Illuminating the Face

Terence Sim, Takeo Kanade

CMU-RI-TR-01-31

Sept. 28, 2001

Abstract

Shape from shading for general objects is a difficult problem, because it is ill-conditioned. However, when the domain is restricted to a specific class of objects, prior knowledge of the class may be used to provide useful constraints, thereby making the problem tractable. This paper presents a novel method for solving the shape from shading problem in the restricted domain of human faces. We show how knowledge of face shapes in general may be incorporated as a statistical model and used to recover the surface normals from a single input image. The model also captures shadow and specularity information without explicitly recovering the 3D shape. New images of the face under novel illumination can then be realistically rendered.

Keywords: shape from shading, face synthesis, class-based synthesis

1 Introduction

Human faces are important objects in computer graphics and vision, because many applications require the detection, transmission or rendering of images of faces. Examples include video conferencing, virtual reality simulations, and security systems. For these applications to work well, they must be able to detect or render faces realistically. One difficulty in dealing with images of faces is caused by changing illumination: the appearance of a face can change dramatically as the light falling it varies. See Figure 1 for an example. A good graphics or vision system must be able to cope with such changes and still preserve the identity of the face.

In this paper, we present a method to solve the following problem: **Given a single image of a face, illuminated under a single point light source of unknown intensity and direction, realistically render the same face under novel illumination.** For simplicity, we will assume that the novel illumination is also from a single point light source, but whose direction and intensity are known. Extending this problem to handle novel illumination coming from multiple or extended light sources is then straightforward: simply model the multiple or extended source as a collection of single point sources, render the face under these single point illumination, and add the resulting images. The additive nature of light justifies this simple addition.

The rest of this paper is organized as follows: Section 2 reviews some related work in this area; Sections 3 and 4 explain the proposed method, giving details of the models and algorithms used; Section 5 shows experimental results; and Section 6 ends with a brief conclusion.

2 Related Work

The general area under which this problem is categorized is called *Shape from Shading*, which has a history going back at least three decades [7]. Shape from shading seeks to recover the 3D shape of a surface from a 2D image of it. This problem is ill-conditioned, because many different 3D shapes produce (project onto) the same 2D image. Many researchers have sought to solve it through a variety of techniques and by imposing general constraints such as surface smoothness and integrability, [8, 15, 10]. A recent survey of the field, [14], reveals that although progress has been made, the state-of-the-art is still not accurate or robust enough when dealing with real objects. As such, some researchers have turned their attention to restricted classes of objects, hoping that the constraints imposed by these restricted classes

help to solve the shape from shading problem. For the class of human faces, related work include [12, 6, 1, 5]. Even within this restricted class, shape from shading is still a challenging problem.

In [12], the quotient image is proposed as an illumination-invariant version of a face image. This is done by exploiting a training set of registered face images, and the assumption that all faces have the same shape, differing only in the albedo (which is a measure of how much a small surface patch reflects incident light). The authors present a method of estimating the unknown light source in a single given image and an algorithm for computing the quotient image. Once this is computed, rendering the face under new illumination is straightforward. In our previous experiments with the Quotient Image, we discovered several problems: (a) the constant shape assumption is not valid for real faces, (b) the choice of the training set affects the quality of the quotient image, and thus of the rendered images, and (c) shadows and specularities (regions of highlight) are not modeled, causing the rendered faces to look unrealistic. Furthermore, it is not clear how additional input images of the novel face may be incorporated to improve the quality of the quotient image.

The illumination cone approach in [6] attempts to recover the 3D shape from at least three images of the same face taken under different (but unknown) illumination conditions, but in the same pose and facial expression. The authors show how integrability constraints, and knowledge about face shapes in general, may be used to recover the shape up to an affine transformation. Thereafter, one can render the face not only under novel illumination, but also from novel viewpoints. Shadows are modeled by ray-tracing techniques, once the shape and new pose are known. This makes the technique computationally expensive, although the authors claim realistic results. Another drawback is the requirement of at least three input images which differ only in illumination. In some applications, it is difficult to impose such requirements because humans are constantly changing facial expression and head pose.

In [5], 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 [12] 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.

Unlike the methods just described, the work in [1] uses a full 3D model of a human head to impose shape constraints. However, this work appears preliminary: the authors assume constant albedo across the whole face, and they give few results to validate their method. Also using a 3D model for imposing constraints is the work reported in [4]. This model, however, is morphable, and is used to great effect to recover the shape, illumination and pose from a single given image. The authors produce impressive results, giving credence to the validity of their sophisticated technique. One minor drawback is the need for the user to approximately align the model to the given image. Although this is not a severe limitation, a fully automatic system would be more convenient to use. The authors also state a need to extend their current morphable model to handle non-Caucasian faces: to include those of other races, as well as faces of children and elderly people.

3 Shape-from-Shading Model

3.1 Augmented Lambertian equation

At the heart of our method is the following equation, which is the standard Lambertian equation [9] 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 $n^\top s$. 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 [12], our model does not assume that every person has the same face shape, nor constant albedo, but allows both to vary from pixel to pixel and from person to person. Solving the full d-pixel equation is generally intractable, so we make some simplifying assumptions. We assume that surface normals at different pixels are

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

independent of one another, or more precisely, we recover the surface normal at each pixel *as if* it is independent of the normals at other pixels. This is clearly not ideal and does lead to some interesting consequences, but it greatly makes the problem tractable. 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. We relax the strong independence slightly because we find that it allows us to render more realistic images, while leaving the problem still computationally tractable. More details will be given in Section 3.2.

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 thus need additional knowledge to solve for the unknowns. To this end we will use knowledge about the class of faces in general. More specifically, we will learn a statistical model for $n(x)$ and $e(x, s)$ from a set of bootstrap images. This model will guide our recovery and synthesis steps, and ensure that we generate reasonable faces. The bootstrap images come from the Yale dataset [6], 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 gray-scale images were then cropped to remove the background and scaled to 120×100 pixels.

3.2 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:

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

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 4.4.

To get a sense for the these quantities, we can display them as a face image. Refer to Figures 11, 8 and 2. Figure 11 displays the surface normals as little arrows on a rectangular array, each positioned over its corresponding pixel. Such a display is called a needle map. Although this is a downsampled version of the full needle map, it is clear that the arrows indicate the local curvature of the face. In fact, the surface normal is *precisely* the derivative of the 3D shape [9], and one may integrate these normals to get back the 3D shape. Figure 8 attempts to show the variation of $n(x)$ at each pixel x across all the bootstrap faces. This is displayed as the trace of the covariance matrix $C_n(x)$: the larger the trace, the greater $n(x)$ varies and the brighter the pixel in the figure. Ignoring the regions outside the face, one can see that the chin, the eyebrows and parts of the forehead exhibit the greatest variation. This in turn means that face shape differ the most in these regions. Figure 2 displays the error term as illumination is varied from right to left. What is shown is actually $|e(x,s)|$: brighter pixels correspond to greater absolute error (i.e., greater deviation from a perfectly Lambertian surface). We can see that the error term does indeed capture shadows and

Figure 2: Displaying $|e(x,s)|$ as illumination is varied from right to left. Brighter pixels denote greater deviation from true Lambertian. Notice how shadows and specularities are captured by the error term.

specularities on the face. Notice how these regions move as the incident illumination is changed.

4 Using the Model

Having learned the statistical model, we want to make use of it in a statistically optimal way. Ideally, given only an input image $i(x)$, we want the most probable image under new illumination s'. That is, we want:

$$\begin{aligned} i^*(x) &= \arg\max_{i'(x)} \Pr(i'(x)|i(x), s') \\ &\text{for every pixel } x = 1, \cdots, d \end{aligned} \tag{3}$$

This formulation does not explicitly recover the unknown illumination s, nor the surface normal $n(x)$, but we recognize that these are *intermediate variables* anyway. Our final goal is another image, and these intermediate variables may or may not be useful toward this goal. Nevertheless, without these variables, we have no real structure to work with, making the problem very difficult. To incorporate the intermediate variables, we can use a little probability algebra and re-write:

$$\begin{aligned} i^*(x) &= \arg\max_{i'(x)} \Pr(i'(x)|i(x), s') \\ &= \arg\max_{i'(x)} \int_n \int_s \Pr(i'(x)|n(x), s, i(x), s') \Pr(n(x), s|i(x), s') \, \mathrm{d}s \, \mathrm{d}n \end{aligned} \tag{4}$$

However, it is not clear that this formulation is consistent in the following sense: the illumination s, although unknown, is *fixed* and must be the same across all pixels x. Any

attempt at recovering s, even implicitly, must ensure that this constraint is satisfied. Given that we are working on each pixel independently of other pixels, we have no way of enforcing this constraint. We intend to explore this more rigorously in the future, but for now we simplify the problem by estimating s in a prior step, and using it consistently across all pixels to recover $n(x)$. Finally we synthesize $i'(x)$. Our algorithm is thus:

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.

4.1 Estimating s

Estimating the unknown illumination is a well-studied problem [14, 15], 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 [2]. 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:

$$\begin{aligned} s &= \sum_{j=1}^{J} w_j s_j / (\sum_{j=1}^{J} w_j) \\ \text{where } w_j &= \exp[-\frac{1}{2}(D(b,a_j)/\sigma_j)^2] \\ \text{and } D(b,a_j) &= \parallel b - a_j \parallel_2 \text{ , the } L_2 \text{ norm} \end{aligned} \tag{5}$$

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. What happens if $b = a_j$ for some j? In this case, the illumination of b should be that of a_j, since the input image is exactly one of the bootstrap images. Equation (5) will give us $w_j = 1$ but other weights $w_{k \neq j}$ will not be zero, giving us the wrong result. We thus need to detect this coincident input condition and set the other weights to zero.

4.2 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 is now:

$$\begin{aligned} \mu_e(x, s) &= \sum_{j=1}^{J} w_j \mu_e(x, s_j)/W \\ \sigma_e^2(x, s) &= \sum_{j=1}^{J} w_j \sigma_e^2(x, s_j)/W \\ \text{where } w_j &= \exp[-\frac{1}{2}(D(s, s_j)/\sigma_j)^2] \\ \text{and } D(s, s_j) &= \parallel s - s_j \parallel_2 \\ W &= \sum_{j=1}^{J} w_j \end{aligned} \tag{6}$$

As before, the coincident condition is detected and one of the weights is set to one while the others are set to zero. 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. Note also that we have chosen to learn and interpolate the correlation coefficients $\{\rho_{jk}\}$ instead of covariances $\{\sigma_{jk}\}$ because our kernel regression method preserves the property $|\rho_{jk}| \leq 1$. This makes the interpolated value a valid correlation coefficient. Mathematically,

$$
\begin{aligned}
\rho &= (\rho_{12} + \rho_{21})/2 \\
\rho_{12} &= \sum w_{jk}\rho_{jk} / (\sum w_{jk}) \\
\rho_{21} &= \sum \omega_{jk}\rho_{jk} / (\sum \omega_{jk}) \\
\text{where } w_{jk} &= \exp[-\frac{1}{2}(D(t_{12}, t_{jk})/\sigma_{jk})^2] \\
\omega_{jk} &= \exp[-\frac{1}{2}(D(t_{21}, t_{jk})/\sigma_{jk})^2] \\
\text{and } D(x, y) &= \| x - y \|_2
\end{aligned}
\tag{7}
$$

4.3 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 $n(x)$ (and s), 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 $\sigma_e^2(x, s)$. Dropping the "(x)" for clarity, we can derive a closed-form solution for the MAP estimate as follows:

$$
\begin{aligned}
n_{\text{MAP}} &= \arg\max_n \Pr(i|n) \Pr(n) \\
&= \arg\max_n \text{Gauss}(n^\top s + \mu_e, \sigma_e^2) \times \text{Gauss}(\mu_n, C_n)
\end{aligned}
$$

Let L be the log probabilities, and ignoring terms constant of n, we get:

$$
\begin{aligned}
\arg\max_n L &= -\frac{1}{2}\left(\frac{i - n^\top s - \mu_e}{\sigma_e}\right)^2 - \frac{1}{2}(n - \mu_n)^\top C_n^{-1}(n - \mu_n) \\
\Rightarrow \quad \arg\min_n L &= \left(\frac{i - n^\top s - \mu_e}{\sigma_e}\right)^2 + (n - \mu_n)^\top C_n^{-1}(n - \mu_n)
\end{aligned}
$$

Taking derivatives w.r.t. n and setting it to 0:

$$
\frac{\partial L}{\partial n} = -\frac{2}{\sigma_e^2}(i - n^\top s - \mu_e)s + 2C_n^{-1}(n - \mu_n) = 0
$$

Re-arranging, we get the following linear equation:

$$\begin{aligned} A\,n &= b \\ \text{where} \quad A &= \frac{1}{\sigma_e^2} s s^\top + C_n^{-1} \\ \text{and} \quad b &= \frac{(i-\mu_e)}{\sigma_e^2} s + C_n^{-1} \mu_n \end{aligned} \tag{8}$$

This form is sufficient for a linear solver: n_{MAP} is the solution to the above linear equation. We can obtain a more explicit solution using Woodbury's identity (page 54 of [13]):

Woodbury's Identity: The inverse of the matrix
$R = R_0 + \gamma^2 u u^\top$ is the matrix:
$R^{-1} = R_0^{-1} - \left(\frac{\gamma^2}{1+\gamma^2 u^\top R_0^{-1} u} \right) R_0^{-1} u u^\top R_0^{-1}$

Applying the identity to the matrix A, and solving for $n = A^{-1} b$:

$$\begin{aligned} n_{\text{MAP}} &= A^{-1} b \\ &= \left(C_n - \frac{1}{\sigma_e^2 + s^\top C_n s} C_n s s^\top C_n \right) \left(\frac{i-\mu_e}{\sigma_e^2} s + C_n^{-1} \mu_n \right) \\ &= \left(\frac{i - \mu_e - s^\top \mu_n}{\sigma_e^2 + s^\top C_n s} \right) C_n s + \mu_n \end{aligned} \tag{9}$$

This final form makes the solution fast: there is no matrix inverse to compute, nor any iteration to perform. We simply compute $n_{\text{MAP}}(x)$ at every pixel x directly. 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 [11], 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, obtained from m input images. 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 (8), 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.

4.4 Illuminating the face

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 coefficient 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 4.2. 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 3, 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.

5 Results

5.1 Accuracy of recovered s

First, note that s is represented using two angles (α, β), denoting the azimuth and elevation angles respectively. All angles are measured in degrees ($^\circ$). This is with reference to a 3D

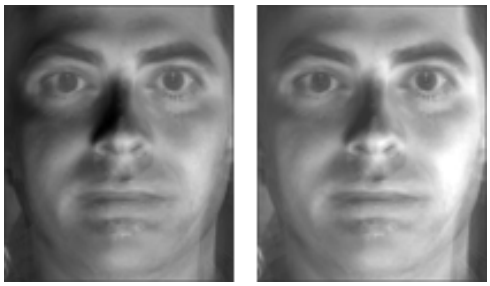

Figure 3: Image rendered using the strict Lambertian equation (left) versus one that uses the error term (right). Specular reflection on the cheeks are more accurately rendered in the right image.

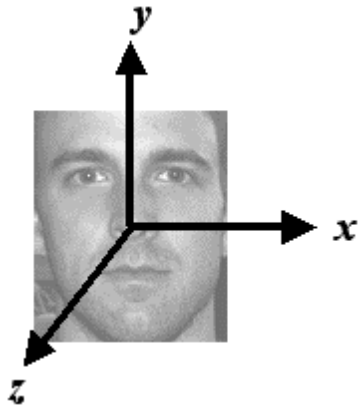

Figure 4: Coordinate axes to measure illumination direction.

coordinate axis whose origin is at the center of the face (Figure 4). The azimuth is the left/right angle, i.e., the angle between the z-axis and the projection of s onto the xz-plane. The elevation is the up/down angle, i.e., the angle between s and the xz-plane. Since s is of unit length, it varies on a sphere of unit radius centered at the origin. As such, this two-parameter representation is sufficient. When used in computations, however, s is converted to a column vector in $\mathcal{R}^3$, using the following equations:

$$\begin{aligned} s_x &= -\cos\beta\sin\alpha \\ s_y &= \sin\beta \\ s_z &= \cos\beta\cos\alpha \\ s &= [s_x, s_y, s_z]^\top \end{aligned} \tag{10}$$

To assess the accuracy of using Equation (5) to estimate s, we compare the recovered

illumination of the 62 images of the test face against their actual values (since all images have known illumination). The actual illumination has azimuth angles ranging between $\pm 130°$, and elevation angles ranging from $-40°$ to $+110°$. The error between the actual and recovered s is computed, and found to have minimum, average, and maximum values of $0.1°, 6.3°$, and $22°$ respectively. Its standard deviation is $3.8°$. This shows that the method is reasonably accurate.

5.2 Rendering results

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 render images, what matters is how realistic our synthetic images are, not how well intermediate parameters are recovered. In these tests, we built the statistical model based on 14 of the 15 subjects in the bootstrap set, and tested on the remaining subject. Figure 5 is an example of rendered versus original images. In this case, the input image (bottom row, first from the left) is frontally illuminated, allowing our algorithm to do a good job in recovering the surface normals and rendering under near-frontal illumination. Note that in one of the original images (bottom row, third from left), the subject has partially closed his eyes. The rendered image cannot mimic this effect, but otherwise has the identity of the person in tact. As the rendered illumination become more extreme, however, our synthetic images start to break down (the last two columns of the figure).

In Figure 6, we have a negative example. The input image is now illuminated from the right (bottom row, fourth from the left). Large portions of the input face are in shadow, making it difficult to recover the surface normal. Our algorithm still recovers a smooth face, however, due to the statistical model, but it is clear that the identity of the subject has been compromised. In fact, we can go the extreme and give a completely black image as the input. Our algorithm proceeds to recover and synthesize a face, in effect hallucinating one into existence [3]. This is a 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 of all the bootstrap images (Figure 7): substituting $s = 0$ and $i = 0$ (because there is no illumination, and the input pixel is black) into Equation (8) or Equation (9) yields $n_{\text{MAP}} = \mu_n$. Note that if s is not exactly 0, then n_{MAP} is the mean μ_n plus a term proportional to $C_n s$. This effect also comes into

Figure 5: Good result. (Top row) Images rendered under the same illumination as the originals. (Bottom row) Original images. Input image is the first from left on the bottom row, where illumination=$(0°, 0°)$. Other than the subject closing his eyes in one instance, the rendered images look reasonably realistic under near-frontal illumination. Things break down as illumination become more extreme.

play when large regions of the face are occluded, or in shadow. For instance, an input image illuminated from one side (and therefore has the other side in the dark) produces a rendered 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 (cf. the fourth and first columns of Figure 6).

For a more objective assessment, we compute the average per-pixel SSD (sum of squared difference) between the rendered images and the originals. This is done for each input test image:

$$\epsilon^2 = \frac{1}{\#\text{pixels}} \sum_{x \in \text{pixels}} \left(i_{original}(x) - i_{rendered}(x)\right)^2$$

This quantity is then plotted against the input illumination, as shown in Figures 9 and 10. For simplicity, the input illumination is arbitrarily sorted into 62 values. In Figure 9, input images under different illumination are used to recover the surface normals. In each case, a single image is synthesized under frontal illumination $(0°, 0°)$ using the recovered surface normals. The SSD between this rendered image and the original is then computed plotted as input illumination varied. The four best and worst SSD values are highlighted.

Figure 6: Bad result. (Top row) Images rendered under the same illumination as the originals. (Bottom row) Original images. Input image is the fourth from left on the bottom row, where illumination=$(-70°, 45°)$. The rendered images exhibit mixed identity: the side of the face in darkness is replaced with an average face, learned from the bootstrap set.

Figure 7: (Left) Hallucinated from an all-black image. (Right) Average of all bootstrap faces.

They correspond to input illuminations of $(0°, 0°), (-5°, -10°), (5°, -10°), (-10°, 0°)$ and $(-70°, 45°), (-70°, 35°), (-35°, 40°), (0°, 45°)$ respectively. Figure 10 shows a similar plot, but this time the input image is fixed at illumination $(0°, 0°)$, while the rendered images are computed for different illuminations. The SSD between the rendered and original images are plotted as output illumination is varied. Again, the four best and worst values are highlighted. They correspond to output illuminations of $(0°, 0°), (5°, \pm 10°), (-5°, -10°)$ and $(-110°, 15°), (130°, 20°), (-120°, 0°), (120°, -20°)$ respectively. These plots show that **as long as both the input image and the output images are under near-frontal illumination, our algorithm does a reasonably good job in recovering and rendering faces**.

6 Conclusion

This report presented a novel method for solving the shape from shading problem within the restricted class of human faces. Using a small bootstrap set of images, we built a statistical model of the surface normals of the human face, and also a model of the deviation from a strict Lambertian reflection. We also presented an algorithm to recover the surface normal, and to render new images under novel illumination. Our algorithm can work with as few as one input image, but is also able to incorporate multiple input images. We showed that we can produce reasonably realistic images as long as the input image and the output images were not under extreme lighting conditions. Some interesting artifacts appeared when illumination was extreme. This shows that there is still room for improvement. The main drawback of our technique is that we recover each surface normal independently of others, i.e. we neglected to use any correlation information between neighboring surface normals, which should help improve our results. This will be investigated in the near future.

Figure 8: Displaying the trace of the covariance of $n(x)$. Colors range from black (denoting small covariance) to red to yellow to white (denoting large covariance).

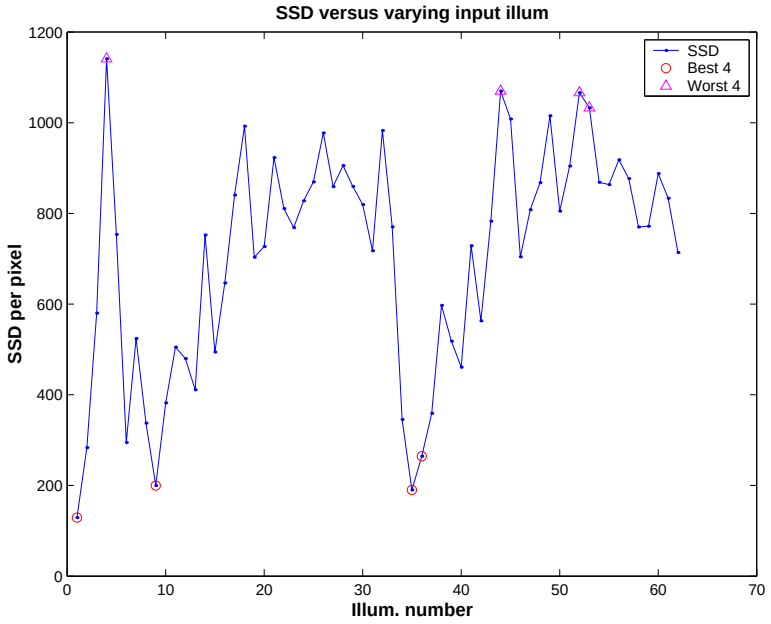

Figure 9: Plot of avg SSD vs. input illumination.

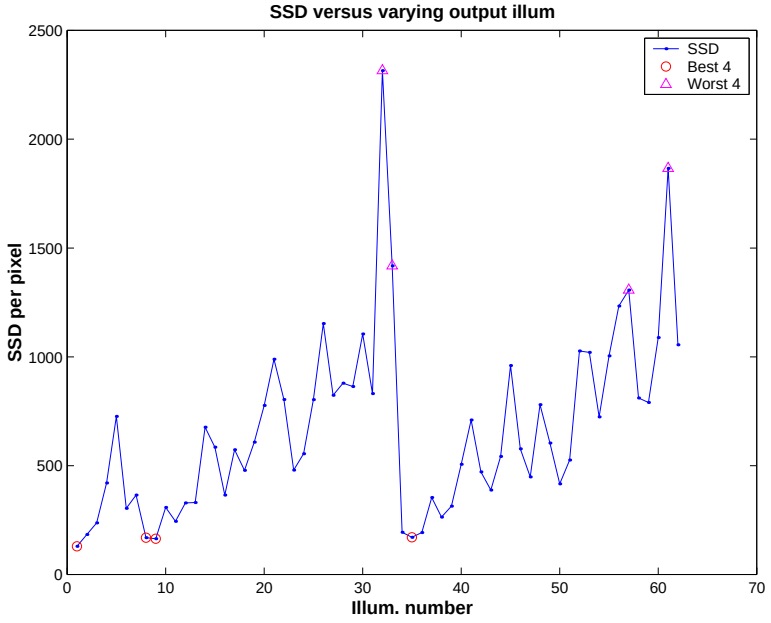

Figure 10: Plot of avg SSD vs. output illumination.

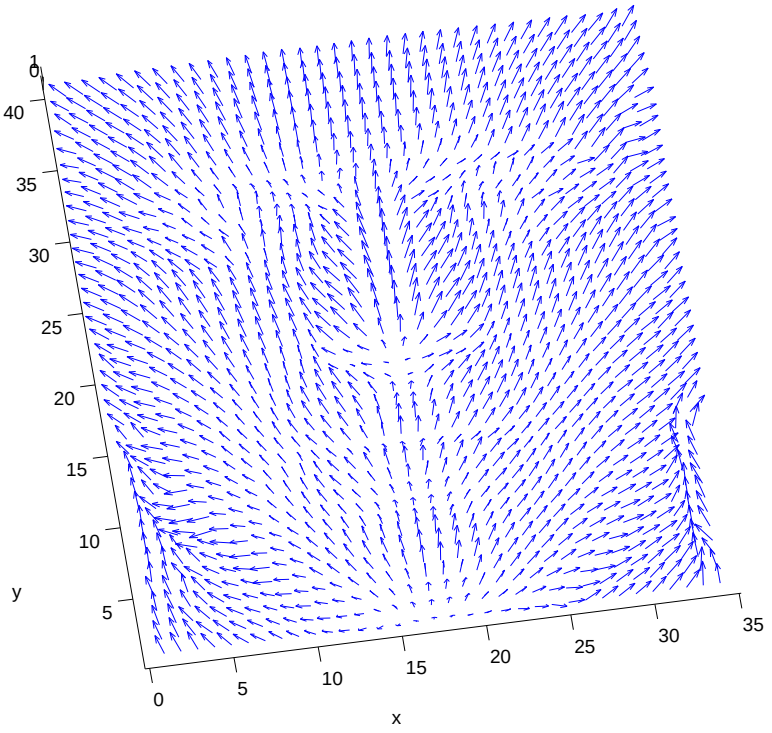

Figure 11: Needle map showing surface normals as little arrows. This is a downsampled version of the full needle map.

References

[1] J.J. Atick, P.A. Griffin, and A.N. Redlich. Statistical Approach to Shape from Shading: Reconstruction of 3-Dimensional Face Surfaces from Single 2-Dimensional Images. *NeurComp*, 8(6):1321–1340, August 1996.

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

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

[4] V. Blanz and T. Vetter. A Morphable Model for the Synthesis of 3D Faces. In *SIGGRAPH*, 1999.

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

[6] A.S. Georghiades, D.J. Kriegman, and P.N. Belhumeur. Illumination cones for recognition under variable lighting: Faces. In *CVPR98*, pages 52–59, 1998.

[7] B.K.P. Horn. Shape from Shading: A Method for Obtaining the Shape of a Smooth Opaque Object from One View. Technical Report AI TR, MIT, 1970.

[8] K. Ikeuchi and B.K.P. Horn. Numerical Shape from Shading and Occluding Boundaries. *Artificial Intelligence*, pages 17:141–184, 1981.

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

[10] C.H. Lee and A. Rosenfeld. Improved Methods of Estimating Shape from Shading Using the Light Source Coordinate System. *AI*, 26(2):125–143, 1985.

[11] 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.

[12] 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.

[13] L. Scharf. *Statistical Signal Processing: Detection, Estimation, and Time Series Analysis*. Addison-Wesley, 1991.

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

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