exemplars for face recognition. We will use models to synthesize additional training images, which can then be used as exemplars in the learning stage of a face recognition system. To illustrate this, we develop a model that recovers the shape of a face from a single image under an arbitrary but unknown illumination, and that re-renders the same face under new illumination [17]. We then synthesize many more images to be used for learning in a simple eigenface classifier.

2. Shape-from-Shading Model

We begin by developing a model that allows us to recover the face shape, in terms of surface normals, from a *single* image. We assume that the image is taken under a single point light source at infinity. The direction of the light source is unknown. This shape-from-shading problem has had a long history [10], and is in general underconstrained: many 3D shapes give rise to the same 2D image. To overcome this, we build a statistical model to guide the shape-recovery process. Similar work is reported in [23, 7, 8], but we differ in a few key areas. Unlike [23], we do not explicitly depend on face symmetry, so our method will work for other non-frontal poses too. And unlike [8], we do not require multiple images to be available. We work with as few as one image, which means we cannot use photometric stereo. In [7], a method is also presented to recover surface normals from a single image, using symmetry and integrability constraints. By contrast, both these contraints are implicit in our statistical model, rather than explicit. We argue that this affords us greater latitude to use our method for other non-symmetric objects, or even to exploit other symmetries in our statistical model which may not be readily apparent. Another difference is that we do not require faces to be strictly Lambertian; instead, we model the non-Lambertian component statistically. In terms of rendering the recovered face under new illumination, our technique is similar to the Quotient Image of [15] in that we also rely on class-based information. In their case, it takes the form of face images taken under 3 linearly independent illumination directions. In ours, the class-based information is a statistical model of the variation of surface normals from person to person, and of the non-Lambertian component of surface reflection. Our statistical model is explained in detail in [17], but for completeness we will describe it here.

2.1. Augmented Lambertian equation

At the heart of our method is the following equation, which is the standard Lambertian equation [11] augmented with an additive term. This is done because the standard Lambertian equation does not handle shadows nor specular reflections, which occur naturally in face images. The augmented model is then:

$$i(x) = n(x)^\top s + e \tag{1}$$

which says that at pixel position x, the pixel intensity, $i \in \mathcal{R}$, is related to dot product of the surface normal (including albedo) at that pixel, $n \in \mathcal{R}^3$, and the single light source, $s \in \mathcal{R}^3$, plus an error term $e \in \mathcal{R}$. The purpose of this error term is to model shadows and specular reflections, without explicitly recovering the full 3D shape (depth) of the face. It is clear that $e = e(x, s)$, since it is just the difference between i and $n^\top s$. We assume that surface normals at different pixels are independent of one another. For the error term however, we assume that $e(x, s)$ is independent of $e(y, s)$ for other pixels y, but correlated with $e(x, t)$ under some other illumination t. This assumption allows us to synthesize more realistic images while remaining computationally tractable. More details will be given in Section 2.2. Note that Equation (1) is really a system of equations, one equation for every pixel position $x = 1, \cdots, d$ in the image. Note also that unlike [15], our model does not assume that every person has the same face shape, nor constant albedo,

Figure 1: Two Yale faces under the same set of four illumination directions.

but allows both to vary from pixel to pixel and from person to person.

Given only a single d-pixel input image, Equation (1) is underconstrained: there are $4d + 3$ unknowns (the components of the vectors n and s, and the scalars e) but only d equations. Moreover, there is an inherent ambiguity, since we may insert any 3×3 invertible matrix A and its inverse to get $n^\top s = n^\top (AA^{-1})s = (n^\top A)(A^{-1}s) = \tilde{n}^\top \tilde{s}$. We will use knowledge about the class of faces in general to solve for the unknowns. More specifically, we will learn a statistical model for $n(x)$ and $e(x, s)$ from a set of bootstrap images. This is the Yale dataset [8], comprising 15 people each taken under roughly 60 known illumination directions (Figure 1). Some basic preprocessing was done: all faces were aligned to an arbitrarily chosen reference face by manually marking 3 points (the centers of the eyes, and the base of the nose) and performing an affine warp; the grayscale images were then cropped to remove the background and scaled to 120×100 pixels.

2.2. Learning the statistical model

Our statistical model is to learn the probability density function (pdf) for $n(x)$ and $e(x, s)$. We assume that the pdf's are Gaussian distributions of unknown means and covariances, which we will estimate using maximum likelihood. This turns out to be the sample mean and sample covariance, computable from the bootstrap images. More precisely, let B be a $d \times m$ matrix whose columns are the d-dimensional images taken under illumination directions $\{s_j\}_{j=1}^m$. Let N be a $3 \times d$ matrix whose columns are the vectors $\{n(x)\}_{x=1}^d$. Also, let S be a $3 \times m$ matrix of the illumination directions, and let E be a $d \times m$ matrix of the error terms. Then for each person in the bootstrap set, we compute the least-squares solution for N and E as follows:

$$\begin{aligned} & B = N^\top S + E \\ \Rightarrow \quad & N = (SS^\top)^{-1} SB^\top \\ \text{and} \quad & E = B - N^\top S \end{aligned} \tag{2}$$

From this we compute the sample mean vector $\mu_n(x)$ and sample covariance matrix $C_n(x)$ for $n(x)$. Since $e(x, s)$ is a scalar, we compute the sample mean $\mu_e(x, s)$ and sample *variance* $\sigma_e^2(x, s)$. To produce better synthetic images, we also compute the correlation coefficient ρ_{jk} between $e(x, s_j)$ and $e(x, s_k)$. Statistically, we are modelling $e(x, s_j)$ and $e(x, s_k)$ as a jointly Gaussian distribution, with correlation coefficient ρ_{jk}. We will make use of this in Section 3.4.

3. Using the Model

Having learned the statistical model, we make use of it in the following algorithm:

1. Given an image, estimate the unknown illumination s (which may be different from those in the bootstrap set).
2. Compute $\mu_e(x, s)$ and $\sigma_e^2(x, s)$
3. Recover $n(x)$ at each x by computing the *maximum a posteriori* (MAP) estimate, $n_{\text{MAP}}(x) = \arg\max_{n(x)} \Pr(n(x)|i(x))$.
4. Synthesize a new image $i'(x)$ under novel illumination s' using $n_{\text{MAP}}(x)$ and the joint statistics of $e(x, s)$ and $e(x, s')$. Note again that s' may be different from the illumination in the bootstrap set.

The next few sections elaborate on these steps.

3.1. Estimating s

Estimating the unknown illumination is a well-studied problem [22, 25], since it is part of the shape-from-shading problem. This turns out to be easier than recovering the shape. For our purposes, we use the simple method of kernel regression [1]. We note that since the bootstrap set is labeled with known illumination, we can recover s by viewing it as a continuous-valued classification problem. More precisely, we first store all the J training images, $\{a_j\}_{j=1}^J$, along with their labeled illumination, $\{s_j\}_{j=1}^J$. Given a new image, b, we recover its illumination s using simple kernel regression:

$$\begin{aligned} s &= \sum_{j=1}^{J} w_j s_j / (\sum_{j=1}^{J} w_j) \\ \text{where } w_j &= \exp[-\frac{1}{2}(D(b, a_j)/\sigma_j)^2] \\ \text{and } D(b, a_j) &= \| b - a_j \|_2 \text{, the } L_2 \text{ norm} \end{aligned} \tag{3}$$

We use Gaussian kernels of widths σ_j, which control the extent of influence of a_j. These values are pre-computed so that approximately ten percent of the bootstrap images lie within $1 \times \sigma_j$ at each a_j. Basically, kernel regression is a smooth interpolation method in which bootstrap images near b get weighted more than those farther away. How accurate is this method of estimating s? We compared the estimated illumination of the 60 images of a test face against their actual values. Details of the test are given in [17], but on average, the estimated s differ from the actual value by $6.3°$, with a standard deviation of $3.8°$. Our method is thus reasonably accurate.

3.2. Computing the statistics of $e(x, s)$

Our statistical model has learned the statistics (i.e. $\mu_e(x, s_j)$ and $\sigma_e^2(x, s_j)$) of $e(x, s_j)$ at the known illuminations s_j. We need a way to compute these same statistics for any new illumination s. Again, we use kernel regression. The mean and variance at s are smoothly interpolated from the known values at $\{s_j\}_{j=1}^{J}$. The kernel regression equation used here is similar to that in (3).

We also need a way to interpolate from the known correlation coefficients ρ_{jk} to obtain a new correlation coefficient ρ_{12} between illuminations s_1 and s_2 (one or both of which may be different from the bootstrap set). To do this, we need a slight modification. We view the problem now as interpolating the (unknown) function $g(t) : \mathcal{R}^6 \rightarrow \mathcal{R}$, where $g(t_{12}) = \rho_{12}$ and t_{12} is the concatenation of s_1 and s_2: $t_{12} = [s_1^\top, s_2^\top]^\top$. We will now use kernel regression to interpolate $g(t)$. Note that because $\rho_{12} = \rho_{21}$, we can actually obtain two estimates $g(t_{12})$ and $g(t_{21})$. By averaging these two values, we obtain a better (smaller variance) estimate.

3.3. Computing the MAP estimate

Using Bayes' rule, the MAP estimate becomes $n_{\text{MAP}}(x) = \arg\max_{n(x)} \Pr(i(x)|n(x)) \Pr(n(x))$. The second term is simply the Gaussian pdf learned in the previous section, while the first term may be obtained from Equation (1). Given $n(x)$, Equation (1) says that $i(x)$ is a scalar random variable with Gaussian pdf of mean $n(x)^\top s + \mu_e(x, s)$ and variance $\sigma_e^2(x, s)$. It can be shown [17] that the MAP estimate may be computed from solving linear matrix equations of the form:

$$\begin{aligned} A * n_{\text{MAP}} &= b \\ \text{where} \quad A &= \frac{1}{\sigma_e^2} s s^\top + C_n^{-1} \\ \text{and} \quad b &= \frac{(i - \mu_e)}{\sigma_e^2} s + C_n^{-1} \mu_n \end{aligned} \tag{4}$$

Note that our algorithm recovers each $n(x)$ independently of other $n(y)$. We have not explicitly used any other constraints such as smoothness or symmetry. These constraints are implicit in our statistical model, and we have found them to be sufficient for our purpose. Imposing the symmetry constraint explicitly, for example, will limit the applicability of our method to symmetric objects only. Besides, faces are not *perfectly* symmetric [13], so this constraint may actually hurt our algorithm.

If we have multiple images from which to recover $n(x)$, we can easily extend the MAP estimation procedure. Let $w(x) = [i_1(x), i_2(x), \cdots, i_m(x)]^\top$ be the vector of m intensity values at pixel x. We now seek $\arg\max \Pr(n(x)|w(x))$, which by Bayes' rule becomes $\arg\max \Pr(w(x)|n(x)) \Pr(n(x))$. The second term is as before, while the first term is now a multivariate Gaussian pdf with mean $S^\top n + \mu$, and covariance matrix C. The new variables are defined as: S, a $3 \times m$ matrix whose columns are the estimated illuminations of the m images; μ, an $m \times 1$ mean vector containing the scalar means $\mu_e(x, s_1), \cdots, \mu_e(x, s_m)$, and C, an $m \times m$ covariance matrix whose diagonal entries are the variances $\sigma_e^2(x, s_1), \cdots, \sigma_e^2(x, s_m)$, and whose off-diagonal entries are the covariances σ_{jk} between $e(x, s_j)$ and $e(x, s_k)$. Again, the solution takes the form of Equation (4), but now the terms are: $A = SC^{-1}S^\top + C_n^{-1}$, and $b = SC^{-1}(w - \mu) + C_n^{-1}\mu_n$.

Note that these m images must differ *only* in illumination. Identity, facial expression, etc., must remain the same, otherwise the method may fail. Since it is difficult in real-world applications to acquire several images of one person that differ *only* in illumination and nothing else, the opportunity to use multiple images is rare. Nevertheless, we are pleased that our statistical model is theoretically able to handle multiple input images.

3.4. Synthesizing new images

We now have the tools to synthesize images under any new illumination s'. We can simply use the standard Lambertian equation and compute $n_{\text{MAP}}(x)^\top s'$. But this generates images that do not look realistic: specularities and shadows are not properly synthesized. This is because real faces are not perfectly Lambertian.

To do better, we note that having computed $n_{\text{MAP}}(x)$, we can obtain the *actual* error at the original illumination s, from $e = e(x, s) = i(x) - n_{\text{MAP}}(x)^\top s$. We now ask for the most probable $e' = e(x, s')$ given knowledge about e. In other words, we want $\arg\max_{e'} \Pr(e'|e)$. Since we have modelled the joint pdf of e' and e as a Gaussian distribution, this turns out to be $\mathcal{E}[e'|e]$, the conditional expectation of e' given e. From basic probability theory, we know that if two random variables (x, y) are jointly Gaussian with means μ_x and μ_y, variances σ_x^2 and σ_y^2, and correlation co-

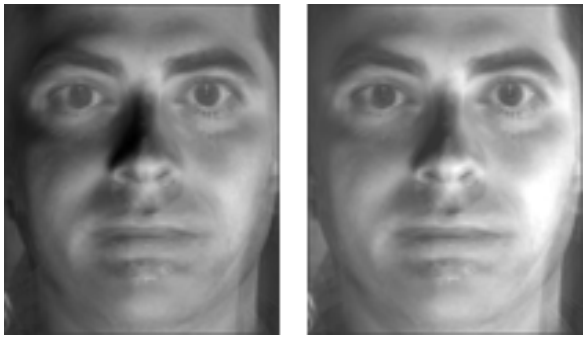

Figure 2: Image rendered using the strict Lambertian equation (left) versus one that uses the error term (right). Specular reflection on the cheeks are more accurately rendered in the right image.

efficient ρ_{xy}, then $y|x$ is also a Gaussian pdf with mean $\mu = \mu_y + \rho_{xy}\sigma_y(\frac{x-\mu_x}{\sigma_x})$, and variance $\sigma^2 = \sigma_y^2(1-\rho_{xy}^2)$. The variances and correlation coefficient terms are computed using kernel regression as described in Section 3.2. The synthesized image is thus: $i'(x) = n_{\text{MAP}}(x)^\top s' + \mathcal{E}[e'|e]$.

The effect of the error term may be seen in Figure 2, which compares an image rendered under strict Lambertian assumption versus one rendered using the additive error term. Clearly, the use of the error term produces a more visually pleasing result.

4. Face Recognition

Let's recall our proposed approach for face recognition. We proceed in two stages: first, we will use models to generate additional synthetic images; these are then be used as exemplars to train a face recognition system. Using our synthesis method of the previous section, we can easily render images under many illumination directions. The effect of multiple light sources is also readily produced: since light is additive, we simply add the images created under single light sources. From our literature survey, we note that having more training images will almost always improve the performance of *any* classification system. It is the dearth of training images that cripples many a classifier. However, we are particularly interested in using these synthetic training images in an exemplar-based classifier.

4.1. Exemplar-based classifiers

The simplest exemplar-based method is to use the training images directly in a k-nearest-neighbor search [19]. The obvious drawback is the enormous amount of storage space required, and the time it takes to perform a single classification. Clearly, this method is not readily scalable. We can improve things somewhat by reducing the dimensionality of the problem. A standard way to do this is via Principal Components Analysis (PCA) [5].

From the set of d-dimensional training vectors $\{x\}_{j=1}^M$, we compute a $d \times k$ projection matrix W and a $d \times 1$ mean vector m. We then project all the training data into a lower-dimensional PCA subspace using $y = W^\top(x - m)$. The benefit is that often, $k \ll d$, allowing us to greatly reduce the dimensionality while preserving the information in the original images. We can then work with $\{y\}$ instead of $\{x\}$, since they are just compressed versions of the original. The problem is that given the size of our images, and the fact that we are synthesizing thousands more, the usual way of computing W from the Singular Value Decomposition (SVD) of $\{x\}$ becomes intractable. The time and space complexity are $\mathcal{O}(M^3)$ and $\mathcal{O}(M^2)$ respectively. Instead, we need a way to compute W incrementally, as new images are synthesized.

We use the method of [4]. The idea is to incrementally update the SVD (and hence the PCA subspace) as more images are added. Furthermore, not every additional image will affect the SVD; only those that are significantly outside the PCA subspace need to be considered, others can be safely ignored since they are well represented in the subspace. Using this technique and requiring that the reconstruction error to be no more than 1%, we found that the PCA subspace so computed spans only 40 dimensions, much smaller than the 12000 dimensions of our original images.

Despite this, working in this reduced subspace still requires a substantial amount of memory. The main culprit is the fact that we are synthesizing so many images. One way to overcome this is to embed the exemplars in their own subspace, one for each person to be recognized. This is readily done by computing *individual* PCA subspaces (W_p,m_p) for each person p. This is different from the global PCA subspace W described above, which is computed from all the exemplars, regardless of identity. Recognition is now performed using reconstruction error, as follows:

1. Given a task image x, project it into each individual subspace using $y_p = W_p^\top(x - m_p)$.
2. Reconstruct the image $x_p = W_p y_p + m_p$, and compute the reconstruction error $\| e_p \|^2 = \| x - x_p \|^2$.
3. Pick the subspace that has the smallest $\| e_p \|^2$.

The memory requirements for this individual eigenspace approach is linear in the number of classes, and does not depend on number of synthetic exemplars generated. Recognition speed is also fast: linear in the number of classes, and independent of the number of exemplars. In terms of pattern recognition theory, each $\| e_p \|^2$ is just a quadratic discrimination function [5].

4.2. Analysis

An important and natural question to ask is this: instead of synthesizing all these training images and then designing a classifier from them, can we exploit the structure of the synthesis procedure to more directly produce an equivalent classifier? If we can do so, we will have completely avoided the synthesis step and its concomitant problems described in the previous section. We shall attempt to do this for the individual PCA approach. We begin by writing the synthesis equation of Section 3.4 in matrix form: $b_i = N^\top s_i + \mu_{s_i|s}$, where b_i is the ith synthesized image (expressed as a d-dimensional column vector) under new illumination s_i; N is the $3 \times d$ matrix of recovered surface normals; and $\mu_{s_i|s} = \mathcal{E}[e'|e]$ is the column vector of the conditional mean of the error term given the error at the original illumination s.

To aid our analysis, we assume that the illumination s_i is moved around in front of the face (there is no need to illuminate from behind). Specifically, assume that s_i is varied on the surface of a hemisphere of unit radius, centered at the face. To compute the individual PCA, we need the eigenvectors of the covariance matrix of the images $B = \{b_i\}$. We should therefore see how varying s_i contributes to the covariance. Appendix A derives the mean and covariance to be:

$$\begin{aligned} \mu_B &= \mathcal{E}[b_i] = N^\top \mathcal{E}[s_i] + \mathcal{E}[\mu_{s_i|s}] \\ cov[B] &= N^\top \mathcal{E}[s_i s_i^\top] N + N^\top \mathcal{E}[s_i \mu_{s_i|s}^\top] + \\ &\quad \mathcal{E}[\mu_{s_i|s} s_i^\top] N + \mathcal{E}[\mu_{s_i|s} \mu_{s_i|s}^\top] - \mu_B \mu_B^\top \\ \text{where} \quad \mathcal{E}[s_i] &= [0, 0, \frac{1}{2}]^\top \\ \mathcal{E}[s_i s_i^\top] &= \begin{bmatrix} \frac{1}{3} & 0 & 0 \\ 0 & \frac{1}{3} & 0 \\ 0 & 0 & \frac{1}{3} \end{bmatrix} \end{aligned} \tag{5}$$

Observe that all expectation terms are constant of N. They do not change for different people, and can therefore be pre-computed. Thus, to determine the individual eigenspace for each person, we simply compute $cov[B]$ by the above equations, and then solve for its eigenvectors. However, there are several problems with this approach. Aside from the two expectations computed as shown, the others are too difficult to obtain analytically as they involve fairly complicated functions of random variables (see Appendix A). We can obtain numerical estimates for them using simulation or Monte Carlo techniques, but even this is problematic because of the huge sizes of the matrices involved. For instance, the term $\mathcal{E}[\mu_{s_i|s} \mu_{s_i|s}^\top]$ is a $d \times d$ matrix, which for our images of dimensionality $d = 12000$ requires about 1 Gb of memory. Assuming we can overcome this problem (e.g. by reducing the size of the images), we would still like a way to quickly compute the eigenvectors. Unfortunately, there is no obvious relationship between the eigenvectors of the individual terms in Equation 5 and those of $cov[B]$. Neither is there an easy way to compute the next set of eigenvectors when N changes for different people.

If we knew the rank of $cov[B]$, or even an upper bound on the rank, we can speed up our eigenvector calculations. Assuming that the upper bound, L, is much smaller than d, there are methods to compute the L dominant eigenvectors quickly, in $\mathcal{O}(Ld)$ time instead of the usual $\mathcal{O}(d^3)$ time for standard eigenvector algorithms. See ARPACK [12] for example. Looking at the above equation, it is tempting, but wrong, to conclude that $cov[B]$ has a rank of at most 5 because it is a sum of five terms containing rank-1 outer products of vectors. The reason is that the expectation operator $\mathcal{E}[\cdot]$ can affect the rank of its argument. A case in point is $\mathcal{E}[s_i s_i^\top]$, which has been computed above to be of rank 3, even though $s_i s_i^\top$ has rank 1. Thus the first three terms of $cov[B]$ each has rank ≤ 3, the fourth term has rank $\leq d$, while the last term has rank 1. We can of course numerically estimate $\mathcal{E}[\mu_{s_i|s} \mu_{s_i|s}^\top]$ and its rank, r, barring memory requirements. Then an upper bound on the rank of $cov[B]$ is $L = r + 10$. If $L \ll d$, we can now resort to ARPACK to compute the eigenvectors; otherwise it does not appear that we can do much better.

5. Experiments and Results

5.1. Synthesis experiments

We tested our shape-from-shading algorithm by comparing the synthesized images against the actual ones. Testing the recovered surface normals is not particularly meaningful because we do not have ground truth data. Even if we used Equation (2) to attempt to recover the ground truth, what we would end up computing is just the least-squares estimate of the true surface normals. Furthermore, since our purpose is to generate more exemplars, what matters is how realistic our synthetic images are, not how well intermediate parameters are recovered. Quantitative results of our synthetic images are reported in [17], and show that the more extreme the illumination (whether in the task image or in the synthetic ones), the greater the error. These quantitative errors, do not, however, tell us about the visual quality of the synthetic images. For this we can only rely on human judgement. Figure 3 shows some synthetic images, compared to the actual ones.

A more interesting and instructive test case is when the task image is completely black. Our algorithm proceeds to recover and synthesize a face, in effect hallucinating one into existence [2]. This is the natural consequence of using class-based prior information. Our algorithm is generating the most probable face, based on other bootstrap faces it has seen. Not surprisingly, this turns out to be the average

Figure 3: Synthesized images (top row) versus actual images under the same illumination (bottom row).

Figure 4: (Left) Hallucinated from an all-black image. (Right) Average of all bootstrap faces.

of all the bootstrap images (Figure 4): substituting $s = 0$ (because there is no illumination) into Equation (4) yields $n_{\text{MAP}} = \mu_n$. This effect also comes into play when large regions of the face are occluded, or in shadow. For instance, a task image illuminated from one side (and therefore has the other side in the dark) produces a synthetic face with mixed identity. The side that is illuminated is one person, while the side that is not visible is replaced with the average face (Figure 5).

Overall, our algorithm synthesizes reasonable faces, especially when illumination is not extreme. For the purpose of face recognition, this seems adequate. A natural question to ask is therefore: how does the quality of the synthetic images affect recognition accuracy? This is something we will investigate in future.

5.2. Recognition experiments

We built three simple exemplar-based classifiers and compared them. The first classifier performs single nearest-neighbor search directly in image space. We call this 1NN. The second classifier searches for the nearest exemplar in

Figure 5: (Left) Input image: face illuminated from one side. (Middle) Face recovered and re-rendered under frontal lighting. The side of the face in shadow is hallucinated from the model, resulting in a mixed identity. (Right) Actual image.

Figure 6: Two PIE faces under four illumination directions. These subjects are different from those in the Yale dataset. The illumination is also different.

the global PCA subspace (dimensionality 40). We denote this as globalPCA. Finally, the third classifier computes individual PCA subspaces from the exemplars (dimensionality between 35 and 45), and classifies using the smallest reconstruction error. Call this indivPCA. All PCA subspaces were computed to have an average reconstruction error of 1%.

For our recognition experiments, we used a subset of ten people from the CMU PIE dataset [16]. These persons are distinct from those in the Yale dataset (see Figure 6). Each person has 21 images taken under different illumination (different from those in the Yale dataset). All images were pre-processed as in the Yale dataset: affinely aligned and tightly cropped. For each person, one image was arbitrarily selected as the training image, and from this 900 additional exemplars were synthesized. These synthetic images were created under illumination directions that range from $-90°$ to $+90°$ in both azimuth and elevation angles on a hemispherical surface in front of the face. The remaining 20 images per person were used to test the classifiers. Note that because our images are large ($d = 12000$), we cannot bypass synthesizing images and compute $cov[B]$ directly using Equation 5.

The results are as follows: 39% recognition accuracy for both 1NN and globalPCA, and 95% accuracy for indivPCA. We postulate that the 1NN and globalPCA classifiers are distracted by "noise": many faces under extreme illumina-

tion look alike because large regions are shadowed. We further speculate that the 40 eigenfaces in the globalPCA subspace capture mostly illumination variation, rather than identity. The eigenfaces beyond these 40 (which have been excluded from globalPCA) are probably more discriminating. As for indivPCA, we pleased to see that it is a viable classifier. For this method to work, large numbers of training images must be available for each person to be recognized. This is required in the computation of the individual PCA subspaces, otherwise the computed subspaces will not be accurate. Since training images are often scarce in face recognition applications, this technique is not commonly used. By using our approach of synthesizing exemplars from even a single training image, we can overcome this limitation. indivPCA now becomes a feasible classifier.

6. Discussion

The results from our simple experiments are encouraging. They show that our model- and exemplar-based approach for face recognition can overcome the limitations that plague other methods. We must emphasize that our approach is not tied to specific synthesis techniques nor classifier designs. We could have used the Quotient Image [15] to generate new face images, or a neural network for our classifier, and it would not change our idea. The synthesis method in Section 3.4 and the indivPCA classifier in Section 5.2 were chosen merely to illustrate our idea of combining models and exemplars, and to show that some mathematical analysis can be done when we know the structures of both the synthesis method and the classifier. In fact, a number of major points may be raised at this juncture.

Firstly, whereas other classification algorithms focus on features – finding discriminating and invariant features, and extracting them from images – we focus on synthesizing training images and designing an exemplar-based classifier that is feature-free.[1] As we have noted in our introduction, finding discriminating features that are truly invariant to all types of appearance variation is very hard, if not impossible. Even if these features are found, it is time-consuming and error-prone to locate them on a face image, extract and process them for recognition. By avoiding features in our classifier, we also avoid these problems. To be sure, our synthesis method does require some features – locating three points on the face for image alignment – but this is confined to the synthesis stage. The classifier itself is feature-free. In general, we expect the model-building and synthesis stage to require features. But as long as the resulting classifier does not explicitly depend on features, we will not be bogged down by feature-processing during recognition.

[1] Although we have not stated it explicitly, it should be obvious that any exemplar-based classifier does not rely on features directly. Rather, the exemplars themselves provide the information necessary to discriminate one pattern from another.

Secondly, our method need not be fully automatic. Note that there is no reason for our model-based synthesis not to require manual intervention from a human expert. As long as our classifier works automatically, the learning stage can use as much manual help as possible to create realistic images. For example, in Figure 3, we could have manually corrected the deficiency of our algorithm and improved the quality of the synthetic images before using them in our classifier. Doing so will only help indivPCA learn a more accurate eigenspace. Other classification schemes try to make both the learning and recognition stage fully automatic, but we feel that this is unnecessarily burdensome.

Thirdly, our approach has the potential to handle many more kinds of appearance variation, simply by synthesizing images that exhibit those variations. For example, to simultaneously cope with pose, illumination and expression changes, synthesize training images under many different combinations of lighting, pose and expression. The exemplar-based classifier will learn them all, regardless of variation. Of course, we have yet to demonstrate this conclusively, but the extension is logical and waiting to be explored. We envisage using different models to synthesize different kinds of variations: the one in this paper for illumination, another for pose, yet a third for facial expression. We can even imagine synthesizing mustaches, beards, and eye glasses. Obviously, the number of synthetic images will grow combinatorially with the kinds of variations to be synthesized. This imposes some constraints on the exemplar-based classifier: (1) it must be able to learn incrementally, as more training images are synthesized; and (2) its storage requirements and classification time must not increase as quickly as the number of exemplars. Classifiers such as 1NN and globalPCA clearly will not work, whereas indivPCA is good candidate.

Finally, we should point out that our approach is applicable to other non-face objects as well. In fact, our approach should be seriously considered for recognizing non-rigid, deformable objects, where finding suitable features has proven to be difficult. As long as models can be built to synthesize images of these objects realistically, we can combine them with an appropriate exemplar-based classifier to do the job.

7. Conclusion

We conclude by summarizing the key points in this paper: We have proposed a new approach for face recognition. The main idea is to utilize models, statistical or otherwise, to synthesize many more images from a given few, which can then be used to train an exemplar-based classifier. We demonstrated this idea by showing how a statistical shape-from-shading model may be used to synthesize images under novel illumination, and next by using this set of aug-

mented training images to build a simple, exemplar-based classifier. We also analyzed the mathematical structure behind the synthesis and classification scheme and suggested ways to improve the construction of the classifier. We note that our synthetic images are not perfect; they degrade under extreme illumination, but we can expect better techniques to correct for this in future. Indeed, the computer graphics community is relentlessly perfecting the art of synthesizing realistic faces. We should be able to leverage their techniques to further our own goals, that of perfecting face recognition.

In the near future, we intend to further develop this approach in several ways. On the synthesis side, we hope to use graphics models to generate pose and expression variations. As for classifier design, we want to explore classifiers that can compactly represent all the synthetic exemplars without combinatorially exploding, or those that were never possible before due to a lack of training data. Where the structure permits, we intend to mathematically analyze both the synthesis algorithm and classification method in order to design better classifiers.

A. Derivation of μ_B and $cov[B]$

We first compute $\mathcal{E}[s_i]$ and $\mathcal{E}[s_i s_i^\top]$. Consider s_i to be a point moving randomly on the surface of a unit hemisphere. We assume that the pdf of s_i is uniform for all points on this surface. Let this be p. Then: $\int_C p \ \mathrm{d}C = 1$, where C is the surface of the hemisphere. Using Spherical coordinates, with α and β denoting azimuth and elevation angles, respectively, and noting that the elemental area on the surface is $\cos\beta \ \mathrm{d}\alpha \ \mathrm{d}\beta$, this may be written as:

$$\int_{-\pi/2}^{\pi/2}\int_{-\pi/2}^{\pi/2} p\cos\beta \ \mathrm{d}\alpha \ \mathrm{d}\beta = 1$$

$$\Rightarrow p\left(\int_{-\pi/2}^{\pi/2} 1. \ \mathrm{d}\alpha\right)\left(\int_{-\pi/2}^{\pi/2} \cos\beta \ \mathrm{d}\beta\right) = 1$$

$$\Rightarrow p(\pi)(2) = 1,$$

$$\text{so that} \quad p = \frac{1}{2\pi}$$

Now, let $s_i = [x, y, z]^\top$ in Cartesian coordinates. Then: $\mathcal{E}[s_i] = [\mathcal{E}[x], \mathcal{E}[y], \mathcal{E}[z]]^\top$. We convert into Spherical coordinates using $x = -\sin\alpha\cos\beta$, $y = \sin\beta$ and $z = \cos\alpha\cos\beta$. So for instance,

$$\begin{aligned}
\mathcal{E}[x] &= p\int_{-\pi/2}^{\pi/2}\int_{-\pi/2}^{\pi/2} -\sin\alpha\cos\beta\cos\beta \ \mathrm{d}\alpha \ \mathrm{d}\beta \\
&= \frac{1}{2\pi}\left(\int_{-\pi/2}^{\pi/2} -\sin\alpha \ \mathrm{d}\alpha\right)\left(\int_{-\pi/2}^{\pi/2} \cos^2\beta \ \mathrm{d}\beta\right) \\
&= \frac{1}{2\pi}(0)(\frac{\pi}{2}) \quad = 0
\end{aligned}$$

Note that the second $\cos\beta$ term comes from the elemental area of the hemisphere. Similarly, we may compute $\mathcal{E}[y] = 0$, and $\mathcal{E}[z] = \frac{1}{2}$. Hence, $\mathcal{E}[s_i] = [0, 0, \frac{1}{2}]^\top$. As for $\mathcal{E}[s_i s_i^\top]$, we get:

$$\mathcal{E}[s_i s_i^\top] = \mathcal{E}\left[\begin{bmatrix} x^2 & xy & xz \\ xy & y^2 & yz \\ xz & yz & z^2 \end{bmatrix}\right]$$

We compute the expectation of each element of the matrix as before, by converting into Spherical coordinates. This results in:

$$\mathcal{E}[s_i s_i^\top] = \begin{bmatrix} \frac{1}{3} & 0 & 0 \\ 0 & \frac{1}{3} & 0 \\ 0 & 0 & \frac{1}{3} \end{bmatrix}$$

Proceeding on, we have $\mu_B = \mathcal{E}[b_i] = \mathcal{E}[N^\top s_i + \mu_{s_i|s}] = N^\top\mathcal{E}[s_i] + \mathcal{E}[\mu_{s_i|s}]$. From Section 3.4,

$$\mu_{s_i|s} = \mu_{s_i} + \rho_{s_i,s}\cdot\sigma_{s_i}\left(\frac{e(s) - \mu_s}{\sigma_s}\right)$$

Note that $\mu_{s_i}, \rho_{s_i,s}$, and σ_{s_i} are computed using kernel regression (see Section 3.2). Because of this, these terms are random variables, since they are functions of the random vector s_i. Recall, however, that s is the estimated illumination of the input image, which is constant as s_i varies on the hemisphere. Thus,

$$\mathcal{E}[\mu_{s_i|s}] = \mathcal{E}[\mu_{s_i}] + \mathcal{E}[\rho_{s_i,s}\cdot\sigma_{s_i}]\left(\frac{e(s) - \mu_s}{\sigma_s}\right)$$

And finally,

$$\begin{aligned}
cov[B] &= \mathcal{E}[b_i b_i^\top] - \mu_B\mu_B^\top \\
&= \mathcal{E}\left[\left(N^\top s_i + \mu_{s_i|s}\right)\left(N^\top s_i + \mu_{s_i|s}\right)^\top\right] - \mu_B\mu_B^\top \\
&= \mathcal{E}[N^\top s_i s_i^\top N + N^\top s_i\mu_{s_i|s}^\top + \mu_{s_i|s}s_i^\top N + \\
&\quad \mu_{s_i|s}\mu_{s_i|s}^\top] - \mu_B\mu_B^\top \\
&= N^\top\mathcal{E}[s_i s_i^\top]N + N^\top\mathcal{E}[s_i\mu_{s_i|s}^\top] + \\
&\quad \mathcal{E}[\mu_{s_i|s}s_i^\top]N + \mathcal{E}[\mu_{s_i|s}\mu_{s_i|s}^\top] - \mu_B\mu_B^\top
\end{aligned}$$

Acknowledgements

We are grateful to Rahul Sukthankar for his helpful comments and prompt feedback on this manuscript, as well as for his early contributions to our ideas.

References

[1] C.G. Atkeson, A.W. Moore, and S. Schaal. Locally Weighted Learning. *Artificial Intelligence Review*, 1996.

[2] S. Baker and T. Kanade. Hallucinating Faces. In *AFGR*, 2000.

[3] P. Belhumeur, J. Hespanha, and D. Kriegman. Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection. *PAMI*, 19(7), 1997.

[4] S. Chandrasekaran, B. Manjunath, Y. Wang, J. Winkeler, and H. Zhang. An Eigenspace Update Algorithm for Image Analysis. In *CVGIP*, 1997.

[5] R. Duda, P. Hart, and D. Stork. *Pattern Classification, 2nd edition*. John Wiley and Sons, 2000.

[6] G.J. Edwards, T.F. Cootes, and C.J. Taylor. Face Recognition Using Active Appearance Models. In *ECCV*, 1998.

[7] R. Epstein, A.L. Yuille, and P.N. Belhumeur. Learning Object Representations From Lighting Variations. *Lecture Notes in Computer Science*, 1144:179–??, 1996.

[8] A.S. Georghiades, P.N. Belhumeur, and D.J. Kriegman. From Few to Many: Generative Models for Recognition Under Variable Pose and Illumination. In *AFGR*, 2000.

[9] D.B. Graham and N.M. Allinson. Face Recognition from Unfamiliar Views: Subspace Methods and Pose Dependency. In *AFGR*, 1998.

[10] B.K.P. Horn and M.J. Brooks. *Shape from Shading*. MIT Press, 1989.

[11] R. Jain, Kasturi R., and B. Schunck. *Machine Vision*. McGraw Hill, 1995.

[12] R.B. Lehoucq, D.C. Sorensen, and C. Yang. *ARPACK Users' Guide: Solution of Large-Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods*. SIAM Publications, Philadelphia, 1998.

[13] Yanxi Liu, R.L. Weaver, Karen Schmidt, N. Serban, and Jeffrey Cohn. Facial Asymmetry: A New Biometric. Technical Report CMU-RI-TR-01-23, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, August 2001.

[14] U.S. Department of Defense. Facial Recognition Vendor Test, 2000. http://www.dodcounterdrug.com/facialrecognition /FRVT2000/frvt2000.htm.

[15] T. Riklin-Raviv and A. Shashua. The Quotient Image: Class Based Recognition and Synthesis Under Varying Illumination Conditions. In *CVPR*, pages II:566–571, 1999.

[16] T. Sim, S. Baker, and M. Bsat. The CMU Pose, Illumination, and Expression (PIE) Database of Human Faces. Technical Report CMU-RI-TR-01-02, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, January 2001.

[17] T. Sim and T. Kanade. Illuminating the Face. Technical Report CMU-RI-TR-01-31, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, 2001.

[18] T. Sim, R. Sukthankar, M. Mullin, and S. Baluja. Memory-based Face Recognition for Visitor Identification. In *AFGR*, 2000.

[19] R. Sukthankar and R. Stockton. Argus: The Digital Doorman. *IEEE Intelligent Systems*, 16(2), 2001.

[20] M. Turk and A. Pentland. Eigenfaces for Recognition. *Journal of Cognitive Neuroscience*, 3(1), 1991.

[21] L. Wiskott, J.M. Fellous, N. Kruger, and C. von der Malsburg. Face Recognition by Elastic Bunch Graph Matching. *PAMI*, 19(7), July 1997.

[22] R. Zhang, P.S. Tsai, J.E. Cryer, and M. Shah. Shape from Shading: A Survey. *PAMI*, 21(8), August 1999.

[23] W. Zhao and R. Chellappa. Robust Face Recognition using Symmetric Shape-from-Shading. Technical Report CARTR -919, 1999., Center for Automation Research, University of Maryland, College Park, MD, 1999.

[24] L. Zheng. A New Model-based Lighting Normalization Algorithm and its Application in Face Recognition. Master's thesis, National University of Singapore, 2000.

[25] Q. Zheng and R. Chellappa. Estimation of Illuminant Direction, Albedo, and Shape from Shading. *PAMI*, 13(7), July 1991.