On the Effect of Illumination and Face Recognition

Jeffrey Ho
Department of CISE
University of Florida
Gainesville, FL 32611
Email: jho@cise.ufl.edu

David Kriegman
Department of Computer Science
University of California at San Diego
La Jolla, CA 92093
Email: kriegman@cs.ucsd.edu

Abstract

Illumination affects the appearance of an object in numerous ways, and the resulting variation in appearance is a major source of difficulty for designing many image-based applications such as face recognition. Partially because of this, the importance of understanding illumination effects has long been appreciated. However, only in the past few years, with the emergence of a wealth of new ideas and insights, a systematic and principled approach to illumination modelling has become possible. Some notable developments include the introduction of spherical harmonics for modelling illumination, the surprising result of non-existence of illumination invariants, and the idea of illumination cone. Accompanying these advances in illumination modelling are a number of recently published face recognition algorithms based on these new understandings of illumination effects. In this paper, we present an informal account of some of these exciting recent developments in studying illumination. In the process, we also provide a brief survey on these new face recognition algorithms, discussing their details, implementations and performances.

0.1 Introduction

Our experience in this predominantly visual world is enriched greatly by the diverse ways the world can be illuminated. While this diversity makes our world fascinating, it also makes recognition from images, face recognition in particular, difficult. As is evident in Figure 1, the effect of illumination on the appearance of a human face can be striking. The four images in the top row are images of an individual taken with the same viewpoint but under different external illumination conditions. The four images in the bottom, on the other hand, are images of four individuals taken under the same viewpoint and lighting. Using the most common measure of similarity between pairs of images, the L^2-difference[1], it is not surprising to learn that the L^2-difference between any pair of images in the bottom is always less than the L^2-difference between any pair of images from the top row. In other words, simple face recognition algorithms based purely on L^2-similarity are doomed to fail for these images. This result corroborates well the sentiment echoed through the often-quoted observation made more than a decade ago that "the variations between the images of the same face due to illumination ... are almost always larger than image variations due to change in face identity"[23].

Needless to say, a robust recognition system must be able to, among other things, identify an individual across variable illumination conditions. For decades, feature-based methods such as [13][18] (see surveys in e.g. [7] and references in [12]) have used properties and relations (e.g. distances and angles) between facial features such as eyes, mouth, nose and chin to perform recognition. However, reliable and consistent extraction of these features can be problematic under difficult illumination conditions, as images in Figure 1 clearly indicate. In fact, it has been claimed that methods for face recognition based on finding local image features and inferring identity by the geometric relations of these features are ineffective [6]. Image-based, or appearance-based, techniques (e.g. [24][32][34]), offer a different approach. For this type of algorithms, local image features no longer play significant roles. Instead, image-based techniques strive to construct low-dimensional representations of images of objects that, in the least square sense, faithfully represent the original images. For the particular problem of face recognition under varying illumination, some of the most successful methods (e.g. [2][8][12][21][31][37]) are image-based. For each person to be recognized, these algorithms use a

[1]For images with same number of pixels n, the L^2-difference between a pair of images is simply the usual L^2-difference between the two corresponding vectors in $\mathbb{R}^n$.

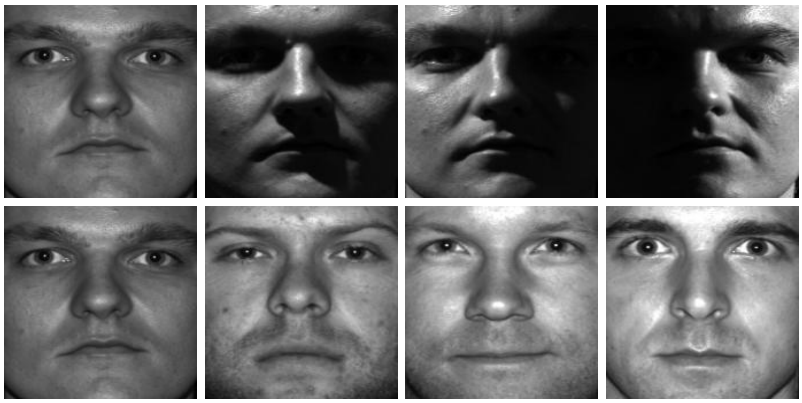

Figure 1: Striking effects of illumination on the appearances of a human face. **Top Row:** Images are taken with the same viewpoint but under different illumination conditions. **Bottom Row:** Images of four different individuals taken under the same viewpoint and illumination condition.

small number of (training) images to construct a low-dimensional representation of images of a given face under a wide range of illumination conditions. The low-dimensional representation is (except [8]) invariably some linear subspace in the image space, and the linearity makes the recognition algorithms efficient and easy to implement. The actual recognition process is straightforward and somewhat trivial: each query image is compared to each subspace in turn, by computing the usual L^2-distance between a linear subspace and an image (in vector form). The recognition result is the person in the database whose linear subspace has the minimal L^2-distance to the query image. What is non-trivial, however, is the discovery of the correct language and mathematical framework to model the effects of illumination [2][4][27], and the application of these illumination models to the designs of efficient face recognition algorithms [2][8][12][21][31][37] that explicitly model lighting variability using only a small number of training images. While pairwise L^2-comparisons between images would have failed miserably, L^2-comparisons between a query image and suitably-chosen (according to the illumination model) subspaces are robust against illumination variation.

Figure 1 illustrates the two main elements in modelling illumination effects: the variation in pixel intensity and the formation of shadows. As lighting varies, the radiance at each point on the object's surface also varies according to its *reflectance* . In general, surface reflectance can be described by a 4D function $\Omega(\theta_i, \phi_i, \theta_o, \phi_o)$, the Bidirectional Reflectance Distribution Function (BRDF), which gives the reflectance of a point on a surface as a function of the illumination geometry (θ_i, ϕ_i) and viewing geometry (θ_o, ϕ_o). (See Figure 2). A full BRDF with four independent variables is difficult to model and work with, e.g.[35]. Fortunately for face recognition, the much simpler Lambertian model [20] has been shown to be both sufficient and effective [4][12] for modelling the reflectance of human faces: the radiance (pixel intensity) I at each surface point is given by the inner product between the unit normal vector $\vec{n}$ scaled by the *albedo* value ρ and the light vector $\vec{L}$, which encodes the direction and magnitude of incident light coming from a distance,

$$I(\vec{L}) = \rho \max(\vec{L} \cdot \vec{n}, 0). \tag{1}$$

The lambertian model effectively collapses the usual 4D BRDF Ω into a constant function with value ρ. In particular, a lambertian object appears equally bright from all viewing directions. We note that Equation 1 is linear on the image level in the sense that the image of an object produced by two light sources is simply the sum of the two images produced by the sources individually. This, of course, is the familiar superposition principle of illumination, and it is the source of linearity appearing in all illumination models discussed below. However, because of the presence of the max term, Equation 1 is only quasi-linear in $\vec{L}$: $I(\vec{L}_1+\vec{L}_2) \neq I(\vec{L}_1)+I(\vec{L}_2)$ in general. This quasi-linearity in $\vec{L}$ is responsible for several tricky points in analyzing illumination effects, and it is related to the formation of attached shadows.

Shadows naturally account for significant portion of the variation in appearances. On a surface,

two types of shadows can appear: attached shadows and cast shadows (Figure 2). An attached shadow is formed when there is no irradiance at a point on surface. In other words, when the light source is on the "back side" of the point. This condition can be summarized concisely as $\vec{n} \cdot L < 0$, with $\vec{n}$ the normal vector. We note that the equality,$I(\vec{L}_1 + \vec{L}_2) = I(\vec{L}_1) + I(\vec{L}_2)$, fails precisely at pixels such that $\vec{n} \cdot L_1 < 0$ or $\vec{n} \cdot L_2 < 0$, i.e., most pixels in attached shadow. Cast shadows, on the other hand, are simply shadows the object casts on itself. Clearly, cast shadows are related to the object's global geometry, and local information such as normals does not determine their formation. Consequently, they are considerably more difficult to analyze (see [19], however).

In this paper, we discuss recent advances in modelling illumination effects [2][4][27] and various face recognition algorithms based on these foundational results [2][8][12][21][31][37]. At first glance, modelling the variability in appearance of a human face under all lighting conditions may seem to be intractable since, after all, the space of all lighting conditions is, in principle, infinite dimensional. However, it turns out that the variability caused by illumination can be effectively captured using low-dimensional linear models. This can be largely attributed to the 1) reflectance and 2) geometry of human faces. While human faces are generally not lambertian (as the often oily and specular forehead demonstrates), they can nevertheless be approximated well by one. That is, except for cast shadow, intensity variation and attached shadow can be succinctly modelled using only Equation 1. In fact, without the presence of cast shadows, illumination modelling for a lambertian object can be formulated under an elegant framework using spherical harmonic functions [2][27], and precise results concerning the dimensionality of the approximating subspaces and the faithfulness of the approximations can be given. Furthermore, image variation due to illumination can be completely characterized by enumerating a finite number of basis images [4]. Although these foundational results only deal with the ideal case of convex Lambertian objects (where cast shadows are absent), they nevertheless form the basis for the subsequent developments. It is largely due to the geometry of human faces that the successful applications of these ideas to face recognition can be made possible. While human face are generally not convex, they are quite nearly so, as the smoothly-curved forehead and almost-planar cheeks attest to (Figure 2). This renders the effect of cast shadows, although appears to be formidable at times, to be manageable. Several empirical results [9][12] have shown that even with cast shadows included, the appearance model is still low-dimensional, and its dimension is only slightly larger than the dimension predicted by the theory [2][27].

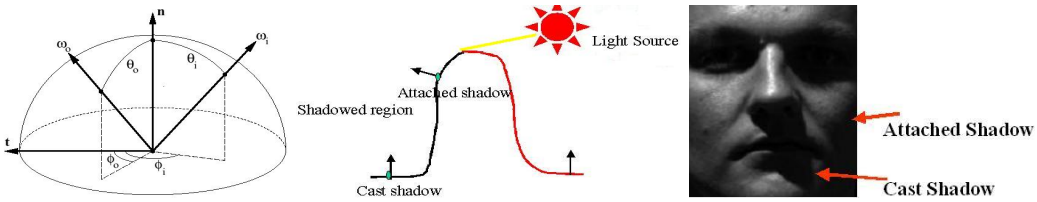

Figure 2: **Right** Coordinates system used in defining the Bidirectional Reflectance Distribution Function (BRDF). $\omega_i = (\theta_i, \phi_i)$ parametrizes the incoming lighting direction. $\omega_o = (\theta_o, \phi_o)$ represents the viewing direction. **n** is the normal vector. **Center** Attached and cast shadows. **Right** The formation of shadows on a human face. Attached shadows are in the upper region of the eye socket. Cast shadows appear in the lower region of the eye socket and the lower part of the face.

This paper is organized as follows. In the following section, we discuss an important result first appeared in [8] concerning the non-existence of illumination invariants. This interesting and somewhat unexpected result raises several subtle issues regarding illumination and face recognition. In the third section, the foundational results on illumination modelling are discussed. Section 4 contains a brief survey on several recently published face recognition algorithms based on these foundational results. Their performances and other experimental results are discussed in section 5. We conclude this paper with a short summary and remark on future work.

0.2 Non-Existence of Illumination invariants

Before delving into the details of various face recognition algorithms, we briefly discuss the important and interesting result of [8] on the non-existence of illumination invariants. Specifically, [8] demonstrates that for any two images, whether they are of the same object or not, there is always

a family of lambertian surfaces, albedo patterns and light sources that could have produced them. One consequence of this surprising result is quite counter-intuitive to our daily experiences: given two images, it is not possible with absolute certainty to determine whether they were created by the same or different objects. More specifically, [8] contains a proof of the following[2]:

Proposition 0.2.1 *Given two images I and J, and any two linearly independent vectors $\vec{s}$, $\vec{l} \in \mathbb{R}^3$, there exists a lambertian surface $\mathcal{S}$ such that the images of $\mathcal{S}$ taken under lighting conditions (point sources at infinity) specified by $\vec{s}$ and $\vec{l}$ are I and J, respectively.*

While the consequences of this proposition can be surprising, the motivation behind its proof, however, is straightforward. We remark that there is no analogous results for three or more images.

Let's assume that the lambertian surface $\mathcal{S}$ is viewed from the direction $(0, 0, 1)$, and $\mathcal{S}$ can be written as $(x, y, z = f(x, y))$, with $z \geq 0$ over some bounded rectangular region $\mathcal{R}$ in xy-plane. The images, I, J, are then considered as some non-negative functions on $\mathcal{R}$, i.e. $I, J \in \mathcal{P}$, where $\mathcal{P}$ denote the space of non-negative functions on $\mathcal{R}$. The space of lambertian objects is then precisely $\mathcal{P} \times \mathcal{P}$, with one factor for the geometry ($z = f(x, y)$) and the other for the albedo values. However, the variability offered by pairs of images is also $\mathcal{P} \times \mathcal{P}$. Therefore, given two fixed lighting conditions and a pair of images, we expect (heuristically) that at least one Lambertian surface can be responsible for the images. When the number of images is greater than two, we see that the variability in images are much larger than the space of lambertian surfaces. Therefore, for a generic triplet of images, one generally does not expect to find a lambertian surface that accounts for these images. Essentially as we will see soon, the proof exploits the following under-determined system of linear equations (in components of $\vec{n}$):

$$I(x, y) = \alpha(x, y)\vec{s} \cdot \hat{n}(x, y) = \vec{s} \cdot \vec{n}(x, y), \tag{2}$$

$$J(x, y) = \alpha(x, y)\vec{l} \cdot \hat{n}(x, y) = \vec{s} \cdot \vec{n}(x, y). \tag{3}$$

Here, $\hat{n}(x, y)$ is the unit normal and $\vec{n}(x, y) = \alpha(x, y)\hat{n}(x, y)$ is the albedo-scaled normal vector. For the case of three images, the analogous system will generally be invertible;therefore, a normal vector can be determined uniquely at any point (x, y). However, the resulting normal vector field formed by these normals determined point-wise will not be integrable [5] in general. So the inconsistency among triplet of images can be detected. For a pair of images and linearly independent $\vec{s}, \vec{l}$, because there is always a family of normal vectors satisfying the above equations at any point, we can produce an integrable normal field by choosing the normal at each point carefully.

To see how the proof works, we assume that the images I, J do not vanish simultaneously, $I(x, y) + J(x, y) > 0$ for all (x, y). This implies in particular that the albedos $\alpha(x, y)$ is also non-vanishing. The general case is only slightly more complicated. In addition, we also assume that $\vec{s} = (-1, 0, 1)$ and $\vec{l} = (1, 0, 1)$. The extension to arbitrary $\vec{s}$ and $\vec{l}$ will become clear later. Under these assumptions, let's consider $\mathcal{S}$ along a scanline, $y = c$ for some constant c (See Figure 3). Let $\mathbb{Y}_c$ denote the plane $y = c$. The intersection between $\mathcal{S}$ and $\mathbb{Y}_c$ defines a curve $\vec{c}(t) = (x(t), y(t), z(t))$. If $\vec{c}(t)$ satisfies the differential equation

$$\frac{d\vec{c}}{dt} = I\vec{l} - J\vec{s} \equiv \begin{cases} \frac{dx}{dt} = I + J \\ \frac{dy}{dt} = 0 \\ \frac{dz}{dt} = I - J \end{cases} \tag{4}$$

then because $\frac{d\vec{c}}{dt}$ is a tangent vector of $\mathcal{S}$, $0 = d\vec{c}/dt \cdot \vec{n} = (I\vec{l} - J\vec{s}) \cdot \vec{n}$ on $\vec{c}(t)$. For the system of ODEs above, a unique solution $\vec{c}$ can be found by integration provided that an initial condition (a point in $\mathbb{Y}_c$ of the form (x, c, y)) is also given. If such initial point is given, $\vec{c}(t)$ does indeed stay on the plane $\mathbb{Y}_c$ because $\frac{dy}{dt} = 0$. Furthermore, because $\frac{dx}{dt} > 0$ by our assumption, $x(t)$ is a strictly monotone function. This implies that $z(t)$ is a function of $x(t)$ on the "slice" $\mathbb{Y}_c \cap \mathcal{S}$. It follows that we can construct one particular $\mathcal{S}$ by specifying initial points along the left edge of the rectangular region $\mathcal{R}$ and integrating across all scanlines. If the initial points are chosen to be a smooth curve, it follows (e.g., [1]) that $\mathcal{S}$ will indeed be in the form $(x, y, z = f(x, y))$ for some smooth function f, and more

[2]In this discussion, we ignore the regularity assumptions. All surfaces and images are assumed to be infinitely differentiable (C^∞).

importantly, $(I\vec{l} - J\vec{s}) \cdot \vec{n} = 0$ everywhere on $\mathcal{S}$. Because $\vec{l} \cdot \vec{n}$ and $\vec{s} \cdot \vec{n}$ can not vanish simultaneously (since $\vec{n}$ is a multiple of $(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, -1)$), we have $I(x,y) = \alpha(x,y)\vec{s} \cdot \hat{n}$ and $J(x,y) = \alpha(x,y)\vec{l} \cdot \hat{n}$ at all (x,y) for some positive function α, the albedos.

This almost completes the proof, except, alas, Equations 2 and 3 are not quite the same as Equation 1. They will be so only if we can show that there is no (x,y) such that $\vec{s} \cdot \vec{n} < 0$ or $\vec{l} \cdot \vec{n} < 0$. Also, we have to show that the lights $\vec{l}$ and $\vec{s}$ do not cast shadows on $\mathcal{S}$; for otherwise, Equation 2 or 3 is not valid. Both can be easy demonstrated. Since $n = (-\frac{\partial z}{\partial x}, -\frac{\partial z}{\partial y}, 1)$ and $\frac{\partial z}{\partial x} = \frac{I-J}{I+J}$ (from Equation 4), a quick calculation gives $\vec{l} \cdot \vec{n} = 1 + \frac{I-J}{I+J}$ and $\vec{s} \cdot \vec{n} = 1 - \frac{I-J}{I+J}$, which are both non-negative everywhere on $\mathcal{S}$. Next, we show that there is no cast shadows. We note that for p to be in cast shadow under $\vec{l}$ (similarly for $\vec{s}$), the ray $p + t\vec{l}$, $t \geq 0$ must intersect $\mathcal{S}$ transversally (Figure 3), i.e., $\mathcal{S}$ must be on both sides of the ray. Since $\vec{l}$ has zero y-component, the ray and p are on the plane $\mathbb{Y}_c$ with c the y-component of p, i.e. we are over a scanline. So points of $\mathcal{S}$ that can cast shadow on p must be on the right of p (and for $\vec{s}$, they must be on the left). Points on the right of p are of the form

$$q = p + \int_0^{t=w} (I\vec{l} - J\vec{s})\, dt = p + \int_0^{t=w} I dt\, \vec{l} - \int_0^{t=w} J dt\, \vec{s} = p + a\vec{l} - b\vec{s}.$$

for some $w > 0$ and non-negative numbers a, b. Because b is non-negative, this immediately shows that $\mathcal{S}$ can not intersect the ray transversally, and hence p is not in shadow.

This completes the proof of the proposition for $\vec{s} = (-1, 0, 1)$ and $\vec{l} = (1, 0, 1)$. For general $\vec{s}, \vec{l}$, the proof above can be modified by defining the planes $\mathbb{Y}_c$ to be the affine planes $\mathbb{Y}_p$ generated by $\vec{s}$ and $\vec{l}$: $\mathbb{Y}_p = \{x | x = p + a\vec{s} + b\vec{l}, p \in \mathbb{R}^3, a, b \in \mathbb{R}\}$. This will ensure that each solution $\vec{c}(t)$ to Equation 4 stays in one such plane. The rest of the proof carries through without much change.

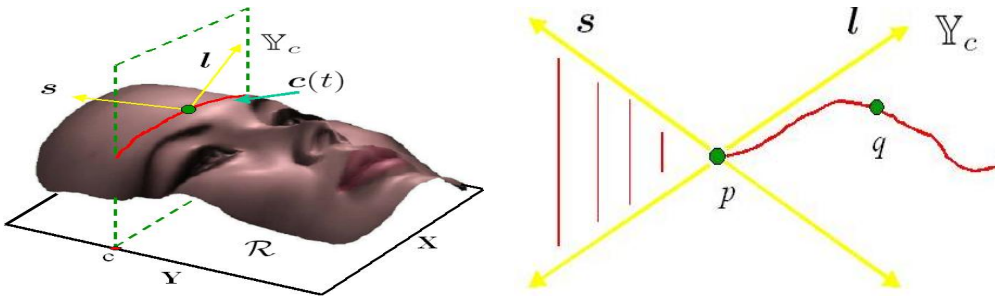

Figure 3: **Left:** The surface $\mathcal{S}$ is defined over a rectangular domain $\mathcal{R}$ in the xy-plane. The intersection $\mathbb{Y}_c \cup \mathcal{S}$ defines the curve $\vec{c}(t)$. **Right:** On each plane $\mathbb{Y}_c$ and any point $q \in \mathbb{Y}_c \cup \mathcal{S}$ that is to the right of p, because $q = p + a\vec{l} - b\vec{s}$ for some non-negative a, b, q has to lie inside the cone generated by $\vec{l}$ and $-\vec{s}$. Similarly, any point q that is to the left of p has to lie in the cone generated by $\vec{s}$ and $-\vec{l}$.

0.2.1 Image Gradients as Illumination Insensitive Measures

The negative result on the existence of illumination invariant is not as devastating as one may have thought. As far as face recognition is concerned, there are at least two ways out of this apparent quandary. While determining whether two images are of the same object is impossible in principle, nothing prevents us from doing so for three or more images. That is, we can increase the number of training images for each person in the database, and if qualitatively and quantitatively sufficient training images are available, un-determinacy can generally be avoided. We will discuss this type of approach in the next section.

On the other hand, Proposition 0.2.1 can be largely attributed to the unrestricted access to the space of lambertian objects, since we can always find some lambertian surface, however bizarre and strange it may be, to account for any two images. For example, the above theorem implies that given an image of Marilyn Monroe and one of Cary Grant, along with the light source directions, there exists a lambertian surface that could produce these images. However, it is unlikely to be face-like. Therefore, it makes sense to limit the space of available lambertian objects, e.g. to face-like objects.

Alternatively, let's consider only *planar* lambertian objects. It follows directly from Equation 1 that the image gradient is a discriminative illumination invariant: given two images I and J of some planar lambertian object taken under same viewpoint, their image gradients $\nabla I, \nabla J$ must be parallel at every pixel where they are defined. This is obvious because for planar object, there is only one surface normal and each image is simply a constant multiple of the albedo values with the constant been determined by the illumination. While the pixel intensity can be any allowable value given appropriate illumination, its derivative, the image gradient, cannot. Probabilistically, the distribution of pixel values under varying illumination may be random, but the distribution of image gradients is not.

Unfortunately, the dependence of image gradient on albedos is only part of the story. For general non-planar surfaces, the image gradient (for a given light source $\vec{s} = (s_u, s_v)$) is related to both the albedos (reflectance) and surface geometry:

$$\nabla I = \overbrace{\hat{u}\kappa_u s_u + \hat{v}\kappa_v s_v}^{geometric} + \underbrace{(\nabla\alpha)\vec{s}\cdot\hat{n}}_{reflectance}. \tag{5}$$

In the above, κ_v, κ_u are the two principal curvatures, and $\hat{u}, \hat{v}$ are the corresponding principal directions at a given surface point[3]. For a planar object $\kappa_v, \kappa_u = 0$ and $\nabla I = (\nabla\alpha)\vec{s}\cdot\hat{n}$. The geometric term in the equation above destroys the simple relation between image gradient and albedo gradient that we have for planar objects. However, for the case of uniform albedos (i.e. $\nabla\alpha = 0$), a deeper analysis using only the geometric term above reveals that the image gradient distribution is still not random. More specifically, for light sources with a directionally-uniform distribution given by

$$\rho_s(\vec{s} = (s_u, s_v, s_n)) = \frac{1}{(\sqrt{2\pi}\sigma)^3} e^{-\frac{1}{2\sigma^2}(s_u^2 + s_v^2 + s_n^2)}, \;\; s_n \in [0, \infty),$$

the image gradient distribution is

$$\rho(u, v) = \frac{1}{\pi^{\frac{3}{2}}\sigma^2\kappa_u\kappa_v} e^{\frac{1}{2\sigma^2}((\frac{u}{\kappa_u})^2 + (\frac{v}{\kappa_v})^2)}.$$

This result strongly suggests that the joint distribution of two image gradients from two different images under two random lighting should not be random either. In its most general form, the probability density function for this joint distribution can be written as [8]:

$$\rho(r_1, \varphi_1, r_2, \varphi_2) = \int \rho(r_1, \phi_1 | \kappa, \vec{\alpha})\varphi(r_2, \varphi_2 | \kappa, \vec{\alpha}) dP(\gamma, \kappa, \vec{\alpha}), \tag{6}$$

where $P(\gamma, \kappa, \alpha)$ is the probability measure on the non-observable random variables that include the surface geometry (κ), albedos (α) and camera viewpoints (γ). In the expression above, r_i and φ_i are the magnitude and orientation of the image gradient, respectively. Equation 6 is only of theoretical interest since the probability measure P on the non-observables is unknown. However, we can try to reconstruct the distribution empirically using images of objects under varying illumination. For this purpose, a slightly different joint distribution (on the angular difference of two image gradients), $\rho(r_1, \varphi = \varphi_1 - \varphi_2, r_2)$, is easier to work with.

In [8], 1280 images of 20 objects under 64 different illumination conditions were gathered. The objects included folded cloth, a computer keyboard, cups, a styrofoam mannequin, etc. $\rho(r_1, \varphi, r_2)$ is estimated directly from a histogram of image gradients. A slice of the joint probability density ρ is shown in Figure 4(Left). Note that for a planar (or piece-wise planar) lambertian object, ρ is a delta function at $\varphi = 0$ (angular difference is 0). It is expected that with contribution from surface geometry and other factors, ρ should be considerably more complicated for general objects. However, the shape of ρ with its prominent ridge at $\varphi = 0$ does resemble that of a delta function. Surface

[3]Equation 5 is really an equation in terms of a coordinates system $\hat{u}, \hat{v}$ at the tangent space of the surface, not the image plane. Following [8], we will ignore the effects of projection, and treat $\hat{u}, \hat{v}$ as directions in the image.

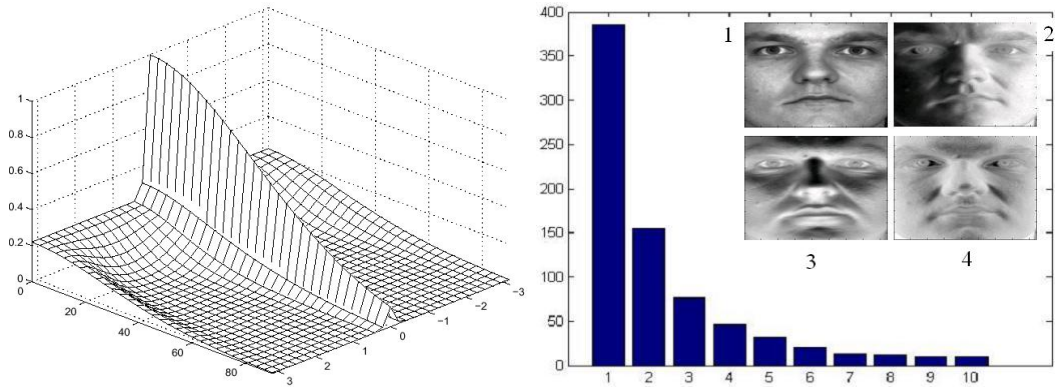

Figure 4: **Left:** (Courtesy of [8]) Empirical joint probability density of two image gradients $\rho(r_1, \varphi, r_2 = 50)$ under two random lighting conditions. **Right:** The magnitudes of first ten singular values for the images in Figure 11. In this example, the first three eigenvalues account for more than 97% of the energy. The four Eigenfaces corresponding to the largest four eigenvalues are also displayed.

geometry accounts for most of the "spread" of the density from the line $\varphi = 0$. This shows that the statistical regularity of scene radiance gradient does reflect the intrinsic geometry and reflectance properties of surfaces, and this regularity can then be exploited for face recognition as we will detail in Section 5.

0.3 Theory and Foundational Results

Let $\mathcal{C}$ denote the set of images of an object $\mathcal{O}$ under all possible illumination conditions. One of the main goals of illumination modelling is to say something about $\mathcal{C}$. We assume that the images were taken under the same viewpoint, i.e. no pose variation, and the images are all of the same size. By the usual rasterization, we can regard $\mathcal{C}$ as a subset of the image space $\mathbb{R}^n$, with n the number of pixels in the image. In this section, we discuss the important results of [2][4][27], which give various characterizations of the set $\mathcal{C}$ when the object $\mathcal{O}$ is lambertian and convex. There are two main themes, the effective low-dimensionality of $\mathcal{C}$ and its linearity.

Before moving on, we fix a few conventions and notations. Inter-reflections will be ignored throughout all subsequent discussions, and all illumination conditions will be assumed to be homogeneous, i.e. generated by distant sources. In particular, if the distant source l is a point source, then l can be represented as a 3-vector $\vec{l}$ such that $|l|$ encodes the magnitude of the source and the unit vector $\vec{l}/|\vec{l}|$ represents its direction. Note that the unit vector is a point on the sphere S^2, and conversely, every point $\vec{p}$ on S^2 can represent some distant point source with direction $\vec{p}$. More generally, any illumination condition can be represented as a non-negative function on S^2. For an image I, we will use the same symbol I to denote both the image *and* its associated vector in the image space.

0.3.1 Early Empirical Observations

Convexity and low-dimensionality are two important properties of $\mathcal{C}$. Convexity is a simple consequence of the superposition principle for illumination. For if I_1 and I_2 are two images taken under two different illumination conditions l_1 and l_2, any convex combination of these two images:

$$J = aI_1 + bI_2, \qquad a, b \geq 0, \quad a + b = 1,$$

is also an image of the same object under a new illumination condition specified by $al_1 \cup bl_2$, i.e. l_1 and l_2 are "turned on" simultaneously with attenuation factors a, b, respectively. This should not be confused with the illumination given by the point distant source $a\vec{l_1} + b\vec{l_2}$, when l_1, l_2 are distant point sources.

The fact that for objects with diffuse, lambertian-like reflectance, the effective dimension of $\mathcal{C}$ is small was also noticed quite early [9][15]. This can be demonstrated by collecting images of an object taken under a number of different illumination conditions. If $\{I_1, \cdots, I_m\}$ are m such images, we can

stack them horizontally to form the intensity matrix $I = [I_1 \cdots I_m]$. Singular value decomposition (SVD) of I [14]:

$$I = U\Sigma V^t, \tag{7}$$

gives the singular vectors as the columns of the matrix U, and the diagonal elements of Σ as the singular values. Let $\{\sigma_1, \cdots, \sigma_m\}$ denote the singular values in descending order. Incidently, the singular vectors are usually called Eigenimages and in the case of face images, they are called, appropriately, *Eigenfaces* [34]. The eigenimages can be used to approximate the original images $\{I_1, \cdots, I_m\}$, and if R denotes the subspace spanned by the k Eigenimages associated to the k largest singular values, the L^2 reconstruction error,

$$\sum_{i=1}^{m} \text{dist}^2_{L^2}(I_i, R) = \sum_{i=k+1}^{m} \sigma_i^2. \tag{8}$$

can be computed directly from the singular values. If σ_i turns out to be negligible for $i > k$, then the entire collection of images can be effective approximated using the subspace R. In particular, the effective dimension of $\{I_1, \cdots, I_m\}$ is simply the dimension of R, which is k.

Figure 4(Right) displays the magnitudes of the first ten singular values obtained by applying SVD to a collection of 45 images of a human face (in frontal pose, and under 45 different point light sources) shown in Figure 11 (Section 5). The magnitudes of the singular values decrease rapidly after the first three singular values. In fact, the first three eigenvalues account for more than 97% of the entire energy. Here, the energy is defined as $\sum_{i=1}^{m} \sigma_i^2$. For a pure lambertian object with simple geometry, this observation can be explained easily. Assuming $\{I_1, \cdots, I_m\}$ contain no shadows, then, the intensity matrix I factored as

$$I = B \cdot S = [\vec{n}_1 \cdots \vec{n}_n]^t \cdot [\vec{s}_1 \cdots \vec{s}_k] \tag{9}$$

where B is a n-by-3 matrix containing the normals and albedos at each pixel, and s_i are the light source directions. Since S can have rank at most 3, I also has rank at most 3, and hence there are at most three non-zero singular values in Σ. For a general collection of images, the object is no longer Lambertian with simple geometry, and the lighting conditions are not describable by point sources. This means that there will be more than three non-zero singular values, and the extent of this "spread of singular values" depends on how many of the idealized assumptions have been violated.

In [9], eigen-analysis similar to the one above was applied to images of non-lambertian objects. These include objects with specular spikes, small cast shadows and some other irregularities such as partial occlusions. The conclusion from this empirical study is surprising in that 5 ± 2 Eigenimages are sufficient to model objects with wide range of reflectance properties. As mentioned earlier, 3 at the lower end of 5 ± 2 can be used to model lambertian objects with simple geometry. In their conclusion, the first few eigenimages describe the lambertian component, and the succeeding eigenimages describe the specular component and specular spikes, shadows and so forth. This result is particularly encouraging because human faces are generally non-lambertian. Still, a low-dimensional linear representation is already sufficient to capture a large portion of possible image variation due to illumination.

0.3.2 Modelling Reflectance and Illumination using Spherical Harmonics.

The effective low-dimensionality of $\mathcal{C}$ that we have just discussed clearly begs for explanations. Somewhat surprisingly, this empirical observation can be elegantly explained via a signal processing framework using spherical harmonics [2, 26, 27]. The key conceptual advance is to treat a Lambertian object as some "low-pass filter" that turns complicated external illuminations into smoothly shaded images. In the context of illumination, the signals are functions defined on the sphere, and spherical harmonics are the analogues of the fourier basis functions.

First, we fix a local (x, y, z) coordinates system F_p at a point p on a convex lambertian object such that the z-axis coincides with the surface normal at p. Let (r, θ, ϕ)[4]denote the spherical coordinates

[4]To conform with the notation used in spherical harmonics literature, θ denotes the elevation angle and ϕ denotes the azimuth angle.

centered at p. Under the assumption of homogeneous light sources, the configuration of lights that illuminates the object can be expressed as a non-negative function $L(\theta, \phi)$ defined on S^2. The reflected radiance at p is then given by

$$r(p) = \rho \int_{S^2} k(\theta) L(\theta, \phi) dA = \rho \int_0^{2\pi} \int_0^{\pi} k(\theta) L(\theta, \phi) sin\theta d\theta d\phi, \tag{10}$$

where ρ is the albedo, and $k(\theta) = \max(\cos\theta, 0)$ is the Lambertian kernel. Note that this equation is simply the integral form of Equation 1, in which we integrate over all possible incident directions at p. Because the normal at p coincides with the z-axis, the Lambertian kernel is precisely the max term in Equation 1. For any other point q on the surface, the reflectance is computed by a similar integral as above. The only difference between the integrals at p and q is the lighting function L: at each point, L is expressed in a local coordinates system at that point. Therefore, considered as a function on the unit sphere, L_p and L_q differ by a rotation $g \in SO(3)$ that rotates the frame F_p to F_q. That is, $L_p(\theta, \phi) = L_q(g(\theta, \phi))$.

Since $k(\theta)$ and $L(\theta, \phi)$ are now functions on S^2, the natural thing to do next is to expand these functions in terms of some canonical basis functions, and spherical harmonics offer a convenient choice. Spherical harmonics, Y_{lm}, are a set of functions that form an orthonormal basis for the set of all square-integrable (L^2) functions defined on the unit sphere. They are the analogue on the sphere to the Fourier basis on the line or circle. Y_{lm}, indexed by two integers l (degree) and m (order) obeying $l \geq 0$ and $-l \leq m \leq l$, has the following form:

$$Y_{lm}(\theta, \phi) = \begin{cases} N_{lm} P_l^{|m|}(cos\theta) cos(|m|\phi) & \text{if } m > 0; \\ N_{lm} P_l^{|m|}(cos\theta) & \text{if } m = 0; \\ N_{lm} P_l^{|m|}(cos\theta) sin(|m|\phi) & \text{if } m < 0; \end{cases} \tag{11}$$

where N_{lm} is a normalization constant that guarantees the functions Y_{lm} are orthonormal in the L^2-sense:

$$\int_{S^2} Y_{lm}\, Y_{l'm'} dA = \delta_{mm'} \delta_{ll'}.$$

$P_l^{|m|}$ is the associated Legendre functions whose precise definition is not important here (however, see [33]). The formal definition of Y_{lm} using spherical coordinates above is somewhat awkward to work with. Instead, it is usually more convenient to write Y_{lm} as a function of x, y, z rather than angles. Each $Y_{lm}(x, y, z)$ expressed in terms of (x, y, z) is a polynomial in (x, y, z) of degree l:

$$Y_{00} = \sqrt{\frac{1}{4\pi}}, \tag{12}$$

$$(Y_{11}; Y_{1-1}; Y_{10}) = \sqrt{\frac{3}{4\pi}}(x; y; z), \tag{13}$$

$$(Y_{21}; Y_{2-1}; Y_{2-2}) = \sqrt{\frac{15}{4\pi}}(xz;\, yz;\, xy),\ (Y_{20}; Y_{22}) = \sqrt{\frac{5}{16\pi}}(3z^2 - 1;\, \sqrt{3}(x^2 - y^2)) \tag{14}$$

In other words, spherical harmonics of degree l are just the restrictions of some homogeneous polynomials (in x, y, z) of degree l to S^2. While degree-two polynomials in x, y, z are six-dimensional $(xy, yz, zx, x^2, y^2, z^2)$, because $x^2 + y^2 + z^2 - 1 = 0$ on S^2, spherical harmonics of degree two are only five-dimensional. Using polynomials, it is straightforward to see that a rotated spherical harmonic is a linear superposition of spherical harmonics of same degree since a rotated homogeneous polynomial of degree l is a polynomial of the same degree. Therefore, for a 3D rotation $g \in SO(3)$,

$$Y_{lm}(g(\theta, \phi)) = \sum_{n=-l}^{l} g_{mn}^l Y_{ln}(\theta, \phi). \tag{15}$$

The coefficients g_{nm}^l are real numbers and are determined by the rotation g.

With these basic properties of spherical harmonics in hand, the idea of viewing a lambertian object as a "low-pass filter" can be made precise by expanding the lambertian kernel $k(\theta)$ in terms of Y_{lm}. Because $k(\theta)$ has no ϕ-dependency, its expansion, $k(\theta) = \sum_{l=0}^{\infty} k_l Y_{l0}$, has no Y_{lm} components with $m \neq 0$ (Equation 11). It can be shown [2, 27] that k_l vanishes for odd values of $l > 1$, and the even terms fall to zero rapidly; in addition, more that 99% of the L^2-energy of $k(\theta)$[5] is captured by its first three terms, those with $l < 3$. See Figure 5(Left). Because of these numerical properties of k_l and the orthogonality of the spherical harmonics, any high-frequency ($l > 2$) component of the lighting function $L(\theta, \phi)$ will be severely attenuated in evaluating the integral in Equation 10, and in this sense, the Lambertian kernel acts as a low-pass filter. Therefore, the reflected radiance computed using Equation 10 can be accurately approximated by the same integral with L replaced by L', obtained by truncating the harmonic expansion of L at $l > 2$, i.e. the spherical harmonics expansion of L' contains no Y_{lm} with $l > 2$. Since rotations preserve the l-degree of the spherical harmonics (rf. Equation 15), the same truncated L' will work at every point. Let $L'(\theta, \phi) = \sum_{i=1}^{9} l_i Y_i$ denote the expansion of L' and Y_i the nine spherical harmonics with degree < 3[6]. At any point q, we have

$$r(q) \approx \rho_q \int_{S^2} k_q(g(\theta)) L'(\theta, \phi) dA = \rho_q \sum_{i=1}^{9} l_i \int_0^{2\pi} \int_0^{\pi} k_q(g(\theta)) Y_i dA, \tag{16}$$

where g is the rotation that rotates the local frame F_q at q to the frame F_p. We can define the nine *harmonic images* I_i whose intensity at each point (pixel) is $I_i(q) = \rho_q \int_{S^2} k_q(g(\theta)) Y_i dA$: images taken under the *virtual* lighting conditions specified by the nine spherical harmonics. Hence, the pixel-wise approximation above translates into the approximation for images.

$$I \approx \sum_{i=1}^{9} l_i I_i. \tag{17}$$

If I is an image taken under some illumination condition L with l_i as the nine coefficients in L's truncated spherical harmonics expansion, I can be approximated by a linear combination of the nine harmonic images using the same coefficients. The far-reaching consequence of this fact is that although lighting conditions are infinite-dimensional (the function space for $L(\theta, \phi)$), the illumination effects on a lambertian object can be approximated by a nine-dimensional linear subspace $\mathcal{H}$, the harmonic subspace spanned by the harmonic images I_i, i.e. $\mathcal{C}$ can be approximated well by $\mathcal{H}$.

Harmonic Images

Equation 17 indicates the great importance of computing the harmonic images. Except for the first spherical harmonic (which is a constant), all others have negative values and therefore, they do not correspond to real lighting conditions. Hence, the corresponding harmonic images are not real images, and as pointed out by [2]: "they are abstractions." Nevertheless, they can be computed quickly if the object's surface normals and albedos are known.

Using the polynomial definition of spherical harmonics, the recipe for computing the nine harmonic images I_i for $(1 \leq i \leq 9)$ is particularly simple: for each pixel p, let $\vec{n}_p = (x, y, z)$ denote the unit surface normal at p and ρ_p the albedo. The intensity value of I_i at p is given by

$$I_i(p) = \rho_p Y_i(x, y, z). \tag{18}$$

Another way to compute the harmonic images is to simulate the images under harmonic lightings by explicitly evaluating the integral $\rho_p \int_{S^2} k_p(g(\theta)) Y_i dA$ at every point p and taking into account the cast shadows:

$$\rho_p \int_{S^2} k_p(g(\theta, \phi)) \nu_p(\theta, \phi) Y_i dA, \tag{19}$$

where $\nu_p(\theta, \phi) = 1$ if the ray coming from direction (θ, ϕ) is not occluded by another point on the surface. Otherwise, $\nu_p(\theta, \phi) = 0$. Figure 5 shows the rendered harmonic images for a face taken from

[5] The energy of $k(\theta)$ is, $\int_{S^2} k^2(\theta) dA$, which is the convergent infinite sum $\sum_{i=0}^{\infty} k_l^2$.

[6] $Y_1 = Y_{00}$, $Y_2 = Y_{11}$, $Y_3 = Y_{1-1}$, $Y_4 = Y_{10}$, $Y_5 = Y_{21}$, $Y_6 = Y_{2-1}$, $Y_7 = Y_{2-2}$, $Y_8 = Y_{20}$, $Y_9 = Y_{22}$

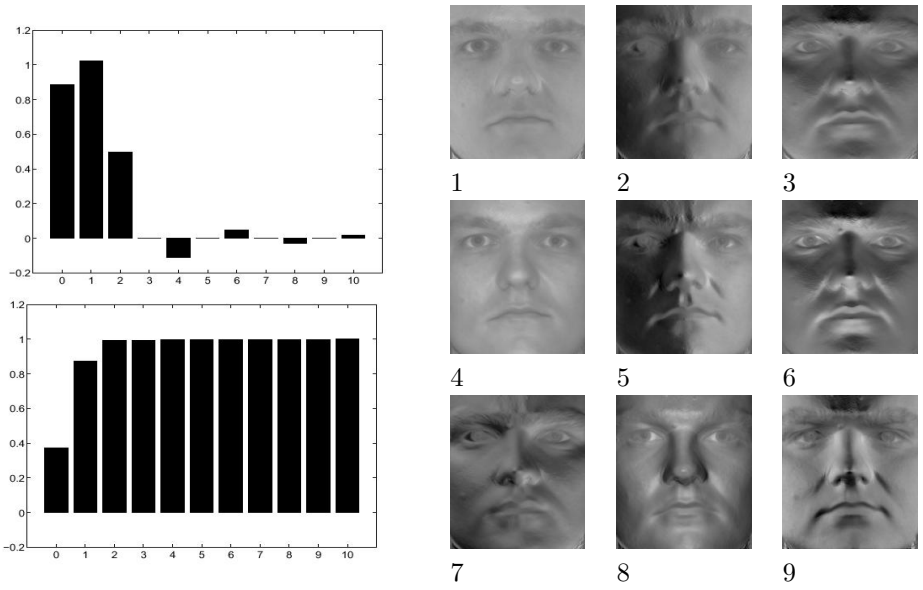

Figure 5: Analysis using spherical harmonics **Left:** (Courtesy of [2]) Top: A graph representation of the first eleven coefficients in the spherical harmonics expansion of the Lambertian kernel $k(\theta)$. Bottom: the cumulative energy. **Right**: The nine harmonic images rendered by ray-tracing. For examples of computing harmonic images without using ray-tracing (Equation 18), see [26].

the Yale Database. These synthetic images are rendered by sampling 1000 rays on a hemisphere, and the final images are the weighted sum of 1000 ray-traced images. By using the 3D information, these harmonic images also include the effects of cast shadows arising from the non-convex human face.

0.3.3 Illumination Cones and Beyond

So far,we have shown that $\mathcal{C}$, the set of images of a convex Lambertian object under all possible illumination conditions, is a convex set in the image space, and $\mathcal{C}$ can be effectively approximated by a nine-dimensional linear subspace. However, we still do not know what $\mathcal{C}$ is. An explicit characterization of $\mathcal{C}$ was first studied and answered in [4]. This paper shows that for a convex Lambertian object, $\mathcal{C}$ is a polyhedral cone in the image space, and its complexity (i.e., number of generators) is quadratic in the number of distinct surface normals. In this subsection, we discuss this result and some of its implications. First, we recall that a cone in $\mathbb{R}^n$ is simply a convex subset of $\mathbb{R}^n$ that is invariant under non-negative scalings: if x is in the cone, then λx is also in the cone for any non-negative λ. A polyhedral cone is simply a cone with a *finite* number of generators $\{e_1, \cdots, e_l\}$: points of the cone are vectors x in $\mathbb{R}^n$ that can be expressed as some non-negative linear combination of the generators, $x = a_1 e_1 + \cdots + a_l e_l$ with $a_1, \cdots, a_l \geq 0$.

For a general object (without assumptions on reflectance and geometry), it is straightforward to establish that the set $\mathcal{C}$ is a convex cone, which is a direct consequence of the superposition property of illumination. This is the simplest and also the only available characterization of $\mathcal{C}$ without any limiting assumptions on reflectance or geometry. However, for the case of a convex and Lambertian object, more can be said about $\mathcal{C}$ [4]:

Proposition 0.3.1 *For a convex Lambertian object, the set $\mathcal{C}$ of images of an object under all possible illumination conditions is a polyhedral cone.*

The finiteness of the number of generators for $\mathcal{C}$ is the proposition's main strength. Instead of just an approximation, we know the entire geometry of $\mathcal{C}$ from a finite number of images. For a general non-lambertian object, there is no analogous result.

Let I denote an image of a convex object with n pixels. Let $B \in \mathbb{R}^{3n}$ be a matrix where each row of B is the product of the albedo with the inward pointing unit normal for a point on the surface

projecting to a particular pixels[7]. Although the proposition deals with all possible illumination conditions, we can actually focus our analysis on single distant sources, thanks to the superposition principle. Thus, we need to examine the set $\mathcal{U}$ of images of a convex Lambertian surface created by varying the direction and strength of a *single* point light source at infinity. It will turn out that $\mathcal{U}$ can be decomposed into a collection of polyhedral subcones $\mathcal{U}_i$ indexed by the set $\mathcal{S}$ of *shadowing configurations*:

$$\mathcal{U} = \{I | I = \max(Bs, 0),\ \forall s \in \mathbb{R}^3\} = \bigcup_{i \in \mathcal{S}} \mathcal{U}_i. \tag{20}$$

As its name suggested, each element $i \in \mathcal{S}$ indexes a particular shadowing configuration, and $\mathcal{U}_i$ indexed by i is a set of images, each with the same pixels in shadow and the same pixels illuminated (images with the same "shadowing configuration"). Since the object is assumed to be convex, all shadows are attached shadows. The technical part of the proof for the above proposition is then to give a bound on the size of the set $\mathcal{S}$ as given in the following Lemma:

Lemma 0.3.2 *The number of shadowing configurations is at most $m(m-1)+2$, where $m \leq n$ is the number of distinct surface normals.*

To see how to count the elements in $\mathcal{S}$, we begin with a definition. As usual, we first ignore the max term in Equation 20. The products of B with all possible light source directions and strengths sweeps out a subspace in $\mathbb{R}^3$, and we call this subspace the illumination subspace $\mathcal{L}$, where

$$\mathcal{L} = \{I | x = Bs,\ \forall s \in \mathbb{R}^3\}.$$

Note that the dimension of $\mathcal{L}$ equals the rank of B. Since B is an $n \times 3$ matrix, $\mathcal{L}$ will in general be a 3-D subspace, and we will assume it to be so in the following discussion. The important point now is to observe that $\mathcal{L}$ slices through different orthants of $\mathbb{R}^n$. The most conspicuous one is the intersection of $\mathcal{L}$ with the non-negative orthant of $\mathbb{R}^n$, and the intersection is non-empty because when a single light source is parallel to the camera's optical axis, all visible points on the surfaces are illuminated, and consequently, all pixels in the image have non-zero values, and the image has no shadow. What can be said about the intersections of $\mathcal{L}$ with other orthants? Let $\mathcal{L}_i$ denote the intersection of $\mathcal{L}$ with an orthant i. Clearly, some components of $x \in \mathcal{L}_i$ are always negative and others always greater than or equal to zero. To turn x into a real image, we have to apply the max term above and this leaves the non-negative components of $x \in \mathcal{L}_i$ untouched, while the negative components of x go to zero. Note that this operation is a linear projection P_i on $\mathcal{L}_i$ that maps $\mathcal{L}_i$ to the closure of the non-negative orthant. We then clearly have the decomposition of $\mathcal{U}$ into subcones:

$$\mathcal{U} = \bigcup_i P_i(\mathcal{L}_i),$$

$P_i(\mathcal{L}_i)$ is a cone because P_i is linear and $\mathcal{L}_i$ is a cone. In fact, we can identify each $P_i(\mathcal{L}_i)$ with $\mathcal{U}_i$ in Equation 20 because we can identify the set $\mathcal{S}$ with the set of orthants having non-empty intersection with $\mathcal{L}$, according to the discussion above. Although there are 2^n orthants in $\mathbb{R}^n$, we will see below that $\mathcal{L}$ can only intersect at most $n(n-1)+2$ orthants.

Representing all possible light source directions by the sphere S^2, we see that for a convex object, the set of light source directions for which a given pixel in the image is illuminated corresponds to an open hemisphere; the set of light source directions for which the pixel is shadowed corresponds to the other hemisphere of points. The boundary is the great circle defined by $\vec{n} \cdot s = 0$, where $\vec{n}$ is the normal at the given pixel. Each of the n pixels in the image has a corresponding great circle on the illumination sphere, and there are m distinct great circles in total, where m is the number of distinct surface normals. The collection of great circles carves up the surface of S^2 into a collection of cells S_i. See Figure 6. The collection of light source directions contained within a cell S_i on the sphere produces a set of images, each with the same pixels in shadow and the same pixels illuminated. This, again, immediately identifies the set $\mathcal{S}$ with the set of cells S_i, and hence, the

[7]Here we effectively approximate a smooth surface normals for the set of points projecting to the same image pixel are identical.

size of $\mathcal{S}$ is the number of such cells on S^2. It is then a simple inductive argument, using the fact that two great circles intersect at two different points, to show that the number of cells S_i can not exceed $m(m-1)+2$. Furthermore, the cone's generators are given by the images produced by light sources at the intersection to two great circles. This then immediately implies that the number of generators of $\mathcal{C}$ is quadratic in the number of distinct surface normal m.

With $\mathcal{U}$ understood, we can now construct the set $\mathcal{C}$ of all possible images of a convex Lambertian surface created by varying the direction and strength of an *arbitrary number* of point light sources at infinity,

$$\mathcal{C} = \{I | I = \sum_{i=1}^{k} \max(Bs_i, 0), \forall s_i \in \mathbb{R}^3, \forall k \in \mathsf{Z}^+\},$$

where Z^+ is the set of positive integers. The above discussion on $\mathcal{U}$ then immediately shows that $\mathcal{C}$ is a polyhedral cone.

Some Properties of an Illumination Cone

Since the illumination cone $\mathcal{C}$ is completely determined by the illumination subspace $\mathcal{L}$, $\mathcal{C}$ can be determined uniquely if the surface normals scaled by albedo B were known. The method from photometric stereo ([36]) allows us to recover B up to an invertible 3×3 linear transformation $A \in GL(3)$

$$Bs = (BA)(A^{-1}s) = B^*s^*$$

by using as few as three images since $\mathcal{L}$ is a 3-D subspace. Although B is not uniquely determined, nevertheless, it is easy to see that B and B^* determine the same illumination subspace, and hence, the same illumination cone.

Another interesting result proved in [4] is that the actual dimension of $\mathcal{C}$ is equal to the number of distinct surface normals. For images with n pixels, this indicates that the dimension of the illumination cone is one for a planar object, is roughly $\sqrt{n}$ for a cylindrical object, and is n for a spherical object. It is to be noted, however, that having a cone span n dimensions does not mean that it covers $\mathbb{R}^n$. It is conceivable that an illumination cone could completely cover the positive orthant of $\mathbb{R}^n$. However, the existence of an object geometry that would produce this is unlikely, since for such an object, it must be possible to choose n light source directions such that each of the n facets (pixels) are illuminated independently. On the other hand, a cone that covers the entire positive orthant can not be approximated by a low-dimensional linear subspace, and this would contradict our analysis in the previous section using spherical harmonics. In particular, the result from the previous section indicates that the shape of the cone is "flat" with most of its volume concentrated near a low dimensional subspace. From a face recognition viewpoint, this is encouraging because it indicates the possibility that the illumination cones for different objects are small (compared with the ambient space $\mathbb{R}^n$) and well separated. Recognition using illumination cone should then be possible, even under extreme lighting conditions.

To compute an illumination cone, we need to obtain the illumination basis (generators of the illumination cone) first. However, these basis images all belong to the boundary of the illumination cone and therefore, compared to images in the interior, these boundary images are closer to images in other illumination cones (from other individuals). From face recognition viewpoint, they are the difficult images to recognize correctly. Conversely, images in cone's interior are relatively easy provided that different illumination cones do not have serious intersections. They typically include images taken under diffused ambient lighting conditions and with little or no shadows on them. [22] contains some preliminary experiment results supporting this observation. Combining these two observations, we have an explanation for the obvious fact that images with shadows are harder to recognize than those without.

Illumination Bases Are Not Equal

While the illumination cone provides a satisfying characterization of $\mathcal{C}$, its computation is, in principle, not feasible for most objects. This is because the number of basis images (generators) for

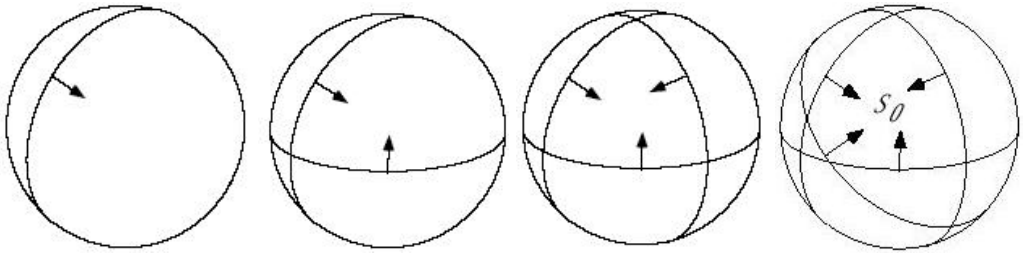

Figure 6: Great Circles corresponding to individual pixels divide the sphere into cells of different shadowing configurations. The arrows indicate the hemisphere of light directions for which the particular pixel is illuminated. The generators (extreme rays) of the cone are given by the images produced by light sources at the intersection of two circles.

an illumination cone is quadratic in the number of distinct surface normals, and for many objects, this number is on the same order as the number of pixels. Both time and space requirements for enumerating all generators can be formidable. For instance, for a typical 200×200 image, there are roughly 1.6 billion generators. Each generator is stored as a 200×200 image, and hence it requires at least 64000 giga-bytes to store all generators. A formidable requirement for just one illumination cone indeed. However, from a face recognition viewpoint, knowing the entire cone is not really necessary. An illumination cone can contain images with unusual appearances taken under some uncommon illumination conditions, such as images with only a few bright pixels. These images are clearly of no significance since they do not contain sufficient information to enable any reasonable recognition result. Instead, what we would like to know is the part of the illumination cone that contains images under common lighting conditions, such as under smooth, ambient illumination.

This idea can be made more precise as follows. While the harmonic subspace $\mathcal{H}$ is a nine-dimensional subspace approximating the illumination cone $\mathcal{C}$, we would like to find a subspace $\mathcal{R}$ (of the same or different dimension), with a *basis* formed by the generators of $\mathcal{C}$, such that $\mathcal{R}$ also approximates $\mathcal{C}$ well. The benefit of replacing $\mathcal{H}$ with $\mathcal{R}$ is that a basis of $\mathcal{R}$ now consists of *real* images taken under *real* lighting conditions. Taking these images as training images, a linear subspace can be immediately computed without recourse to estimating surface normals and albedos and without rendering images.

The discussion above can be formulated as a computational problem [21]. Let $\mathbb{D}$ be a collection of lighting conditions, and we want to determine a subset $\{s_1, \cdots, s_n\}$ of $\mathbb{D}$ such that images $\{I_{s_1}, \cdots, I_{s_n}\}$ taken under $\{s_1, \cdots, s_n\}$ span a subspace $\mathcal{R}$ that approximates $\mathcal{C}$ well. $\mathbb{D}$ can be, for instance, the set of generators of an illumination cone, or a set of points sampled on S^2. An algorithm for computing the subset $\{s_1, \cdots, s_n\}$ is presented in [21][22]. One possible way to solve the problem is to enumerate all possible subspaces "generated" by points in $\mathbb{D}$ and compute how good of a fit it has to the original cone. However, in practice, it is not possible to do so, therefore, [21][22] choose a different solution. Instead, a nested sequence of linear subspaces, $R_0 \subseteq R_1 \subseteq \ldots \subseteq R_i \cdots \subseteq R_9 = R$, with R_i an i-dimensional subspace and $i \geq 0$, is computed. The idea is to make sure that all subspaces are as close to the harmonic subspace as possible. Since $\mathcal{H}$ approximates $\mathcal{C}$ and if R_i is close to $\mathcal{H}$, then it should also approximate $\mathcal{C}$ well. In [21][22], the nested sequence of linear subspaces is computed iteratively by finding $s \in \mathbb{D}$ at each iteration that satisfies:

$$s_i = \arg \max_{s \in \mathbb{D}_{i-1}} \frac{\text{dist}(s, R_{i-1})}{\text{dist}(s, \mathcal{H})}. \tag{21}$$

dist is the usual L^2-distance between a subspace and a vector. Notice that here we are using the same notation to denote the lighting condition in $\mathbb{D}$ as well as the corresponding images. $\mathbb{D}_{i-1}$ denotes the set obtained by deleting i elements from $\mathbb{D}$. The next subspace R_i in the sequence is the subspace spanned by R_{i-1} and s_i.

To actually solve the optimization problem above, we have to know the images under lighting conditions in $\mathbb{D}$. Assuming human faces are Lambertian, this can be accomplished as before by rendering images under novel lighting conditions $s \in \mathbb{D}$ if surface normals and albedos are known. In [21], a collection of 1005 sampled points on S^2 is used to define the domain $\mathbb{D}$ for the optimization

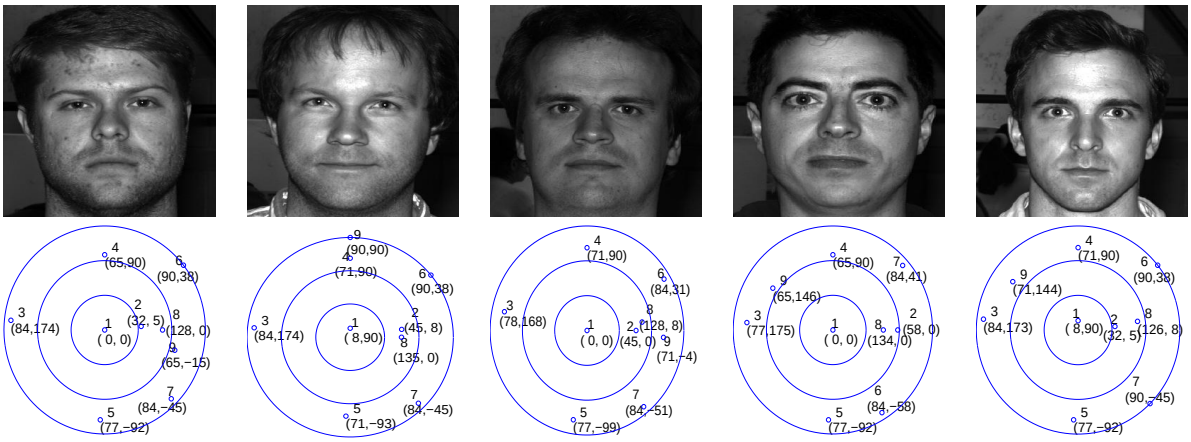

Figure 7: Experiment results for selecting basis images. **Top Row:** Five of the ten faces in the Yale database used in the experiment. **Bottom Row:** The nine lighting directions found by maximizing Equation 21 for the five faces above. The directions are represented in spherical coordinates (ϕ, θ) centered at the face. See [21]

problem posed in Equation 21. They worked with the well-known Yale Face database, which contains faces images as well as 3D models (surface normals and albedos) for ten individuals.

For the five faces shown in Figure 7, the results of computing the 9-dimensional linear subspace R are shown beneath their respective images. Since all lights are sampled from S^2, we use spherical coordinates to denote the light positions. The coordinates frame used in the computation is defined such that the center of the face is located at the origin, and the nose is facing toward the positive z-axis. The x and y axes are parallel to the horizontal and vertical axes of the image plane, respectively. The spherical coordinates are expressed as the pair (ϕ, θ) (in degrees), where ϕ is the elevation angle (angle between the polar axis and the z-axis) with range $0 \leq \phi \leq 180$, and θ is the azimuth angle with range $-180 \leq \theta \leq 180$. It is worthwhile to note that the set of nine lighting directions chosen by the algorithm has a particular type of configuration. First, the first two directions chosen are frontal directions (with small values of ϕ). The first direction chosen, by definition, is always the image that is closest to H and in most cases, it is the direct frontal light given by $\phi = 0$. Second, after the frontal images are chosen, the next five directions are from the sides (with $\phi \approx 90$). By examining the θ values of these directions, we see that these directions spread in a quasi-uniformly manner around the lateral rim. And finally, the last chosen direction seems to be random. It is important to note that it is by no mean clear a priori that our algorithm based on maximizing Equation 21 will favor such type of configurations. Furthermore and most importantly, the resulting configurations across all individuals are very similar.

This similarity strongly suggests that we can extrapolate the results here for all individuals. That is, there may exists a configuration of nine (or fewer) lighting directions such that the subspace spanned by images taken under these lighting conditions approximates $\mathcal{C}$. As we will discuss in the following section, the face recognition algorithm proposed in [21][22] is simply an algorithm for computing one such configuration of lighting directions. For face recognition, this configuration is particularly beneficial because it tells us the kind of training images (and the lighting conditions under which to gather them) that are needed.

0.4 Menagerie

In this section, we discuss several recently-published algorithms [2][8][12][21][31][37] for face recognition under varying illumination. These are all image-based methods, and except [8], the common feature among them is the ability to produce a low-dimensional linear representation that models the illumination effect using only a handful of training images. The justification for such generalization is provided by the discussions in the previous sections.

The six algorithms proposed in [2][8][21][12][31][37] can roughly be categorized into two types, algorithms that explicitly estimate surface normals as well as albedos [2][12][31][37]and algorithms

that do not [8][21]. Knowing surface normals allows one to recover the 3D structure by integration and therefore, it is possible to simulate images of a human face under different illumination conditions. In particular, 3D information allows the modelling of cast shadows by using, for example, ray tracing [12]. The dimensionality of the datum (images) can be reduced, for example, using Principal Component Analysis ([31]). On the reduced space, various different classifiers (such as Support Vector Machine and Nearest Neighbors Classifier) can be brought to bear on these simulated images. The method of harmonic subspaces [2][37] provides one way of utilizing surface normals without explicitly computing the 3D structure. Since the basis images are simply polynomials of the surface normals and albedos, they can be easily computed if the normals and albedos are known. Obviously, a technical part of these algorithms is the recovery surface normals and albedos using a few training images. This can be accomplished either using photometric stereo techniques [12] or employing probabilistic methods using some learned prior distributions of normals and albedos [31][37].

Papers [8] and [21] offer algorithms that do not require normal and albedo information. [8] proposes an algorithm based on their joint probability density distribution (Equation 6). The joint pdf is obtained empirically, and recognition is by calculating maximum likelihood using this pdf. [21] is perhaps the simplest algorithm. The paper shows that a set of training images (as few as five) taken under prescribed lighting conditions is sufficient to yield good recognition results. The key here is to obtain a configuration of lighting conditions such that the (training) images taken under these lighting conditions form a basis of a linear subspace that approximates the illumination cone well.

Georghiades et. al. [12] In this algorithm, surface normals and albedos of the face are recovered using photometric stereo techniques [5][36], and the 3D shape of the face is obtained by integrating the normal vector fields. Once the normals and albedos are known, it is possible to simulate images under new lighting conditions by applying Equation 1 directly. However, to account for the cast shadows, a simple ray tracer is employed to render images with cast shadows. In both cases, the simulated images are all under distant point sources, and they can be interpreted as the generators of the illumination cone, and the process of simulating images can be considered as sampling generators. After sufficiently many images have been sampled, there are two ways to produce appearance models. One can apply Principal Component Analysis (PCA) to produce a low-dimensional linear representation, or one can use the cone generated by the sampled images directly. The difference between subspace and cone models is how the projection of each query image x is computed: in both cases, the projection is defined by minimizing the reconstruction error:

$$\min \|x - (a_1 e_1 + \cdots, a_s e_s)\|^2.$$

In the subspace model, e_i's are the basis vectors, and the coefficients a_i's are real numbers. In the cone case, e_i's are the generators, and the coefficients are subjected to the non-negativity constraint, $a_i \geq 0$. Because of this constraint, determining a_i's becomes a convex programming problem, which, fortunately, can be solved efficiently.

Next, we mention briefly how 3D reconstruction is accomplished in this paper. Using photometric stereo [36], the problem is the following: given a collection of training images $\{I_1, \cdots, I_k\}$, we want to find matrices B and S that minimize the "reconstruction error":

$$\min_{B,S} \|X - BS\|^2 \tag{22}$$

where $X = [I_1, \cdots, I_k]$ is the intensity matrix of k images (in vector form) and S is a $3 \times k$ matrix whose columns s_i are the light source directions scaled by their intensities for all k images. B is a $n \times 3$ matrix whose rows are the normal vectors. Given X, B and S can be estimated using Singular Value Decomposition (SVD) [14]. However, there are three complications. First, a straightforward application of SVD is not robust since minimizing Equation 22 at shadowed pixels (both attached and cast shadows) is incorrect. The solution is to consider entries of X corresponding to shadowed pixels as missing values, and SVD with missing values ([17][29]) is used instead of regular SVD. The second complication arises from the fact the the normal vector field B estimated by SVD is in generally not integrable, i.e., it is not the normal vector field of a smooth surface. However, it is possible to efficiently compute an integrable normal vector field that has minimal L^2-distance to B using the

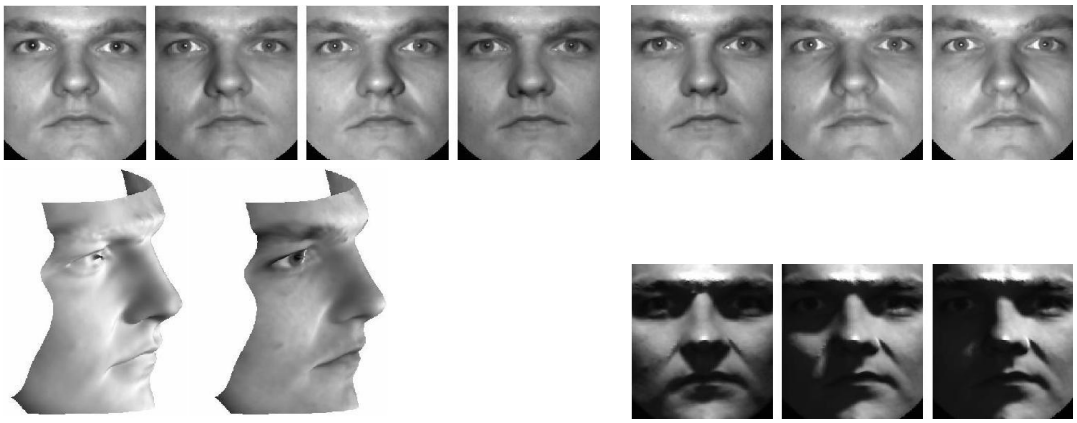

Figure 8: 3D reconstruction of a human face. **Top**: The seven training images. **Bottom, Left**: Reconstruction Results. Left: The surface is rendered with flat shading (constant albedo). Right: rendered using estimated albedos. **Bottom, Right**: Three synthesized images with new lighting conditions. Note the large variations in shading and shadowing as compared to the seven training images above.

Discrete Cosine Transform [10]. The overall strategy to minimize Equation 22 is to minimize B and S separately and iteratively: each time B has been estimated, we find an integral normal vector field that is closest to B in the L^2-sense, and then S is estimated using this integrable field. The third complication arises from the fact that the pair B, S estimated from the factorization $X = BS$ is not unique. In fact, for any non-singular 3×3 matrix G, the product of BG and $G^{-1}S$ is also X. Integrability of BG and B requires that G belongs to a three-dimensional subgroup of $GL(3)$, the Generalized Bas-Relief transformations (a GBR transformation scales the surface and introduces an additive plane) [5]. Therefore, the reconstruction of the surface geometry outlined above is only up to some (unknown) GBR transformation. In [12], symmetries and similarities in faces are exploited to resolve this ambiguity. Some reconstruction results in this paper are shown in Figure 8. A related and more sophisticated reconstruction algorithm using non-Lambertian reflectance functions have been proposed recently [11].

The method proposed in [12] is a generative algorithm in that images under new illumination *and* pose conditions can be simulated. The single-pose recognition algorithm we just discussed can be generalized immediately to multiple poses by associating different poses with different illumination cones. Each query image is then tested against all these illumination cones to determine the recognition result.

Basri & Jacobs [2] The face recognition algorithm proposed by the authors is a straightforward application of their illumination model based on spherical harmonics. Similar to the preceding algorithm, it is also a subspace-based algorithm in that the appearance model for each individual in the database is a nine-dimensional linear subspace spanned by the nine harmonic images. Assuming Lambertian reflectance, this subspace will capture more than 99% of the variance in pixel intensities. Since a harmonic image is simply a product of albedos and a polynomial (with degree less than three) in the components of the normal vectors, the nine basis images can be immediately obtained once the normals and albedos are known. The analytic description of the subspace is the strength of this algorithm, and it enables us to compute the subspace without simulating any images.

Let $B = [b_1, \cdots, b_9]$ be the matrix whose columns are harmonic images (of an individual). The face recognition algorithm is based on computing the L^2 reconstruction error, and for a query image x, it is given by

$$\min_a \|Ba - x\|^2, \tag{23}$$

where a can be any 9×1 vector. Experiment results reported in [2] have shown that the recognition algorithm based on this minimal L^2 reconstruction error has good performance. However, without any constraint on a, it is possible that the illumination condition implied by a is not physically

realizable, i.e. the function $l = a_1 Y_1 + \cdots + a_9 Y_9$ has negative values somewhere on S^2. The constrained version of Equation 23 in this context is slightly harder to formulate. We start with a lighting configuration given by a collection of J point lights represented by the delta function $\delta_{\theta_j \phi_j}$,

$$l = \sum_{j=1}^{J} a_j \delta_{\theta_j \phi_j} = \sum_{j=1}^{J} a_j \sum_{n=0}^{\infty} \sum_{m=-n}^{n} Y_{nm}(\theta_j, \phi_j) Y_{nm}.$$

As before, any physically realizable lighting conditions can be approximated to an arbitrary precision using sufficiently large J and appropriate delta functions. The point, of course, is that a_i in the above equation are all non-negative, and we can rewrite equations Equation 23 above using l. Specifically, we need a matrix H that relates the delta functions $\delta_{\theta_i \phi_j}$ and the spherical harmonics. Let H be a matrix that contains a sampling of the harmonic functions, with its rows contain the transforms of the delta functions. Equation 23 can be re-written as

$$\min_a \|BH^t a - x\| \text{ s.t. } a \geq 0.$$

This gives the constrained version of the linear problem, and it guarantees that the resulting lighting configuration is physically realizable. However, the experimental results reported in [2] do not indicate any visible difference between the performances of the two slightly different algorithms above.

Surface normals and albedos are unexpendable components in the previous two algorithms, and photometric stereo is a commonly used technique for estimating normals and albedos. However, photometric stereo generally requires more than three images (under different lighting conditions) in order to unambiguously estimate the surface normals at every pixel. What is needed is an algorithm that estimates the normals and albedos from as few training images as possible, and Sim and Kanade's algorithm below does that for just one image.

The non-existence of illumination invariant discussed previously has shown that it is impossible to recover the normals and albedos from one image directly. However, as we pointed out earlier, the results presented in Section 2 were derived without any assumption on the geometry and reflectance of the object. It is possible, however, to estimate the normals and albedos reasonably accurately if some useful prior has been given, and this is precisely what the following two algorithms strive to accomplish: given one image of an individual and some learned priors, normals and albedos are estimated based on some maximal likelihood estimates.

Sim & Kanade [31] In this method, the illumination model is the usual Lambertian model augmented with an exterm term e:

$$i(x) = n(x)^t s + e(x, s). \tag{24}$$

Here, as before, the $i(x)$ stands for the intensity at pixel x, $n(x)$ the albedo-scaled normal and s is the direction of some single distance lighting. The extra e term models the effecive ambient illumination and it depends both on x and s. With aligned images, it is assumed that the normals of human faces at pixel x forms a Gaussian distribution with some mean $\mu_n(x)$ and covariance matrix $C_n(x)$. Similarly, $e(x, s)$ is also assumed to form a Gaussian distribution, with mean $\mu_e(x, s)$ and variance $\sigma_e^2(x, s)$. All these parameters can be estimated from a collection of images with known normals and lighting directions. In addition, normals at different pixels are assumed to be independent, and this assumption makes the following MAP procedure much simpler.

Once the distributions for $n(x)$ and $e(x, s)$ have been obtained, we can estimate the normals at each pixel of a given image using Equation 24. Specifically, for an given image, we first estimate the unknown illumination s ([38][39]). This allows $\mu_e(x, s)$ and $\sigma_e^2(x, s)$ to be computed. $n(x)$ can be recovered as a *maximum a posteriori* (MAP) estimate, $n_{\text{MAP}}(x) = \arg\max_{n(x)} \Pr(n(x)|i(x))$, where $i(x)$ is given by Equation 24. Simulated images under new illumination can be rendered using the estimated normals and ambient illumination term e. See Figure 9. Face recognition proceeds exactly as before: for each individual in the database with one training image, we first estimate the normals and the error term e. Images under novel illumination conditions are then simulated, and a linear subspace is computed by applying PCA to the simulated images.

Figure 9: Hallucinated Images (Courtesy of [31]). **Left:** Image rendered using strict Lambertian equation (without e in Equation 24) and one that uses the error term e, where the specular reflection on the left cheek is more accurately rendered. **Right:** Four synthetic images using estimated normals $n(x)$ and $e(x, s)$ (top row) and actual images under the same illumination (bottom row).

Zhang & Samaras [37] Note that using the estimated normals above, we might as well compute the nine-dimensional harmonic subspaces using the normals, and therefore, avoid simulating images. In fact, we can do better by directly estimating the nine harmonic images from just one training image, and an algorithm for doing this appeared in [37]. The starting point is an equation that is similar to Equation 24:

$$i(x) = b(x)^t \alpha + e(x, \alpha) \tag{25}$$

the new $b(x)$ is a 9×1 vector that encodes the pixel values of the nine harmonic images at x and e models the ambient illumination as before. In place of s, we have a 9×1 vector α that represents the nine coefficients in the truncated spherical harmonics expansion of s. We can assume that $b(x)$ form a Gaussian distribution at each pixel x, with some mean $\mu_b(x)$ and covariance matrix $\mathcal{C}_b(x)$. As before, these parameters can be estimated from a collection of training (harmonic) images. Once they have been computed, for any given image, we can estimate $b(x)$ at each pixel.

Chen et. al. [8] Unlike the previous algorithms, this algorithm does not estimate surface normals and albedos, and it requires only one single training image. It is essentially probabilistic, similar to **Zhang & Samaras** and **Sim & Kanade**, in the sense that the algorithm depends critically on a prior distribution. In this case, the distribution is on the angles between image gradients, and it is obtained empirically, instead of analytically. As we discussed in Section 2, the joint density distribution ρ for two image gradients can be used as an illumination insensitive measure. If we treat each pixel independently, the joint probability of observing the images gradients, $\nabla I, \nabla J$, of two images I and J of the same object is

$$P(\nabla I, \nabla J) = \prod_{i \in \text{Pixels}} \rho(\nabla I_i, \nabla J_i) = \prod_{i \in \text{Pixels}} \rho(r_1(i), \phi(i), r_2(i)), \tag{26}$$

where $r_1(i) = |\nabla I(i)|, r_2(i) = |\nabla J(i)|$, and ϕ is the angle between the two gradient vectors. With this probability value, it is then quite straightforward to come up a face recognition algorithm. Given a query image I, we compute $P(\nabla I, \nabla J)$ for every training image J using the empirically determined probability distribution ρ. The one training image having the largest P value is considered to be the likeliest to have come from the same face as the query image I. Therefore, no subspace is involved for this algorithm, and the computation is exceptionally fast and efficient. The obvious drawback is that we need to know how to evaluate (at least empirically) the joint density function ρ, and determining ρ accurately may require great efforts.

Lee et. al. [22] Implementation-wise, this is perhaps the simplest algorithm. In this algorithm, surface normals and albedos are not needed and there is no need to simulate images under novel lighting conditions. The main insight here is to obtain certain configuration of lighting positions such that images taken under these lighting positions can serve as basis vectors for the subspace. This particular configuration, named the "universal configuration" in [21], can be computed for a small number of models (faces) and it can be applied to all faces.

Suppose l face models are available with sufficient information (normals and albedos) that we can simulate these faces under any new lighting condition. Given a set of sampled directions $\mathbb{D} \subset \mathrm{S}^2$,

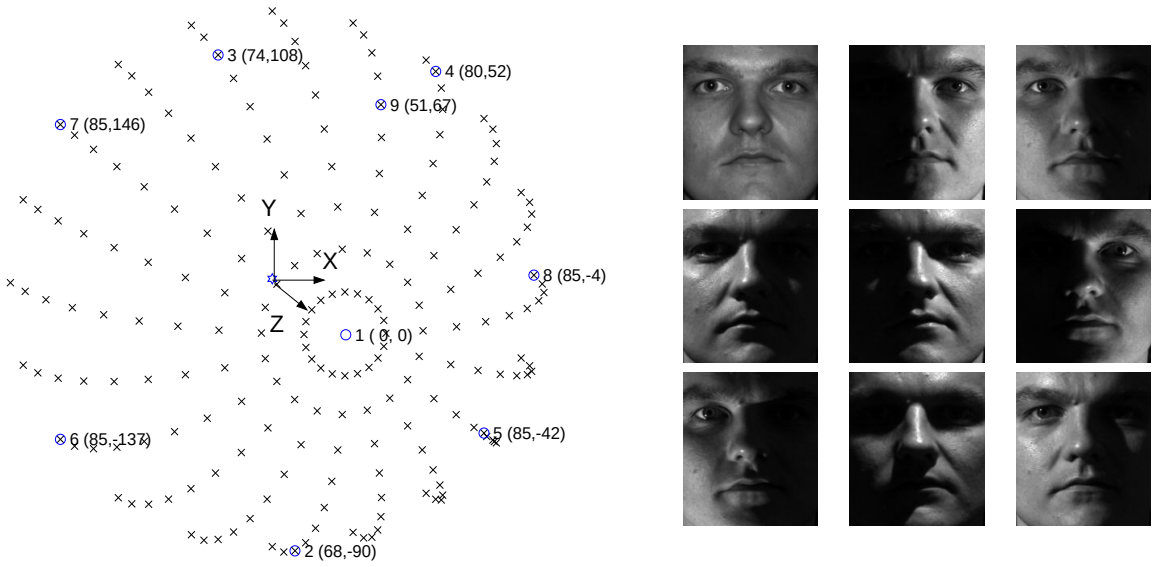

Figure 10: **Left**: The universal configuration of nine light source directions with all 200 sample points plots on a hemisphere. Spherical coordinates (ϕ, θ) (in degrees) are used here, and the nine directions in spherical coordinates are $\{\ (0,0),\ (68,-90),\ (74,108),\ (80,52),\ (85,-42),\ (85,-137),\ (85,146),\ (85,-4),\ (51,67)\}$. **Right**: Nine images of a person illuminated by lights from the universal configuration.

we seek a fixed configuration of nine lighting directions for all l faces such that for each face, on average, the linear space spanned by the images taken under these lighting conditions is a good linear approximation to the illumination cone. To find such a configuration, [21] tries to obtain a nested sequence of linear subspaces, $R_0 \subseteq R_1 \subseteq \ldots \subseteq R_i \cdots \subseteq R_9 = R$, by iteratively maximizing the average of the quotient in Equation 21 over all the available faces:

$$x_i = \arg \max_{x \in \mathbb{D}_{i-1}} \sum_{k=1}^{l} \frac{\text{dist}(x^k, R^k_{i-1})}{\text{dist}(x^k, H^k)}. \tag{27}$$

Since we are computing Equation 21 for all the available face models (indexed by k) simultaneously, for each $x \in \mathbb{D}_{i-1}$, x^k denotes the image of model k taken under a single light source with direction x. $\mathbb{D}_{i-1}$ denotes the set obtained by deleting i elements from Ω. With k indexing the available face models, H^k denotes the harmonic subspace of model k, and R^k_{i-1} represents the linear subspace spanned by the images $\{x^k_1, \cdots, x^k_i\}$ of model k under light source directions $\{x_1, \cdots, x_i\}$. [21] computes a R using a set of 200 uniformly sampled points on the "frontal hemisphere" (the hemisphere in front of the face). The resulting configuration as well as the 200 samples on the hemisphere are plotted in Figure 10.

0.5 Experiments and Results

In this section, we discuss the performances of the face recognition algorithms summarized in the previous section. With one exception ([8]), all algorithms are subspace-based algorithms. Since they are all image-based algorithms, low level imaging processing such as edge and feature detections are unnecessary. Experiment results reported below will demonstrate that these recognition algorithms are quite robust against illumination variation. In addition, because L^2-differences can be quickly computed using a small number of matrix operations, they are efficient and easy to implement as well. However, the algorithms differ from each other in two fundamental ways: by the number of training images they require and by ways the subspaces are computed from the training images.

The algorithms are tested below using two face databases, Yale Face Database B and the PIE (Pose, Illumination and Expression) database from CMU. In the past few years, they have become

Subset 1

Subset 2

Subset 3

Subset 4

Figure 11: Example images of a single individual in frontal pose from the Yale Face Database B showing the variability due to illumination. The images have been divided into four subsets according to the angle the light source direction makes with the camera axis – Subset 1 (up to 12°), Subset 2 (up to 25°), Subset 3 (up to 50°), Subset 4 (up to 77°).

the de facto standards for researchers working on illumination effect and face recognition. Both databases contain many images of different individuals taken under various illumination and viewing (pose) conditions. In the case of CMU PIE database, expression variation is also included. For the experiments below, only the illumination part of the databases will be used. The PIE database (see [30] for more information) contains 1587 images of 69 individuals and 23 different illumination conditions. The original (older) Yale database contains 10 individuals and each individual has 45 different illumination conditions (a sample of Yale database is shown in Figure 11 and see [12] for more details). Later, the number of individual has increased to 38 in the the extended Yale database. The images in the Yale database are grouped into four subsets according to the lighting angle with respect to the camera axis. The first two subsets cover the angular range 0° to 25°, the third subset covers 25° to 50°, and the fourth subset covers 50° to 77°. As the lighting direction moves from the frontal to the lateral positions, shadows, both attached and cast shadows, develop prominently in the resulting images. These heavily-shadowed images (subset four) are the most challenging for face recognition.

0.5.1 Results

Table 1 summarizes the experiment results for all the algorithms (except **Sim & Kanade**) discussed in the previous section. The original Yale face database (10 individuals, 450 images) is used in this experiment. The first five rows contain the results of using "quick-fix" algorithms without significant illumination modelling. The next eight rows display the results of using more sophisticated illumination modelling. The difference in performances between these two categories of algorithms is apparent: while the total error rates for the former category hover above 20%, algorithms in the later category can achieve less than 1% in error rates. Note that different algorithms require different numbers of training images and in evaluating algorithm performances, we have tried to use the same number of training images whenever possible.

Before going further, we briefly describe these six "quick-fix" algorithms [12]. Correlation is a nearest-neighbor classifier in the image space [6] in which all of the images are normalized to have zero mean and unit variance. In this experiment, we take several frontally-illuminated images as training images and calculate the correlations between these normalized training images and each (normalized) query image. "9NN" is a straightforward implementation of the nearest neighbor classifier using nine training images for each individual. The nine training images are images taken under the lighting conditions specified in the universal lighting configuration discussed in the previous section. Therefore, unlike Correlation, the nine training images contain both frontally and laterally illuminated images. Eigenfaces uses PCA to obtain a subspace from training images. One proposed method for handling illumination variation using PCA is to discard the first three most significant principal components, which, in practice, yields better recognition algorithm [3]. The linear subspace method is a simple subspace-based method. The subspace is a three-dimensional subspace built on the x, y, z components of the surface normals. This is a variant of the photometric alignment method proposed in [28] and is related to [16][24]. While this method models the variation in shading when the surface is completely illuminated, it does not model shadowing, neither attached nor cast shadows.

In Table 1, there are two slightly different versions of the harmonic subspace method [2] and the illumination cone method [12]. In "Harmonic Subspace-attached", the nine harmonic images that form the basis of the linear subspace are rendered directly according to the formulas in Equations 12-14. "Harmonic Subspace-cast" uses a simple ray tracer to simulate the harmonic images of a 3D face under harmonic lightings. Similarly, in "Cone-attached" and "Cone-cast" methods, we use images without and with cast shadows to compute the illumination cones, respectively. Harmonic exemplars is the method proposed in [37] and the result here is taken directly from that paper. "Gradient Angle" comes from [8] and finally, "9PL" is the algorithm first proposed in [21] that uses nine training images taken under the nine lighting conditions specified by their universal configuration. In this experiment, the 3D structure of the face is used to render the nine images under these nine lighting conditions (which are not included in the Yale Database).

There are several ways to understand the results in Table 1. First, images taken under frontal illuminations are generally easier to recognized. As expected, the laterally-illuminated images (those from Subsets 3 and 4) are the main challenges. As the first five "quick-fix" algorithms clearly

Table 1: The error rates for various recognition methods on subsets of the Yale Face Database B. Some of the Each entry is taken directly from a published source indicated by citation.

Comparison of Recognition Methods						
Method	Number of Training Images	Estimate Normals	Error Rate (%) vs. Illum.			
			Subset 1&2	Subset 3	Subset 4	Total
Correlation [12]	6-7	No	0.0	23.3	73.6	29.1
Eigenfaces [12]	6-7	No	0.0	25.8	75.7	30.4
Eigenfaces w/o 1st 3 [12]	6-7	No	0.0	19.2	66.4	25.8
9NN [22]	9	No	13.8	54.6	7.0	22.6
Linear subspace [12]	6-7	Yes	0.0	0.0	15.0	4.6
Cones-attached [12]	6-7	Yes	0.0	0.0	8.6	2.7
Harmonic Exemplars [37]	1	Yes	0.0	0.3	3.1	1.0
9PL (simulated images)[22]	9	No	0.0	0.0	2.8	0.87
Harmonic Subspace-attached (no cast shadow)[22]	6-7	Yes	0.0	0.0	3.571	1.1
Harmonic Subspace-cast (with cast shadows)[22]	6-7	Yes	0.0	0.0	2.7	0.85
Gradient Angle [8]	1	No	0.0	0.0	1.4	0.44
Cones-cast [12]	6-7	Yes	0.0	0.0	0.0	0.0
5PL (real images)[22]	5	No	0.0	0.0	0.0	0.0
9PL (real images)[22]	9	No	0.0	0.0	0.0	0.0

demonstrated, it is difficult to robustly recognize these images without any significant illumination modelling. Second, linear subspace models are indeed the right tool to use for modelling illumination. This, of course, is the main result we discussed in Section 3 and here we observe it empirically by comparing the recognition results using "9PL" and "9NN". While they use the same training images and compute the same norm (L^2 norm), the ability of the subspace to correctly extrapolate images under novel illumination conditions is the only explanation for the discrepancy in performances between "9PL" and "9NN". Finally, we see that the difference in performance between "Harmonic images-attached" and "Harmonic images-cast" (and likewise for "Cone-attached" and "Cone-cast") are not significant. This implies, as we mentioned in the introduction, that the degree of non-convexity of human faces is not too severe as to render the effect of cast shadows on human faces unmanageable.

While the on-line recognition processes for algorithms listed in Table 1 are pretty much the same, they differ significantly, however, in their off-line training processes. For algorithms that required surface normals, at least three training images are needed in order to determine the normals and albedos. In this experiment, we require typically six frontally-illuminated images to estimate the surface normals and albedos using photometric stereo techniques. Although "Harmonic Exemplar" can get by with just one training image, it requires the priors on harmonic images that can only be obtained using an off-line training process that typically requires a large number of training images. Same goes for "Gradient Angle" in which a prior on the angles between image gradients has to be estimated empirically. Perhaps, the simplest algorithm among the bunch, both implementation-wise and conceptually, is "9PL". Since there is practically no training involved here, the work is almost minimal: we simply need to obtain images of a person taken under nine specified lighting conditions. Further experiments have also shown that a five-dimensional subspace ("5PL") is already sufficient for robust face recognition. While "9PL" is sufficient for single-view (frontal view) face recognition, without 3D reconstructions, it can not deal with multi-view face recognition such as **Georghiades et. al.**.

Experiments have also been carried out using CMU PIE database. In [22], it has been demon-

strated that using only a five-dimensional subspace for each individual , i.e. five training images per person, the overall recognition error rate of 3.5% can be achieved for the CMU PIE dataset using the algorithm of **Lee et.al.**. In [31], Sim and Kanade have compared the performance of two different algorithms, between the nearest neighbor (NN) classifier, and the classifier based on individual PCA subspaces using their algorithm (as discussed in the previous section). The result reported for NN has recognition error rate of 61% while it is just 5% for the proposed method in [31]. The dimensions of the PCA subspace used in their experiment range from 35 to 45.

0.5.2 Further Dimensionality Reduction

Although subspace-based algorithms have done well in the preceding experiment, they all have used subspaces with dimension greater than or equal to nine. While the numerological fixation on nine has its origin (or justification) in spherical harmonics, it is desirable to have subspaces with still smaller dimension without suffering from significant degradation in recognition performance.

Further dimensionality reduction is particularly straightforward for **Lee et. el. [22]**. Here, the subspace is determined through a nested sequence of linear subspaces with increasing dimension: $R_0 \subseteq R_1 \subseteq \ldots \subseteq R_i \cdots \subseteq R_9 = R$ with R_i, an i-dimensional subspace and $i \geq 0$. All these subspaces R_i can be used for recognition, and surprisingly, the experiments reported in [22] have demonstrated that the five-dimensional subspace R_5 is already sufficient for face recognition under large illumination variation. Figure 12(left) shows that the recognition error rate is negligible when R_i, $i \geq 5$, is used as the subspace. Specifically, they have tested their algorithm (with R_5 as the subspace) on the extended Yale face database (1410 images of 38 individuals). Using real images as training images this time, they have reported an error rate of 0.2%. Considering the lighting distribution specified by R_5 (the first five directions in the universal configuration), this result corroborates well with our discussion in Section 3.1, where the empirical observation was that using 5 ± 2 eigenimages is sufficient to provide a good representation of the images of a human face under variable lighting.

Under spherical harmonics framework, dimension reduction is less straightforward. For example, to define a seven-dimensional subspace R_7, presumably, we can find a basis for R_7 using some linear combinations of the nine harmonic images. [25] has proposed a method of determining such linear combinations. However, the simplest way is to use the first four spherical harmonics (i.e. ignoring spherical harmonics with degrees greater than one). By themselves, these four spherical harmonics have accounted for at least 83% of the reflected energy, and the four corresponding harmonic images encode already the albedos and surface normals. Figure 12(right) displays the Receiver Operating Characteristic (ROC) curves for 9D harmonic subspace method, and 9D and 4D harmonic subspace methods with non-negative light conditions [2]. The experiment in [2] uses a database of faces collected at NEC, Japan, which contains 42 faces with seven different poses and six different lighting. The ROC curves show the fraction of query images for which the correct model (person) is classified among the top k closest models (persons) as k varies from 1 to 40. As expected, the 4D positive lighting method performs less well than the other two methods employing the full 9D subspace. However, it is much faster, and seem to be quite effective under simpler pose and lighting conditions [2].

0.6 Conclusion

Looking back at images in Figure 1 in the introduction, we now have, at our disposal, a number of face recognition algorithms that can comfortably handle these formidable-looking images. Barely more than a decade ago, these images would have been problematic for face recognition algorithms at the time. The new concepts and insights introduced in studying illumination modelling in the past decade has bore many fruits in the form of face recognition algorithms that are robust against illumination variation. In many ways, we are very fortunate because human faces do not have more complicated geometry and reflectance. Coupled with the superposition nature of illumination, they allow us to utilize low-dimensional *linear* appearance models to capture large portion of image variation due to illumination. The linearity makes the algorithms efficient and easy to implement, and the appearance models make the algorithms robust.

While great strides have been made, many problems are still awaiting our formulations and solutions. From the face recognition perspective, there is the important problem of alignment and

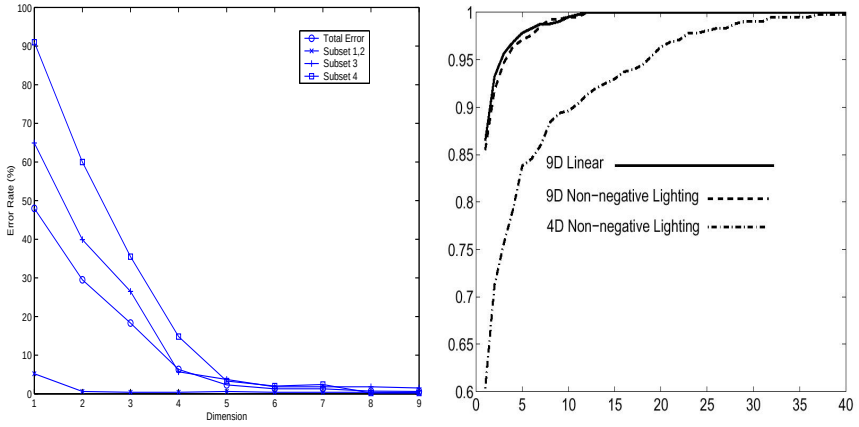

Figure 12: Further Dimensionality Reduction. **Left:** (Courtesy of [22]) The error rates for face recognition using successively smaller linear subspaces R_i. The abscissa represents the dimension of the linear subspace while the ordinate gives the error rate. **Right:** (Courtesy of [2]) ROC curve for using nine-dimensional and four-dimensional harmonic subspaces.

registration, which has been completely ignored in our discussion. How to make these processes robust under illumination variation is a difficult problem and a solution to this problem would have significant impacts on other related research areas such as video face recognition. Because face tracking is an integral and indispensable part of video face recognition, it is also a challenging problem to develop a tracker that is robust against illumination variation. Other important and interesting problems include photo-realistic simulation of human faces as well as face recognition using lighting priors.

Bibliography

[1] V. Arnold. *Ordinary Differential Geometry*. MIT press, 1973.

[2] R. Basri and D. Jacobs. Lambertian reflectance and linear subspaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(6):383–390, 2003.

[3] P. Belhumeur, J. Hespanha, and D. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):711–720, 1997.

[4] P. Belhumeur and D. Kriegman. What is the set of images of an object under all possible lighting conditions. *Int. J. Computer Vision*, 28:245–260, 1998.

[5] P. Belhumeur, D. Kriegman, and A. Yuille. The bas-relief ambiguity. *Int. J. Computer Vision*, 35(1):33–44, 1999.

[6] R. Brunelli and T. Poggio. Face recognition: Features vs. templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(10):1042–1053, 1993.

[7] R. Chellappa, C. Wilson, and S. Sirohey. Human and machine recognition of faces : A survey. *Proc. IEEE*, 83(5):705–740, 1995.

[8] H. Chen, P. Belhumeur, and D. Jacobs. In search of illumination invariants. In *Proc. IEEE Conf. on Comp. Vision and Patt. Recog.*, pages 1–8, 2000.

[9] R. Epstein, P. Hallinan, and A. Yuille. 5+/-2 eigenimages suffice: An empirical investigation of low-dimensional lighting models. In *PBMCV*, 1995.

[10] R. Frankot and R. Chellapa. A method for enforcing integrability in shape from shading algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(4):439–451, 1988.

[11] A. Georghiades. Incorporating the torrance and sparrow model of reflectance in uncalibrated photometric stereo. In *Proc. Int. Conf. on Computer Vision*, pages 816–825, 2003.

[12] A. Georghiades, D. Kriegman, and P. Belhumeur. From few to many: Generative models for recognition under variable pose and illumination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6):643–660, 2001.

[13] A. Goldstein, L. Harmon, and A. Lesk. Identification of human faces. *Proc. IEEE*, 59(5):748–760, 1971.

[14] G. Golub and C. van Loan. *Matrix Computation*. The John Hopkins Univ. Press, 1989.

[15] P. Hallinan. A low-dimensional representation of human faces for arbitrary lighting conditions. In *Proc. IEEE Conf. on Comp. Vision and Patt. Recog.*, pages 995–999, 1994.

[16] P. Hallinan. A deformable model for face recognition under arbitrary lighting conditions. *Ph.D. Thesis, Harvard Univ.*, 1995.

[17] D. Jacobs. Linear fitting with missing data: Applications to structure from motion and characterizing intensity images. In *Proc. IEEE Conf. on Comp. Vision and Patt. Recog.*, 1997.

[18] T. Kanade. *Ph.D Thesis*. Kyoto Univ., 1973.

[19] D. Kriegman and P. Belheumer. What shadows reveal about object structure. *Journal of the Optical Society of America*, pages 1804–1813, 2001.

[20] J. H. Lambert. Photometria sive de mensure de gratibus luminis, colorum umbrae. *Eberhard Klett*, 1760.

[21] K. Lee, J. Ho, and D. Kriegman. Nine points of lights: Acquiring subspaces for face recognition under variable lighting. In *Proc. IEEE Conf. on Comp. Vision and Patt. Recog.*, pages 519–526, 2001.

[22] K.-C. Lee, J. Ho, and D. Kriegman. Acquiring linear subspaces for face recognition under variable lighting. *to appear in IEEE Transactions on Pattern Analysis and Machine Intelligence*.

[23] Y. Moses, Y. Adini, and S. Ullman. Face recognition: The problem of compensating for changes in illumination direction,. In *Proc. European Conf. on Computer Vision*, pages 286–296, 1994.

[24] S. Nayar and H. Murase. Dimensionality of illumination in appearance matching. *Proc. IEEE Conf. on Robotics and Automation*, 1996.

[25] R. Ramamoorthi. Analytic PCA construction for theoretical analysis of lighting variability in images of a lambertian object. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:1322–1333, 2002.

[26] R. Ramamoorthi and P. Hanrahan. An efficient representation for irradiance environment. In *Proceedings of SIGGRAPH*, pages 497–500, 2001.

[27] R. Ramamoorthi and P. Hanrahan. A signal-processing framework for inverse rendering. In *Proceedings of SIGGRAPH*, pages 117–228, 2001.

[28] A. Shashua. On photometric issues in 3D visual recognition form a single image. *Int. J. Computer Vision*, 21:99–122, 1997.

[29] H. Shum, K. Ikeuchi, and R. Reddy. Principal component analysis with mising data and its application to polyhedral object modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(9):854–867, 1995.

[30] T. Sim, S. Baker, and M. Bsat. The cmu pose, illumination and expression (pie) database. In *Proc. IEEE Conf. on Auto. Fascial and Gesture Recog.*, pages 53–58, 2002.

[31] T. Sim and T. Kanade. Combining models and exemplars for face recognition: An illuminating example. In *Proceedings of Workshop on Models versus Exemplars in Computer Vision*, 2001.

[32] L. Sirovitch and M. Kirby. Low-dimensional procedure for the characterization of human faces. *J. of Optimal Soc. Am. A*, 2:519–524, 1987.

[33] W. Strauss. *Partial Differential Equations*. John Wiley & Sons, Inc, 1992.

[34] M. Turk and A. Pentland. Eigenfaces for recognition. *J. of Cognitive Neuroscience*, 3(1):71–96, 1991.

[35] S. Westin, J. Arvo, and K. Torrance. Predicting reflectance functions from complex surfaces. In *Proceedings of SIGGRAPH*, pages 255–264, 1992.

[36] A. Yuille and D. Snow. Shape and albedo from multiple images using integrability. In *Proc. IEEE Conf. on Comp. Vision and Patt. Recog.*, pages 158–164, 1997.

[37] L. Zhang and D. Samaras. Face recognition under variable lighting using harmonic image examplars. In *Proc. IEEE Conf. on Comp. Vision and Patt. Recog.*, volume 1, pages 19–25, 2003.

[38] R. Zhang, P. Tsai, J. Cryer, and M. Shah. Shape from shading: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(8):690–706, 1999.

[39] Q. Zheng and R. Chellappa. Estimation of illuminant direction, albedo and shape from shading. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(7):680–702, 1991.