The 4-source photometric stereo technique for 3-dimensional surfaces in the presence of highlights and shadows [1]

Svetlana Barsky and Maria Petrou
School of Electronics and Physical Sciences,
University of Surrey
Guildford, GU2 7XH, UK

[1] Acknowledgement. This work was supported by EPSRC grant GR/M73361. The photometric stereo images used were collected by G. McGunnigle and M.J. Chantler in Heriot-Watt University, UK.

Abstract

We present an algorithm for separating the local gradient information and Lambertian colour by using 4-source colour photometric stereo in the presence of highlights and shadows. We assume that the surface reflectance can be approximated by the sum of a Lambertian and a specular component. The conventional photometric method is generalised for colour images. Shadows and highlights in the input images are detected using either spectral or directional cues and excluded from the recovery process, thus giving more reliable estimates of local surface parameters.

Index terms: photometric stereo, surface orientation and colour recovery, highlights, shadows

1 Introduction

The motivation for this paper is the problem of illumination-invariant[1] characterisation of three-dimensional surfaces with unknown reflectivity, which, however, can be assumed to be well approximated by a Lambertian component plus a specular component.

Characterisation of 3-dimensional surfaces from 2-dimensional images is not an easy task. The 2-dimensional images depend on variation in both surface reflectance and surface relief. While the reflectance properties are intrinsic to a surface, the surface relief produces a pattern of shadings that depends strongly on the direction of the illumination. The appearance of a 3D surface changes drastically with illumination. It is often useful to recover some surface parameters that are independent of the illumination direction, for example, local surface orientation and reflectance.

The photometric stereo (PS) technique [20] uses several images of the same surface taken from the same viewpoint but under illuminations with different directions. Thus the changes of the intensities in the images depend only on the local surface orientation, which can be recovered by combining the information from all images. Let us consider a surface patch. When imaged under a particular illumination, it produces a camera intensity value which depends on the properties of the illumination configuration, the surface reflectance and the surface orientation, according to some photometric equation (hence the term "photometric stereo"). Having several images of the same surface patch gives us a system of such equations which can be solved for surface orientation and possibly some reflectance parameters. For Lambertian surfaces it is enough to have 3 images to recover both local surface normal and albedo. Various developments of the method were offered in later years, utilising different reflectance models and assumptions about the surface (see Section 2 for more details). One of these methods ([3]) suggests using 4 images to detect specularities. Under the assumption of nearly Lambertian behaviour outside the specularity region, one can exclude a highlighted pixel from consideration, and recover the local surface gradient from the remaining three pixels.

[1] Under "illumination-invariant" we understand "illumination direction invariant", leaving the problem of spectral illumination invariance outside the scope of this paper.

We propose a generalisation of an existing greyscale photometric stereo (GPS) technique for use with colour images. We proceed to show how the method mentioned above is susceptible to errors in the presence of shadows. We make use of spectral information as an additional cue for detecting highlights. For the case when the spectral information is not sufficient for the detection of highlights, an alternative method, based on comparing a recovered normal with the corresponding specular direction, is presented. As a result, we propose a combined method for recovering local gradient and colour for 3-dimensional non-Lambertian surfaces from colour images, capable of coping with shadows and highlights. This paper is structured as follows. In Section 2 we discuss previous work and relate our approach with the bulk of surface reconstruction problems tackled by other researchers. In Section 3 we state our assumptions and establish terms and notation. In Section 4 we describe the linear photometric stereo method for grey-scale images, and generalise it to colour images. In Section 5 we discuss the effect shadows and highlights have on the outcome of the linear algorithm. Section 6 is devoted to highlight and shadow detection, and ways of dealing with them are suggested. Section 7 presents the results of experiments with synthetic and real data. Conclusions are drawn in Section 8.

2 Previous work

The photometric stereo method has been around for 20 years now, and it has received an extensive theoretical and experimental treatment. It was conceived by Woodham [19][20] who first used it to recover local surface orientation. The method was based on the use of the so called *reflectance maps* in the form of look-up tables. These tables were obtained by means of a calibrating sphere made of the same material as that of the imaged surface, which allowed one to map obtained sets of intensities directly to surface normals. Since then the idea was extended to recover not only surface orientation but also some reflectance parameters for a number of reflectance models. In what follows we discuss the existing PS techniques for both greyscale and colour images.

Greyscale images The vast majority of the existing body of work in the PS field deals with greyscale images.

For Lambertian surfaces the photometric equations are linear, which allows one to formulate the problem in a matrix form. Inverting a system of linear equations makes it possible to recover the unknown albedo as well as the gradient from three image intensities for every surface patch (see Section 4.1). Therefore by using three images of a 3-dimensional Lambertian surface in the absence of shadows one can successfully separate the surface shape and the pattern on the surface produced by varying albedo.

In the case of Lambertian surfaces with spatially uniform albedo the system of photometric equations becomes overconditioned, and the "surplus" information may be used in a variety of ways, for example, to find outliers [20], or reconstruct unknown illumination directions and strengths [18].

A large amount of research is devoted to the recovery of reflectance parameters (of some particular reflectance model) along with the local gradient. Estimation of the reflectance parameters can be performed locally (provided we have a sufficient number of images), and therefore such algorithms are suitable for surfaces with spatially variable reflectance. Nayar, Ikeuchi and Kanade [12] applied PS using a so called *hybrid reflectance model*. Tagare and deFigueiredo [17] developed the theory of photometric stereo for the class of *m-lobe reflectance maps*. Their research was continued by Kay and Caelly [8] who investigated the problem from a practical point of view, applying non-linear regression to a large number of input images.

The above methods recover both surface orientation and reflectance parameters but they require quite a lot of images, and the algorithms are fairly complicated. Since many non-Lambertian surfaces exhibit near-Lambertian behaviour outside their regions of specularity, it is a very attractive option to apply the linear algorithm, developed for Lambertian surfaces, to surfaces with non-Lambertian reflectance, and treat highlights as deviations from the Lambertian law. This technique was proposed by Coleman and Jain [3], who used the fourth image to detect and exclude highlights. We use a modification of this method in this paper, and discuss it in more detail later.

Colour images The information in the colour image of a Lambertian surface illuminated by a single light source is redundant since the photometric equations for individual colour bands are linearly dependent. An efficient way to use this redundancy is to perform a conventional PS method using a single colour image of a Lambertian surface under a complex illumination rather than three greyscale images [5][6][4][13]. The surface should be illuminated by several light sources which are spectrally distinct and their directions do not lie in the same plane. This method is called Shape-from-Colour. In [7] it was applied to the task of face recognition. Kontsevich *et al* [10] considered a similar approach.

Christensen and Shapiro [2] introduced the method of *colour photometric stereo* (CPS) for surfaces with an arbitrary reflectance. The method is a generalisation of [19], and also uses look-up tables. The disadvantage of this method is that the surface should be either uniformly coloured, or its colours should form distinct separable clusters in the colour space, which significantly restricts the choice of acceptable surfaces. Another disadvantage is the need for a preliminary calibration. On the plus side, this method is not restricted by any specific reflectance model.

According to the *dichromatic reflection model* [9], a general reflectance function can be modelled as the sum of a matte (Lambertian, body) and a specular (surface) component. Incorporating spectral information into conventional Shape-from-Shading techniques gives a welcome advantage. See, for example, Lee and Bajcsy [11], who used a spectral differencing algorithm to detect specularities from multiple images. They, however, varied the pose of the object rather than the illumination.

Assuming a dichromatic reflection model, Schlüns and Wittig [15] also used colour information to develop a colour PS technique for non-Lambertian surfaces without precalibration. They attempted to recover the surface parameters directly from the three input colour images using colour histograms. They worked out the illumination and body chromaticities directly from the histogram, and then decomposed the image pixels into linear combinations of matte and specular components. This method was not tried on real surfaces. In an ideal case all image pixels lie on a plane spanned by the chromaticity vectors of the body and the illumination colours. Such pixels indeed are easy to decompose into linear combinations according to the dichromatic reflection theory. If, however, histograms are not planar, the decomposition coefficients are not reliable. In real surfaces there are always variations in the colour, and there are always errors, so histograms are rarely planar. Another difficulty is presented by saturated pixels. They appear when the irradiance of a surface facet exceeds the capacity of image sensors, and the pixel becomes white. A white pixel can only be decomposed uniquely, thus all saturated pixels yield the same decomposition. In real pictures highlights often are saturated, and in this method the highlighted areas will be recovered as flat patches.

The contribution of this paper Our ultimate goal is to describe 3-dimensional surfaces in a way that is invariant to the direction of illumination. We assume that a surface is not necessarily Lambertian, and that the reflectance parameters may vary along the surface. We do not assume any prior knowledge about the surface, so we can not use preliminary calibration. We also want to keep the algorithm practical and easy to implement, therefore the number of images in a photometric set should be kept relatively low. For these reasons the 4-source PS method proposed by Coleman and Jain [3] is a rather appealing option.

We use colour rather than greyscale images, because for non-Lambertian surfaces the spectral content of the images gives an additional cue for the detection of specularities. Usually, to detect highlights in colour images by spectral difference, some global technique is employed, which often involves building colour histograms of the input images (e.g. [15]). We, on the other hand, compare colour pixels locally, for each surface facet individually. This allows us to consider rough surfaces with spatially variable reflectance (i.e. surface textures). However, for surfaces whose colour is close to the colour of the illuminant (e.g. grey surfaces in white light), the spectral difference method does not work. For such surfaces we propose an alternative technique which compares the recovered normals with specular directions. This method is less reliable than the spectral difference method, but it also gives good results.

3 Assumptions, terms and notation

We assume that both camera and light source are far away from the surface, so the viewing direction and illumination direction are constant across the surface. We choose the coordinate system so that the image plane coincides with the xy plane, and the z axis coincides with the viewing direction. Then the surface can be described by a 2D height function $z = S(x, y)$.

For every point on the surface we can define its gradient components: $p \equiv \frac{\partial S}{\partial x}$, $q \equiv \frac{\partial S}{\partial y}$, and the normal unit vector $\mathbf{n}$[2]:

$$\mathbf{n} \equiv \frac{1}{\sqrt{p^2 + q^2 + 1}}(p, q, -1)^T$$

We assume that the surface in question can be approximated by a collection of flat patches (each corresponding to an image pixel). Then for each surface patch a local normal $\mathbf{n}$ refers to its slope with respect to the camera-based coordinate system.

We do not assume smoothness or even integrability of the surface. Each surface patch is considered in isolation from the others, thus enabling us to apply the algorithm to fairly rough surfaces. However, we assume that the roughness of the surface manifests itself at scales larger than the pixel resolution, so we assume that each surface patch of pixel size is smooth.

The illumination is described by vector $\mathbf{L}$ which points from the surface towards the illumination source. Illumination vector $\mathbf{L}$ can be represented as the product of unit vector $\mathbf{l}$ which defines the illumination direction, and a scalar μ, proportional to the illumination strength, so that $\mathbf{L} \equiv \mu\mathbf{l}$. If we have several illumination sources, we denote them using a superscript: $\mathbf{L}^1, \ldots, \mathbf{L}^K$, where K is the number of light sources. We assume that all the illuminants have the same spectral content, but their strengths may vary. The case of spectrally different illumination requires different treatment and is not considered in this paper.

In this work we consider four different sources at directions such that no three of them lie in the same plane. Additional requirements to the illumination set-up are discussed in

[2] Throughout the paper small bold letters refer to unit vectors

Appendix A.

A pixel, obtained by a camera with $\mathcal{A}$ sensors in each cell, can be represented by a vector in an $\mathcal{A}$-dimensional *colour space*. A colour pixel, obtained by the kth illuminant, is denoted by $\mathbf{I}^k = (I_1^k, \ldots, I_{\mathcal{A}}^k)$. A grey-scale pixel, obtained under the kth illuminant, is denoted by I_0^k.

The notion of Lambertian colour we use in this paper incorporates not only the surface reflectance properties, but also the spectral properties of the illumination, and the sensitivity of the camera sensor. Thus one should keep in mind that the recovered "colour" depends on the illumination and the camera used for experiments.

Suppose that the surface is illuminated by light with spectral distribution $\mu\mathcal{E}(\lambda)$, where μ is a parameter proportional to the strength of the light. The Lambertian surface reflectance function is $(\mathbf{l}\cdot\mathbf{n})\mathcal{R}(\lambda)$ (where $\mathbf{l}$ is the illumination direction, and $(\cdot)$ represents the dot product of two vectors), and the sensitivity function of the αth camera sensor is $\mathcal{Q}_\alpha(\lambda)$. Then the value recorded by the αth sensor is:

$$I_\alpha = \int_{-\infty}^{\infty} \mu\mathcal{E}(\lambda)(\mathbf{l}\cdot\mathbf{n})\mathcal{R}(\lambda)\mathcal{Q}_\alpha(\lambda)d\lambda = (\mathbf{L}\cdot\mathbf{n})\int_{-\infty}^{\infty} \mathcal{E}(\lambda)\mathcal{R}(\lambda)\mathcal{Q}_\alpha(\lambda)d\lambda \qquad (1)$$

The factor $C_\alpha \equiv \int_{-\infty}^{\infty} \mathcal{E}(\lambda)\mathcal{R}(\lambda)\mathcal{Q}_k(\lambda)d\lambda$ does not depend on the geometry of the scene, and we refer to vector $\mathbf{C} = (C_1, \ldots, C_{\mathcal{A}})^T$ as the "body colour". For a more rigorous treatment of the subject see, for example, [4]. When talking about greyscale images, we have only one equation (1), and we shall use the term "albedo" for $\rho \equiv \int_{-\infty}^{\infty} \mathcal{E}(\lambda)\mathcal{R}(\lambda)\mathcal{Q}(\lambda)d\lambda$, where $\mathcal{Q}(\lambda)$ is the sensitivity of the camera sensor. Note that factor ρ is proportional to the true surface albedo in the case of a white illuminant, and flat camera spectral response.

4 Photometric Stereo for Lambertian surfaces

4.1 GPS for Lambertian surfaces

Let us consider a Lambertian surface patch with albedo ρ and normal $\mathbf{n}$, illuminated in turn by three illumination sources with directions $\mathbf{L}^1$, $\mathbf{L}^2$, and $\mathbf{L}^3$. In this case we can express the intensities of the obtained (greyscale) pixels as:

$$I_0^k = \rho(\mathbf{L}^k \cdot \mathbf{n}), \qquad \text{where} \quad k = 1, 2, 3. \tag{2}$$

We stack the pixel intensities to obtain the pixel intensity vector $\mathbf{I_0} = (I_0^1, I_0^2, I_0^3)^T$. We also stack the illumination vectors row-wise to form the *illumination matrix* $[L] = (\mathbf{L}^1, \mathbf{L}^2, \mathbf{L}^3)^T$. (The square brackets are used throughout the paper to denote matrices). Then (2) could be rewritten in matrix form:

$$\mathbf{I_0} = \rho[L]\mathbf{n} \tag{3}$$

If the three illumination vectors $\mathbf{L}^k$ do not lie in the same plane, then matrix $[L]$ is non-singular and can be inverted, giving:

$$[L]^{-1}\mathbf{I_0} = \rho\mathbf{n}$$

Since $\mathbf{n}$ has unit length, we can estimate both the surface normal (as the direction of the obtained vector) and the albedo (as its length). If we have more than three input images, the illumination matrix $[L]$ is not square and cannot be inverted directly. Following the standard Least Square Error procedure, we can recover the albedo and the normal as:

$$\rho\mathbf{n} = ([L]^T[L])^{-1}[L]^T\mathbf{I_0}$$

Extra images allow one to recover the surface parameters more robustly.

4.2 CPS for Lambertian surfaces

Let us now assume that we have 3 colour images of the same Lambertian surface from the same viewpoint, illuminated in turn by 3 illumination sources, which are described by the illumination matrix $[L]$. The intensity triplets $\mathbf{I}^1$, $\mathbf{I}^2$, and $\mathbf{I}^3$, produced by a surface patch (under each of the 3 illumination arrangements), are described by:

$$\mathbf{I}^k = (I_1^k, I_2^k, I_3^k)^T = (\mathbf{L^k} \cdot \mathbf{n})\mathbf{C}$$

where vector $\mathbf{C}$ is the colour of the surface patch, $\mathbf{C} = (\mathcal{C}_r, \mathcal{C}_g, \mathcal{C}_b)$, in the sense we discussed earlier. Let us denote the scalar product $(\mathbf{L^k} \cdot \mathbf{n})$ by s^k, so that we can form a *"shading"* *vector* $\mathbf{S} = (s^1, s^2, s^3)^T = [L]\mathbf{n}$, which shows the dependence of image intensities on the strength and direction of illumination. If we stack the pixel vectors row-wise to obtain the *intensity matrix* $[I] = (\mathbf{I}^1, \mathbf{I}^2, \mathbf{I}^3)^T$, we can write:

$$[I] = \begin{bmatrix} I_1^1 & I_2^1 & I_3^1 \\ I_1^2 & I_2^2 & I_3^2 \\ I_1^3 & I_2^3 & I_3^3 \end{bmatrix} = \begin{bmatrix} s^1\mathcal{C}_r & s^1\mathcal{C}_g & s^1\mathcal{C}_b \\ s^2\mathcal{C}_r & s^2\mathcal{C}_g & s^2\mathcal{C}_b \\ s^3\mathcal{C}_r & s^3\mathcal{C}_g & s^3\mathcal{C}_b \end{bmatrix} = \mathbf{S}\mathbf{C}^T \tag{4}$$

Note that while the kth row of matrix $[I]$ is the kth input pixel $\mathbf{I^k}$, its lth column is the intensity vector $\mathbf{I_l}$ for the lth colour band. Equation (4) describes the intensity matrix in the ideal, noiseless case. However, in real data there is always a certain degree of noise, and the *observed* intensity matrix differs from the "ideal" matrix. We want to find such estimates of $\mathbf{n}$ and $\mathbf{C}$ for which the error between their matrix product on the right hand side of equation (4) and the observed matrix $[I]$ is minimal. This can be done by applying the Least Square Error technique, which results in the desired estimates for colour and shading vectors being the principal eigenvectors of matrices $[I]^T[I]$ and $[I][I]^T$ respectively. Intuitively these estimates can be interpreted as follows. For a Lambertian surface patch the three colour pixels corresponding to it are collinear in the RGB space, and differ only by a scalar factor, the shading of the patch under a particular illumination. Introduced errors may disturb the collinearity, therefore we use Principal Component Analysis to find their principal direction. The principal direction gives us the chromaticity[3] of the body colour.

By determining the chromaticity of the body colour, and projecting all input pixels on the principal colour line, we reduce the problem to the grey-scale case, where the projections play the role of grey-scale intensities. Applying the GPS algorithm to this intensity vector, we get the optimal estimation of the surface gradient and the norm of $\mathbf{C}$.

This method can be easily extended to more than 3 input images, say, M images. We estimate the surface chromaticity using all M colour pixels by finding the principal eigenvector of the corresponding colour correlation matrix. Using this chromaticity, we produce M "intensities" by projecting all pixels on the principal colour line. These intensities are then used as the input for the appropriate GPS method.

[3] The chromaticity is the unit vector, collinear with body colour in the colour space

5 The problem of highlights and shadows

If the algorithm of Section 4.2 is applied to a triplet which has a highlight or a shadow, the recovery will be affected: the recovered colour will appear different than it would in the absence of highlights and shadows, and the recovered normal will lean more towards the light source which produced the highlight, or away from the source which produced a shadow.

The method proposed by Coleman and Jain [3] uses 4 images of the same surface to detect highlights in the absence of shadows. This is done by comparing the albedos recovered from all four possible triplets of pixels, under the assumption that the specularity regions do not intersect. If the albedos differ significantly, it should be due to a highlight. The three largest albedos must be affected by the highlight, therefore the triplet producing the smallest albedo contains only the Lambertian component, and is used for recovery.

However, many natural surfaces produce cast and self-shadows when illuminated by directional light. The variation of the above method, proposed by Solomon and Ikeuchi [16], takes self-shadows into consideration. Solomon and Ikeuchi considered a unit hemisphere of surface normals, illuminated by all four illuminants at once. The hemisphere was naturally divided into regions: those illuminated by all four illuminants, by three illuminants, and only by two illuminants. Different strategies were suggested for detecting specularities and local surface recovery for each of the regions. Their algorithm effectively used self-shadows as an aid to local gradient recovery.

This method has, however, several shortfalls. First of all, it excludes cast shadows. Cast shadows will be interpreted by the algorithm as self-shadows. The gradient of a self-shadowed facet is restricted, whereas the gradient of a cast-shadowed facet has no restrictions at all, and using erroneous restrictions leads to an incorrect gradient reconstruction. The second problem which prevents us from using this method is that there is no indication as to how to detect shadows. In real images, shadows are rarely perfectly black, so they cannot be identified easily by simple thresholding. The shadow value depends on the strength and direction of illumination, so there is always a range of shadow values even for surfaces with spatially uniform albedo. For surfaces with varying albedo the uncertainty in shadow detection is even bigger: shadow values in bright areas may be brighter than non-shadows in dark areas.

Both highlights and shadows are unexpected changes in pixel intensities, the only difference being that highlights elevate the affected value, and shadows lower it. Let us consider a surface patch with albedo ρ and normal $\mathbf{n}$. Under a 4-source image configuration (and for grey images) this patch gives rise to four pixel values, I^k, $k = 1, \ldots, 4$, one in each image:

$$\begin{aligned} \mathbf{I}^{\{1\}} &= (I^2, I^3, I^4)^T \qquad \mathbf{I}^{\{2\}} = (I^3, I^4, I^1)^T \\ \mathbf{I}^{\{3\}} &= (I^4, I^1, I^2)^T \qquad \mathbf{I}^{\{4\}} = (I^1, I^2, I^3)^T \end{aligned} \tag{5}$$

Each triplet $\mathbf{I}^{\{u\}}$ has illumination matrix $[L]^{\{u\}}$ associated with it, which is made up from the corresponding illumination vectors. Using the algorithm from Section 4.2, we obtain four recovered vectors $\mathbf{T}^{\{u\}} = \left[[L]^{\{u\}}\right]^{-1} \mathbf{I}^{\{u\}}$. If all values are non-shadowed Lambertian, the recovered vectors should be identical. Suppose, however, that one of the values is not a non-shadowed Lambertian (without loss of generality, let it be I^4). Then:

$$I^4 = \rho(\mathbf{n} \cdot \mathbf{L}^4) + \epsilon$$

If I^4 is a highlight, ϵ is positive. If I^4 is a self-shadow, the scalar product $(\mathbf{n} \cdot \mathbf{L}^4)$ is negative, and ϵ is positive, because $I^4 \geq 0$. If I^4 is a cast shadow, then ϵ is negative. Let us now compute $\mathbf{T}^{\{u\}}$:

$$\mathbf{T}^{\{1\}} \equiv \left[[L]^{\{1\}}\right]^{-1} \mathbf{I}^{\{1\}} = \rho\mathbf{n} + \epsilon \left[[L]^{\{1\}}\right]^{-1} (0,0,1)^T$$

$$\mathbf{T}^{\{2\}} \equiv \left[[L]^{\{2\}}\right]^{-1} \mathbf{I}^{\{2\}} = \rho\mathbf{n} + \epsilon \left[[L]^{\{2\}}\right]^{-1} (0,1,0)^T$$

$$\mathbf{T}^{\{3\}} \equiv \left[[L]^{\{3\}}\right]^{-1} \mathbf{I}^{\{3\}} = \rho\mathbf{n} + \epsilon \left[[L]^{\{3\}}\right]^{-1} (1,0,0)^T$$

$$\mathbf{T}^{\{4\}} \equiv \left[[L]^{\{4\}}\right]^{-1} \mathbf{I}^{\{4\}} = \rho\mathbf{n}$$

Comparing the various $\mathbf{T}^{\{u\}}$, we can see that the difference in the recovered vectors depends only on the value and the sign of ϵ. For fixed ϵ the recovered albedos will exhibit the same variance either they were affected by shadows or by highlights. Therefore one can not distinguish between highlights and shadows using variance in the recovered albedos as the only cue. Nevertheless, since highlights appear under rather specific circumstances, we have several other cues for separating shadows from highlights.

6 Detection of highlights and shadows

We assume that shadows and highlights can be treated as disturbances in non-shadowed Lambertian photometric quadruples. We also assume that surface reflectance can be modelled adequately by the dichromatic reflection model.

6.1 Identifying quadruples unaffected by shadows and highlights

In the 3-dimensional world any 4 vectors are linearly dependent. So any 4 illumination vectors $\mathbf{L}^k$ are also linearly dependent, i.e. there are real coefficients a_k, $k = 1, \ldots, 4$, such that:

$$a_1\mathbf{L}^1 + a_2\mathbf{L}^2 + a_3\mathbf{L}^3 + a_4\mathbf{L}^4 = 0$$

If we multiply both sides of this vector equation by a local surface normal $\mathbf{n}$ and albedo ρ, we obtain:

$$a_1\rho(\mathbf{L}^1 \cdot \mathbf{n}) + a_2\rho(\mathbf{L}^2 \cdot \mathbf{n}) + a_3\rho(\mathbf{L}^3 \cdot \mathbf{n}) + a_4\rho(\mathbf{L}^4 \cdot \mathbf{n}) = 0$$

This is equivalent to:

$$a_1 I^1 + a_2 I^2 + a_3 I^3 + a_4 I^4 = 0 \qquad (6)$$

In other words, linear dependence of the illuminant vectors leads to the same linear equation for the corresponding pixel intensities, if the Lambertian assumption holds.

We can rewrite equation (6) in vector form:

$$\mathbf{a} \cdot \mathbf{I} = 0$$

where $\mathbf{a} \equiv (a_1, a_2, a_3, a_4)^T$. This means that any non-shadowed Lambertian quadruple of pixel intensities is perpendicular to $\mathbf{a}$, i.e. for a specific illumination configuration all non-shadowed Lambertian quadruples form a hyperplane in the 4-dimensional intensity space, no matter what albedo and normal the corresponding surface facets have.

The hyperplane is defined by the coefficients vector $\mathbf{a}$, which could be computed directly from the known illumination vectors, or, in the case when the illumination configuration is unknown, by using the Least Square Errors approximation, provided, of course, that the number of quadruples affected by shadows and highlights is comparatively small.

Note that the value $\mathbf{a} \cdot \mathbf{I}$ is exactly (up to the sign) equal to the error ϵ we saw in the previous section. Therefore we propose a simpler version of the Coleman-Jain method, where we detect large values of ϵ directly from intensity values rather than from large variance in the recovered albedos.

For colour images the same relationship is true for each colour band. However, there could be weaker and stronger colour bands, therefore we compare the values of projections of the input colour pixels along the principal colour line rather than use the actual values in each colour band separately.

6.2 Highlight detection using colour

One of the important cues for distinction between highlights and shadows is the colour of the input pixels.

According to the dichromatic reflection theory [9], a highlighted pixel $\mathbf{I}$ can be decomposed into the sum of the matte (body) component $\mathbf{C}$ and specular (surface) component $\mathbf{C}_S$:

$$\mathbf{I} = \mathbf{C} + \mathbf{C}_S = m\mathbf{c} + m_S\mathbf{w} \tag{7}$$

where $\mathbf{c}$ is the chromaticity of the body colour, $\mathbf{w}$ is the chromaticity of the illuminant, and m and m_S are geometric scaling factors, or *strengths* of the corresponding components. If we consider highlights as deviations from the Lambertian law, we can use the specular strength m_S to measure the "specular" error.

We estimate the surface chromaticity $\mathbf{c}$ using the method of Section 4.2. We can uniquely decompose $\mathbf{I}$ along $\mathbf{c}$ and $\mathbf{w}$, obtaining:

$$m_S = \frac{(\mathbf{I} \cdot \mathbf{w}) - (\mathbf{I} \cdot \mathbf{c})(\mathbf{c} \cdot \mathbf{w})}{1 - (\mathbf{c} \cdot \mathbf{w})^2} \tag{8}$$

We can detect highlights by an appropriate thresholding of m_S: having chosen a threshold T_C, we declare that the brightest pixel is a highlight if m_S calculated by (8) exceeds T_C. Note that though we attempt to decompose a highlighted pixel, we do not use the decomposition for recovery as in [15]; we use it only to detect deviation from body colour.

6.3 An alternative solution

The method described above works rather well when the colour of the surface and the incident light are distinctly different. However, if the difference between chromaticities of the body colour and the incident light is small, then the variation in pixel colour due to highlighting becomes indistinguishable from the variation due to the imaging process. In terms of equation (8) this means that the denominator is getting close to zero, and the spectral difference algorithm for highlight detection is not applicable.

Then we must use an alternative solution. We define for each light source with direction $\mathbf{l}^k$ its specular direction $\mathbf{v}^k$:

$$\mathbf{v}^k = \frac{\mathbf{l}^k - \mathbf{z}}{|\mathbf{l}^k - \mathbf{z}|}$$

where $\mathbf{z}$ is the unit vector directed along the z axis. This is the direction with which the surface normal should coincide in order to reflect light specularly towards the camera. Having identified a quadruple as problematic, we exclude the brightest pixel (as a possible highlight) from consideration, and reconstruct the surface colour and local gradient using the three darkest pixels. It can be shown (see Appendix B) that if there is a perfect shadow in the triplet, then the recovered normal will be forced onto the shadowline of the corresponding source. Brightening the shadow lifts the recovered normal towards the source. Therefore if the reconstructed normal direction is close to the corresponding specular direction and is sufficiently far from the shadowlines of the other light sources, we conclude that the brightest pixel is indeed a highlight. We measure the closeness of the recovered normal and the specular direction by their dot product, and make decision by thresholding its value.

Notice, however, that a highlighted pixel is not always the brightest in the quadruple (if, for example, the intensity of the source which produces the highlight is lower than one of the other sources). Therefore to determine potentially highlighted or shadow pixels, we should normalise their values by dividing them by the corresponding illumination strengths. However, since such normalisation amplifies errors in dimly lit images, we cannot use the normalised values for reconstruction purposes. We use the original unnormalised values for that.

6.4 Which method to use?

The colour difference method is more reliable than the alternative method, so it should be given preference in situations where it is applicable. To decide which method to use, we need to compare the chromaticities of the body colour and the incident light. We determine whether they are close or not by calculating and appropriately thresholding the square of chromatic distance $D^2 \equiv 1 - (\mathbf{c} \cdot \mathbf{w})^2$. The threshold T_D depends, in particular, on the level of camera noise and is chosen empirically over several surfaces with different chromatic characteristics.

If D^2 exceeds T_D, the colour of the surface and the illumination are sufficiently different, and we apply the spectral difference method to highlight detection, otherwise the alternative method should be applied.

6.5 Summary of the algorithm

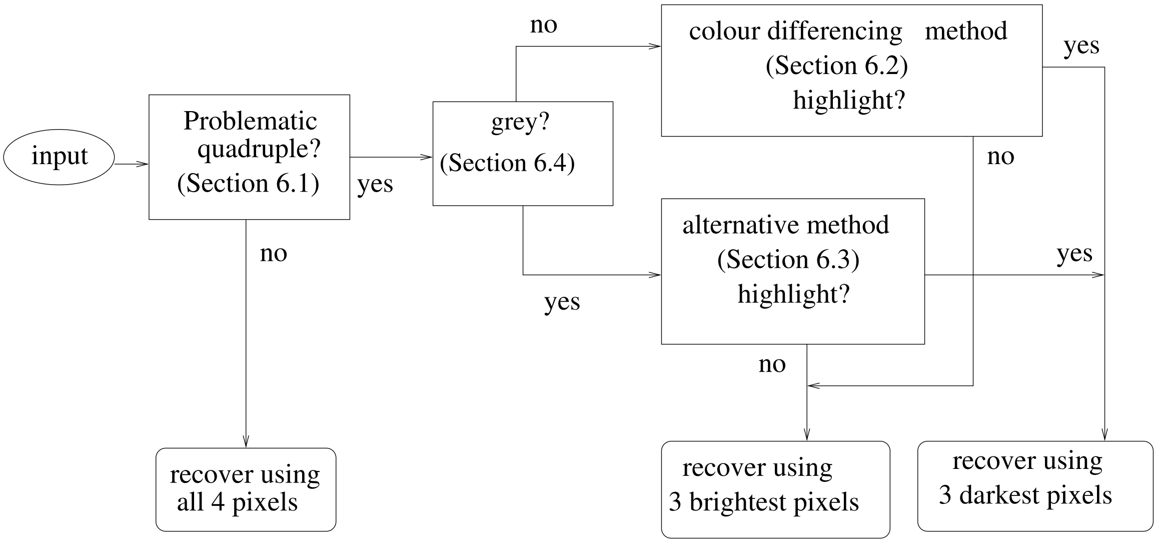
We present now the full algorithm for 4 light sources (see Figure 1).

1. Construct the input matrix $[I]$ of size 4×3 by stacking the pixel values $\mathbf{I^k}$ as rows.

2. Compute the colour correlation matrix $[Q] = [I]^T[I]$. Compute its normalised principal eigenvector $\mathbf{c}$. Project all 4 colours that correspond to the same pixel along the direction of $\mathbf{c}$, and thus define a single grey value for each pixel, $\mathbf{J} = [I]\mathbf{c}$.

3. For each quadruple of pixel values compute the "Lambertian error" $\epsilon_L \equiv (\mathbf{J} \cdot \mathbf{a})^2$ to decide whether it contains a highlight or a shadow.

   If $\epsilon_L \leq T_L$:



input
Problematic quadruple? (Section 6.1)
yes
no
grey? (Section 6.4)
no
yes
colour differencing method (Section 6.2) highlight?
yes
no
alternative method (Section 6.3) highlight?
yes
no
recover using all 4 pixels
recover using 3 brightest pixels
recover using 3 darkest pixels

Figure 1: Principal scheme of the algorithm

3.1. Apply the photometric stereo algorithm for a set of 4 grey images using all available information, as described in Section 4.1, to derive the local gradient vector and surface albedo ρ.

If $\epsilon_L > T_L$:

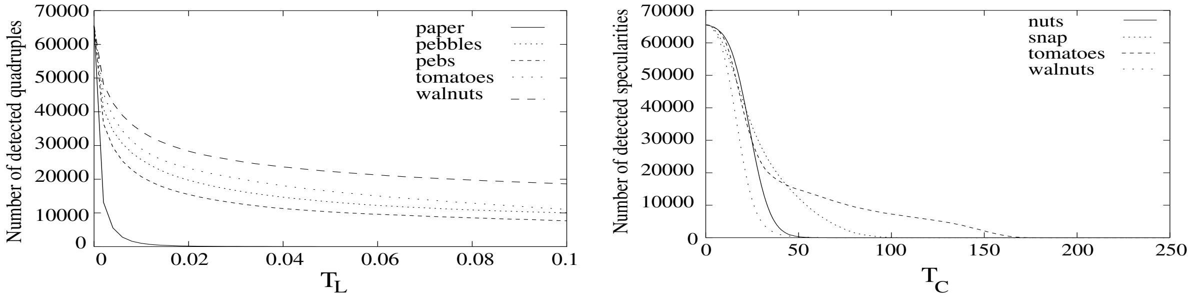
3.2. Measure the chromatic difference D^2 between the surface colour and the colour of the illumination.

If $D^2 \geq T_D$: Apply the method of Section 6.2 to determine whether the quadruple contains a highlight or not.

If $D^2 < T_D$: Apply the method of Section 6.3 to determine whether the quadruple contains a highlight or not.

3.2.1. If the quadruple contains a highlight, use the 3 darkest components of vector $\mathbf{J}$ to recover the local surface normal and the albedo according to the algorithm of Section 4.1.

3.2.2. If the quadruple does not contain a highlight, conclude that it must contain a shadow, and use the 3 brightest components



(a) (b)

Figure 2: Number of detected pixels as a function of the threshold: (a) for "Lambertian" error (b) for spectral difference

of vector $\mathbf{J}$ to recover the local surface normal and the albedo according to the algorithm of Section 4.1.

4. In all cases the local colour is recovered as $\rho\mathbf{c}$

Note that the algorithm can be simplified: we can *always* perform a recovery using only the three brightest pixels except when there are highlights, thus skipping the stage of detecting the "problematic" quadruples.

6.6 Threshold choice

An important design issue is the choice of the threshold values we use. Thresholds T_L, T_C and T_D must be chosen empirically according to guidelines given in Section 7.1. The value of threshold T_N can be determined analytically.

Thresholds T_L, T_C and T_D: Experimental evaluation showed that thresholds T_L and T_C depend on the level of noise in the images. This dependence is demonstrated in the experimental section with synthetic data. Further, it turned out that threshold T_D depends on the choice of threshold T_C and the level of noise [1]. To demonstrate the effect of these thresholds on the algorithm, we plot here the number of picked problematic quadruples as a function of T_L for several sets of real data (Fig. 2a). The values were divided by 255^2 to avoid dealing with large numbers. For real data the ground truth is not known, and so we can not tell whether the detected quadruples are indeed problematic.

However, for all surfaces the number of detected points drops sharply until $T_L \approx 0.02 - 0.04$, and then stabilises. Note that the algorithm is fairly robust to the choice of T_L as long as T_L is larger than 0.02. In the results presented in Section 7.2 we used $T_L = 0.03$. In a similar way, in Fig.2b we plot the number of pixels, classified as highlights, as a function of threshold T_C. For our experiments in Section 7.2 we chose $T_C = 50$ and $T_D = 0.01$.

Threshold T_N : This threshold is used to compare the direction of the recovered normal with the specular direction. We label a pixel as highlighted if the normal reconstructed from the other three pixels is close to the corresponding specular direction. We measure this closeness using the dot product of the normal and the vector of the specular direction. Thresholding the dot product defines the width of a possible specularity region. We need to exclude the possibility of misclassification which can be of two kinds:

1. A highlighted patch misclassified as shadowed. This happens when the specularity cone is too narrow.
2. A shadowed patch misclassified as highlighted, i.e. the cone is too wide.

To achieve an adequate trade-off, we need to consider several factors which can affect the choice:

1. The overlapping of specularity cones associated with different lighting directions.
2. The closeness of the possible specularity region and the shadow lines from other illumination sources.
3. The effect of brightening of shadows.

Under the assumption that all shadows are black, the thresholds which separate a specularity region from the shadow lines of the other sources take care of both cast and self shadows: as already mentioned, a perfectly black shadow forces the recovered normal to fall on the corresponding shadow line, lifting it for the case of self-shadows, and lowering it for cast shadows (see Appendix B). Therefore for an arbitrary illumination configuration the width of a specular cone will be defined by the minimum distance between a specular direction and the shadow lines of the rest of the illuminants. It is necessary that none of the specular directions lies in the shadow of another source. The sufficient condition for such an illumination is discussed in Appendix A.

The perfectly black shadows do not present much of a problem anyway, since they could be detected by merely thresholding the intensity value. The biggest problem is the detection of brightened shadows, whose intensity could be quite high. This effect may appear due to secondary illumination, from the environment and/or neighbouring parts of the (non-convex) surface. The degree of secondary illumination depends among other things on the illumination direction, roughness and surface albedo etc., and it can not be easily modelled. Brightening of the shadows elevates the recovered vector, moving it away from the shadow line and towards the specular direction. Therefore the threshold value T_N should "pad" the specularity region securely from such elevated recovered vectors. Appendix C suggests a strategy for choosing the threshold for an arbitrary acceptable set-up.

In Appendix D we consider the thresholding values for a cross-like illumination set-up, where the 4 illumination sources have the same elevation angle [4], θ, and are positioned as if at the corners of a square. According to this calculation, in this specific case the threshold value T_N must be chosen to be:

$$T_N = max\left[\frac{1+T_1}{2}, \sqrt{\frac{1+\sin\theta}{2}}\right].$$

7 Experimental results

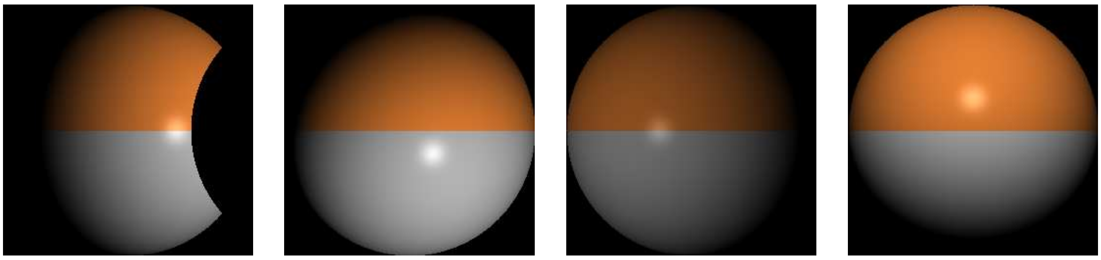


Figure 3: The synthetic input images with $A = 0.3$ and $n = 200$

[4] The elevation angle of the illumination is the angle between the illumination direction and the horizon.

7.1 Experiments with synthetic images

Synthetic images are used for evaluation of the algorithm, showing its advantages and limitations. We have rendered a two-coloured sphere under 4 illuminants (Fig. 3). We used the Phong reflectance model [14] and rendering with the same coefficients for both halves. The top half of a sphere has orange colour ($\mathbf{C}_{top} = (230, 128, 50)^T$) and the bottom one is grey ($\mathbf{C}_{bottom} = (180, 180, 180)^T$), so we can test both variations of the algorithm. In one of the images we added a "cast shadow". We assume the illumination is white. The illumination matrix is:

$$\begin{bmatrix} 0 & \cos\frac{\pi}{4} & \sin\frac{\pi}{4} \\ \cos\frac{\pi}{4}\cos\frac{\pi}{3} & \sin\frac{\pi}{4}\cos\frac{\pi}{3} & \sin\frac{\pi}{3} \\ 0 & -0.6\cos\frac{\pi}{3} & 0.6\sin\frac{\pi}{3} \\ -\cos\frac{\pi}{3} & 0 & \sin\frac{\pi}{3} \end{bmatrix}$$

so that we have illuminants with different strengths, different elevation angles, and they are not arranged in a cross-like configuration. The Phong model adds a specular term in the form of an $A\cos^n\Phi$ component to the Lambertian model, where Φ is the angle between the surface normal and the specular direction. We performed experiments with $A = 0.3$ and 0.15, and $n = 200$, 100 and 50. In addition, we assumed a component of secondary illumination from the zenith, of strength f times the strength of the main illumination in each image, and additive Gaussian noise of standard deviation σ in all images.

Problematic quadruples detection and thresholds The first series of experiments is intended to investigate the role of a threshold in the detection of problematic quadruples (section 6.1). For given model parameters (i.e. the reflectance parameters A, n, level of noise and secondary illumination) the images of the sphere were rendered. Then for each threshold value we calculated the proportion of undetected problematic quadruples, and the proportion of misclassified non-problematic quadruples. Figure 4 shows typical results for a noiseless and noisy cases. One can clearly see the trade-off between the false positives and false negatives even in the noiseless case. For noisy conditions the proportion

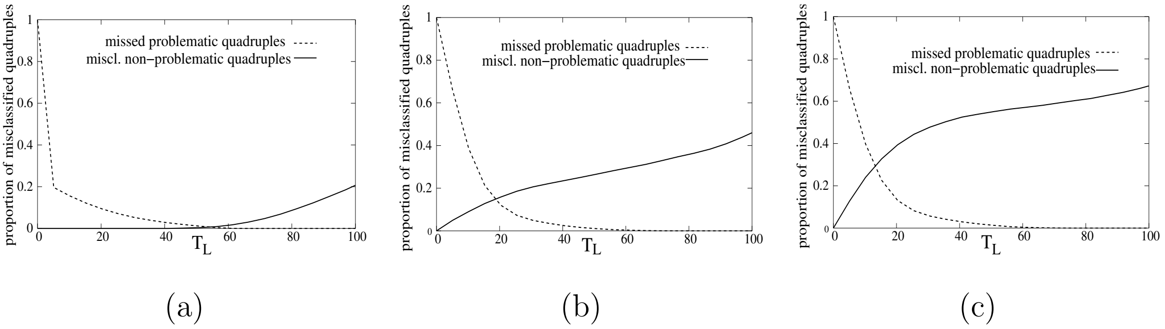


Figure 4: Proportion of undetected problematic quadruples (solid line) and misclassified non-problematic quadruples (dashed line) as a function of a threshold value T_L. (a) for noiseless Lambertian case (b) for Lambertian case with $f = 0.1$, and $\sigma = 10$ (c) for the case with reflectance parameters $A = 0.3$, $n = 200$, and noise parameters $f = 0.1$, $\sigma = 10$

of misclassified quadruples of both kinds grows for any pre-set threshold.

Assessment of colour differencing method To assess the performance of highlight detection using colour, we rendered several spheres with different body colour under white light. Fig. 5 presents results for spheres with $\mathbf{C}_1 = 255(1,0,0)^T$ and $\mathbf{C}_2 = 255(1,0.8,0.8)^T$. As before, each time we plotted two lines: one shows the proportion of Lambertian pixels misclassified as highlights (solid line), and the other - the proportion of highlighted pixels misclassified as Lambertian (dashed line) as a function of the value T_C by which we threshold the specularity strength m_S of a pixel. Fig. 5a presents a typical result for the noiseless case. There is a whole range of values T_C between 1 and 10 where we have no errors of either kind even when the body and illumination colours are close. However, in the presence of noise we start misclassifying both highlights and Lambertian pixels. Fig. 5b shows the performance of the highlight detection algorithm on noisy images of the sphere with body colour $\mathbf{C}_1$. Finally, when the angle between the illuminant and the body colours gets more narrow, the classification is almost random (see the two straight lines on Fig. 5c, which shows the results for the sphere with body colour $\mathbf{C}_2$).

Overall performance Both the straightforward linear algorithm and the proposed algorithm were applied to each set of images to evaluate the two algorithms.

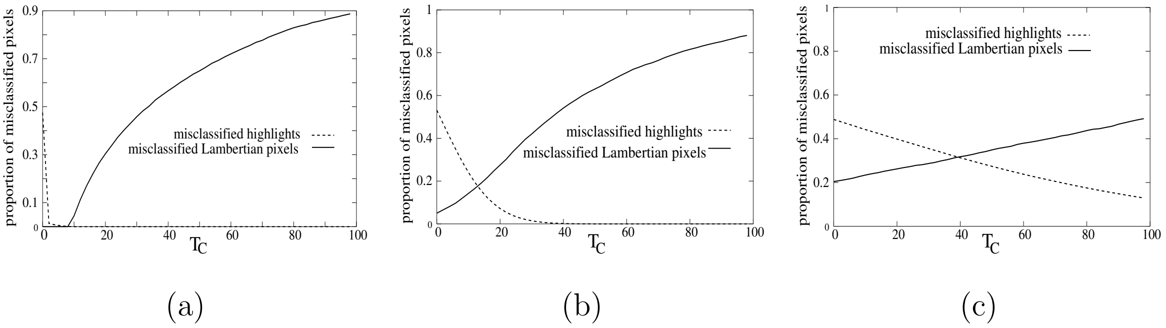


Figure 5: Proportion of misclassified Lambertian pixels (solid line) and misclassified highlights (dashed line) as a function of a threshold value T_C

secondary	standard deviation of noise					
illumination	0	1	2.5	5	7.5	10
0	0.101	0.108	0.115	0.167	0.291	0.427
1/100	0.109	0.115	0.120	0.178	0.301	0.435
1/50	0.117	0.123	0.131	0.201	0.330	0.458
1/30	0.143	0.154	0.169	0.264	0.391	0.498
1/20	0.272	0.284	0.308	0.393	0.480	0.556
1/10	0.565	0.575	0.585	0.615	0.648	0.675

Table 1: Fraction of pixels with erroneously reconstructed colour recovered by the proposed algorithm

We measure the colour error as the length of the vector difference between the original and the recovered colour vectors. The error in normals is measured as $1 - \mathbf{n}_{recovered} \cdot \mathbf{n}$. Figure 6 presents some example results for the case $A = 0.3$ and $n = 200$. The detected errors in the results of the two algorithms have been scaled in the same scale so that the same grey tone in the different panels indicates the same level of error. The recovered surface normals are used to produce an image just for visualisation purposes.

The results of the experiments with $A = 0.3$ and $n = 200$ are presented in tables 1-4. The tables show the proportion of pixels for which the absolute errors in the recovered values exceed some predetermined acceptance value (0.05 for colour, 0.005 for normals). We chose this error measure rather than conventional mean and standard deviation of error distribution because the artefacts in the recovered parameters are localised and rather

secondary	standard deviation of noise					
illumination	0	1	2.5	5	7.5	10
0	0.284	0.285	0.287	0.303	0.371	0.463
1/100	0.289	0.290	0.291	0.311	0.382	0.474
1/50	0.296	0.297	0.300	0.329	0.406	0.497
1/30	0.316	0.318	0.325	0.378	0.459	0.535
1/20	0.411	0.417	0.433	0.482	0.534	0.588
1/10	0.644	0.645	0.646	0.656	0.670	0.688

Table 2: Fraction of pixels with erroneously reconstructed colour recovered without highlight and shadow correction

secondary	standard deviation of noise					
illumination	0	1	2.5	5	7.5	10
0	0.070	0.080	0.083	0.095	0.127	0.189
1/100	0.078	0.084	0.087	0.098	0.128	0.193
1/50	0.079	0.087	0.088	0.098	0.131	0.191
1/30	0.080	0.088	0.090	0.100	0.133	0.195
1/20	0.083	0.091	0.093	0.103	0.137	0.202
1/10	0.093	0.099	0.102	0.119	0.171	0.242

Table 3: Fraction of pixels with erroneously reconstructed normals recovered by the proposed algorithm

different in nature, and therefore their error distributions are far from Gaussian. If, for example, highlights are not detected correctly, we get big errors in a small area, but elsewhere the algorithm might work reasonably well.

The errors in the results of the proposed algorithm which are due to strong secondary illumination are mostly concentrated in the grey part of the sphere, where the areas shadowed in more than one image are sometimes falsely classified as highlights. Secondary illumination also leads to larger overall errors because the actual illumination differs from the one given by the illumination matrix (this is also a problem for the straightforward

	standard deviation of noise					
	0	1	2.5	5	7.5	10
0	0.260	0.261	0.262	0.266	0.274	0.294
1/100	0.261	0.261	0.262	0.267	0.274	0.294
1/50	0.262	0.262	0.263	0.267	0.276	0.295
1/30	0.264	0.264	0.265	0.267	0.277	0.298
1/20	0.266	0.266	0.267	0.269	0.278	0.303
1/10	0.272	0.272	0.273	0.277	0.298	0.334

Table 4: Fraction of pixels with erroneously reconstructed normals recovered without highlight and shadow correction

algorithm). Errors due to high levels of noise also induce some misclassification. The results of the series of experiments with different parameters of the Phong reflectance model are similar to these: predictably, they worsen slightly for more diffuse surfaces when the width of the specularity regions grows. They also get slightly worse if the strength of the specular component gets higher. For example, for the zero noise and zero secondary illumination case, the fraction of the pixels with wrongly recovered colour instead of being 0.101 is 0.110 ($A = 0.3$, $n = 50$), 0.123 ($A = 0.3$, $n = 100$), 0.115 ($A = 0.15$, $n = 100$), and 0.173 ($A = 0.15$, $n = 50$). All other recorded errors for all cases remain in similar levels for those presented in the tables.

7.2 Experiments with real images

We have applied the algorithm described above to a number of test surfaces with a variety of reflectance properties. Here we present just a small selection of them in order to demonstrate various aspects of the algorithm and its limitations: *tomatoes*, *walnuts*, *pebbles1*, and *pebbles2*. Fig. 7 presents the input images, one from each set of four test images. Images were taken by Kodak DC 290, distance from camera to surface 1m, illumination configuration cross-like with elevation angle 60 degrees. These images have been chosen because they demonstrate different aspects of the algorithm when we have shadows and

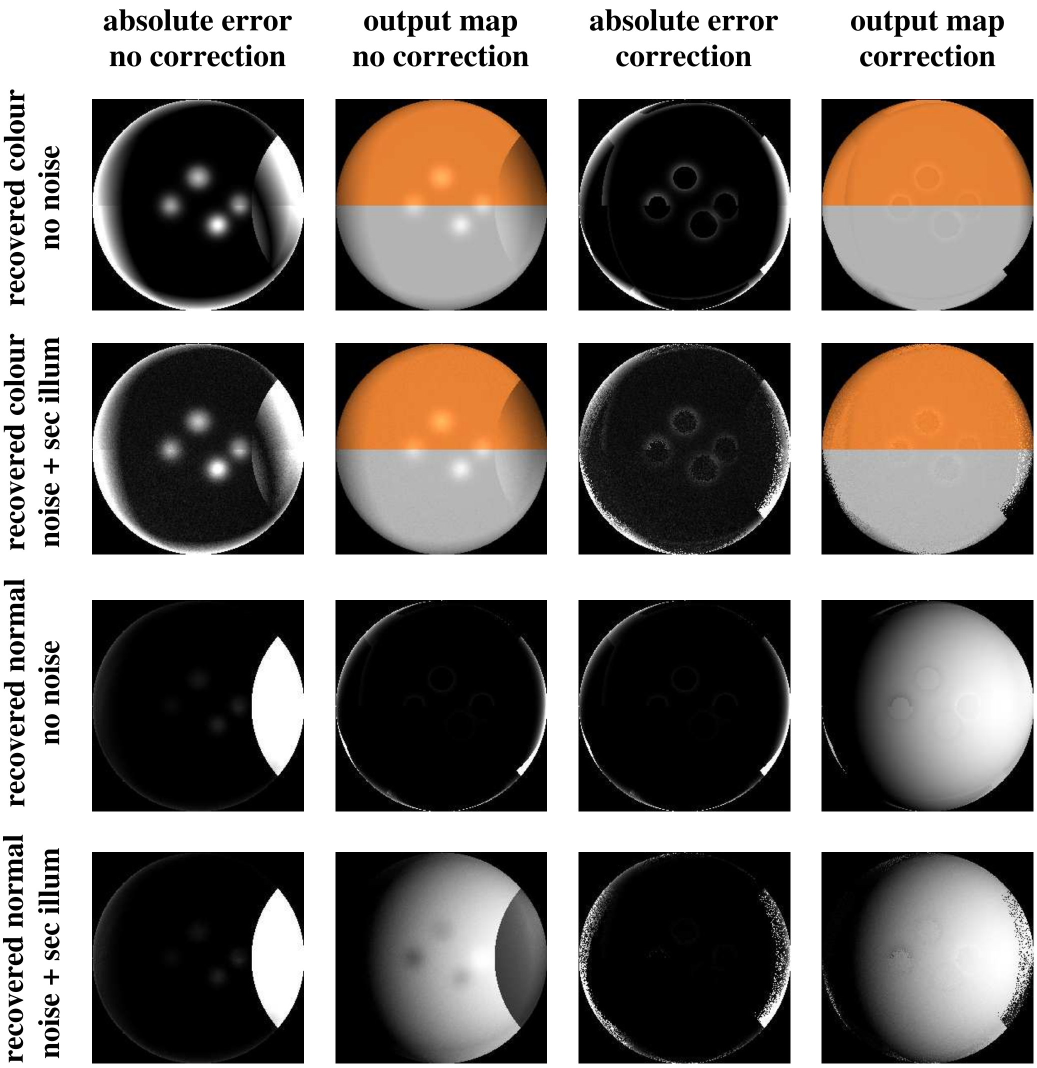


Figure 6: Results with and without shadow and highlight corrections. The recovered normals are presented rendered with the first illumination direction of Figure 3.

highlights. For example, the tomato images have very high levels of secondary illumination. Both tomatoes and walnuts have colour quite different from grey, whereas both sets with pebbles are grey. In addition all these images have several parts that are shadowed in more than one of the captured images.

Figs. 8 and 9 present the results obtained by the CPS algorithms with (upper row) and without (lower row) highlights and shadow correction. Fig. 8 shows the recovered colour maps for all surfaces with and without correction. Fig. 9 shows the reconstructed surfaces rendered under one of the original illuminants. Note from Fig. 5 that in the presence of noise it is impossible to define a threshold that allows us to detect all highlight and

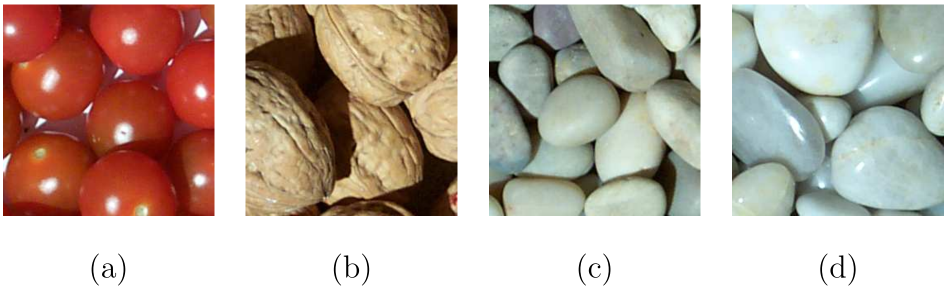


Figure 7: Input images, one from each set of four images: (a) *tomatoes*, (b) *walnuts*, (c) *pebbles1*, (d) *pebbles2*

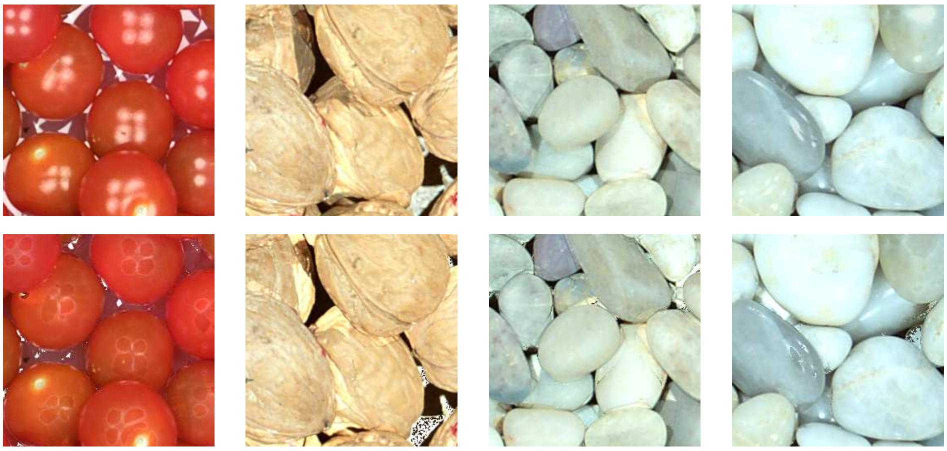


Figure 8: Colour maps, recovered with and without correction

shadow pixels without errors. Due to this fact we observe the rings in the *tomatoes* images shown in Figs. 8 and 9. Although the results of both algorithms have wrongly recovered parts, the proposed algorithm copes much better with highlights and shadows than the ordinary CPS. In addition these results show the places where the proposed algorithm fails, as expected.

8 Conclusions

Shadows and highlights in the input images pose a problem for surface reconstruction. We propose a modification of a well-known photometric stereo algorithm, which uses 4 images to detected highlights and shadows in the input images, and, excluding them from the recovery process, allows one to obtain more reliable estimates of surface parameters. Using pixel-wise estimates of colour, we are able to detect highlights locally. Then the

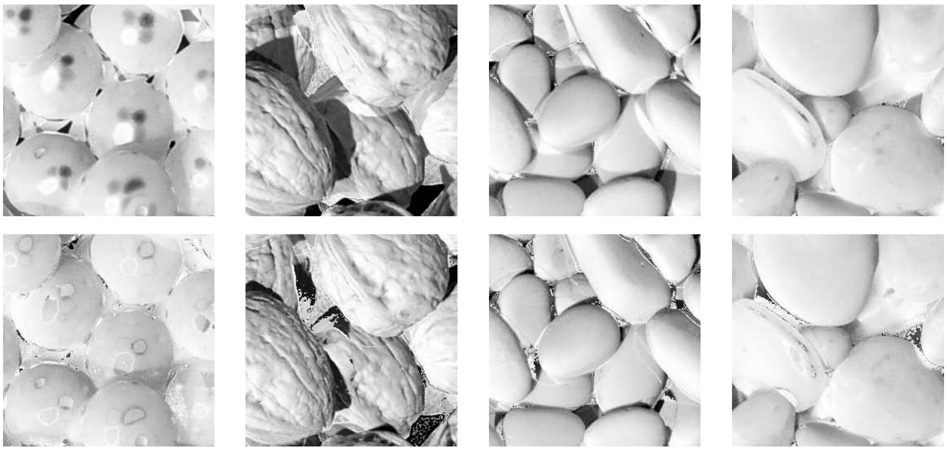


Figure 9: Rendered surfaces, recovered with and without correction

shadows can be detected as disturbances of linearity on the input quadruples of pixels.
In the greyscale case, when there are no spectral cues for highlight detection, we resolve to comparing the recovered normals with shadowlines and specular directions. Comparing the recovered normals with the shadowlines can be considered as a variation of the thresholding process. However as it uses both the local surface albedo and the illumination strength in the process of recovery, it does not rely on a global shadow thresholding value.

The performance of the proposed algorithm may be improved if one uses some post-processing technique to reduce "pepper noise" which appears when we apply thresholding on various stages of the algorithm. The method fails in the greyscale case for pixels which are shadowed in more than one images and are affected by secondary illumination. To deal with misclassifications of this kind, one may detect multiple shadows in a quadruple, using an already derived local shadowing threshold.

A Sufficient condition for acceptable illumination configuration

To be able to use the closeness of a recovered normal to a specular direction for highlight detection in an arbitrary illumination configuration, we should make sure, first, that no three of the illumination vectors lie in the same plane, and second, that no specular direction lies in the shadow of any other illuminant.

Consider, for example, illumination and specular directions of the ith illumination source:

$$\mathbf{l}^i = (\cos\phi_i \cos\theta_i, \sin\phi_i \cos\theta_i, \sin\theta_i)$$

where θ_i the the elevation angle of the ith source, and ϕ_i is its tilt (azimuth) angle. Then the corresponding specular direction is:

$$\mathbf{v}^i = \frac{\mathbf{l}^i - \mathbf{z}}{|\mathbf{l}^i - \mathbf{z}|} = \frac{(\cos\phi_i \cos\theta_i, \sin\phi_i \cos\theta_i, 1 + \sin\theta_i)}{\sqrt{(\cos^2\theta_i^2 + 1 + 2\sin\theta_i + \sin^2\theta_i)}} = \frac{(\cos\phi_i \cos\theta_i, \sin\phi_i \cos\theta_i, 1 + \sin\theta_i)}{\sqrt{2(1 + \sin\theta)}}$$

For any i and j we want the scalar product $\mathbf{l}^i \cdot \mathbf{v}^j > 0$, that is, no specular direction lies in the shadow of any other source:

$$\mathbf{l}^i \cdot \mathbf{v}^j = \frac{1}{\sqrt{2(1 + \sin\theta_j)}} \times$$

$$(\cos\phi_i \cos\theta_i \cos\phi_j \cos\theta_j + \sin\phi_i \cos\theta_i \sin\phi_j \cos\theta_j + \sin\theta_i (1 + \sin\theta_j)) =$$

$$\frac{1}{\sqrt{2(1 + \sin\theta_j)}} (\cos\theta_i \cos\theta_j \cos(\phi_i - \phi_j) + \sin\theta_i (1 + \sin\theta_j)) > 0$$

Since the denominator is positive and all elevation angles are in $(0, \pi/2]$, we can rewrite the inequality as:

$$\sin\theta_i (1 + \sin\theta_j) > -\cos\theta_i \cos\theta_j \cos(\phi_i - \phi_j)$$

Provided that none of the two sources is from zenith (in which case the inequality is satisfied anyway), we can divide both sides by the cosines:

$$\tan\theta_i \frac{1 + \sin\theta_j}{\cos\theta_j} > -\cos(\phi_i - \phi_j)$$

The inequality $\mathbf{l}^j \cdot \mathbf{v}^i > 0$ should be treated in the same way.

Notice that:

$$\tan\theta_i \frac{1 + \sin\theta_j}{\cos\theta_j} > \tan\theta_i \tan\theta_j \quad \text{and} \quad \frac{1 + \sin\theta_i}{\cos\theta_i} \tan\theta_j > \tan\theta_i \tan\theta_j$$

Therefore the following inequality is sufficient:

$$\tan\theta_i \tan\theta_j > -\cos(\phi_i - \phi_j)$$

This inequality can be tightened for specific illumination set-ups.

B Recovered vector and the shadow line

We shall show that for any illumination configuration if there is a perfectly black shadow(s) in a pixel triplet, then the vector recovered by the PS method falls on the shadow line(s) of the corresponding source(s).

Consider an illumination configuration which consists of 3 lights with directions $\mathbf{L^1}$, $\mathbf{L^2}$, and $\mathbf{L^3}$, and has an illumination matrix $[L]$ associated with it. Let one of the pixels produced under this configuration (say I^1) be perfectly black: $I^1 = 0$. Then applying the linear GPS algorithm we obtain the recovered vector $\mathbf{T}$:

$$\mathbf{T} = [L]^{-1}(0, I^2, I^3) = \rho\mathbf{n} - \rho(\mathbf{L^1} \cdot \mathbf{n})[L]^{-1}(1,0,0)^T$$

$[L]^{-1}(1,0,0)^T$ is the first column of matrix $[L]^{-1}$. Let us now multiply both sides of this equation by $\mathbf{L^1}$:

$$(\mathbf{L^1} \cdot \mathbf{T}) = \rho(\mathbf{L^1} \cdot \mathbf{n})\left[1 - (\mathbf{L^1} \cdot [L]^{-1}(1,0,0)^T)\right]$$

By definition of the inverse matrix the term in square brackets is equal to zero, and therefore the recovered vector $\mathbf{T}$ indeed falls on the shadow line of $\mathbf{L^1}$. If there are two perfectly black shadows, the recovered vector falls on the intersection of the corresponding shadow lines.

C The strategy for choosing threshold T_N in an arbitrary acceptable illumination configuration

If we have only one defect in the quadruple (the brightest pixel as a highlight or the darkest pixel as a shadow), then we only have to distinguish between these two cases.

Let us assume that the brightest pixel was obtained under the ith illuminant, and the darkest pixel under the jth illuminant. We want to define a specularity cone around $\mathbf{v}^i$ such that it is still sufficiently far from the shadowline of $\mathbf{l}^j$. Consider angle γ between these directions:

$$\gamma = \arccos(\mathbf{v}^i \cdot \mathbf{l}^j)$$

Then the angle between the $\mathbf{v}^i$ and the closest point on the shadowline is $\pi/2-\gamma$ (consider the plane which contains both $\mathbf{v}^i$ and $\mathbf{l}^j$). The angle, which defines the width of the specularity cone, should not exceed $\pi/2-\gamma$. To make the "padding", necessary for brightened shadows, we use parameter $\tau \in (0,1)$, which depends on the brightness of shadows and may be "tweaked" during calibration. The width of the specularity cone can be defined then as $\beta = \tau(\pi/2-\gamma)$, widening when τ grows and narrowing when it diminishes. The thresholding value $T_N(i,j)$ can then be defined as

$$T_N(i,j) = \cos\beta = \cos(\tau(\pi/2-\gamma)) = \cos(\tau(\pi/2 - \arccos(\mathbf{v}^i \cdot \mathbf{l}^j))) \qquad (9)$$

However, this method is prone to misclassification if there is more than one shadow, so it may be beneficial to try and detect the second shadow (from the second darkest pixel) in a similar way (with, perhaps, smaller τ as this shadow will be brighter). If we have more than one shadow in the quadruple, the method is inapplicable, and recovering surface parameters from the brightest three pixels is probably the best estimate we can get.

D The bounding conditions for T_N in the cross-like configuration

If we measure the closeness of a normal and a specularity direction in terms of their dot product, then thresholding this value is in fact determining a cone of directions which can possibly produce highlights. We want this cone not to intersect the shadow lines of the other illuminants.

Let us suppose that our illuminants are arranged in a cross-like configuration. Let us also suppose that the brightest pixel is obtained under $\mathbf{L}^4$, and we reconstruct the normal using $\mathbf{L}^1$, $\mathbf{L}^2$, and $\mathbf{L}^3$. Let us denote the reconstructed normal by $\mathbf{n} \equiv (n_x, n_y, n_z)^T$.

Possible specular cone and the shadow line of the opposite light source The widest specularity cone which still does not intersect the shadow line of $\mathbf{L}^2$ is such that the cone touches the shadow line, i.e. intersects it at exactly one point. We should also make sure that the cone is outside the shadowed region. To ensure that, we must require that the angle between $\mathbf{v}^4$ and $\mathbf{L}^2$ is less than $\pi/2$. It is easy to see that θ must exceed $\pi/6$: The angle between $\mathbf{v}^4$ and the vertical is exactly half of the angle between $\mathbf{L}^4$ and the vertical. Therefore the angle between $\mathbf{v}^4$ and $\mathbf{L}^2$ can be expressed as:

$$\frac{3}{2}\left(\frac{\pi}{2}-\theta\right) < \frac{\pi}{2} \qquad \Leftrightarrow \qquad \theta > \frac{\pi}{2} - \frac{2}{3}\frac{\pi}{2} = \frac{\pi}{6}$$

To find the threshold T_1 defining such a cone, we should solve a system of equations so that the system has exactly one solution:

$$\begin{cases} \mathbf{n}\cdot\mathbf{v}^4 = T_1 \\ \mathbf{n}\cdot\mathbf{L}^2 = 0 \\ ||\mathbf{n}|| = 1 \end{cases} \tag{10}$$

It is easy to see that $\mathbf{v}_4$ is:

$$\mathbf{v}^4 = \frac{\mathbf{L}^4 - \mathbf{z}}{|\mathbf{L}^4 - \mathbf{z}|} = \frac{(-\cos\theta, 0, -[1+\sin\theta])}{\sqrt{\cos^2\theta + (1+\sin\theta)^2}} = \frac{(-\cos\theta, 0, -[1+\sin\theta])}{\sqrt{2(1+\sin\theta)}} \tag{11}$$

Then we can rewrite (10) as:

$$\begin{cases} -n_x\cos\theta - n_z(1+\sin\theta) = T_1\sqrt{2(1+\sin\theta)} \\ n_x\cos\theta - n_z\sin\theta = 0 \\ n_x^2 + n_y^2 + n_z^2 = 1 \end{cases}$$

Solving the first two equations, we get:

$$n_x = -\frac{T_1\sqrt{2(1+\sin\theta)}}{2\sin\theta+1}\tan\theta \quad n_z = -\frac{T_1\sqrt{2(1+\sin\theta)}}{2\sin\theta+1}$$

Substituting into the third equation we get:

$$\frac{2T_1^2(1+\sin\theta)}{(2\sin\theta+1)^2}(\tan^2\theta+1) + n_y^2 = 1$$

For this equation to have a unique solution the normal vector must be coplanar with vectors $\mathbf{v}^4$ and $\mathbf{L}^2$, i.e. we must require $n_y = 0$. Then:

$$\frac{2T_1^2(1+\sin\theta)}{(2\sin\theta+1)^2}(\tan^2\theta+1) = 1$$

Solving for T_1^2:

$$T_1^2 = \frac{(2\sin\theta+1)^2}{2(1+\sin\theta)}\frac{1}{\tan^2\theta+1} = \frac{(2\sin\theta+1)^2}{2(1+\sin\theta)}\cos^2\theta$$

T_1 should be positive, therefore:

$$T_1 = \frac{(2\sin\theta + 1)}{\sqrt{2(1+\sin\theta)}}\cos\theta \tag{12}$$

Thus if the dot product between a normal and the specular direction exceeds T_1, we are sure that the normal cannot be self-shadowed by the opposite illumination source.

Possible specular cone and the shadow line of the neighbouring light source

In a similar way we can determine the threshold which ensures that we do not pick facets that are self-shadowed under the neighbouring light sources.

For the sake of simplicity let us find this threshold for $\mathbf{L}_1$. It will be exactly the same for $\mathbf{L}_3$ due to the symmetry of the system.

To find the threshold T_2, we use the same approach as before. It is easy to see that $(\mathbf{v}^4 \cdot \mathbf{L}^1)$ is always positive, that is, the specularity cone cannot lie inside the shadowing region. We need to solve the following system of equations in such a way that it has exactly one solution:

$$\begin{cases} \mathbf{n}\cdot\mathbf{v}^4 = T_2 \\ \mathbf{n}\cdot\mathbf{L}^1 = 0 \\ ||\mathbf{n}|| = 1 \end{cases} \tag{13}$$

This system becomes:

$$\begin{cases} -n_x\cos\theta + n_z(1+\sin\theta) = T_2\sqrt{2(1+\sin\theta)} \\ n_y\cos\theta - n_z\sin\theta = 0 \\ n_x^2 + n_y^2 + n_z^2 = 1 \end{cases}$$

Solving the first two equations in terms of n_z, we get:

$$n_x = \frac{n_z(\sin\theta+1) - T_2\sqrt{2(1+\sin\theta)}}{\cos\theta} \qquad n_y = n_z\tan\theta$$

Upon substitution to the third equation we obtain:

$$\frac{\left(n_z(\sin\theta+1) - T_2\sqrt{2(1+\sin\theta)}\right)^2}{\cos^2\theta} + n_z^2\frac{\sin^2\theta}{\cos^2\theta} + n_z^2 = 1$$

$$\left(n_z(\sin\theta+1) - T_2\sqrt{2(1+\sin\theta)}\right)^2 + n_z^2(\sin^2\theta + \cos^2\theta) = \cos^2\theta$$

which leads to:

$$n_z^2\left[1 + (1+\sin\theta)^2\right] - 2n_zT_2\sqrt{2(1+\sin\theta)}(\sin\theta+1)$$

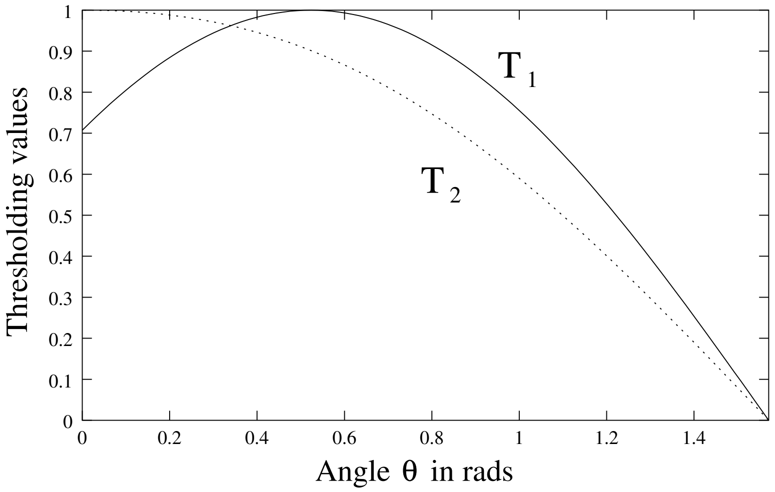


Figure 10: Thresholding values T_1 and T_2.

$$+(T_2^2 2(1+\sin\theta)-\cos^2\theta)=0$$

This is a quadratic equation with respect to n_z. For T_2 to be the desired threshold, this equation has to have only one solution, that is its discriminant $\mathcal{D}$ should be zero:

$$\mathcal{D}=2T_2^2(\sin\theta+1)^3-\left[1+(1+\sin\theta)^2\right]\left[T_2^2 2(1+\sin\theta)-\cos^2\theta\right]=0$$

Solving this equation, we get:

$$2T_2^2(1+\sin\theta)=\cos^2\theta\left[1+(1+\sin\theta)^2\right]$$

Therefore the threshold T_2, which ensures that we do not pick up normals that could be self-shadowing under the neighbouring illuminant, is:

$$T_2=\frac{\cos\theta}{\sqrt{2(1+\sin\theta)}}\sqrt{1+(1+\sin\theta)^2} \tag{14}$$

T_2 never exceeds T_1 for acceptable elevation angles (see Fig. 10).

References

[1] S. Barsky. *Surface Shape and Colour Reconstruction using Photometric Stereo.* PhD thesis, School of Electronics and Physical Science, University of Surrey, UK, 2003.

[2] P.H. Christensen and L.G. Shapiro. Three-dimensional shape from colour photometric stereo. *International Journal of Computer Vision*, 13(2):213–227, 1994.

[3] E.N. Coleman and R. Jain. Obtaining 3-dimensional shape of textured and specular surfaces using four-source photometry. *Computer Graphics and Image Processing*, 18:309–328, 1982.

[4] M. Drew and M. Brill. Color from shape from color: a simple formalism with known light sources. *J. Opt. Am. Soc. A*, 17(8):1371–1381, August 2000.

[5] M.S. Drew. Shape from color. Technical Report CSS/LCCR TR 92-07, Simon Fraser School of Computer Science, 1992.

[6] M.S. Drew. Photometric stereo without multiple images. *Human Vision and Electronic Imaging, SPIE/IS&T*, 3016:369–380, February 1997.

[7] G.D. Finlayson, J. Dueck, B.V. Funt, and M.S. Drew. Colour eigenfaces. In *IEEE 3rd International Workshop on Image and Signal Processing*, Manchester, England, November 4-7 1996.

[8] G. Kay and T. Caelly. Estimating the parameters of an illumination model using photometric stereo. *Graphical Models and Image Processing*, 57(5):365–388, September 1995.

[9] G.J. Klinker. *A Physical Approach to Color Image Understanding.* A K Peters, 1993.

[10] L.L. Kontsevich, A.P. Petrov, and I.S. Vergelskaya. Reconstruction of shape from shading in colour images. *J. Opt. Soc. Am. A*, 11(3):1047–1052, March 1994.

[11] S.W. Lee and R. Bajcsy. Detection of specularity using color and multiple views. *Image and Vision Computing*, 10:643–653, 1992.

[12] S.K. Nayar, K.Ikeuchi, and T.Kanade. Determining shape and reflectance of hybrid surfaces by photometric sampling. *IEEE Trans. on Robotics and Automation*, 6(4):418–431, August 1990.

[13] A.P. Petrov and G.N. Antonova. Resolving the color-image irradiance equation. *Color Research and Application*, 21:97–103, 1996.

[14] B.T. Phong. Illumination for computer generated pictures. *Communications of the ACM*, 18(6), 1975.

[15] K. Schlüns and O. Wittig. Photometric stereo for non-lambertian surfaces using colour information. In *Proc. 5th Int. Conference on Computer Analysis of Images and Patterns*, pages 444–451, Budapest, Hungary, September 13-15 1993.

[16] F. Solomon and K. Ikeuchi. Extracting the shape and roughness of specular lobe objects using four light photometric stereo. *IEEE Trans. Patt.Anal. Machine Intell.*, 18(4):449–454, April 1996.

[17] H.D. Tagare and R.J.P. deFigueiredo. A theory of photometric stereo for a class of diffuse non-lambertian surfaces. *IEEE Trans. Patt. Anal. Machine Intell.*, 13(2):133–152, February 1991.

[18] R. J. Woodham, Y. Iwahori, and R. A. Barman. Photometric stereo: Lambertian reflectance and light sources with unknown direction and strength. Technical report, University of British Columbia Department of Computing Science, 1991.

[19] R.J. Woodham. Photometric method for determining surface orientation from multiple images. *Opt. Eng.*, 19(1):139–144, 1980.

[20] R.J. Woodham. Gradient and curvature from the photometric-stereo method, including local confidence estimation. *J.Opt.Soc.Am*, 11(11):3050–3068, November 1994.