Surface Dependent Representations for Illumination Insensitive Image Comparison

Margarita Osadchy∗, David W. Jacobs†, Michael Lindenbaum‡

Abstract

We consider the problem of matching images to tell whether they come from the same scene viewed under different lighting conditions. We show that the roughness of a surface determines the type of image comparison method that should be used. Previous work has shown the effectiveness of comparison of the direction of the gradient for rough surfaces. We show analytically that two other widely used methods, normalized correlation and comparison of multiscale oriented filters, can essentially compute the same thing. Then we show that for smooth surfaces, comparison of the output of whitening filters is most effective. This suggests that comparison of general objects should employ a combination of these strategies. We discuss indications that Gabor jets use such a mixed strategy effectively, and propose a new mixed strategy. We validate our results using both synthetic and real images.

1 Introduction

Image comparison is central to computer vision tasks such as tracking and object recognition. Lighting variation significantly affects the appearance of a surface, and makes image comparison difficult. We will show that the right way to solve this problem depends on the nature of the surfaces in the scene. Specifically, methods based on the direction of the gradient work well for rough surfaces that contain discontinuities, or regions of sharp change, either in their shape or albedos. Methods that use whitening filters are better suited to smooth surfaces that do not have these sharp changes. This suggests that for general image comparison we should use methods that combine these two approaches.

∗M. Osadchy is with the Computer Science Department, Technion, Haifa 32000, Israel, e-mail: ritaol@cs.technion.ac.il
†D.W.Jacobs is with the Computer Science Department, University of Maryland, College Park, Maryland, e-mail:djacobs@cs.umd.edu
‡M. Lindenbaum is with the Computer Science Department Technion, Haifa 32000, Israel, e-mail: mic@cs.technion.ac.il
In this paper, we make a number of specific contributions, integrated with prior work, to make this picture clear. First, in Section 3, we discuss smooth objects (an earlier version of this work was presented in [28]). We begin by providing a simplified, analytic model of smooth surfaces. Then we use results from signal detection theory to show that for this model, the optimal method of image comparison involves applying a whitening filter to the images before comparison. We show how to learn a whitening filter for a specific domain, and also point out that standard filters, such as a Laplacian of Gaussian, provide an approximation to the optimal whitening filter. This section therefore provides a new image comparison method that we will experimentally demonstrate to be superior to previous approaches.

Second, in Section 4, we consider rough surfaces (a partial version of these results appears in [29]). Many previous authors ([18, 35, 6, 7, 9]) have proposed using the direction of the gradient as an image comparison method. Chen et al. [6], in particular, provide a statistical analysis that shows that this is the most effective gradient-based image comparison method for rough surfaces. Our primary contribution in this section will be to show a connection between direction of gradient methods and comparison methods based on image normalization. We show that methods based on the direction of the gradient are equivalent to normalized correlation in the limit, when the correlation window is small. Next we show that there is a weaker, but interesting relationship between comparison of gradient directions and comparisons using histogram equalization or mutual information; we describe conditions under which they judge the same image pairs to be identical. We also demonstrate that comparison of the direction of gradient is equivalent to comparison using the normalized outputs of oriented derivative filters (as done, for example, by [34]). These approaches have been increasingly popular in image comparison. This section, then, does not propose a new comparison method, but rather helps to unify many different approaches under the banner of the direction of gradient.

Given that two quite different comparison methods succeed in different domains, it is natural to consider combining them into one, complete method. In Section 5 we begin this endeavor. We present some simple approaches to combining whitening and the direction of the gradient. Perhaps the most elegant is using a jet of Laplacian of Gaussians, which
combines direction of gradient effects due to normalization with whitening effects from the Laplacian filter. We analyze the effectiveness of this method. We also show that Gabor Jets can be understood also as a simple method of combining whitening with the direction of the gradient. These combined methods are all somewhat simple, and there is clearly room for more effective methods to be developed in future work. However, the take-home message of our paper is that combined methods should be used for image comparison.

2 Background

Dealing with the effect of lighting variation has been one of the most significant challenges of image matching. We first review some of these effects and then focus on three groups of approaches: 1) methods that derive sparse, illumination insensitive features, such as edges, prior to matching; 2) methods for normalizing images to reduce the effects of lighting change; 3) methods that filter or transform the image into a dense representation that is less sensitive to lighting variations, such as the direction of the gradient. Finally, we will briefly mention other methods that are less directly relevant to this paper.

In this work we will consider the effects of directional light sources on Lambertian objects (see [30] for some discussion of non-Lambertian objects). Let \( \hat{N} \) denote the surface normal of a point, let \( s \) denote a vector encoding the direction to a visible point light source, scaled by its intensity, and let \( \rho \) denote the albedo of the object (the fraction of light that it reflects). Then we can write the intensity the surface in an image as: \( i = \rho \hat{N} s \). This simple model of reflection rules out cast shadows, light sources such as slide projectors and surfaces such as mirrors, which can have an arbitrarily complex effect.

Some lighting effects are simple. For example, when light sources change their intensity but do not move, all image intensities are simply scaled. For a planar object and distant light sources, changes in lighting conditions also linearly scale the image intensities, because all surface normals have the same angle relative to any light source. For convex objects of uniform albedo, a diffuse light source in which constant intensity light comes from all directions has the effect of adding a constant to all image intensities. In these simple cases, lighting variation has an additive (offset) or multiplicative (gain) effect on the image. Since camera parameters also effect offset and gain, invariance to these effects is often the focus of
lighting insensitive image matching.

Changes in lighting can have a much more complex effect on three-dimensional objects, however. This is because a change in the direction of lighting effects the cosine of the angle between lighting and surface normal differently at every scene point, making some points lighter and some dimmer. This simplest case is that of a scene, such as a roof, containing two planar surfaces. By varying the angle and strength of the light, the two sides of the roof can take on any pair of values. In this case, lighting can change the polarity of the roof edge, that is, it changes which side is brighter. Methods invariant to offset, gain and polarity are invariant at roof edges.

Lighting has a more complex effect on more complex 3D shapes. Chen et al. [6] illustrate this complexity by showing that given one image of a scene, it is impossible to predict anything definite about how this scene will appear under different lighting, making it difficult to compensate for lighting changes in a way that is completely general.

However, when a scene contains discontinuities in shape or albedo, this generally leads to discontinuities in the image. This motivates the use of edges for image matching. Edges due to albedo changes are particularly stable. However, discontinuities in shape, such as at the edges of polyhedron, can produce weak image gradients when the light is at a similar angle to both polyhedron’s faces that form the edge. Smooth 3D surfaces such as faces produce edges that are even more illumination dependent (eg., [37]). For this reason, edge based methods have been very successful, but only for limited classes of objects, or for matching under controlled lighting conditions.

Another method of coping with illumination is to normalize the intensities of images prior to comparison, in hopes of reducing lighting effects. The simplest such approach is to adjust the offset and gain of the image to standard values. For example, one can subtract a constant value from the image, or a window in the image, to give it zero mean, and scale the image to give it unit variance. *Normalized cross-correlation* follows this normalization with correlation, and is a standard way to manage the effects of lighting change (eg., [19]). It is invariant to additive or multiplicative changes within a window, and can therefore handle variations in lighting intensity, in lighting direction for locally planar objects, or some changes in the intensity of diffuse lighting. Brunelli et al. [5], for example, use normalized correlation
for face recognition, and Nayar and Bolle\cite{27} normalize using ratios for lighting invariant recognition. However, it has not been claimed that for 3D scenes normalized correlation should be invariant to complex changes in lighting, such as changes in the direction of a light source, or the addition of new sources.

Another approach related to normalization is to transform image intensities to make the image histogram constant (histogram equalization) or to give it a specific profile (histogram specification). These methods are described in standard texts (eg., \cite{15}). Phillips and Vardi\cite{32} describes the use of histogram specification based on prior knowledge of faces, for use in face recognition. Kittler et al.\cite{23} compare a number of normalization methods, including histogram equalization, for face recognition. Like normalized correlation, histogram normalization methods are clearly invariant to lighting changes that scale or add a constant to intensities. In fact they are invariant to any monotonic change in intensities throughout the image. They are also not invariant to complex lighting changes.

Many approaches match images using the output of multiscale oriented filters instead of raw pixel intensities (eg., \cite{25, 34, 16, 24, 36}). By using filters that integrate to zero, such as difference of Gaussians or Gabor filters, such approaches become invariant to additive changes in the image. Normalizing the total magnitude of all filter responses produces invariance to multiplicative image changes. In addition to their other advantages, this invariance to offset and gain often motivates the use of multiscale oriented filters (eg., \cite{41, 42, 8, 4, 21}).

A third approach to handling lighting is to derive a representation of images that is insensitive to lighting variation. Specifically, a number of authors have proposed the direction of the image gradient for this purpose (\cite{2, 3, 18, 35, 10, 6, 7, 9}). In some cases, this has been motivated by the fact that the direction of gradient, too, is invariant to changes in offset and gain in the image (eg., \cite{7, 9}) and indeed to any monotonic change in image intensity (\cite{35}). Chen et al.\cite{6} provide a statistical analysis of the behavior of the direction of gradient, in terms of scene structure. They show that the direction of gradient is invariant to lighting changes at scene discontinuities. Moreover, it is also relatively insensitive to lighting changes for a surface in which the ratio of one principal curvature to the other is high. The most extreme example of such a surface is a cylinder, in which the direction of gradient is invariant to lighting.
Figure 1: A cylinder illuminated from the left (left) and from the right (middle). On the right we plot a cross-section of the intensities of the two images, after normalization (left, solid line; middle, dashed line). Note that normalization does not mitigate the effects of lighting variation. Note also that the direction of the gradient is invariant (horizontal in both images) although its polarity may change with lighting.

There are many possible ways to compare images using the direction of the gradient. Perhaps the simplest is to compute the sum of squares of the differences between two gradient direction images. The difference between the direction of gradient in two images can range from 0 to $\pi$, or, if we compare the images in a manner insensitive to the polarity of the gradient, it ranges from 0 to $\frac{\pi}{2}$. We will call the polarity sensitive comparison 'DIRP', and the polarity invariant comparison 'DIR', for short. Chen et al. also suggest a somewhat superior method, relying on the distribution of direction differences using the statistics of image pairs that come from the same scene, under different lighting conditions.

To illustrate the potential differences between the direction of the gradient and normalization methods, consider the images produced by a uniform albedo, Lambertian cylinder, illuminated by a point light source (see Figure 1). A circular cross-section of the cylinder will produce intensities that look like a cosine function (but are clipped at zero so that they do not become negative). The phase of the cosine varies so that its peak is at the surface normal that points directly towards the light. Even if we normalize two of these intensity patterns from two images, their different phases can cause them to be largely uncorrelated. However, because a cylinder has zero curvature in one direction, the nonpolar direction of the gradient of its image is always the same, always in the direction of the principal curvature of the cylinder. Therefore, for objects such as cylinders, normalization seems to have little ability to compensate for lighting variations, while the direction of the gradient can be completely insensitive to them. We will show, though, that when image normalization is local, this apparent difference between the power of these methods disappears.
Finally, we note that there has been much work on lighting insensitive object recognition that uses cues outside the scope of this paper, such as 3D knowledge of the scene (e.g., [38, 14, 1]), color (e.g., [39, 13]) and multiple images (e.g., [26, 43]).

3 Image Comparison of Smooth Surfaces

While good methods exist for rough surfaces, previous research has not focused on smooth surfaces. It is difficult to compare images of smooth surfaces, because they do not produce edges or gradients whose directions are insensitive to lighting variation.

To motivate our approach, we will first consider some of the difficulties that arise when we simply compare images directly using sum-of-squared-differences (SSD). Let $I_d = I_1 - I_2$ be the difference image between two images, $I_1$ and $I_2$. Comparing $I_1$ and $I_2$ using SSD when the pixels of $I_d$ are independent, identically distributed Gaussian random variables, is equivalent to making a maximum likelihood decision and may be considered as theoretically optimal.

We will analyze the true nature of $I_d$ by considering a Lambertian surface illuminated by distant point sources. Neglecting shadows, we can model the images of pixel $j$ as: $I_{1,j} = \rho_j \hat{N}_j s_1$ and $I_{2,j} = \rho_j \hat{N}_j s_2$, where $\hat{N}_j$ is a surface normal, $\rho_j$ is its albedo, and $s_1$, $s_2$ are light sources in two images. The difference image then is an image itself, associated with the same object geometry and a difference lighting:

$$I_{d,j} = \rho_j \hat{N}_j s_1 - \rho_j \hat{N}_j s_2 = \rho_j \hat{N}_j (s_1 - s_2)$$

In smooth objects, nearby albedos and surface normals are highly correlated. This leads to dependencies in $I_1, I_2$ and $I_d$. These dependencies imply that comparing two images using SSD is problematic due to two related but different reasons:

1. The gray levels in nearby pixels of the difference image are correlated. Consequently, a comparison based on SSD (that implicitly treat neighboring pixels as independent), is not statistically valid.

2. Moreover, correlations between nearby pixels of the same image imply that even images of different objects can be relatively similar; see section 3.4 and e.g., [22] for a discussion.
Fortunately, both problems can be handled in the same way. The statistical model of the dependencies between neighboring parts of smooth shapes, models dependencies in their difference images as well. We can then design whitening operators to lessen the dependencies in these images. Whitening is a common technique in signal processing and statistics that relies on known second order statistical characterization of the noise. It is the optimal approach when the difference between images of the same object consists of colored (non-independent) Gaussian noise [40]. We show that for a simple model of smooth surfaces, this is a good characterization. Furthermore, we also show that whitening makes the distance between images associated with different objects, much larger.

3.1 The Whitening Technique

First we describe some necessary background on whitening. Let \( n \) represent the intensity of pixels in the difference image, written as a vector. We will shortly motivate a model of \( n \) as Gaussian colored noise, in which different pixels of the difference image are correlated. This implies that \( n \) is fully characterized by its first and second order statistics. In particular, a whitening filter may be designed using the covariance matrix. Let \( C = E[nn^T] \) be the covariance matrix characterizing the distribution of \( n \) (\( E \) denotes expected value). Let \( e_i \) denote the \( i \)'th eigenvector of \( C \), and let \( \lambda_i \) denote the corresponding eigenvalue. Let \( W \) be a matrix in which the \( i \)'th row is the \( i \)'th scaled eigenvector of \( C \), \( \frac{1}{\sqrt{\lambda_i}} e_i \). Then, the components of \( y = Wn \) are independent, as implied from their Gaussianity and their covariance:

\[
E[yy^T] = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_m).
\]

That is, multiplication by the matrix \( W \) “whitens” the vector \( n \).

The whitening procedure described above is computationally infeasible for large signals \( n \) but is actually not needed if the signals’ dependency is local and stationary [20]. In such a case it is common to model the signal as an Autoregressive (AR) process. A sequence \( x_n \) is called an AR process of order \( p \) if it can be generated as the output of the recursive causal linear system

\[
x_n = \sum_{k=1}^{p} a_kx_{n-k} + \varepsilon_n, \forall n
\]
where $\varepsilon_n$ is white noise, and the sum $\bar{x}_n = \sum_{k=1}^{p} a_k x_{n-k}$, is the best linear mean squared (MS) predictor of $x_n$ based on the previous $p$ samples. Given a random sequence (with possible dependencies), an AR model can be fitted using SVD to estimate the overdetermined parameters $a_k$ that minimize the empirical MS prediction error $\sum_n (x_n - \bar{x}_n)^2$. For Gaussian signals the prediction error sequence: $\varepsilon_n = x_n - \bar{x}_n$ is white, implying that a simple convolution with the filter $W' = (1, -a_1, \ldots, -a_p)$ is a whitening process for $x_n$.

### 3.2 A Covariance Model of Smooth Surfaces

To whiten a surface’s images, we must understand its covariance structure. We will do this for a simple model of a smooth surface. We model these surfaces as locally *approximately planar*, with surface normals that make small random perturbations about a common direction (without loss of generality the $z$ axis). Such a surface will be smooth when nearby surface normals are highly correlated. This is a generalization of the common facet model which relies on a strictly planar local characterization [17].

**A covariance model of an approximately planar Lambertian Surface**: We consider a simplified, 1D, variant, of an approximately planar surface. In this case, the “surface” is described by a function $z = f(x)$. The normals at every point $x$ are random (but not independent!). Each of them is specified by a single parameter $\theta$, which is its angle relative to the $z$ axis (Figure 2). Quantitatively we characterize the function $\theta(x)$ as a wide sense (w.s.) stationary Gaussian random process [31]. That is, we assume that the expected value at every point is constant $\mu_\theta = 0$, that the variance $C_\theta(x, x) = \sigma_\theta^2$ is constant as well, and that the auto-correlation $C_\theta(x_1, x_2)$ depends only on the spatial distance $|x_1 - x_2|$ between the two points. It is convenient to express the covariance using the variance $\sigma_\theta$ and the correlation coefficient $r(\cdot)$, $C_\theta(x_1, x_2) = r(|x_1 - x_2|)\sigma_\theta^2$. We also assume that the surface is Lambertian, and that its albedo $\rho$, is constant, at least locally.

**Proposition 1**: Let $I(x)$ be the image associated with the stationary surface, specified above, and with a distant light source illuminating the surface at angle $\phi$, with no shadows. Then $I(x)$ is a random w.s. stationary process. Its expected value, variance and auto-
Figure 2: The roughly planar (random) surface is specified (in 2D approximation) by the angle \( \theta(x) \) that the normal makes with the \( z \) direction.

correlation are:

\[
E[I(x)] = \rho \cos \phi e^{-\sigma_\theta^2/2}
\]

\[
\sigma_i^2 = \frac{1}{2} \rho^2 (\sin^2 \phi (1 - e^{-2\sigma_\theta^2}) + \cos^2 \phi (1 - e^{-\sigma_\theta^2})^2)
\]

\[
C_I(x_1, x_2) = \frac{1}{2} \rho^2 (\sin^2 \phi e^{-\sigma_\theta^2} (e^{r\sigma_\theta^2} - e^{-r\sigma_\theta^2}) + \cos^2 \phi (e^{-\sigma_\theta^2} (e^{r\sigma_\theta^2} + e^{-r\sigma_\theta^2}) - 2e^{-\sigma_\theta^2}))
\]

where \( x_1, x_2 \) are two points for which the correlation coefficient of the tangent direction is \( r = r(|x_1 - x_2|) \). (For proof see Appendix B.)

As expected, for \( r = 1 \) (\( x_1 \) and \( x_2 \) are the same pixel), the covariance reduces to the variance expression, and for \( r = 0 \) (they are distant pixels), the covariance becomes 0. For rougher surfaces (larger \( \sigma_\theta^2 \) and correlation decreasing faster in \( |x_1 - x_2| \)) the correlation is lower for the same distance \( |x_1 - x_2| \). For the (impossible) white surface (independent normals, \( r = 0 \) for every distance \( |x_1 - x_2| \)), the image is white as well.

The covariance in eq. 3 may be written as

\[
C_I(x_1, x_2) = \sin^2 \phi f_1(r(|x_1 - x_2|)) + \cos^2 \phi f_2(r(|x_1 - x_2|))
\]

where

\[
f_1(r(|x_1 - x_2|)) = \frac{1}{2} \rho^2 e^{-\sigma_\theta^2} (e^{r\sigma_\theta^2} - e^{-r\sigma_\theta^2})
\]

\[
f_2(r(|x_1 - x_2|)) = \frac{1}{2} \rho^2 (e^{-\sigma_\theta^2} (e^{r\sigma_\theta^2} + e^{-r\sigma_\theta^2}) - 2e^{-\sigma_\theta^2})
\]

The covariance in eq. 4 is not a constant and it varies with \( \phi \), which is the angle between the light and the predominant surface normal in a planar patch of the object. Note, however, that \( f_1() / f_2() \approx (2 + \sigma_\theta^2) / \sigma_\theta^2 \), implying that for say, \( \sigma_\theta = 0.1 \) (a smooth planar patch), and
Figure 3: Covariance estimates for different patches of a real objects: left – non-normalized covariances differ by multiplicative factors; right – covariances which are normalized by the variance are almost the same for all angles.

for $\phi > 12$ degrees, the first term in eq. 4 is ten times larger than the second term and thus dominant.

Therefore, we can conclude:

**Proposition 2:** The second order statistical behavior of a smooth, approximately planar, Lambertian surface, illuminated by a single source, is characterized by an autocorrelation function which, for nearly every illumination, is approximately invariant to the illumination direction up to a multiplicative factor.

To confirm the multiplicative behavior of the covariance model described above, we took a high resolution image of a real, approximately Lambertian sphere, illuminated by a point source. We divided the image into $50 \times 50$ pixel patches, and calculated covariance in every patch using $5 \times 5$ neighborhoods. Figure 3 shows the estimated covariance as a function of the distance. Each curve represents a different patch. The plots confirm that covariance in different patches differs by a multiplicative factor as was claimed in Proposition 2.

**A covariance model of a smooth surface:** We assume that smooth surface locally behaves as an approximately planar one. When we move across the smooth surface the average surface normal changes, but the light source direction stays the same. For a Lambertian surface this is the same as keeping the surface normals constant, but changing the light source direction. According to Proposition 2 this doesn’t change the structure of the autocorrelation function, but only changes the scale factor. This still implies that the autocorrelation is not stationary, but its special structure allows us to compute a single whitening filter directly by fitting a parametric Autoregressive (AR) model, without explicitly estimating covariance.
3.3 Whitening the difference image using AR models

To decorrelate the difference image we propose to model it as a 2D AR process. Then we can compare images by measuring the magnitude of their decorrelated differences. We have adopted a 2D “causal” model described in [20], where a gray level $x_n$ is predicted from the previous gray levels in a $p \times p$ neighborhood in a column by column scan. (A simple generalization of the 1D neighborhood in eq. 2.) Using a non-causal neighborhood leads to a difference image with a smaller magnitude, but the prediction error sequence is not white [20]. The whitening filter coefficients are set by minimizing the empirical MS prediction error $\sum_n (x_n - \bar{x}_n)^2$ using SVD. The filter size may be optimized as well.

Note that scaling all the grey levels by the same factor would give a correlation function which is the same up to a multiplicative constant. This is essentially what happens when the angle between the average normal and the illumination direction changes. Fortunately, this does not change either the AR coefficients, or the resulting whitening filter, implying that it can be space invariant.

The whitening filter depends on the image statistics. Intuitively, for smoother images the correlation is larger and decorrelating it requires a wider filter. For images that are not so smooth the decorrelation may be done over a small range, and the filter looks very much like the Laplacian, which is also known to have some whitening effect. Therefore, for rougher images, we do not expect to perform better than an alternative procedure using the Laplacian. As we shall see later, for smooth objects the performance difference is significant.

3.4 Whitening Images from Different Objects

One reason that it’s difficult to distinguish smooth objects is that their grey levels are spatially correlated. This correlation implies that if images of two different smooth objects are similar at a single pixel, they are likely to be similar in a substantial neighborhood about that pixel. For this reason, it is easier for different smooth objects to produce similar images than for different rough objects to look the same. The second advantage of whitening is that it makes images of smooth objects less correlated and hence more distinctive.
Figure 4: Distribution of correlations between images of different objects before (dark on the right) and after (light on the left) whitening.

We start by illustrating the type of change caused by whitening. Let $S$ denote a 3D surface, with normals $\tilde{N}_{i,j}$ and albedos $\rho_{i,j}$. Denote $N_{i,j} \equiv \tilde{N}_{i,j}\rho_{i,j}$. Let $I_{1,i,j} = N_{i,j}^T s_1$ be one of its images, associated with illumination vector $s_1$. Let $L$ be a whitening filter, represented discretely as a matrix with elements $L_{k,l}$. Without loss of generality we assume that $L$ is square and $-n \leq k, l \leq n$. Applying this filter to the image $I_1$ we get the output image $I_1$:

$$I_{1,i,j} = \sum_{k=-n}^{n} \sum_{l=-n}^{n} L_{k,l} I_{1,i-k,j-l}$$

Define now a new surface, $S$, such that its scaled surface normals are:

$$N_{i,j} = \sum_{k=-n}^{n} \sum_{l=-n}^{n} L_{k,l} N_{i-k,j-l}$$

Intuitively, $S$ can be thought of as the surface filtered by $L$. According to our model, while the original normals are highly correlated, the whitened normals will be white noise, with randomized directions and sizes. The randomization of sizes is analogous to changing the smooth surface by splattering it with gray paint in random locations, which makes the surface visually distinctive from other surfaces. Of course, whitening does not add differences to signals, it makes explicit the differences that are already there.

Two white noise images are not necessarily uncorrelated. A white noise image, for example, is fully correlated with itself. The correlation of a white noise image with any nonzero translation of it is zero, however. Therefore, it is especially unlikely that two unrelated objects will be highly correlated after whitening. We have tested this observation empirically on images of smooth real objects (see Section 6.1.2 for the description of the database). We took 91 pairs of images (normalized to unit length) associated with different objects and same
illumination and computed their inner products (correlations) before and after whitening. Figure 4 shows the resulting distributions. As expected, the images are highly correlated before whitening and are much less correlated after whitening.\footnote{Some decorrelation is done by simply removing the mean, which is a part of the whitening process. However whitening further reduces the correlation.} Decorrelating the images of unrelated objects is important because from communication theory we know that discriminating between correlated models is difficult and for best performance, the correlation coefficient between any pair of models should be as low as possible\cite{40}.

### 3.5 Approximate Whitening Filters

We have shown how to construct a filter that optimally removes correlations from a difference image. Such whitening filters should be learned for a specific domain. A standard filter, such as a Laplacian of Gaussian, may serve as an approximation to the optimal whitening filter; see, e.g. \cite{33}. Such an approximation is suboptimal, but is more general in the sense that it doesn’t require prior knowledge of the domain. Our experiments show (Section 6.1.3) that Laplacian of Gaussian performance is close to that of a domain specific whitening filter. On very smooth objects the difference is apparent, but in general a Laplacian of Gaussian is a good approximation to the optimal whitening filter.

### 4 Image Comparison of Rough Surfaces

In this section we focus on methods of comparing images of rough surfaces. By rough we refer to objects associated with images containing rapid change in grey levels or discontinuities, and also to surfaces in which ratio of one principle curvature to the other is high. The task of comparing images of rough surfaces is significantly easier than comparing smooth images, largely because there is an effective quasi-invariant: the direction of the gradient. We observe that several other effective approaches to image comparison are based on some kind of normalization. In this section we will show that these methods are closely related to the direction of gradient. We will consider three methods based on normalization: 1) locally normalized correlation, 2) histogram equalization and 3) jets of linear filters that integrate to zero (specifically, difference of Gaussians \cite{24} and odd component of Gabor Jets, which are...
both widely used for image comparison).

4.1 Normalized Correlation

Proposition 3: Consider a small window of an image in which intensities can be approximated as a linear function of location coordinates. Under the above conditions, normalized correlation computes the cosine of the difference in the direction of the gradient.

Proof: Normalized correlation starts by subtracting the mean from each window. Assume \( I_1 \) and \( I_2 \) are zero mean windows that satisfy the above conditions. Without loss of generality let \( I_1 = ax, I_2 = bx + cy \). Then, the corresponding gradients are \( a\hat{x} \) and the \( b\hat{x} + c\hat{y} \), and the angle between them, \( \theta \), satisfies

\[
\cos(\theta) = \frac{ab}{a\sqrt{b^2 + c^2}} = \frac{b}{\sqrt{b^2 + c^2}}
\]

(5)

The correlation between \( I_1 \) and \( I_2 \) is \(^2\):

\[
\int_{-1}^{1} \int_{-1}^{1} (abx^2 + acxy)dx\,dy = \frac{4}{3}ab.
\]

The normalization factor, corresponding to \( I_1 \) and \( I_2 \), are

\[
\int_{-1}^{1} \int_{-1}^{1} a^2x^2dx\,dy = \frac{4}{3}a^2 \quad \int_{-1}^{1} \int_{-1}^{1} (b^2x^2 + c^2y^2 + 2bcxy)dx\,dy = \frac{4}{3}(b^2 + c^2)
\]

Then the normalized correlation is

\[
\frac{4ab}{3\sqrt{\frac{3}{4a^2} \sqrt{\frac{3}{4(b^2 + c^2)}}}} = \frac{b}{\sqrt{b^2 + c^2}}
\]

which is the same as eq.5 \( \Box \)

This demonstrates that with small windows, normalized correlation compares images by summing the cosine of the difference in the direction of the gradient at corresponding image points. This is similar to DIRP since like the square of the difference in angle, the cosine function is also monotonic in this difference, and changes more rapidly as the difference increases (up to a difference of \( \frac{\pi}{2} \)). We note that normalized correlation is not invariant to the polarity of the gradient direction. However, it is simple to change this, by taking the

\(^2\)The size of the integration interval is chosen arbitrarily, because its magnitude will be cancelled by normalization.
absolute value of the normalized correlation computed for each window. In this case, we compute the cosine of the difference in angle, allowing this difference to range from 0 to $\frac{\pi}{2}$. Normalized correlation is, of course, an increasing function of the match quality while DIR is a decreasing function. Therefore, in our experiments we use a simple modification: $1 - (\text{absolute value of the normalized correlation})$, as a distance function. We will denote this comparison method, 'NC'.

### 4.2 Histogram Equalization and Mutual Information

We will now show that there is an interesting connection between comparisons based on the direction of the gradient and those based on histogram equalization (HE) or mutual information (MI). Specifically, we will show that if two images are judged to be identical after histogram equalization, they will have gradients with identical directions and polarity. And, given one image, a second image that has maximum mutual information with the first image will also have identical gradient directions. The converse is not necessarily true, however, except in the case of simple images.

First, consider histogram equalization that consists of a monotonic transformation of pixel intensities so that the new image has a flat histogram. To obtain lighting insensitivity, images are often compared after HE. We note that histogram equalization does not change the shape of isoluminant contours in the image (these are contours along which the intensity is constant). So two images are identical after histogram equalization only if the have identical isoluminant contours. The direction of the gradient is always orthogonal to the isoluminant contours, implying that the gradient directions are identical at all locations. HE also does not alter the polarity of a gradient, because it transforms intensities monotonically. Therefore, whenever two images are identical after HE, the original images will be judged identical by DIRP.

A similar relationship exists between mutual information (MI) and DIR. Given an image A, a new image B will maximize mutual information with A whenever any set of pixels that have identical intensities in A all have a single (possibly different) intensity in B. Therefore B has maximum mutual information with A only if it has identical isoluminant contours, and therefore identical gradient directions with A.
The converse relationships need not hold. Even if two images have identical gradient directions and polarities, they may be different after histogram equalization. For example, one image may have two isoluminant contours that have identical intensities, while in the other image the same isoluminant contours have different intensities. In this case, the isoluminant contours will have different intensities before and after HE, but they will still give rise to identical gradient directions. Similar comments apply to mutual information.

It is possible to show that certain simple pairs of image patches with identical gradient directions and polarity will also be maximally similar according to HE and MI. This will be true when there exists a curve through each image patch that intersects every isoluminant contour, so that along the curve intensities vary monotonically. In this case, if gradient directions are the same for each patch, the isoluminant contours will have the same shape. If gradient polarity is the same, then the intensities of the isoluminant contours will have the same monotonic ordering in each image. This means the image patches will be the same after histogram equalization, and that one of the patches will maximize mutual information with respect to the other. Figure 4.2 illustrates this condition.

Our main goal in this section, however, is not to show a true equivalence between gradient direction and histogram equalization or mutual information. Rather, it is to provide some intuition about the connection between these methods. Intuitively, we can see that all three methods, in different ways, measure the similarity of the isoluminant contours in the image. The chief difference between them is that HE and MI are more global methods that will be influenced more by non-uniform changes in lighting. For example, consider an object that in one image is brightly lit from the left and dimly lit from the right, and in a second image is dimly lit from the left, and brightly lit from the right. Gradient direction ignores the magnitude of the gradient, and so it is not directly influenced by the overall brightness of an image region. HE and MI are more influenced by such changes, which can effect the relative brightness of distant portions of the image.

4.3 Jets of Derivatives of Gaussians

Many authors have remarked that jets of multi-scale, oriented filters provide some insensitivity to lighting variation. We now analyze a simple version of these approaches using a jet of
Figure 5: We show four sets of isoluminant contours. Each contour is labeled with an intensity. The left two images contain isoluminant contours that produce identical image gradients directions and polarities, but the images will not be identical after histogram equalization. (Note that the gradient direction is not defined for the dashed line.) Each of the two images on the right illustrates a set of isoluminant contours that will be invariant after histogram equalization under any change of intensity that does not alter image gradient direction or polarity.

Oriented difference of Gaussian filters at one scale. We show that comparison of the output of these filters effectively computes the cosine of the difference in the direction of gradient of the two images. We call this method, which is sensitive to polarity, ‘DOGP’. With a slight variation, it computes a similar quantity that is invariant to polarity, which we call ‘DOG’.

Let $D_\theta(\vec{x})$ denote the result at position $\vec{x}$ of convolving an image, $I$, with a difference of Gaussian filter oriented in the direction $\theta$, where $\theta$ indicates the angle relative to the $x$ axis. $D_\theta(\vec{x})$ is the directional derivative of $G \ast I$ in the direction $\theta$, where $G$ denotes a Gaussian filter, and $\ast$ denotes convolution. A jet is a vector of the output of these filters at $\vec{x}$: $D(\vec{x}) = (D_{\theta_1}(\vec{x}), D_{\theta_2}(\vec{x}), ... D_{\theta_k}(\vec{x}))$. A typical number of oriented filters is eight, eg., $k = 8$ with equal spacing. We will also consider a common, polarity insensitive variation in which the absolute value of the jet is used: $D_a = (|D_{\theta_1}(\vec{x})|, |D_{\theta_2}(\vec{x})|, ... |D_{\theta_k}(\vec{x})|)$. The resulting jets are normalized prior to comparison. The simplest comparison is correlation, in which case we compute $\frac{D_1 \cdot D_2}{||D_1|| ||D_2||}$.

Suppose that the direction of the gradient at $\vec{x}$ is along the $x$ axis (this will be without loss of generality when our analysis moves to the continuous domain), and the magnitude of the gradient is $M_x$. Then: $D_\theta(\vec{x}) = M_x \cos(\theta)$ and we have:

$$D(\vec{x}) = M_x (\cos(0), \cos\left(\frac{2\pi}{k}\right), ... \cos\left(\frac{2(k-1)\pi}{k}\right))$$

That is, $D(\vec{x})$ is a vector that discretely samples the cosine function, scaled by $M_x$. If we compute difference of Gaussians at a point, $\vec{y}$, in another image, at which the direction of
Table 1: Coefficients obtained by expanding each function in terms of Legendre polynomials.

<table>
<thead>
<tr>
<th>Coefficient</th>
<th>Function</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>[\frac{n}{\pi}[(\frac{\pi}{2} - \alpha) \cos(\alpha) + \sin(\alpha)]]</td>
<td>1.146</td>
<td>-.179</td>
<td>.007</td>
<td>.020</td>
<td>-.002</td>
<td>-.0003</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>[\frac{n}{\pi} \cos(2\alpha) + \frac{n}{\pi}]</td>
<td>1.157</td>
<td>-.180</td>
<td>0</td>
<td>.022</td>
<td>0</td>
<td>-.0007</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

the gradient is \(\alpha\) and its magnitude is \(M_y\) we have:

\[
D(\vec{y}) = M_y(\cos(-\alpha), \cos\left(\frac{2\pi}{k} - \alpha\right), \ldots \cos\left(\frac{2(k - 1)\pi}{k} - \alpha\right))
\]

DOGP compares jets by computing:

\[
\frac{D(\vec{x}) \cdot D(\vec{y})}{\|D(\vec{x})\| \|D(\vec{y})\|}
\]

To analyze this, it is useful to approximate the discretely sampled cosine with a continuous function. So we take:

\[
\|D(\vec{x})\| \approx M_x \sqrt{\int_0^{2\pi} (\cos(\theta))^2 d\theta} = M_x \sqrt{\pi}
\]

Similarly, \(\|D(\vec{y})\| \approx M_y \sqrt{\pi}\) Therefore:

\[
\frac{D(\vec{x}) \cdot D(\vec{y})}{\|D(\vec{x})\| \|D(\vec{y})\|} \approx \frac{1}{\pi} \int_0^{2\pi} \cos(\theta) \cos(\theta - \alpha) d\theta = \cos(\alpha)
\]

This is exactly the same comparison measure that normalized correlation performs when a small window is used.

Next, we consider what happens when we take the absolute values of filter outputs. Since a difference of Gaussian oriented in the direction \(\theta\) produces a result with the same magnitude as one oriented towards \(\theta + \pi\) we only apply filters in a range of directions from 0 to \(\pi\). We obtain:

\[
\frac{D_a(\vec{x}) \cdot D_a(\vec{y})}{\|D_a(\vec{x})\| \|D_a(\vec{y})\|} \approx \frac{2}{\pi} \int_0^{\pi} |\cos(\theta)| |\cos(\theta - \alpha)| d\theta
\]

We can assume, without loss of generality that \(0 \leq \alpha \leq \frac{\pi}{2}\). We obtain:

\[
\frac{2}{\pi} \int_0^{\pi} |\cos(\theta)| |\cos(\theta - \alpha)| d\theta = \frac{2}{\pi} \left(\frac{\pi}{2} \cos(\alpha) + \frac{\sin(\alpha)}{2} - \alpha \cos(\alpha) + \frac{\cos(\alpha) \sin(2\alpha)}{2} - \frac{\sin \alpha \cos(2\alpha)}{2}\right)
\]

\[
= \frac{2}{\pi} \left[\frac{\pi}{2} - \alpha \right] \cos(\alpha) + \sin(\alpha)\]

This first equality is obtained by breaking the integral into intervals of constant sign. The second equality follows from trigonometric identities.
In fact, it turns out that \( \frac{2}{\pi} \left( \frac{\pi}{2} - \alpha \right) \cos(\alpha) + \sin(\alpha) \approx \frac{2}{11} \cos(2\alpha) + \frac{9}{11} \). To show this, we can expand the two functions with Legendre polynomials. These form an orthonormal basis for the polynomials. The coefficients of the two functions in this basis are shown in Table 1. They are almost identical. The two functions are plotted in Figure 6.

Therefore, DOG essentially compares image gradients by taking the cosine of twice the difference in angle. This comparison is insensitive to the polarity of this difference, since it is periodic with a frequency of \( \pi \). Within this range it is monotonic with the difference in angle, and qualitatively identical to the comparison method in DIR.

Note that these functions are a continuous approximation to the discrete functions actually computed. However, in experiments (not shown) we have verified that the continuous and discrete versions perform identically, with \( k = 8 \).

This analysis addresses the question of whether it is important that DOG jets use a redundant set of filters at each scale. By using a set of eight filters we obtain a highly redundant feature set which, by definition, is not needed to capture the information present in the image. However, if we compare these jets by taking inner products, we can see that this is a discrete approximation to a comparison of a continuous function of the gradient. A non-redundant set of two oriented filters would provide a poor approximation to this continuous function, while eight orientations provide an excellent approximation.
5 Combining Methods

So far we have focused on two types of methods: methods based on whitening and methods equivalent to direction of gradient. We have shown that whitening is very effective on smooth surfaces while direction of gradient is effective on rough surfaces and discontinuities. We therefore expect to improve performance by combining these methods. In this section we consider some simple, intuitive methods of combining whitening with the direction of the gradient.

The simplest method is to sum the outputs of two filters with proper normalization. Direction of gradient is naturally normalized to the $[0, \pi]$ range. Whitening, however, requires normalization prior to combining. Let $s_1, s_2, ..., s_n$ denote the distances between the query image and $n$ reference images after whitening. We normalize them to the $[0, 1]$ range by dividing all the distances by $\max|s_i|$. Different regions in the image can have different roughness levels. We compensate for this effect by choosing the normalization factor adaptively in small local areas instead of once for the whole image. Our experiments show that adaptive normalization yields better results. We have also scaled the direction of gradient output to the $[0, 1]$ range. Our experiments on rough and smooth data show (see Section 6.2) that the simplest combination of whitening and direction of gradient performs better than either whitening or direction of gradient alone.

The above approach is trivial, but it already verifies the benefit of combining whitening with direction of gradient. We now turn to less straightforward combining methods: jet of Laplacian of Gaussians and Gabor jets. As far as we know Laplacian of Gaussian jet has not been considered before. Gabor jets have been extensively used in image comparisons, but not viewed as a combined method.

5.1 Jet of Laplacian of Gaussian

Let $L_\theta(\vec{x})$ denote the result at position $\vec{x}$ of convolving an image, $I$, with a second derivative of a Gaussian filter, in which the second derivative is taken in the direction $\theta$. By varying $\theta$ we produce a set of filters and by convolving an image with these filters, we obtain a vector of values at each pixel; one for each orientation. We form a jet of the magnitudes
of these outputs, \( L(\vec{x}) = (|L_{\theta_1}(\vec{x})|, \ldots |L_{\theta_k}(\vec{x})|) \). We call this a Laplacian of Gaussian jet (LGJ), because the oriented second derivative filters can be thought of as oriented Laplacians. Similar to other jets we compare points in two images by computing the LGJ \( J \) and \( I \) at corresponding points, and taking their normalized correlation: \( \frac{J \cdot I}{\|J\|\|I\|} \).

Next we will give an intuitive explanation of the behavior of LGJ on different types of surfaces.

**Behavior on discontinuities:** LGJ, like all similar jets, is invariant to offset, gain and polarity, so it is invariant to lighting changes at discontinuities.

**Behavior on rough surfaces:** In Appendix A we show that LGJ essentially computes the difference in the direction of the second derivative, while DIR computes the difference in the first derivative. For many rough surfaces, close to the areas where the first derivative is large, the second derivative will be large as well. This suggests that LGJ should have similar behavior to DIR on rough surfaces.

**Behavior on smooth surfaces:** Recall that the Laplacian of Gaussian (LAP) is an approximation to a whitening filter. This makes LAP effective in the case of images of smooth surfaces. The primary difference between LAP and LGJ is that a LAP is rotationally symmetric, while filters used in LGJ are oriented, 1D second derivative operators applied to a Gaussian at multiple orientations. Because of this similarity, it is natural to predict that LGJ will perform image comparison well for smooth surfaces.

Note also that LGJ may be expected to perform better than symmetric LAP in the presence of cast shadows. Both methods apply a filter suitable for whitening, using either a 2D whitening filter or oriented 1D whitening filters. Normalized correlation with a 2D filter means normalizing the output of the filters across the whole image, which achieves invariance to changes in global illumination levels. In LGJ, filters’ outputs are normalized at each pixel, which provides invariance when the intensity of the illumination varies locally, as happens inside cast shadows.

**To summarize:** LGJ is an effective image comparison method that naturally combines the direction of gradient with whitening. This makes it robust against illumination changes on both rough and smooth surfaces. When applied to rough surfaces, due to normalization it behaves similarly to the direction of gradient, and when applied to smooth surfaces it derives
5.2 Gabor Jets

Gabor[12] introduced the windowed Fourier transform. By taking the product of a Gaussian and a harmonic function such as a sine or cosine, one derives a filter that is localized in time and frequency. Lades et al.[25] suggested the use of Gabor filters as a local representation of features in an image, for use in face recognition. Specifically, following their notation, we may write the impulse response of a Gabor filter as:

\[
\psi_{\vec{k}}(\vec{x}) = \frac{\vec{k}^2}{\sigma^2} \exp\left(-\frac{\vec{k}^2 \vec{x}^2}{2\sigma^2}\right) \left[\exp(i\vec{k} \vec{x}) - \exp\left(-\frac{\sigma^2}{2}\right)\right]
\]

\(\sigma\) is a parameter that controls the scale of the Gaussian. The magnitude and the angle of the 2D vector \(\vec{k}\) determines the frequency of the harmonic function and its orientation, respectively. The final term \(\exp\left(-\frac{\sigma^2}{2}\right)\) is a normalizing constant added so that the Gabor integrates to zero. Lades et al.[25] then form these oriented Gabor filters into a jet.

The Gabor filter is complex. The real component of the Gabor is the product of a cosine and a Gaussian. For an orientation of zero, this component is symmetric about the \(y\)-axis, and we call it the even component of the Gabor. The imaginary part of the Gabor is the product of a sine and a Gaussian, and is called the odd component. The magnitude of the complex Gabor filter is taken when forming the output of Gabor filters into a jet. This captures the amplitude of the image in a band of frequencies, but discards the phase.

These and similar representations have been very widely used throughout the object recognition community. At the same time, they have become the dominant representation used for texture synthesis and comparison (for a general discussion, see [11]). Gabor Jets have been motivated by considerations unrelated to lighting change, but some authors (eg., Wiskott et al.[42]) have noted that they are invariant to changes in offset and gain, as are all normalized jets.

It has been noted that the odd components of the Gabor jet are quite similar to a difference of Gaussian operator. Figure 7 plots a comparison of the two functions. This similarity suggests that a jet composed of just the odd components of a Gabor will behave similarly to a jet of difference of Gaussians, and therefore to a comparison based on the direction of
Figure 7: Left: A cross-section of the odd component of a gabor filter compared to a difference of Gaussian. The gabor is shown as a dashed line, while the difference of Gaussian is solid. Right: The even component of a gabor filter (dashed) compared to a Laplacian of Gaussian.

Figure 8: Normalized, absolute values of the output of eight horizontal, odd Gabors (GO) as an image with a vertical line is shifted by zero, two and four pixels. For reference, the output of difference of Gaussians is plotted in black solid lines (when normalized, this is invariant to shifting).

gradient. We will call image comparison based on odd Gabors GO. Our experiments indeed show that the behavior of GO and DOG are very similar qualitatively.

We have found, however, that GO does perform somewhat better than DOG (see Section 6.2). To gain some intuition about this, we consider the simple example of an image containing a step edge, slightly blurred by a narrow Gaussian. (See also [16] for discussion of the difference between these two, and the representational adequacy of multiple oriented Gabors.) The direction of the gradient is constant throughout this image patch. Therefore, the response of DOG will also be constant. DOG is capable of discriminating between two image patches that contain step edges with different orientations. However, it cannot discriminate between two image patches if they contain step edges of the same orientation, but different positions. In contrast, GO is sensitive to the position of a step edge as well as to its orientation. This is because as the odd Gabor shifts in the image, its secondary peaks
away from the center of the filter also respond to the step edge. To illustrate this, in Figure 8, we plot the outputs of oriented difference of Gaussian and odd Gabor filters as a step edge is translated in the image. The Gabor filters’ outputs vary considerably with small shifts. This allows GO to make finer distinctions about whether two images really match. It also makes GO more sensitive to image misalignments or deformations. It is beyond the scope of this paper to determine how lighting invariant image measures interact with image deformations, but we note that while GO achieves better lighting insensitive performance than DOG or DIR, it may not be superior in realistic applications. In fact, it is precisely to overcome sensitivity to distortion that Gabor jets remove the phase of these filter responses.

We have seen that a jet of odd Gabor components behaves much like a jet of difference of Gaussians, and therefore like the direction of the gradient. Similarly, a jet of even gabor components (we will call it GE) behaves much like a jet of Laplacian of Gaussians (and indeed this is the inspiration for our method). While we do not analyze the differences between the even components of Gabors and oriented Laplacians, their similarity is clear from Figure 7-right.

We can therefore see that Gabor jets contain elements of two different methods, one which compares images based on the direction of the gradient, and the second of which is itself similar to a combination of whitening and a method sensitive to the direction of the images second derivative. We now examine the method by which the odd and even Gabor components may be combined.

By taking the magnitude of the output of a complex Gabor filter prior to taking the inner product between two jets, we are extracting the amplitude of the response and discarding the phase. The motivation for this is to achieve quasi-invariance to small deformations in shape (phase has been used separately for fine alignment of images [42]). However, it is not clear that discarding phase should help achieve illumination invariance. We conjecture that Gabor jets are illumination insensitive largely because they combine even and odd components of Gabor jets, which separately produce illumination insensitivity. To test this conjecture, we experimentally compare the performance of Gabor jets (GJ) and a method we call GO+GE. GO+GE separately compares the even and odd components of Gabor jets, and then adds the resulting comparison measures. The details of these experiments are given in Section
6.2, and the results can be seen in Figures 12. In brief, we compare GJ with GO+GE on both smooth and rough objects. In every case, GO+GE outperforms GJ, except when the performance of both measures is quite poor. This supports our conjecture and suggests that Gabor jets are illumination insensitive because they are a simple combination of two different comparison methods, GO and GE. These two methods, between them, possess illumination insensitivity for both rough and smooth objects.

6 Experiments

We have designed experiments to test our main claims:

1. Different types of surfaces require different representations for illumination insensitive image comparisons. Specifically:
   - Direction of gradient and other methods based on normalization are most effective on discontinuities and rough surfaces and their performance drop as the smoothness of the surface increases.
   - Whitening methods (including the Laplacian of Gaussian as an approximation to a whitening filter) have the best performance on smooth surfaces.

2. Combining surface dependent representations is beneficial to illumination insensitive image comparisons.

6.1 Testing Surface Dependent Representations

To test the first contention we applied whitening and the Laplacian of Gaussian (expected to be good representations of smooth surfaces), and DIR, DOG, and normalized correlation (expected to be good representations of rough surfaces), for recognizing different collections of objects.

6.1.1 Recognition Methods

We applied the following recognition scheme as an application of our approach to image comparison. For a given collection of objects, we took reference images of each object under
the same lighting conditions. Then we took query images of these objects with different lighting directions. A query image was compared to each reference image, and matched to the one that minimized the appropriate comparison measure. We have tested the following comparison measures:

**LAP:** We filtered each image with a Laplacian of Gaussian, normalized the filtered images to unit length and measured the SSD between the query and each reference image.

**Whitening:** For every set of images we learned a whitening filter as 2D causal filter that minimizes the MS prediction error. A size of the filter varied depending on the smoothness of the images. The whitening filter was trained on the difference images between reference images and corresponding images associated with the same object under varying illumination. When tested on different images, we whitened each image, normalized the whitened images to unit length and measured the SSD between the query and each reference image.

**DIR:** We first smooth all images with a Gaussian. We define the direction of the gradient of the smoothed image, $I$, as:

$$ r(\vec{x}) = \text{mod}(\text{atan}\left(\frac{\partial G * I}{\partial y} / \frac{\partial G * I}{\partial x}\right), \pi) $$

Note that by taking the direction mod $\pi$ we are discarding the polarity of an edge. Then, to compare the directions of gradients in two images, $r_I, r_J$, we took: $\min((r_I - r_J)^2, \text{mod}(\pi - |r_I - r_J|, \pi)^2)$.

**DOG, GO, GE, GO+GE, GJ, LGJ**³: These methods use jets of difference of Gaussians (GO), the odd (GO), or even (GE) components of Gabor Jets (GJ), the sum of these two (GO+GE) or jets of oriented second derivatives (LGJ). All these methods involve a Gaussian, which we select so that the amount of smoothing is the same as with DIR.

**NC:** We perform normalized correlation in every $3 \times 3$ window and took its absolute value to discard the edge polarity. This allows fair comparison with DIR which is also polarity insensitive. To integrate the results over the whole image we computed the magnitude of the vector that contains these absolute normalized correlations.

³All these methods are tested in the next section where we discuss the combined strategy.
Figure 9: Images used in our experiments. The first two columns show two different objects under the same lighting conditions. The third column shows a query image of the first object under different lighting conditions. Rows, from top to bottom, show smooth synthetic objects, smooth real objects, and rough real objects.

6.1.2 Data Sets

We compared the above methods on images produced by three groups of objects: synthetic smooth textureless objects, smooth real and rough real objects (sample images of objects from these sets are shown in Figure 9). Since the Whitening method requires training, we divided each image set into training and test sets.

**Synthetic smooth set:** Every scene was created as a sum of random harmonic functions, with fixed amplitudes but random directions and phases. This provides an ensemble of images with similar statistical properties. These were rendered as Lambertian surfaces with point sources.

The training set included 2000 images with a fixed illumination, deviating 67.5 degrees from the vertical direction. Since the synthetic images are very smooth we trained a large whitening filter of 265 coefficients inside a $23 \times 23$ window. The test was done on 5000 triplets of images. Two of each triplet were reference images produces from random scenes illuminated by the same nearly vertical illumination. The third image was a query image synthesized from the first scene, with a different illumination, deviating up to 67.5 degrees from the vertical direction.
**Real Smooth Set:** We created eighteen real, smooth objects from clay and illuminated them by a single light source moving along a half circle, so that its distance from the object was roughly fixed. We used a camera placed vertically above the object, and took 14 images of every object with different lighting directions at angles in the range \([-65, 65]\) degrees to the vertical axis. One image of each object, associated with a nearly vertical illumination, was chosen as the reference image. The whitening filter was trained on the difference images between reference images and corresponding images associated with the same object and six other illuminations. Twelve images associated with 2 objects (out of 18) were used. We learned the whitening filter as a 2D causal filter with 25 coefficients inside \(7 \times 7\) windows. (We found this size to be optimal for the given domain). All images of the 18 objects except the reference images were used as query images (234 images).

**Real Rough Set:** For rough objects, we used the Yale database [6], which contains 20 objects with abrupt changes in albedo and shape. The database consists of 63 images of each object with lighting direction deviating up to 90 degrees from the frontal. The whitening filter was trained on 20 difference images associated with two objects and 10 illuminations. Varying the size of the whitening filter from 3x3 to 7x7 window had no real effect on recognition performance. Similar to the previous set, one image of each object, associated with frontal illumination, was chosen as the reference image. All the rest of the images where used as queries.

**6.1.3 Results**

We found that in all our experiments, DOG and NC behaved almost identically to DIR (this was also shown analytically in sections 4.1 and 4.3). Therefore we do not show plots for the DOG and NC methods. The results of the experiments (Figure 10) show that Whitening outperforms other methods on smooth objects. LAP performs quite well on real smooth objects. However on very smooth surfaces (synthetic images) its performance degrades, because its size is insufficient to handle the high correlations between the grey levels present in such smooth surfaces. DIR is less good, relative to Whitening and LAP, on smooth objects, but outperforms them on rough objects. Whitening and LAP are almost identical
on rough objects because decorrelation in the rough objects occurs over a small range, and the whitening filter looks very much like the Laplacian.

The above results support our conjecture that DIR and other methods based on normalization (DOG, NC, etc...) are best for image comparisons of rough surfaces, while Whitening and LAP are best for images produced by smooth objects.

6.2 Testing Combined Methods

We tested the combined methods discussed in the paper (simple combining, GJ, GO+GE and LGJ) on three sets of real objects: rough, smooth and combined. The rough and smooth sets are the same as above. The combined set contains 27 clay objects with varying roughness (Figure 11). The database consists of 32 images of each object with lighting direction deviating up to 86 degrees from the frontal. The results, described in Figure 12, are plotted for illumination angles in [0, 65] available for all data set.
Figure 11: Sample images from the combined set: top line – rough objects, bottom – similar smooth objects.

Figure 12: Experimental results of combined methods compared against surface dependent representations. The results are from real objects under varying illumination: top left – rough, top right – smooth, bottom left – combined. Bottom right plot compares the performance of GE against LGJ and GE against DOG. (These figures are best viewed in color.)
First, we see that even though there are some differences in performance of the combined methods on different sets, in general, they are better than surface dependent methods, specifically Whitening and DIR.

Next, we point out that GO+GE performs very similarly to GJ. We interpret this as support for our conjecture that Gabor jets work well under variable lighting because of the individual performance of the their constituents GE and GO, and not because of the particular way in which GJ combines these two filters. We noted earlier that qualitatively GO is very similar to DOG and GE is very similar to LGJ. We verified the posited similarity by comparing the performance of these methods on the combined set (Figure 12 bottom right).

6.3 Summary of Experimental Results

Our experiments support our prior analysis. Specifically, we showed that:

- Whitening methods work well on smooth surfaces, while methods related to the direction of the gradient work well on rough surfaces.
- A learned whitening filter outperforms other approaches on very smooth surfaces.
- Normalized correlation and a jet of difference of Gaussians show the same performance as the direction of gradient method.
- Smooth and rough methods can be combined to achieve generally superior performance on all types of objects.
- Gabor jets display strong lighting insensitive matching on a variety of surfaces because they combine a whitening type operator and a direction of gradient operator.
- LGJ outperforms, or performs as well as other combining methods.

7 Conclusions

In this paper we considered a variety of issues concerning lighting insensitive image matching. We have shown analytically that a variety of recognition methods are either identical to, or
closely related to comparisons based on the direction of the gradient, including normalized correlation, histogram equalization, and comparisons using the output of oriented difference of Gaussian filters. These are well suited to matching images of rough surfaces; however we have shown that rough and smooth surfaces present different challenges to recognition. Using ideas from signal detection we demonstrate that whitening filters can be used to compare images of smooth surfaces. These results lead to the idea of combining direction of gradient and whitening methods, to achieve illumination invariance on a wider range of surfaces. Indeed this works well. Intriguingly, the already popular Gabor Jets seem already to combine such operations.

The combining methods that we explore in this paper are quite simple, yet they perform very well. We hope that by elucidating the reasons for their good performance we will also help pave the way for the development of better combining methods.

A Appendix: LGJ

A 2D Gaussian is defined as follows:

\[
G(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)
\]

The derivatives of the Gaussian are:

\[
G_x(x, y) = -\frac{x}{\sigma^2}G, \quad G_y(x, y) = -\frac{y}{\sigma^2}G
\]

\[
G_{xx} = \frac{1}{\sigma^2}G\left(\frac{x^2}{\sigma^2} - 1\right), \quad G_{yy} = \frac{1}{\sigma^2}G\left(\frac{y^2}{\sigma^2} - 1\right), \quad G_{xy} = \frac{xy}{\sigma^4}G
\]

The first oriented derivative is \(G_x \cos \theta + G_y \sin \theta\). The second oriented derivative is then

\[
(G_{xx} \cos \theta + G_{yx} \sin \theta) \cos \theta + (G_{xy} \cos \theta + G_{yy} \sin \theta) \sin \theta
\]

\[
= G_{xx} \cos^2 \theta + 2G_{xy} \sin \theta \cos \theta + G_{yy} \sin^2 \theta
\]

We will start by analyzing LGJ for a specific case, and then we will consider a more general case. Laplacian detects curvatures \(ax^2 + 2bxy + cy^2\) in image intensities (it’s response to gradients \(ax + by\) is zero). Without loss of generality we can rotate the axes so that \(b = 0\). Consider a special case of paraboloid elongated along the y axis: \(|a| \gg |c|\) The following
analysis also holds for a cylinder with the radius $\gg \sigma$. For this special case the general form of the paraboloid is reduced to $x^2$ and the direction of the maximum of the second derivative is along the x axis. Convolving this image with the Laplacian of Gaussian oriented in direction $\theta$ will result in:

$$L_\theta(x) = \int \int (G_{xx} \cos^2 \theta + 2G_{xy} \sin \theta \cos \theta + G_{yy} \sin^2 \theta)x^2 \, dx \, dy$$  \hspace{1cm} (6)

Next we compute each term in eq. (6):

$$\int \int G_{xy}x^2 \, dx \, dy = \int \int \frac{x^3}{\sigma^2} \, G \, dx \, dy = 0$$

$$\int \int G_{yy}x^2 \, dx \, dy = \int \int (\frac{1}{\sigma^2} \frac{y^2}{\sigma^2} - 1)x^2 \, G \, dx \, dy = \frac{1}{\sigma^2} (E[(x^2)^2] - E[x^2]) = \frac{1}{\sigma^2} (\frac{E[x^2]}{\sigma^2} - E[x^2]) = 0$$

If we denote $\int \int G_{xx}x^2 \, dx \, dy = M_{xx}$, then eq. (6) reduces to

$$L_\theta(x) = M_{xx} \cos^2 \theta$$

and it’s magnitude is:

$$\|L_\theta(x)\|^2 = M_{xx}^2 \int_0^\pi \cos^4 \theta \, d\theta = \frac{3\pi}{8} M_{xx}^2$$

If we compute the response of the oriented Laplacian at a point, $\vec{y}$, in another image, at which the direction of the maximum of second derivative is $\alpha$, we obtain

$$L_\theta(\vec{y}) = M_{yy} \cos^2(\theta - \alpha)$$

LGJ are compared at $\vec{x}$ and $\vec{y}$ by computing:

$$\frac{L(\vec{x})L(\vec{y})}{\|L(\vec{x})\|\|L(\vec{y})\|} = \frac{8}{3\pi M_{xx}} \frac{1}{3\pi M_{yy}} M_{xx} M_{yy} \int_0^\pi \cos^2 \theta \cos^2(\theta - \alpha) \, d\theta$$

$$= \frac{8}{3\pi} \frac{\pi}{8} (\cos(2\alpha) + 2) = \frac{1}{3} (\cos(2\alpha) + 2)$$  \hspace{1cm} (7)

Consider a more general case of paraboloid $I = ax^2 + cy^2$ where we don’t have the assumption
that $\|a\| \gg \|c\|$. We can normalize the coefficients such that $I = x^2 + ky^2$, where $k$ controls the elongation. Assume that

$$
\int \int G_{xx}x^2dxdy = \int \int G_{yy}y^2dxdy = \mu
$$

then

$$
L_\theta(\bar{x}) = \mu [\cos^2 \theta + k \sin^2 \theta]
$$

$$
f_{k,s}(\alpha) = \frac{L(\bar{x})L(\bar{y})}{\|L(\bar{x})\| \|L(\bar{y})\|} = \frac{\int_0^\pi [\cos^2 \theta + k \sin^2 \theta][\cos^2(\theta - \alpha) + s \sin^2(\theta - \alpha)]d\theta}{\sqrt{\int_0^\pi [\cos^2 \theta + k \sin^2 \theta]^2d\theta} \sqrt{\int_0^\pi [\cos^2(\theta - \alpha) + s \sin^2(\theta - \alpha)]^2d\theta}}
$$

$$
f_{k,s}(\alpha) = \frac{2(k + 1)(s + 1) + (k - 1)(s - 1)\cos(2\alpha)}{\sqrt{3 + 2k + 3k^2} \sqrt{3 + 2s + 3s^2}} \quad (8)
$$

In this general case not only the direction of the second derivative is allowed to change, but the elongation factor can be different in two images ($k$ in the first image and $s$ in the second). Figure ?? shows the dependencies between $f_{k,s}(\alpha)$ and the elongation factor.

For the elongated case when $k, s \to 0$, eq. 8 reduces to eq. 7 that essentially computes $\cos(2\alpha)$. Hence in the places of high gradients LGJ is very similar to DOG and DIR.
Appendix: Proof of Proposition 1

Proof: The reflected light function $I(x) = \rho \cos(\phi - \theta)$ is a random process. Its expected value is:

$$E[I(x)] = \rho (\sin \phi E[\sin \theta(x)] + \cos \phi E[\cos \theta(x)]) = \rho \cos \phi E[\cos \theta(x)] = \rho \cos \phi e^{-\sigma^2 \phi^2}.$$  \hfill (9)

In computing the variance, we make use of the Gaussian integral

$$\int_{-\infty}^{\infty} \cos x e^{-x^2/2a^2} \, dx = \sqrt{2\pi} a e^{-a^2/2} \quad \hfill (10)$$

The variance of $I(x)$ is:

$$\sigma_I^2 = E[(I(x) - E[I(x)])^2]$$

$$= \rho^2 E[(\sin \phi \sin \theta + \cos \phi \cos \theta - \cos \phi E[\cos \theta])^2]$$

$$= \rho^2 (\sin^2 \phi E[\sin^2 \theta] + \cos^2 \phi E[\cos^2 \theta] - \cos \phi E[\cos \theta]^2 + \sin \phi \cos \phi E[\sin \theta (\cos \theta - E[\cos \theta])])$$

$$= \frac{1}{2} \rho^2 (\sin^2 \phi (1 - e^{-2\sigma^2 \phi}) + \cos^2 \phi (1 - e^{-\sigma^2 \phi}))$$  \hfill (11)

The last term in the third line integrates to zero. Simple trigonometric expressions like $\sin^2 \theta = \frac{1}{2} (1 - \cos 2\theta)$ and the Gaussian integral in eq. 10 suffice to derive this expression.

Consider now the autocorrelation. Let $x_1, x_2$ be two points for which the correlation coefficient of the tangent direction is $r = r(|x_1 - x_2|)$. Then,

$$C_I(x_1, x_2) = E[(I(x_1) - E[I(x_1)])(I(x_2) - E[I(x_2)])]$$

$$= \rho^2 E[(\sin \phi \sin \theta_1 + \cos \phi \cos \theta_1 - \cos \phi E[\cos \theta_1]) \cdot$$

$$\quad \quad \quad \quad \quad \quad \quad \quad \quad \quad (\sin \phi \sin \theta_2 + \cos \phi \cos \theta_2 - \cos \phi E[\cos \theta_2])]$$

$$= \rho^2 (\sin^2 \phi E[\sin \theta_1 \sin \theta_2] + \cos^2 \phi E[\cos \theta_1 \cos \theta_2] - \cos^2 \phi E[\cos \theta]^2$$

$$= \frac{1}{2} \rho^2 (\sin^2 \phi e^{-\sigma^2 \phi^2} (e^{r \sigma^2 \phi} - e^{-r \sigma^2 \phi}) + \cos^2 \phi (e^{-\sigma^2 \phi} (e^{r \sigma^2 \phi} + e^{-r \sigma^2 \phi}) - 2e^{-\sigma^2 \phi}))$$

Note that all $\sin \theta_1 \cos \theta_2$ terms vanish due to symmetry. The rest of the derivation requires us to change variables to the sum and difference of $\theta_1$ and $\theta_2$, which are independent. Simple trigonometric expressions and the Gaussian integral in eq. 10 are used as well. \hfill \Box
References


