ILLUMINANCE FLOW

(met een samenvatting in het Nederlands)

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de Universiteit Utrecht
op gezag van de rector magnificus, prof.dr. J.C. Stoof, ingevolge het
besluit van het college voor promoties

in het openbaar te verdedigen op vrijdag 15 januari 2010
des middags te 4.15 uur

door

(Dan) <u>Stefan</u> Mikael Karlsson

geboren op 3 september 1978 te Stafsinge, Zweden

Promotor: prof.dr. J.J. Koenderink

Co-promotor: dr. S.C. Pont

To Oscar,
in memory of my friend

(Oscar van Hoof, 1975 – 2009)

Contents

vi

CHAPTER 1

Introduction

A significant part of perceived reality is induced from vision where shape and texture play important roles. If we see an image, even without color or motion, we can "instantly" perceive the scene that it depicts. We also assume that other people in the world will have roughly the same percept, as far as the 3D contents go.

Trying to construct a machine that performs the same task as the human visual system is no easy task. Bear in mind that I am only considering interpretation of a single monochrome image here, while the brain obviously does more. It uses both color and motion to help us achieve the much greater goal of navigating through a complex world, and to ultimately make sense of it. Humans excel at this, assuming no cognitive impairments such as, e.g., motion blindness or over-exposure to the philosophy of Sartre[15].

This thesis is concerned with the task of estimating illuminance flow, which is used to derive shape from single grayscale images. It investigates using texture for this purpose, but deviates significantly from most other works (especially in computer vision) in that texture is given a much richer definition. In general, texture is an elusive phenomenon when it comes to description; there are various, sometimes contradictory definitions in the literature. One could easily dedicate several pages on the topic, as done by many works connected to texture analysis (see for instance[16] for a good overview).

Computer vision approaches to 3D reconstruction commonly model objects as consisting of shape (the surface of the object) and texture (the variation of surface albedo forming a pattern on the surface). While this definition is very useful for cases such as a box of cereals, a race car with commercial printing or a wall with graffiti, it is (almost) useless for objects such as the stem of a tree, a boulder or a uniformly colored sculpture. I adapt a more general notion of texture, which is more in line with the intuitive human notion of it. While surface variation in albedo (painted pattern on the surface say) is one part of what makes up texture, another part is fine-scale surface variation. This kind of texture is often called 3D texture. This should not be confused with the kind of "texture" found in 3D volumetric data, which confusingly is often referred to as 3D texture as well[26].

1.1   Texture, shape and scale

If one takes a perfect sphere, and covers it with a uniformly colored, fine scale, plastered surface, then nobody (except some of the more unreasonable computer vision academics) would object to the statement: "I see a sphere with some rough texture on it". The unreasonable academic (lets call him Simplicio from now on) will point out to you that if you look close at a small surface patch, you only see shading and shadowing due to shape variation. He will continue to state that this, in itself, is proof that the object is not a perfect sphere, and that you should really learn what texture is before engaging in discussion with real scientists.

In this context, it is appropriate to consider the planet we live on. To Simplicio, one can explain that mount Everest is a barely noticeable part of the texture of the planet and that on this scale he himself is part of neither shape nor texture, but is utterly irrelevant, as is his opinion.

This is not to say that mount Everest does not have a shape, but it is to point out that for the problem of shape inference, what is shape and what is texture is completely scale dependent. A mountain is a shape when considering how to land a space shuttle, but it is not when maintaining geostationary orbit. Going back to the sphere with plaster, it is to ordinary people a sphere, because if asked, they can easily grab it as a sphere, roll it as a sphere, throw it as a sphere, but when asked to count the number of plaster indentations and protrusions on the entire object, they will be unable to (of course, Simplicio can help out here, as he can with ease provide a protrusion detector/counter, as long as he can find the right Bayesian priors

for his vision algorithm).

Even though observers cannot easily count the protrusions, they are still aware of them. A keen ball player will know how to better roll and grab the sphere due to their distribution across the surface, and a visual artist will be able to reproduce a picture of the sphere that the ball player will recognize. If the artist is skillful enough he can depict the sphere in different illumination conditions, where 3D texture will have widely different appearance. The depiction will most likely have no protrusions similar to that of the original sphere, and still no human will at a glance see the difference. The artist has performed the remarkable task of 3D texture synthesis[22][21]. The exact protrusions and indentations across the depicted sphere will probably be different from the original. A reasonable hypothesis is that the statistics of the texture that matter to human interaction (such as surface directionality and spikiness/smoothness) would stay the same.

I model objects as having surface variation on 3 independent scales of importance: *macro*, *meso* and *micro*.

Macro scale is the scale of the object as a whole, which is why no one raises an eyebrow when I say that the Earth is round. Only a modern day Simplicio would disagree. A planetary scientist could point out that the planet is slightly elongated along the equator, and not a perfect sphere, but Simplicio can dig a hole his entire life without any changes occurring on the macro scale.

Meso scale is the scale of texture, especially when it is stochastic in nature. Ridges of mountains are variations in the crust of the planet Earth, and yields a noticeable texture when viewed from space. If we discuss the Earth as an object, then we can talk of the specific peaks and valleys only if we are in the meso scale.

Micro scale, finally, is the scale which is too fine for us to perceive as a local shape change, and *only* makes sense to analyze statistically. In terms of a camera, it is the sub-pixel resolution context. This is where we find Simplicio on the crust of the planet and there is no lower limit to this scale, even algae in the example of the Earth. A handful of algae is not observable from space but a huge blooming might be.

With this close dependence between scale, shape and texture, the risk of misunderstandings is a very real possibility. In the frame of thought introduced here, it is quite possible to talk of mount Everest as a "shape within the texture". This is to do the mental exercise of switching scales within the same sentence, and does not add to the clarity of the text. Nonetheless, I

hope the reader at this stage understands the meaning of a "shape within the texture", and I will henceforth avoid the temptation of "surfing the scales". Furthermore, I will try to keep myself to the range of scales in which we humans perceive with our unaided original senses (i.e. middle world as coined by Richard Dawkins[1]).

1.2 A haptics analogy

In the area of haptics, the issue of defining scale in order to differentiate between texture and shape is less complicated[32] (although some unique complications do arise). One uses the natural aperture of the finger or the hand, and relates it to a measure of scale of surface variation. This keeps any haptically inclined Simplicio at bay (by giving him the finger, so to speak, as an example), and researchers can safely say "I feel a sphere with some rough texture on it". As for the micro-scale, it is the scale so fine that any particular protrusion cannot be felt haptically, but statistics from this scale nonetheless affect the percept of the surface. For example, if there is directionality in the height profile on the micro-scale, then there will be more friction when sliding the finger in one direction compared to another. This kind of surface-to-surface interaction on the micro-scale is studied in the area of tribology[34].

With the finger, as with the retina, we perceive reality in $2^1/_2$ dimensions[57]. Beyond that, unfortunately, the finger or hand makes for a very poor analogy with the retina. One might be tempted to define the scales based on variations over angular units (parameterizing the visual field). This breaks down instantly as one moves the object closer to the observer. Even in an unchanging scene, what is texture and what is not is entirely dependent on what is given attention to. Objects are often built up hierarchically of smaller repeating objects in any of the three scales. Clearly, in the context of object appearance we must let the object of interest define the scales, relative to the global size of the object and the resolution of the imaging device. This is assuming that the object is contained within and takes up a fair amount of the visual field, in which case the general object size gives the macro-scale, the pixel resolution gives the micro-scale and the meso-scale occupies a range in-between these two. This does not solve all problems with scale definition. One problem occurs if we consider an object with one part closer to a camera with a wide-angle lens. Another, more fundamental problem with the separation of macro-, meso and micro scales is

that global shapes become limited to smooth (and most certainly differentiable) manifolds. It is impossible to have a macro-scale shape that is, for example, a mathematical cube, because the edges and corners are infinitely fine.

An interesting point of view to the three scales can be had for the special case when the object can be parameterized as a spherical function. This is when all surface points are projectable to a single point inside the object (this is a so-called "star shape"). In this case, one could use the spherical harmonics expansion[17] to reason about the three scales. Spherical harmonics can be explained as frequency analysis on a spherical topology, and is a generalization of the 1D Fourier Series expansion. The 2D Fourier series expansion is connected; it is essentially equivalent, but works on a toroidal topology instead of a spherical one. With the harmonics expansion, one has a spectrum of surface variations, and one can chop this up into 3 intervals, and label them as macro, meso and micro. Of course, the limit between meso and micro is dependent on the resolution of the camera and how much of the visual field the object is occupying. One has decomposed the object as a sum of the 3 scales of interest, and the problem with non-differentiable manifolds becomes apparent; for a cube, we get a smoothed out version for macro shape, and visible bleed off into both meso and micro scales for the edges and corners.

I hasten to point out that spherical harmonics are not used to define the different scales, but are merely a convenient tool for reasoning about them. Spherical harmonics come with a range of problems in this context, including, but not limited to, allowing negative values and only allowing spherical functions.

1.3 3D texture vs. flat texture

The variation in appearance of 3D texture due to illumination change is a fascinating subject. It fundamentally sets 3D surface texture apart from its flat 2D counterpart (painted patterns) in terms of how it can be analyzed and ultimately handled by any vision system[7]. Consider, if you will, a plane with texture illuminated by collimated beams (the sun on a cloudless day, or what some call a point source at infinity). If the texture is 2D Lambertian, then only the elevation angle (the slant) of the incoming beam of light matters to its appearance; variation in the azimuthal angle of light is irrelevant. The variation of the appearance is remarkably simple: the global intensity of the

pattern is uniformly scaled as the elevation (slant) angle changes. Phrased differently, for 2D texture, if we simultaneously increase the intensity of the light as we decrease its slant then it is possible to leave the appearance of the texture unchanged. With 3D texture, the case is naturally quite different. On the meso-scale, the protrusions and indentations of the height variation will appear differently depending on the elevation angle (slant), but more critically, the appearance will vary based on the *azimuthal* (tilt) angle of the light direction[7].

This observation has been exploited to great effect in the computer graphics community. The collection of techniques called "bump mapping"[19] allows a designer to provide two separate 2D arrays for a given object facet: **a**) the 2D flat texture, and **b**) the meso-scale height variation across the facet[19][18]. Objects rendered this way appear much more realistic and the scene is generally more pleasing to the eye. More importantly (and probably fundamentally connected) is that the observer is given a richer stimulus to deduce the setup of the scene. There are now additional cues in the image for the light field and the shape of the objects!

The addition of 3D texture to an object aids the human visual system tremendously, and it is especially the possibility to infer shape from this information that is the ultimate goal of observing the wonderful phenomenon of illuminance Flow. This brings us finally to the specifics of my thesis work, which is where the next section starts off. This, I promise, will be less philosophically inclined and more technical then what this introduction would have you believe.

1.4 Theoretical background

The position of the light source relative to the camera (the light vector) is given by:

$$\boldsymbol{\ell} = \{\ell_1, \ell_2, \ell_3\}^{\mathrm{T}} = \{\cos\theta\cos\phi, \cos\theta\sin\phi, \sin\theta\}^{\mathrm{T}}$$

where θ is the illuminant elevation angle (slant) and ϕ is the azimuthal angle (tilt). The light vector $\boldsymbol{\ell}$ is *not* given in the tangential frame of the surface, but in the frame of the camera. There is no spatial dependence, it is constant throughout the scene. Fig. 1.1 illustrates the geometry.

To arrive at illuminance flow, we first define the twice projected light vector:
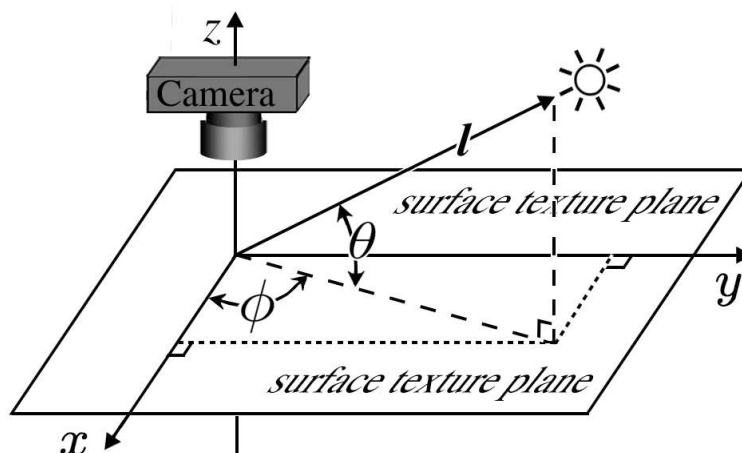
Figure 1.1: The imaging geometry where θ is the slant (elevation) and ϕ is the tilt (azimuth).

$$\tilde{\boldsymbol{f}}(\boldsymbol{x}) = \boldsymbol{P}(\boldsymbol{\ell} - (\boldsymbol{\ell}^{\mathrm{T}}\boldsymbol{n})\boldsymbol{n})$$

where the 2-by-3 matrix $\boldsymbol{P}$ is the projection into the camera frame. In our case $\boldsymbol{P}$ will simply discard the third dimension (the camera is along the z-axis): $\boldsymbol{P} = (\boldsymbol{e}_1 \vdots \boldsymbol{e}_2)^{\mathrm{T}}$. Illuminance flow is defined as the normalized projected light vector, with a sign ambiguity:

$$\boldsymbol{f}(\boldsymbol{x}) = s\frac{\tilde{\boldsymbol{f}}(\boldsymbol{x})}{|\tilde{\boldsymbol{f}}(\boldsymbol{x})|} \tag{1.1}$$

where $s \in \{-1, 1\}$ represents the sign ambiguity which allows us to treat illuminance flow as an orientational (axial) rather then a directional flow field. The reason for doing so is because of the sign ambiguity in shading patterns, often called the convex-concave ambiguity[14]. Consider any Lambertian rough surface texture distributed on a plane viewed frontally. Roughly the same image will result if simultaneously the tilt light angle is rotated one half revolution ($\phi \rightarrow \phi \pm 180°$) while the texture height is inverted ($h_{meso}(\boldsymbol{x}) \rightarrow -h_{meso}(\boldsymbol{x})$). The sign ambiguity can sometimes be resolved using inter-reflections or shadows, but these effects are difficult to use in general, and are most often negligible when dealing with a low relief surface texture[22]. The sign ambiguity is illustrated even clearer in section 1.5 below.

Fig. 1.2 illustrates ground-truth on a sphere. The figure also shows how we envision shape from illuminance flow to work. After estimation, any
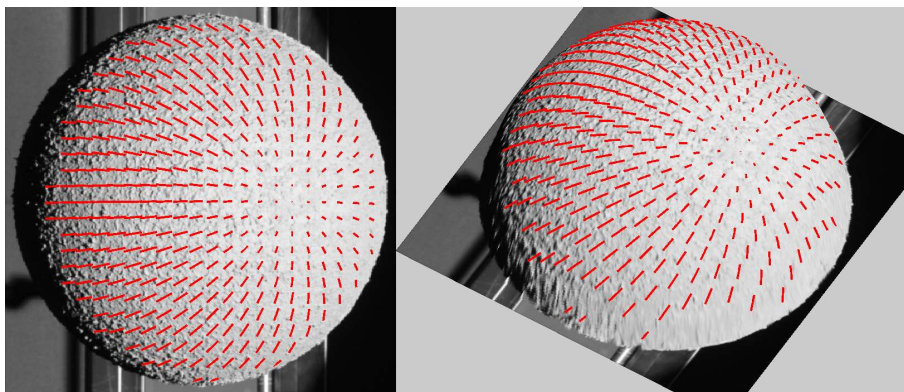
7

Figure 1.2: Illustration of ground-truth illuminance flow over the surface of a sphere. Left: original image, with flow illustrated with red line segments. Right: illustration of how a shape from illuminance flow algorithm should perform given the correct flow. The image can be inflated.

algorithm should have the power to inflate the image, as illustrated in the figure.

An important example is a flat plane illuminated with collimated beams of light; $\boldsymbol{f}(\boldsymbol{x})$ is then a constant vector throughout the image, and is detectable only in the presence of 3D texture. If only 2D texture is present, then there is no dependence on the tilt angle.

How would one go about measuring illuminance flow from an image of the surface? To attack this problem (which is a novel topic), it is convenient to start with a simpler and more studied issue, that of *illuminant tilt estimation.*

1.5 Illuminant tilt estimation

Illuminant tilt (ϕ) is the azimuthal light direction relative to the camera frame. Now, illuminance flow is *fundamentally different* from illuminant tilt for all cases except for the frontally viewed flat plane. Illuminance flow is (in general) a spatially varying entity, while illuminant tilt is a global constant. One might be tempted to think of illuminance flow as a local illuminant tilt estimator, where the "tilt" angle is local with respect to the tangential plane. This is still not correct, for it does not factor in the change in orientation given by the final projection into the camera plane. Illuminance flow is the twice projected light vector, and it will be equivalent to a local illuminant tilt, only if the frame of the camera is aligned with the tangential frame of

the object (which happens when the shape is a plane viewed head on, i.e. fronto-parallel). The distinction is very clear cut, and very important!

With that said, we know that for the singular case of a fronto-parallel patch, illuminant tilt estimation is what we are trying to achieve with any illuminance flow estimation (especially such approaches as found in[35][9]). One could approach illuminance flow estimation from the perspective of generalizing illuminant tilt estimation, which is one way of explaining the work done on illuminance flow by use of the structure tensors. This is the focus of chapters 2 and 3, although this is not the way in which the topic is there presented. When reading chapters 2 and 3 one could easily replace every occurrence of the (gradient-based) "structure tensor" with "any local unsupervised illuminant tilt estimator" (This is addressed in more detail in Section 2.4). Chapter 4 shows that the gradient based structure tensor emerges as the optimal estimator in a regression setting, while chapter 5 goes beyond the structure tensor, and assumes that ground-truth flow is already given, and forms the basis of shape inference[1].

Unsupervised illuminant tilt estimation is all about image directionality although not always stated so explicitly[35][9]. Before we go on, I will share with the reader a small simple simulation experiment that is easily reproducible.

1.6 Image-based, fronto-parallel 3D texture synthesis

I hasten to point out that the thesis work has no connection with 3D texture synthesis. The following cartoon example is to illustrate a few points connected to the appearance of 3D texture and illuminant tilt estimation. For a work on proper 3D texture synthesis, see for instance[22], and[21] for a thorough review of this fascinating rather recently considered problem.

We will generate images randomly, as the sum of sinusoidal wave planes. Each wave-plane takes the form of $\cos(a(\boldsymbol{k}_b^{\mathrm{T}}\boldsymbol{x} + c))$ for $\boldsymbol{k}_b = \{\cos(b), \sin(b)\}$. Here, the frequency of the wave is $a \in \mathbb{R}^+$, the direction is $b \in [-90°, 90°)$ and the phase is $c \in [-90°, 90°)$. One single such wave plane can be seen in the top left of Fig. 1.3. We form a sum of N such wave-planes with individually

---

[1]More detail on the connection between the chapters and how they form a coherent body of work is made clear from the thesis summary.

generated random (a, b, c) and transform the resulting image such that its gray values are translated and scaled to fit the range of 0 to 1.

For each plane we generate, lets draw independent random values for (a, b, c). The script draws a from a chi-square distribution, with parameter value 5. This is a strictly positive distribution, that is skewed (any positive skewed distribution would do for our example, yielding different kinds of perceived surfaces in Fig.1.3). The direction of the waves, b is drawn from a normal distribution with mean direction $\mu_b = 45°$ and standard deviation $\sigma_b = 25°$ (this is actually a "wrapped" normal distribution because b is a circular variable). The phase c is drawn from a uniform distribution.

The result for up to 1024 waveplanes is shown in Fig. 1.3. Interestingly, what comes out in the end seems to be a rough surface texture frontally viewed, illuminated with tilt angle $\phi = \mu_b = 45°$. We can note the sign (convex-concave) ambiguity, by rotating the images 180° (the reader is encouraged to this). When doing so, the images appear to still have illumination from the direction $\phi = 45°$ but each perceived convexity has been replaced by a concavity, and vice versa. Also interesting is that the procedure of adding these planes converges to a stationary process, that is: as we add more planes, the specific illuminated surface indentations and protrusions change, but it is statistically a similar kind of surface illuminated from the same direction. This simple experiment illustrates the central principle which lies behind all the algorithms for illuminant tilt estimation from texture. The directionality of the image will be highly correlated with the tilt of the light vector. If the light vector is in the position of the camera, that is if the slant angle is 90°, then there will be no directionality in the image to exploit, and this is when the problem of tilt estimation is undefined (as is illuminance flow).

The reader who is comfortable with frequency analysis will likely think of this approach of adding waveplanes as a frequency based approach, and realize that a very similar algorithm can be implemented in the 2D discrete Fourier (FFT) domain. In the 2D continuous Fourier domain, the above approach is equivalent to randomly placing points of energy (Dirac delta distributions) in the spectrum. The statement above regarding convergence to a stationary process is easily shown by realizing that the image we create will converge towards a bimodal power spectrum.
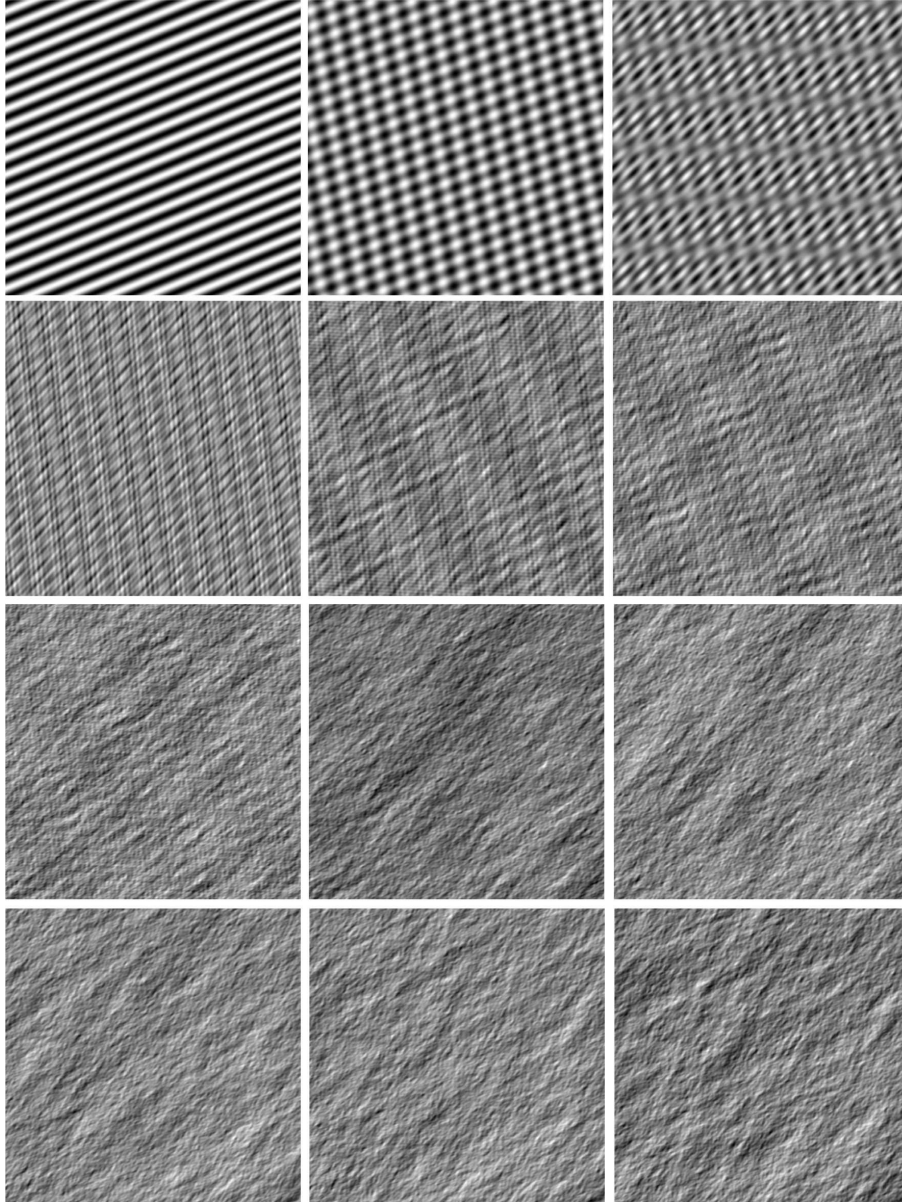
Figure 1.3: Illustration of simplistic 3D synthesis by addition of waveplanes with random phase, frequency and direction. From top left to bottom right: increasing amount of waveplanes, doubling for every image (1, 2, 4, 8, ... , 2048).

The distribution of the orientations of the waveplanes will determine the shape and size of the modes in the angular dimension, and the distribution of the frequency will likewise determine the size and shape of the modes in the radial directions. For certain choices of distributions for the orientations and frequencies, the power spectrum will resemble closely the directional derivative of some underlying two dimensional signal. This is precisely when the resulting image will appear to us as an obliquely illuminated surface texture (the underlying 2D signal is the meso-scale surface).

The results of my simple experiment with adding random wave planes to yield image texture, must have some explanation from physical reality. It seems improbable that it happens to resemble surface 3D texture just by chance. There is indeed a physical rationale, the first to observe and make use of it were Kube and Pentland[30]. In their elegant original work, the focus was on fractal surfaces which I will briefly discuss here before continuing on the physical rationale.

1.7 Surface models

A fractal surface is a 2D function whose second order (and possibly higher) statistics are invariant of scaling. Thus, one can zoom into, or out of, the surface indefinitely and find self-repeating structure. The focus of Kube and Pentland[30] claimed to be the so-called Brownian surface, which can be defined as a surface with a power spectrum following a "power law". If $h(\boldsymbol{x})$ is the surface, and the Fourier transform of it is $H(\boldsymbol{\omega})$, then for the Brownian surface:

$$|H(\boldsymbol{\omega})| \propto |\boldsymbol{\omega}|^{-\beta}$$

for $\beta \in \mathbb{R}^+$. The so-called fractal dimension (denoted D, a standard measure of the self-repeatedness) correlates with both β and perceived surface roughness[28].

A Brownian surface is a most non-intuitive entity. It is continuous, but not differentiable, it is defined by its power spectrum that has infinite energy, and it does not have an auto-covariance function. There are a lot of inherent problems with this surface model but it is popular mainly because of the esthetic value of the generated approximations (which do resemble real world surfaces, as Fig. 1.4 illustrates). The Brownian surface is especially not suited for illuminant tilt estimation of 3D texture analysis. One of many problem is that the surface variation goes into the micro scale, and
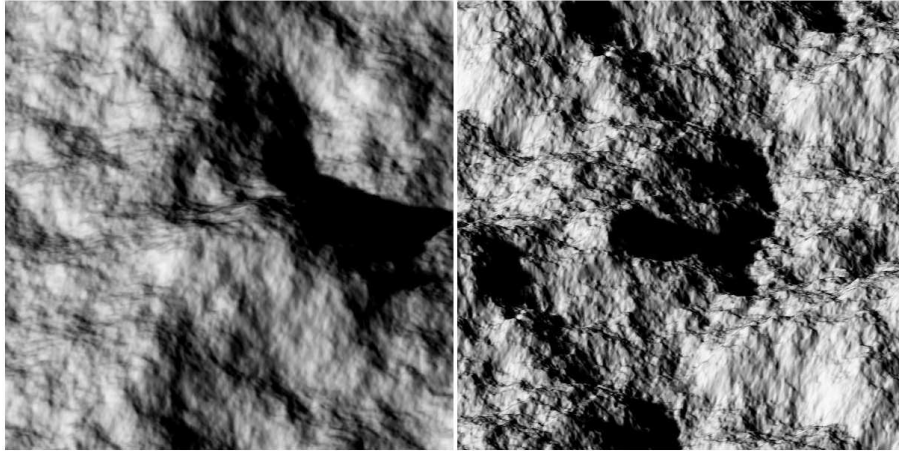
Figure 1.4: Renderings of two approximations of Brownian surfaces, viewed obliquely. They have varying fractal dimension. Left: $\beta = 1.1$. Right: $\beta = 1$. Gaussian smoothing (of varying degrees) was applied on the surface height profiles. This was done sufficiently so as to make them pleasing to the subjective opinion of this author.

thus changes the surface reflectance properties[13] (exactly how is dependent on β). In practice, approximations of the model are used, where the power spectrum is cut off towards the lower and higher frequencies (as done in Fig. 1.4). If one does not need infinite repeatability, but one does need differentiability and a convenient auto-correlation function, then the Brownian surface should not be an option. The work of Kube and Pentland is most certainly not intended for non-differentiable manifolds, as the very entry point of the theory developed begins with differentiation to form the locally varying surface normal. The esthetically pleasing appearance that the approximation provides can equally well be described as a sum of differentiable surfaces of the following type (which we will henceforth call "Gaussian surfaces"):

$$|H(\boldsymbol{\omega})| = \exp(-\sigma|\boldsymbol{\omega}|^2)$$

These surfaces are differentiable, and have the benefit of containing energy in only one scale (given by σ). The Brownian surface contains infinite amount of energy in all scales (macro, meso and micro, and above and beyond to coarser then galactic scales, and finer then string theory scales). If one wants Brownian looking surfaces as a model, then one can use a sum of Gaussian surfaces to approximate the behavior of the Brownian model, within the scale of interest (the relevant frequency band). This way, we have a model that is differentiable and has a well defined auto-correlation function.

13

Of course, other surface models exists, like those based on Markov models[20] and other kinds of Power spectra[25]. It is important that the reader does not confuse the meso-scale surface height profile (not observable) with the image texture (observable).

1.8 From image based to physics-based modeling

Now I return to the approach of adding random waveplanes in the image based synthesis (section 1.6). Kube and Pentland were the first to realize that the image of a random surface would quite often have a bimodal power spectrum, and there is indeed a physics based explanation that we will now investigate. First, the underlying assumptions of the theory are:

- Lambertian reflectance

- Uniform albedo

- Low relief (also results in no inter-reflections nor shadows/vignetting)

- Normal view (fronto-parallel to x-y plane)

- Light field of dominant directional component (in the extreme: collimated beams, light source at infinity)

- The light vector must not be close to the viewing direction.

We start by considering the spatially varying normal of the surface as given by:

$$\boldsymbol{n}(\boldsymbol{x}) = \frac{\{-h_x(\boldsymbol{x}); -h_y(\boldsymbol{x}); 1\}^{\mathrm{T}}}{\sqrt{h_x^2(\boldsymbol{x}) + h_y^2(\boldsymbol{x}) + 1}}$$

where suffixes indicate partial derivatives and $h(\boldsymbol{x})$ is the surface height function.

According to Lamberts law, the observed radiant intensity is proportional to

$$I(h_x, h_y) \propto \boldsymbol{\ell}^{\mathrm{T}} \boldsymbol{n}$$

14

With the assumptions listed above, this quantity is also proportional to the image of the surface[2] (thus $I(h_x, h_y)$ will henceforth be treated as the image). Now, the expression for $I(h_x, h_y)$ is a very uncomfortable non-linear expression, so Kube and Pentland performed a Taylor series expansion around $(h_x, h_y) = (0, 0)$, and arrived at their model by only using the linear terms. I will henceforth use $\boldsymbol{g} = (h_x, h_y)$.

I will do a small modification to their approach. Instead of allowing two degrees of freedom for the gradient of the surface, I will allow for only one: the magnitude $|\boldsymbol{g}| = \sqrt{h_x^2 + h_y^2} \in \mathbb{R}^+$ such that $(h_x, h_y) = |\boldsymbol{g}|(\cos a, \sin a)$ for some arbitrary local angle of the gradient a. We allow the derivatives to change uniformly only with $|\boldsymbol{g}|$. We perform the one-dimensional Taylor expansion around $|\boldsymbol{g}| = 0$ and show up to second order as[3]:

$$I(h_x, h_y) = -\ell_3 + (h_x\ell_1 + h_y\ell_2) + \frac{1}{2}\ell_3(h_x^2 + h_y^2) + O\left(|\boldsymbol{g}|^3\right) \qquad (1.2)$$

The Kube and Pentland linear model is $-\ell_3 + (h_x\ell_1 + h_y\ell_2)$, which we also find as the zeroth and first order terms above. We recognize that $(h_x\ell_1 + h_y\ell_2)$ (which we can also write as $\cos\theta(h_x\cos\phi + h_y\sin\phi)$) is proportional to a directional derivative in the direction of the illuminant tilt angle (ϕ). Now, what will a directional derivative look like in the power spectrum? The answer is given in Fig. 1.5. The derivative of Fig. 1.5 is along the y-axis, so if the Kube and Pentland model would be correct, then we should see a surface illuminated from below. We actually see surfaces illuminated from above, because our vision system has a strong prior for that setting. We see again the sign ambiguity in illumination direction and surface convexity/concavity. Turning the images upside down will give rise to a perceptual flip. The rationale of the Kube and Pentland model is the explanation to why the technique of bump-mapping works[18].

Going back to Eq. 1.2, we notice the second order term in addition to the Kube and Pentland linear model[4]. The first term is a weighted directional

---

[2]We can also mention that, strictly speaking, a linear camera transfer function is needed to actually produce a proper image for this theory to work on. In practice, a monotonic transfer function will do.

[3]This approach is very close to that of Section 3.3. The height scaling used there is somewhat different, in that it is not spatially varying, and thus makes the connection to global roughness measures. Here we just use the gradient magnitude because it makes for a nicer expression that should be more intuitive.

[4]The second order term comes out this nicely because we used $|\boldsymbol{g}|$ in the Taylor series expansion.
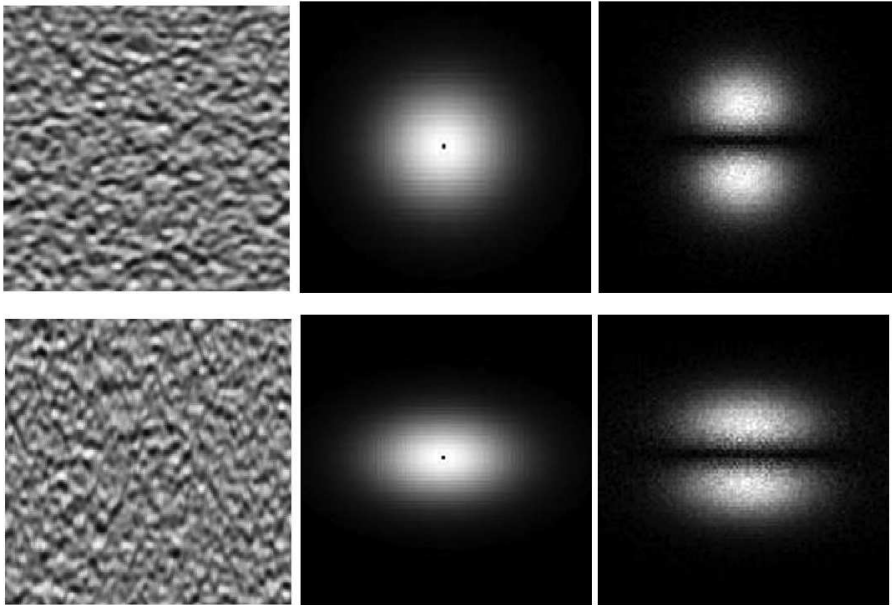
Figure 1.5: Illustration of the directional derivative of Gaussian surfaces. Observe that these are not rendered surfaces but derivatives of height profiles. Top row: Isotropic surface. Bottom row: Anisotropic surface. Left column: the partial derivative (in the y-direction). Middle column: Surface Power spectrum. Right column: Power spectrum of the derivative image (from the left row).

derivative, whose weight $(\cos\theta)$ gets larger as the illumination approaches grazing ($\theta = 0$, the light approaches the horizon). This is called first order shading. The second term is the square magnitude of the gradient, weighted with $\ell_3 = \sin\theta$, and thus gets weighted more as the illumination approaches normal direction ($\theta = 90°$, the light approaches zenith). This is called second order shading.

The truncated Taylor series expansion is elegant in that it correctly orders the different kinds of shading, and gives an accurate description of which type of shading will occur more dominantly then the other. We see the two main contributing factors explicitly in the formulation: the illumination slant angle θ, and the surface gradient magnitude $|\boldsymbol{g}|$.

CHAPTER 2

Illuminance Flow over Anisotropic Surfaces

---

abstract

A theory is presented to analyze images of anisotropic fine-scale surfaces. We investigate the estimates of illuminance flow using structure tensors. For anisotropic surfaces, both the gradient-based and the Hessian-based tensors will yield deviations from the true illumination orientation. Our theory predicts this deviation. To show the use of this theory, an algorithm is derived that uses both tensors simultaneously to compensate for small amounts of anisotropy. Experimental results with rendered surfaces are shown to conform well to our theory.

2.1   Introduction

The light field[52] in which objects are situated is important in machine vision. Often slight changes in illumination produce large changes in the image of an object, especially with regard to meso scale surface variations across the object (so-called 3D or surface texture).

A great deal of work has been done regarding methods for handling image variation due to translation, rotation and scale[23][42][3]. Only in the last decade or so has the subject of lighting variation been given widespread attention within the machine vision field[53][54]. The focus has been mainly on arriving at a set of lighting invariant features. The traditional approach has been to vary the light field and gather statistics that are insensitive to light change (often resulting in edges and corners in the albedo map).

In this chapter, we investigate illuminated, Lambertian anisotropic surface textures of uniform albedo, and try to infer the so-called illuminance flow[10][11]; defined as the set of vectors resulting from projecting the light vector into the tangent planes of a surface. The illuminance flow over the surface of an object constitutes a flow field. An example of this flow field (for a sphere) is shown in the left pane of Fig. 2.1. The surface illuminance flow is a continuous set of 3D vectors (one for each surface patch that is illuminated) and does not depend on the camera position.

Geometry of an object exists on several scales. The global scale constitutes the shape of the object as a whole (e.g. a sphere as in Fig. 2.1) and the meso scale makes up the surface texture. At even finer detail are the micro scales with variations so fine that they influence the reflectance properties of the surface rather than observable geometry. Where one scale starts and the other ends is usually not given, and can vary locally within the image and depends on the camera position. The global scale geometry defines a set of tangent planes, into which the light vector is projected to form the illuminance flow.

In this chapter, we consider the case where the object is a plane on the global scale, and where the camera is in normal position (illustrated by the right pane of Fig. 2.1). Assuming a collimated light source then the illuminance flow is a unidirectional flow field with direction ϕ (the azimuthal angle of the incident light, as illustrated in Fig. 2.1). For this case, the problem is reduced to estimating the azimuthal direction ϕ from the image.

Any algorithm that can estimate ϕ (locally) can be used to estimate the illuminance flow in normal viewing. This puts our work close to surface tex-
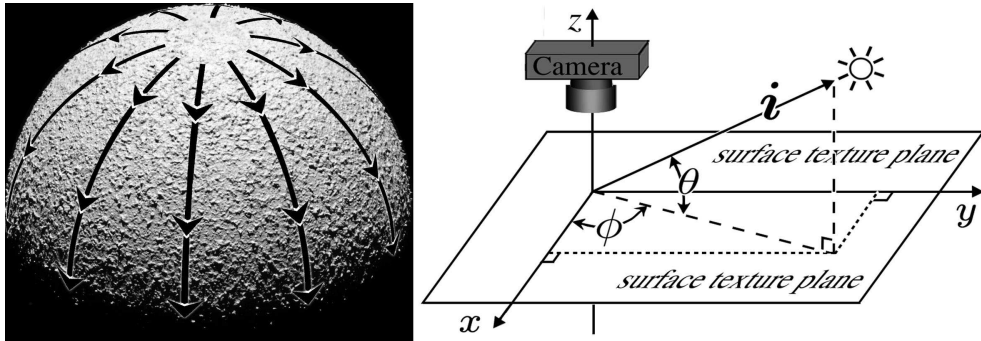
Figure 2.1: Left:Illustration of illuminance flow on a sphere. Two scales of interest: meso-scale, surface variation due to texture, and object-scale, defining the geometry of the object (here a sphere), Right: the imaging geometry. We assume an object that is flat on the global scale for this chapter. The elevation angle θ, and the tilt angle ϕ define the direction of illumination.

ture classification and photometric stereo, where illuminant azimuth (or tilt) estimation plays an important part[7]. In this context, this chapter presents a novel theory that generalizes the Kube and Pentland imaging model[30], by use of structure tensor statistics to account for surface anisotropy. After presenting our theory and a proposed estimator, we will relate it to a few algorithms of illuminant estimation from texture analysis (section 2.4) and show that our theory is relevant to them all. With respect to illuminance flow, this chapter is in line with the study by Pont and Koenderink[10] where image structure tensors were used to estimate the light direction. The current chapter is concerned primarily with the extension of the theory to the case of anisotropic surface textures.

When the light field[52] has a dominant directional component (a collimated beam being the extreme case) anisotropy will be imposed in the image of the surface. In some cases this can be used to estimate the illuminance flow. It has been shown[10] that using the structure tensor of either the Hessian or the gradient yields good estimates. However, formally, this only applies under a set of assumptions, namely:

- Isotropic surface

- Lambertian reflectance

- Uniform albedo

- Low relief

- Normal view

- Light field[52] of dominant directional component (*e.g.* a collimated beam)

Experimental results have been presented for real world surfaces[10]; the structure tensors yielded promising results, with orientation estimates within a few degrees of the veridical orientation. The illumination elevation was not estimated due to the bas-relief ambiguity[55].

All assumptions are important in their own right. Some work has been done on extending the theory to account for non-normal views [11] and non-uniform albedo[33]. In this chapter, we focus on the isotropy assumption, which, as far as we know, has not been given any attention so far. For anisotropic surfaces, both the gradient and Hessian based tensors will yield deviations from the true orientation of the illuminance flow. The theory presented here predicts these deviations, which are shown to conform well to experiments on rendered surfaces.

To show the use of this framework, an algorithm is proposed that uses both tensors simultaneously. Extensive experimental results on rendered surfaces are presented, as well as a few estimations on real-world textures.

2.2 Theoretical background

Derivatives will be denoted as $h_{pq} = \frac{\partial^{p+q} h(x,y)}{\partial x^p \partial y^q}$, the gradient as $\boldsymbol{g}_h = \{h_{10}, h_{01}\}^T$ and the Hessian as:

$$H_h = \begin{pmatrix} h_{20} & h_{11} \\ h_{11} & h_{02} \end{pmatrix}$$

$\hat{h}(\boldsymbol{k})$ is the Fourier transform of $h(\boldsymbol{r})$ (spatial coordinates $\boldsymbol{r} = \{x, y\}^T$ and frequency coordinates $\boldsymbol{k} = \{u, v\}^T$) and its square magnitude is referred to as the power spectrum, denoted $\hat{\rho}_h(\boldsymbol{k}) = |\hat{h}(\boldsymbol{k})|^2$.

$\langle h \rangle$ will denote averaging over the function $h(\boldsymbol{r})$, and m_h^{pq} denotes moments of the power spectrum defined as:

$$m_h^{pq} = \langle u^p v^q \hat{\rho}_h \rangle \tag{2.1}$$

For correlations between derivatives we have (because $\langle h_{p_1 q_1}\rangle = 0$) :

$$\langle h_{p_1 q_1} h_{p_2 q_2}\rangle = m_h^{(p_1+p_2),(q_1+q_2)}(-1)^{(p_1-p_2+q_1-q_2)/2} \tag{2.2}$$

Eq. 2.2 is derived easily by use of the following properties of the Fourier transform:

$$h_{pq} \;\Leftrightarrow\; (-ju)^p(-jv)^q \hat{h}(\boldsymbol{k})$$
$$\langle f(\boldsymbol{r})g(\boldsymbol{r})\rangle \;=\; \left\langle \hat{f}(\boldsymbol{k})\hat{g}^*(\boldsymbol{k}))\right\rangle$$

where * indicates complex conjugate. Because $\hat{\rho}_h$ is symmetric (due to hermitian symmetry $\hat{h}(\boldsymbol{k}) = \hat{h}(-\boldsymbol{k})^*$) all moments for which $p+q$ is an odd integer will equal zero.

Gaussian Surfaces and the Structure Tensors

Henceforth $h(\boldsymbol{r})$ will denote a Gaussian surface, which is a 2D Gaussian (stochastic) process with several convenient properties[50][51] making it a good candidate for a surface model. It is (strongly) stationary with a constant autocorrelation function $\rho_h(\boldsymbol{r})$ within its ensemble. The covariance function $\mathrm{cov}_h(\boldsymbol{r})$ is related to $\rho_h(\boldsymbol{r})$ by $\mathrm{cov}_h(\boldsymbol{r}) = \sigma^2 \rho_h(\boldsymbol{r})$ where σ is the standard deviation of surface height. The spread of $\rho_h(\boldsymbol{r})$ is denoted s, *i.e.* its width as measured by second-order moments. s is a measure of scale, and the quantity $\frac{\sigma}{s}$ equals the standard deviation of the derivative processes $h_{10}(\boldsymbol{r})$ and $h_{01}(\boldsymbol{r})$.

The power spectrum $\hat{\rho}_h(\boldsymbol{k})$ is related to $\rho_h(\boldsymbol{r})$ through a Fourier transform, and thus $\hat{\rho}_h(\boldsymbol{k})$ is also constant within the ensemble, as are all the moments m_h^{pq}.

We will assume a Gaussian-shaped $\rho_h(\boldsymbol{r}) = \exp\left(-\boldsymbol{r}^T G_h \boldsymbol{r}\right)$ yielding an equivalent Gaussian for the power spectrum: $\hat{\rho}_h(\boldsymbol{k}) = C \exp\left(-\boldsymbol{k}^T G_h^{-1} \boldsymbol{k}\right)$, where C is a normalizing constant and G_h is the gradient-based structure tensor, a symmetric, positive semi-definite matrix consisting of the second order moments:

$$G_h = \begin{pmatrix} m_h^{20} & m_h^{11} \\ m_h^{11} & m_h^{02} \end{pmatrix} = \left\langle \boldsymbol{g}_h \boldsymbol{g}_h^T \right\rangle \tag{2.3}$$

The assumption that the autocorrelation function is Gaussian-shaped is of no critical importance, it is only for convenience. All power spectra of

interest have well-defined second-order moments, and thus also a structure tensor, though not necessarily explicit in closed form as with the Gaussian.

For an isotropic surface, $\rho_h(\boldsymbol{r})$ is isotropic and G_h is a scaled identity matrix (*i.e.* $m_h^{20} = m_h^{02}$ and $m_h^{11} = 0$). For an anisotropic surface the eigenvalues/eigenvectors of G_h yield the amount and orientation of anisotropy. We denote the maximum and minimum eigenvalues and their corresponding eigenvectors as $\lambda_{G_h}^{max}$, $\lambda_{G_h}^{min}$, $\boldsymbol{v}_{G_h}^{max}$ and $\boldsymbol{v}_{G_h}^{min}$. The gradient confidence is a measure of anisotropy defined as:

$$\xi\{G_h\} = \frac{\lambda_{G_h}^{max} - \lambda_{G_h}^{min}}{\lambda_{G_h}^{max} + \lambda_{G_h}^{min}} \in [0, 1]$$

$\xi = 0$ occurs when $h(\boldsymbol{r})$ is a completely isotropic surface, whereas $\xi = 1$ indicates degeneracy (the power spectrum is reduced to a line and the surface contains only isolines). The two cases $\xi = 0$ and $\xi = 1$ will be referred to as the extrema, or extremal cases.

We are interested in the orientation of anisotropy. Here, we choose to make the distinction between orientation and direction, the latter being free to assume any angle, *i.e.* any real number modulo $360°$. Orientation, on the other hand, is a quantity that treats directions that differ by $180°$ equally, defined by taking any directional angle modulo $180°$ (a two-to-one mapping $[0°, 360°) \rightarrow [0°, 180°)$).

The structure tensor was originally introduced to image analysis as a method for orientation estimation[4]. The orientation angle of anisotropy is given by the highest eigenvector and denoted:

$$\mu\{G_h\} = \angle\boldsymbol{v}_{G_h}^{max} \in [0°, 180°)$$

Closely related to G_h is the Hessian-based structure tensor defined as:

$$T_h = \begin{pmatrix} m_h^{40} + m_h^{22} & m_h^{31} + m_h^{13} \\ m_h^{31} + m_h^{13} & m_h^{04} + m_h^{22} \end{pmatrix} = \langle H_h H_h^T \rangle$$

A measure of anisotropy (similar to $\xi\{G_h\}$) is the Hessian confidence: $\xi\{T_h\}$. The orientation of anisotropy is given by $\mu\{T_h\}$.

Analogous to the $\xi\{G_h\}$ measure, $\xi\{T_h\} = 0$ corresponds to the isotropic case and $\xi\{T_h\} = 1$ indicates degeneracy. However, for the non-extremal cases, the two measures differ in general (*e.g.* $\xi\{G_h\} = \frac{1}{2}$ and $\xi\{T_h\} = \frac{1}{2}$ have different interpretations). Also, important for our purposes, $\mu\{G_h\}$ and $\mu\{T_h\}$ will differ in general for the non-extremal cases. We will make

frequent use of $\xi\{G_h\}$ and for the sake of convenience will write simply ξ, where it should be clear that $\xi = \xi\{G_h\}$, the confidence with respect to the gradient tensor of the surface $h(\boldsymbol{r})$, a well-defined and natural measure of the anisotropy of a surface.

2.3 Illuminated Gaussian surfaces

In this section we will derive relationships between the hidden tensors of the surface and the observable ones of the image. The key property here is that surface anisotropy is encoded in the $\xi\{G_h\}$ statistic (equivalently, we could have used $\xi\{T_h\}$, yielding slightly more cumbersome expressions). Thus, the goal of this section is to express $\mu\{G_I\}$ and $\mu\{T_I\}$ in terms of $\xi\{G_h\}$, and ultimately to predict the deviations of the $\mu\{\cdot\}$ observables from the true orientation of illumination ϕ. The orientation of surface anisotropy, encoded by the hidden $\mu\{G_h\}$, will be addressed at the end of the section.

The unit normal of a surface can be written:

$$\boldsymbol{n} = \frac{\{-h_{10}, -h_{01}, 1\}^T}{\sqrt{h_{10}^2 + h_{01}^2 + 1}}$$

The irradiance by a collimated beam is described by a single vector:

$$\boldsymbol{i} = c\{\cos\theta\cos\phi, \cos\theta\sin\phi, \sin\theta\}$$

where θ and ϕ are the elevation and azimuthal angles of the light source respectively and c is the intensity of the light (normal irradiance caused by the beam).

The assumption of a collimated beam is convenient for the derivation, but not critical since extended sources can be exactly accounted for by equivalent collimated sources if vignetting is not present. The following theory holds also for light fields[52] where a dominant directional component is present; the $\boldsymbol{i}$ vector can be replaced by this dominant direction (for Lambertian surfaces this is equivalent, when self-shadowing is not present).

We will consider the image $I(\boldsymbol{r})$ that is formed by viewing the surface $h(\boldsymbol{r})$ normally, and will assume that its intensity is proportional to the irradiance of the surface. The image of a Lambertian low-relief surface will then be given by:

$$I(h_{01}, h_{10}, \theta, \phi) = \boldsymbol{i} \cdot \boldsymbol{n} = c\frac{\sin\theta - \cos\theta(h_{10}\cos\phi + h_{01}\sin\phi)}{\sqrt{h_{10}^2 + h_{01}^2 + 1}} \quad (2.4)$$

We assume a low relief surface (which also entails no shadows or inter-reflections) with low values of h_{10} and h_{01} on average, making it reasonable to approximate Eq. 2.4 around $(h_{10}, h_{01}) = (0, 0)$. A Taylor series approximation including the first order terms of Eq. 2.4 yields

$$I(h_{01}, h_{10}, \theta, \phi) = c \sin \theta - c \cos \theta (h_{10} \cos \phi + h_{01} \sin \phi) + O\left(\boldsymbol{g}^T \boldsymbol{g}\right) \quad (2.5)$$

which is a weighted directional derivative of the surface plus a constant, $c \sin(\theta)$, of little interest. Eq. 2.5 is a model of image formation first suggested by Kube and Pentland[30] and has been used in studies into texture segmentation[6][8] and light estimation[10][8].

The approximation is valid only under oblique lighting, which is evident from the fact that the approximation yields a constant image for $\theta = 90°$, and for lower values $\theta \to 0$ we enter the shadow region.

We consider the Taylor series expansion of the surface around any point of interest (expressed in a local frame where the origin has been aligned with the point of interest)[1]:

$$h(\boldsymbol{r}) = x h_{10} + y h_{01} + x^2 h_{11}/2 + \dots \quad (2.6)$$

Now, we truncate Eq. 2.6 up to third-order terms, and insert it into Eq. 2.5 yielding a closed form approximation of the imaging process valid under the assumptions of low relief and oblique lighting of Gaussian surfaces:

$$
\begin{aligned}
I(\boldsymbol{r}, \theta, \phi) \simeq \ & c(\sin(\theta) - \frac{1}{2} \cos(\theta)(\sin(\phi)(h_{21} x^2 + \\
& 2 h_{11} x + 2 y h_{12} x + 2 h_{01} + 2 y h_{02} + y^2 h_{03}) + \\
& \cos(\phi)(h_{30} x^2 + 2 h_{20} x + 2 y h_{21} x + \\
& 2 h_{10} + 2 y h_{11} + y^2 h_{12})))
\end{aligned} \quad (2.7)
$$

The approximation in Eq. 2.7 can be used to derive expressions of the structure tensors of the image, which will equal:

$$G_I(1,1) = \frac{c^2}{2} \cos^2 \theta \left[m_h^{22} + m_h^{40} + (m_h^{40} - m_h^{22}) \cos 2\phi + 2 m_h^{31} \sin 2\phi \right] \quad (2.8)$$

$$G_I(1,2) = \frac{c^2}{2} \cos^2 \theta \left[m_h^{31} + m_h^{13} + (m_h^{31} - m_h^{13}) \cos 2\phi + 2 m_h^{22} \sin 2\phi \right] \quad (2.9)$$

$$G_I(2,2) = \frac{c^2}{2} \cos^2 \theta \left[m_h^{22} + m_h^{04} + (m_h^{22} - m_h^{04}) \cos 2\phi + 2 m_h^{13} \sin 2\phi \right] \quad (2.10)$$

---

[1]The alignment of the frame implies h(0)=0. There is no loss of generality as the final results depend on the partial derivatives of h (with minimum order of 1).

$$
\begin{aligned}
T_I(1,1) \;=\;& \frac{c^2}{2}\cos^2\theta\,\big[m_h^{60} + 2m_h^{42} + m_h^{24} + \\
& (m_h^{60} - m_h^{24})\cos 2\phi + 2(m_h^{51} + m_h^{33})\sin 2\phi\big] \qquad (2.11)
\end{aligned}
$$

$$
\begin{aligned}
T_I(1,2) \;=\;& \frac{c^2}{2}\cos^2\theta\,\big[m_h^{51} + 2m_h^{33} + m_h^{15} + \\
& (m_h^{51} - m_h^{15})\cos 2\phi + 2(m_h^{42} + m_h^{24})\sin 2\phi\big] \qquad (2.12)
\end{aligned}
$$

$$
\begin{aligned}
T_I(2,2) \;=\;& \frac{c^2}{2}\cos^2\theta\,\big[m_h^{42} + 2m_h^{24} + m_h^{06} + \\
& (m_h^{42} - m_h^{06})\cos 2\phi + 2(m_h^{15} + m_h^{33})\sin 2\phi\big] \qquad (2.13)
\end{aligned}
$$

The left-hand sides of equations 2.8-2.13 are the elements of the structure tensors (equations 2.3 and 2.4) *with respect to the image $I(\boldsymbol{r})$*. The right-hand side contains power spectrum moments *with respect to the surface $h(\boldsymbol{r})$*.

Now, the moments of the surface will be affected by the anisotropy. We assume a Gaussian-shaped surface power spectrum $\hat\rho_h(\boldsymbol{k}) = C\exp\left(-\boldsymbol{k}^T G_h^{-1}\boldsymbol{k}\right)$. Anisotropy is introduced as a non-zero $\xi\{G_h\}$. Assuming (for now) that the orientation of anisotropy is in the orientation of the x axis ($0°$ or equivalently $180°$), we can write the power spectrum as:

$$
\hat\rho_h(u,v) = g(u, s^{-1})g\!\left(v, \frac{1 + \xi\{G_h\}}{1 - \xi\{G_h\}}s^{-1}\right)
$$

where $g(x,a)$ is a 1D normalized Gaussian with standard deviation a. The parameter s remains a general measure of the scale of the surface.

Applying Eq. 2.1 will yield (for $\xi = \xi\{G_h\}$):

$$
m_h^{40} = \frac{12\pi^2(\xi+1)^5}{s^4(\xi-1)^4}
$$

$$
m_h^{22} = \frac{4\pi^2(\xi+1)^3}{s^4(\xi-1)^2}
$$

$$
m_h^{04} = \frac{12\pi^2(\xi+1)^5}{s^4(\xi-1)^4}
$$

$$
m_h^{60} = \frac{60\pi^2(\xi+1)}{s^6}
$$

$$
m_h^{42} = \frac{12\pi^2(\xi+1)^3}{s^6(\xi-1)^2}
$$

$$
m_h^{24} = \frac{12\pi^2(\xi+1)^5}{s^6(\xi-1)^4}
$$

$$
m_h^{06} = \frac{60\pi^2(\xi+1)^7}{s^6(\xi-1)^6}
$$

The remaining moments $(m_h^{31}, m_h^{13}, m_h^{51}, m_h^{33}$ and $m_h^{15})$ all equal zero. Insertion into equations 2.8 to 2.13 yields:

$$G_I(1,1) = C_1 \frac{2\left(\xi^3 + 1\right) + (\xi + 1)((\xi - 4)\xi + 1)\cos 2\phi}{(\xi - 1)^2} \tag{2.14}$$

$$G_I(1,2) = C_1 \frac{(\xi + 1)^3 \sin 2\phi}{(\xi - 1)^2} \tag{2.15}$$

$$G_I(2,2) = C_1 \frac{(\xi + 1)^3 \left(2\left(\xi^2 + \xi + 1\right) - (\xi(\xi + 4) + 1)\cos 2\phi\right)}{(\xi - 1)^4} \tag{2.16}$$

$$T_I(1,1) = C_2 \left(\frac{2\xi^5 - 2\xi^4 + 4\xi^3 + 4\xi^2 - 2\xi + 2}{(\xi - 1)^4} + \right.$$
$$\left. \frac{\left(\xi^5 - 5\xi^4 - 5\xi + 1\right)\cos 2\phi}{(\xi - 1)^4} \right) \tag{2.17}$$

$$T_I(1,2) = C_2 \frac{(\xi + 1)^3 \left(\xi^2 + 1\right)\sin 2\phi}{(\xi - 1)^4} \tag{2.18}$$

$$T_I(2,2) = C_2 \left(\frac{2\xi^7 + 10\xi^6 + 26\xi^5 + 42\xi^4 + 42\xi^3 + 26\xi^2 + 10\xi + 2}{(\xi - 1)^6} - \right.$$
$$\left. \frac{\left(\xi^7 + 9\xi^6 + 27\xi^5 + 43\xi^4 + 43\xi^3 + 27\xi^2 + 9\xi + 1\right)\cos 2\phi}{(\xi - 1)^6} \right) \tag{2.19}$$

for $C_1 = \frac{4c^2\pi^2\cos^2(\theta)}{s^4}$ and $C_2 = \frac{24c^2\pi^2\cos^2(\theta)}{s^6}$. We are interested in the eigen analysis of the tensors. Multiplication by arbitrary positive constants does not affect the eigenvalues or vectors, so we can safely choose to ignore C_1 and C_2 - they only become relevant when we have $s = \infty$, $\theta = 90°$ or $c = 0$, for which both C_1 and C_2 are zero. For $s = \infty$ the surface has degenerated to a plane (for which no variation is present in the image), for $\theta = 90°$ the surface is illuminated strictly from above (no anisotropy imposed by the illumination) and $c = 0$ indicates complete absence of light.

We now investigate the statistics that will be given by the eigen analysis

of the tensors. For the $\mu\{\cdot\}$ measures we get:

$$\mu\{G_I\} = -\tan^{-1}\left[\frac{a}{b + \sqrt{c}}\right] \tag{2.20}$$
$$a = 2\left(\xi^2 - 1\right)\sin(2\phi)$$
$$b = \left(4\xi^2 + 2\right)\cos(2\phi) - 6\xi$$
$$c = 10\left(\xi^2 + 4\right)\xi^2 + 4 + 6\left(\xi^2 + 2\right)\xi^2\cos(4\phi)$$
$$-24\left(2\xi^3 + \xi\right)\cos(2\phi)$$

$$\mu\{T_I\} = -\tan^{-1}\left[\frac{a}{b + \sqrt{c}}\right] \tag{2.21}$$
$$a = 2\left(\xi^2 - 1\right)\sin(2\phi)$$
$$b = \left(8\xi^2 + 2\right)\cos(2\phi) - 2\xi^3 - 8\xi$$
$$c = 4\xi^6 + 66\xi^4 + 76\xi^2 + 4 + 10\left(3\xi^2 + 2\right)\xi^2\cos(4\phi)$$
$$-8\left(\xi^2 + 4\right)\left(4\xi^2 + 1\right)\xi\cos(2\phi)$$

The $\xi\{\cdot\}$ measures will yield:

$$\xi\{G_I\} = \frac{\sqrt{a-b}}{c}$$
$$a = 5\left(\xi^2 + 4\right)\xi^2 + 2 + 3\left(\xi^2 + 2\right)\xi^2\cos(4\phi)$$
$$b = 12\left(2\xi^3 + \xi\right)\cos(2\phi)$$
$$c = \sqrt{2}\left(\xi^2 - 3\xi\cos(2\phi) + 2\right)$$
$$\xi\{T_I\} = \frac{\sqrt{a-b}}{c}$$
$$a = 2\xi^6 + 33\xi^4 + 38\xi^2 + 2 + 5\left(3\xi^2 + 2\right)\xi^2\cos(4\phi)$$
$$b = 4\xi\left(\xi^2 + 4\right)\left(4\xi^2 + 1\right)\cos(2\phi)$$
$$c = \sqrt{2}\left(3\xi^2 + 2 - \left(\xi^2 + 4\right)\xi\cos(2\phi)\right)$$

If the surface is isotropic (*i.e.* $\xi = 0$) then $\mu\{T_I\} = \mu\{G_I\} = \phi \bmod 180°$, the correct estimate of the light orientation. For the anisotropic case, a non-zero deviation will occur:

$$\mathrm{dev}_G(\phi) = (\mu\{G_I\} - \phi) \bmod 180°$$

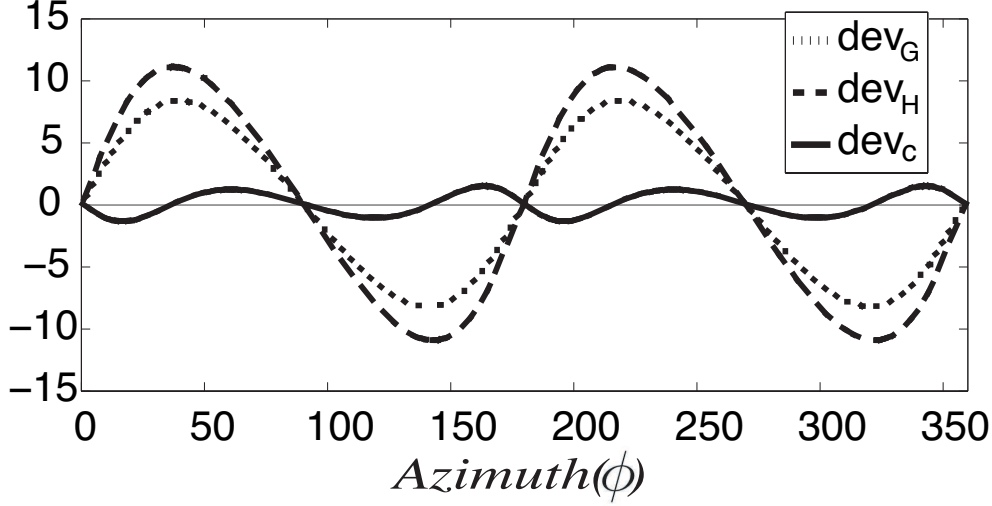with equivalent definition for the Hessian-based estimate of the image $\mathrm{dev}_H(\phi)$.

Figure 2.2: Theoretical predictions for deviations of the three estimates, dev_H (dashed), dev_G (dotted) and dev_C (drawn), as a function of true azimuthal direction of illumination (ϕ), all three for $\xi\{G_h\} = 0.1$.

We have so far assumed that $\mu\{G_h\} = 0$ (i.e. that the orientation of surface anisotropy is along the x-axis). Generalizing to anisotropy of different orientations is easily achieved by shifting the deviations, *i.e.* $\text{dev}_G(\phi - \mu\{G_h\})$. We will continue to assume $\mu\{G_h\} = 0$, knowing that the results are valid for all possible $\mu\{G_h\}$.

The deviations dev_G and dev_H are plotted in Fig. 2.2 for $\xi = 0.1$, a low amount of anisotropy and yet deviations are quite noticeable. This is indeed expected and shows the impact of anisotropy. The expressions for the deviations are very unwieldy which is why we choose to approximate them with a truncated Taylor series around the point $\xi = 0$:

$$\text{dev}_G(\phi) = 3\xi \sin 2\phi + O(\phi^2) \qquad (2.22)$$

$$\text{dev}_H(\phi) = 4\xi \sin 2\phi + O(\phi^2) \qquad (2.23)$$

The aim is to have an estimate with no deviation, so Eq. 2.22 and Eq. 2.23 are good arguments for a measure that combines both the Hessian-based and the gradient-based tensors of the image. A most simple example of such combination would be:

$$\mu_C = 4\mu\{G_I\} - 3\mu\{H_I\} \qquad (2.24)$$

Expressed in terms of its lower order Taylor approximation (equations 2.22-2.23), μ_C will have zero deviation. When the expressions based on equations 2.20-2.21 are used, the resulting function $\text{dev}_C = (\mu_C - \phi) \bmod 180°$ is plotted in Fig. 2.2

While simple and straightforward to derive (given the theory we have presented), the μ_C measure suffers from the drawback of introducing another non-linearity. The multiplicative factors of -3 and 4 can be expected to amplify any errors that are already present in $\mu\{G_I\}$ and $\mu\{H_I\}$. We stress that this chapter focuses on the theory we have presented, i.e. the physics based modeling of the imaging process of anisotropic surface textures. With the combined tensor estimate of Eq. 2.24 we wish to exemplify the use of this theory.

2.4 Connection to other illuminant estimators

To our knowledge, no other local and unsupervised illuminant estimator has been suggested that accounts for surface anisotropy. We will mention a few estimators that are based on the Kube and Pentland modeling, namely those proposed in[29][35][36][9][37]. The premise is a surface texture normally viewed, where a local estimate of ϕ, the illuminant azimuthal angle (tilt) is to be estimated. Iterative non local algorithms, such as[39], that simultaneously estimates the surface texture height profile are not considered, neither are supervised algorithms that deal with a finite set of textures, such as[8].

The early approaches of[36] and[29] are based on means of directional derivatives, and not variances. For an object that is globally a plane, this means that they will not work (both Knill[35] and Chantler[9] discusses this). Many modern approaches still explore the same premiss of a convex or spherical surface (for example[37]).

Two algorithms that deal explicitly with the problem of unsupervised estimation of illumination direction from a textured plane are Knill's[35] and Chantler's[9]. Both of these schemes are equivalent to the gradient based structure tensors, which emphasize the general nature of the theory we have presented.

Chantler's method[9] consists of a circular Fourier series expansion of the power spectrum. This requires the Power spectrum to be collapsed (averaged along the radial frequency direction) into a 1D function in the angular fre-

quency variable (a polar frequency plot). Depending on how this collapsing is done, Chantler's method can be made equivalent to either the gradient or the Hessian tensors (given that equivalent selections of derivative filters and scales are selected for the tensor calculations). In Chantler's method, weighting higher frequencies more gives the Hessian tensor estimate, and less the gradient. From Chantler's perspective, the combined measure that we suggest could be implemented by simultaneously making two polar plots of the same power spectrum, one weighting the higher frequencies more, and then combining the two estimates.

The relationship with Chantlers method is most easily seen through the use of so called complex moments of the local image power spectrum, especially the properties presented in[2]. Structure tensors can be viewed as a circular Fourier series expansion of the image power spectrum (just as in Chantler's method[9]). From the complex moments it is also easy to show that the eigenvector angle $\mu\{G_I\}$ is equivalently calculated as Knill's estimator[35]. Knill, in fact, arrives at the complex moment description explicitly, which was proposed years earlier by Bigun and Granlund[4], thus connecting it for the first time to illumination direction.

2.5 Experiments

An experiment on rendered surfaces was performed to test the theoretical predictions, and to analyze the behavior of the theory for violations of the assumptions in terms of higher relief, non-oblique lighting and larger amounts of anisotropy.

Rendering

The rendering was implemented without any 3D graphics libraries, due to the simple nature of the scene and because we are not aware of any libraries that render surfaces defined with torus topology (described in the following subsection). Images (400x400 pixels) were generated as the orthographic projection of the radiance map of a Gaussian surface illuminated by a collimated beam of light. This was rendered as the product of two entities:

$$I_{rend}(\boldsymbol{r}) = I(h_{01}(\boldsymbol{r}), h_{10}(\boldsymbol{r}), \theta, \phi)S(\boldsymbol{r})$$

where $I(h_{01}(\boldsymbol{r}), h_{10}(\boldsymbol{r})), \theta, \phi)$ is the formulation of the surface radiance for the low relief case(Eq. 2.4) and $S(\boldsymbol{r}) \in \{0, 1\}$ is the binary shadow function

that was calculated in full (0 indicates self-shadowing, for both attached and cast shadows).

Inter-reflections were not rendered (discussed in section 2.8).

Surface Generation

The surfaces were generated randomly in the Fourier domain by using the specified power spectrum and pairing it with a randomly generated phase spectrum of independently generated values from a uniform distribution $\in [0, 2\pi)$. This way of generating the surfaces ensures constancy of the estimated autocorrelation function $\rho_h(\boldsymbol{r})$ and of the height probability distribution function.

As is the case for all signals generated smoothly in the discrete Fourier domain, the surface will be defined on a topological torus, *i.e.* it can be viewed as having infinite area by repeating itself, without losing its stationarity or continuity at any point. This topology is desirable for our purposes as there will be no concern about the surface boundaries (there are none!).

Every aspect of the rendering, surface generation and estimation of statistics was done without violating the torus topology, *i.e.* $(x, y) \rightarrow \{x \bmod width, y \bmod height\}$ when the surface is represented as a finite 2D array.

Setup

Parameters in the experiments were:

- σ: Surface height. Scales the height of the surface linearly.

- θ: elevation angle of the light source.

- ξ: the anisotropy, *i.e.* the value assumed for $\xi\{G_h\}$

For every configuration of these variables a sequence of 360 images is created by varying $\phi \in [0°, 360°)$ in integer steps (two such images of the same surface are shown in Fig. 2.3, for $\phi = 90°$ and $0°$ (left and center of figure respectively)). The Hessian-based and the Gradient-based structure tensors are applied to each image, and the two different estimates of Eq. 2.24, $\mu\{G_I\}$ and $\mu\{H_I\}$ are collected from the eigen analysis outlined in section 2.3. This allows for relatively smooth curves of the deviations from the true estimates to be compared with the theoretical prediction of Fig. 2.2.
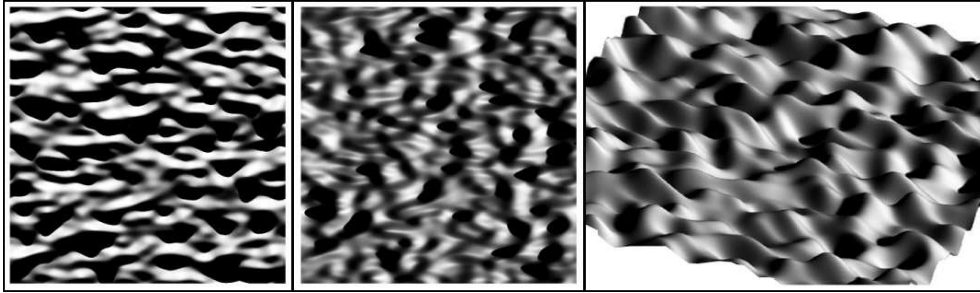
Figure 2.3: A rendered Gaussian surface with $\xi = 0.6$, $\sigma = 100$ and $\theta = 30$. Left: $\phi = 90°$. Middle: $\phi = 0°$. Right: $\phi = 0°$ and viewed from another direction.

2.6 Results

Typical results are illustrated in Fig. 2.4 (each point of the graphs was calculated on images like those in Fig. 2.5). The output is bounded in $[-90°, 90°)$, as it is orientation estimation and not direction estimation. Comparing the top panel of Fig. 2.4 with Fig. 2.2 yields, as expected, a strong resemblance. In this range of the parameters the variation in curves, from one surface to the other, is very low. For the case where the surface is rough, as in the second panel of Fig. 2.4, and the lighting angle is still relatively low, shadows will make a big impact, as is evident from the noisy curves.

In the experiment of the third panel of Fig. 2.4 a large amount of anisotropy is present. All three estimates deviate towards the axis of anisotropy, as expected. Consider, as an extreme case, a completely anisotropic surface ($\xi = 1$); say a sinusoidal wave surface. It is impossible to determine the light orientation from the information given by the irradiance of the wave surface, by analogy with the aperture problem in optical flow. At some point, while anisotropy increases, any algorithm must falter. It is worth noting, however, that the combined estimate still performs noticeably better than separate estimates in the high anisotropy case. Finally, the fourth (bottom) panel of Fig. 2.4 shows what happens when the light elevation is high. The deviation of the combined estimate worsens, but the overall shape of the three curves remains quite consistent with the theoretically predicted output.
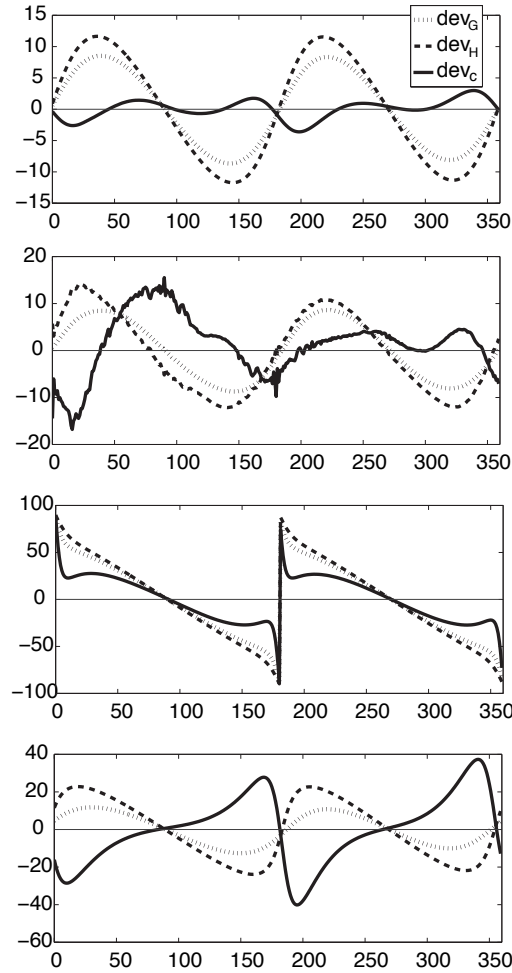
Figure 2.4: Typical outputs from the renderings for dev_H (dashed), dev_G (dotted) and dev_C (drawn). X and Y axes: same as Fig. 2.2 (Note: The ranges of the Y axes are different in the figure). First (Top): close to the assumptions of the theory ($\sigma = 30$, $\theta = 30°$, $\xi = 0.1$). Second: higher relief ($\sigma = 60$, $\theta = 30°$, $\xi = 0.1$). Third: strongly anisotropic surface, ($\sigma = 30$, $\theta = 30°$, $\xi = 0.6$). Fourth (Bottom): high light elevation (non-oblique lighting), ($\sigma = 30$, $\theta = 70°$, $\xi = 0.1$).
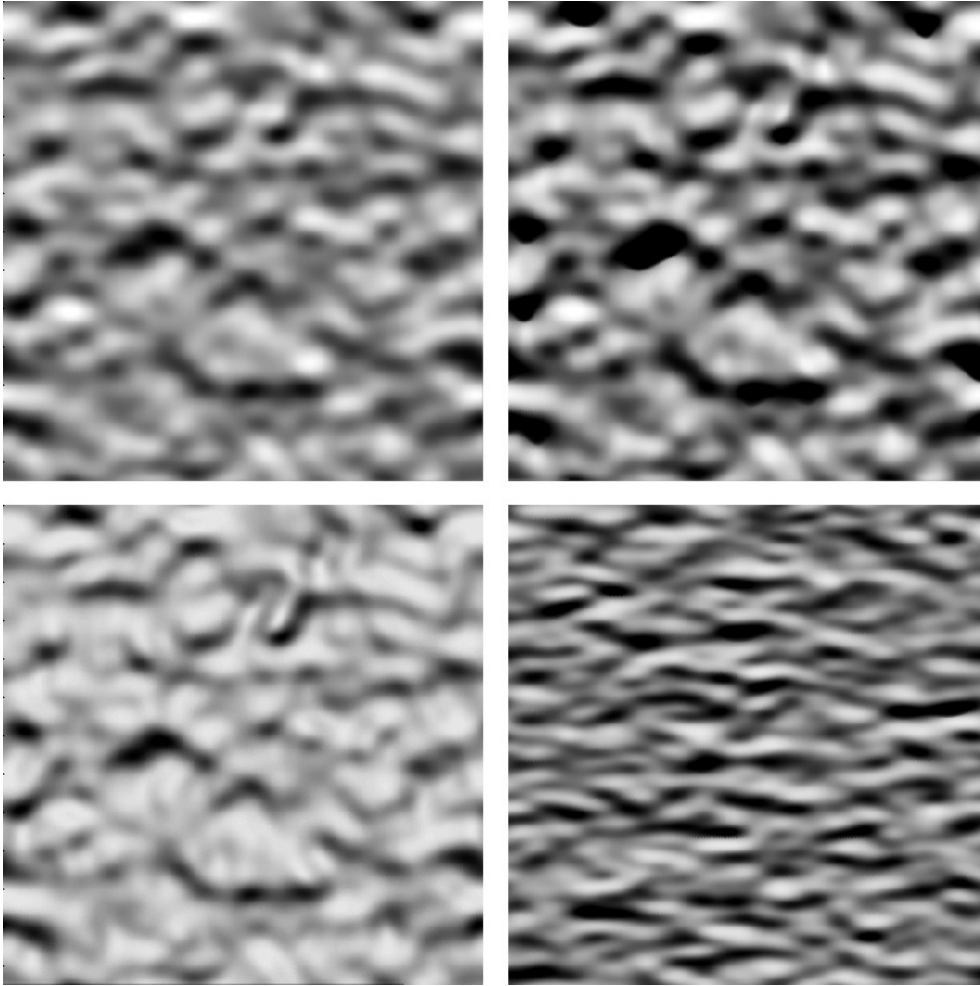
Figure 2.5: Example images corresponding to $\phi = 90°$ (top illumination), each one correspond to one out of 360 images used to draw the graphs in Fig. 2.4. The images have been histogram normalized for visualizing details. Top left: ($\sigma = 30$, $\theta = 30°$, $\xi = 0.1$). Top right: ($\sigma = 60$, $\theta = 30°$, $\xi = 0.1$). Bottom left: ($\sigma = 30$, $\theta = 70°$, $\xi = 0.1$). Bottom right: ($\sigma = 30$, $\theta = 30°$, $\xi = 0.6$).

We generated a large amount of data by varying $\xi \in \{0, 0.1, \ldots, 0.8\}$, $\theta \in \{30°, 35°, \ldots, 80°\}$ and $\sigma \in \{30, 35, \ldots, 100\}$. Furthermore, we averaged the results over 5 independent surfaces. The absolute value of each curve of the type in Fig. 2.4 was averaged, yielding one scalar error measure (average error) for every configuration of θ, ξ and σ. The results are 3D, and thus cannot be plotted in paper form. Fig. 2.6 shows contour plots over pairs of dimensions, while the third dimension is kept fixed, close to the assumptions of the theory.

The average error should be bounded $\in [0°, 45°]$; if it is any higher, the output is worse than random guesses (*e.g.* an average error of 90 indicates an output that is always perpendicular to the true orientation). The main observation to make from Fig. 2.6 is low error and stable monotonic behavior around the point corresponding to the assumptions ($\xi = 0$, $\sigma = 0$ and $\theta \approx 45°$). As we go further away from the assumptions in parameter space, the output gradually becomes worse. We also note that ξ completely dominates as source of error in the plots, with the exception of the elevation angle θ when it comes close to $90°$.

For an accepted error of say $6°$, one should stay within the bounds of $\xi < 0.2$, $\sigma < 65$ and expect even better behavior for oblique lighting of around $\theta = 45°$.

As expected, for large amounts of anisotropy, the measure becomes inaccurate, especially when the illumination orientation is in the direction of the anisotropy axis of the surface (at $\phi = 0°$ and $180°$ in the graphs of Fig. 2.4).

Also expected was the increase in deviation due to higher relief. When the surface is rough (σ is high), and as θ decreases, the surface tends to cast shadows. Shadows affect the outcome of the tensor estimates, because they give rise to new edges in the image. Shadow edges are non-local and more noisy with respect to ϕ changes than are regular shading patterns. On the other hand, as θ increases it comes closer to $90°$ where the problem is undefined.

For high values of θ, the modeling of the imaging process as a directional derivative is less accurate; other structure becomes evident as is seen in Fig. 2.5. The images on the left of Fig. 2.5 are images of the same surface, but only the top one, with low θ, resembles a directional derivative. The structure of the top image resembles more the square magnitude of the gradient, and is often referred to as second-order shading. Second-order shading is independent of the orientation of illumination, and will inherit anisotropy from the surface (if the surface contains anisotropy, then so will the square

magnitude of its gradient).

For anisotropic surfaces with a higher relief, we expect to see higher average errors when θ approaches $0°$ and $90°$, due to shadows and second-order shading respectively. This explains the concave shape of the lower graph of Fig. 2.6 as σ increases, varying θ; oblique lighting at around $\theta = 45°$ will allow for greater violations of the assumptions.

We mentioned in section 2.2 that we assumed (for convenience) Gaussian shaped power spectrum for the surface. All the renderings so far has been on such Gaussian surfaces. Preliminary experiments were performed on Brownian surfaces[30], i.e. power spectrum of the shape $\hat{\rho}_h(\boldsymbol{k}) = \boldsymbol{k}^T \boldsymbol{k}^{-\frac{\beta}{2}}$ where we used $\beta = 2$. Anisotropy was introduced in the same way as for the Gaussian surfaces, i.e. as a binomial form with the structure tensor: $\hat{\rho}_h(\boldsymbol{k}) = (\boldsymbol{k}^T G_h^{-1} \boldsymbol{k})^{-1}$. The Brownian surfaces needed to be smoothed to be implemented (the roll-off frequencies were suppressed by multiplying the power spectrum with a gaussian with identical G_h). Preliminary results on the anisotropic fractal surfaces were identical to that of the regular anisotropic Gaussian ones.

2.7 Real-world textures

The question naturally arises of how the combined measure will fare on real-world textures. We tested it on a selection of the photex database[6][5]. From this database, we selected 26 textures, namely: aaa, aab, aaf, aai, aaj, aam, aan, aao, aap, aar, aas, aba, abj, abk, acc, acd, acf, acg, ack, acl, acm, afa, afb, afd, afe and afg. We picked these because they appear to have only slight variations in albedo, with diffuse reflectance and varying degrees of anisotropy. We didn't tweak any of the many hidden parameters (such as implementations of derivatives) for the benefit of the combined estimate, nor did we pre-process the images in any way. We ran the algorithm with the same parameters as for the rendered textures, except that we now implemented smooth local averaging windows (2 by 2 as seen in Fig. 2.7). Fig. 2.7 illustrates typical outputs for the texture 'aab' (one of the textures where the combined estimate performed noticeably better).
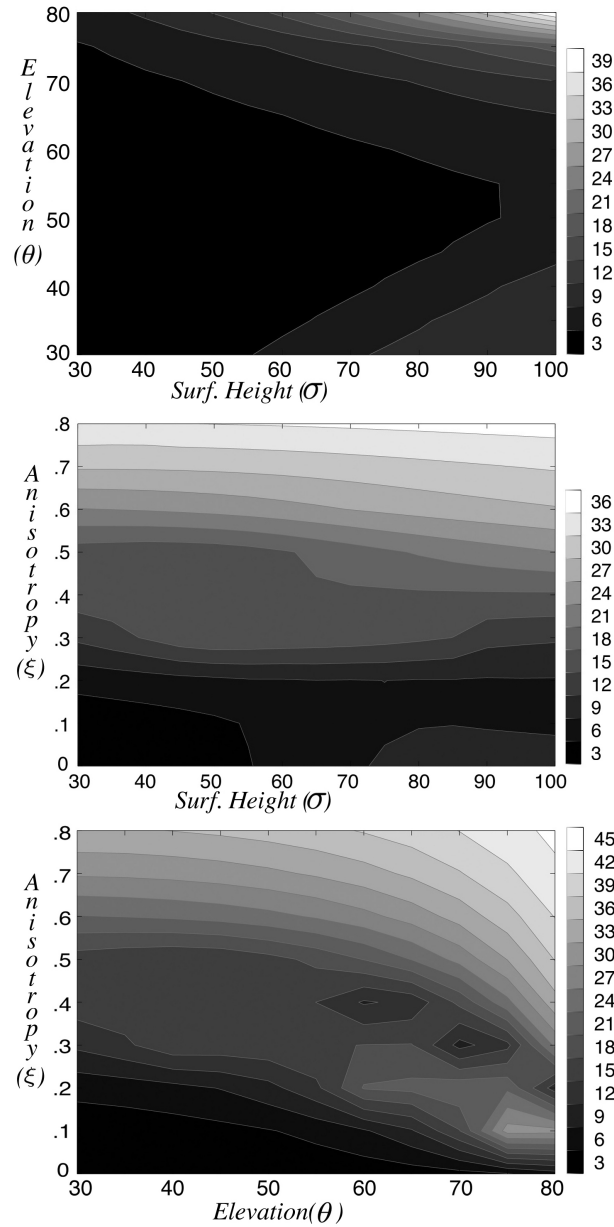
Figure 2.6: Contour plots of the average magnitude deviations. The absolute value of each curve of the type in Fig. 2.4 is averaged (one non-negative scalar for each curve). The gray value of each region indicates an upper bound on the error, specified by the bars on the right of each figure. Top: θ vs. σ with $\xi = 0.1$. Middle: ξ vs. σ with $\theta = 30°$. Bottom: ξ vs. θ with $\sigma = 30$.
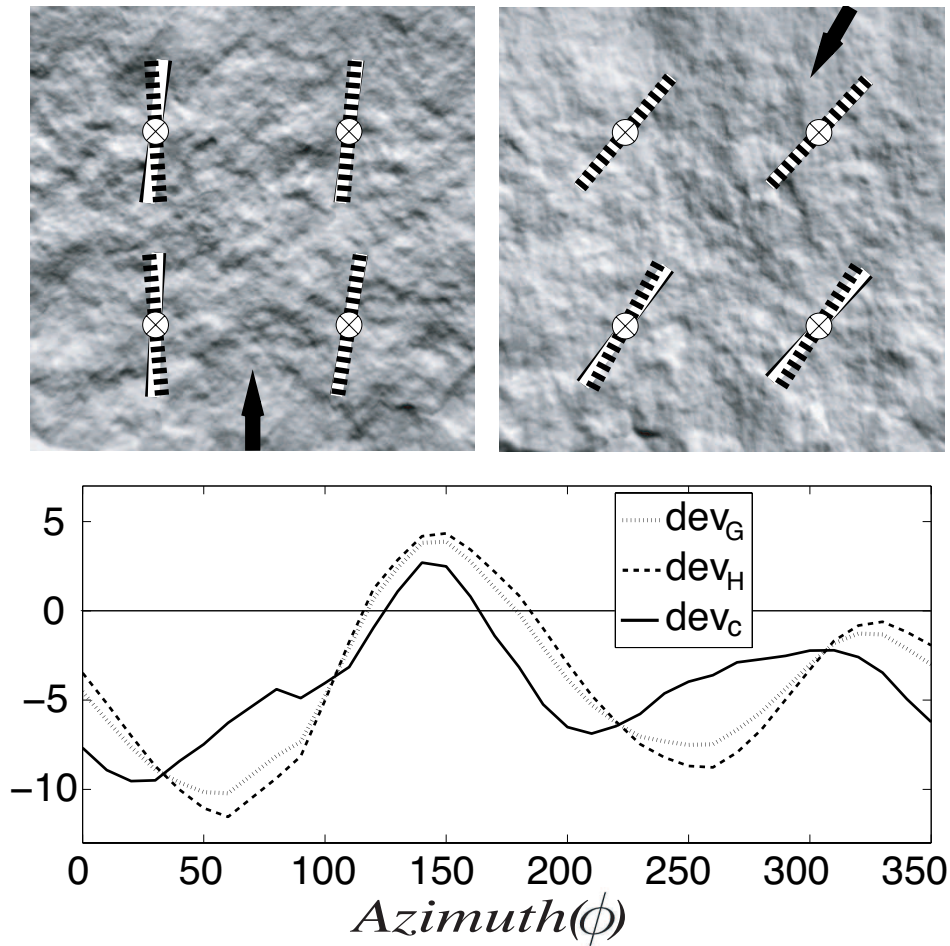
Figure 2.7: Example of results from real-world textures. Top row: two images of the texture 'AAB' with light elevation $(\theta) = 45°$, they vary in azimuthal light angle (ϕ). Dotted line is the combined measure, black line the gradient and white line the Hessian, black arrow indicates global light direction, top left: $\phi = -90°$top right: $\phi = 60°$; Bottom graph shows the results for the 36 different images corresponding to equidistant sampling of ϕ.

Noteworthy is that ground truth for ϕ is slightly inaccurate for local estimates. The black arrows in Fig. 2.7 illustrates the direction of illumination as indicated by the photex database, but the light source was likely positioned quite close to the texture, resulting in slight variation in the illuminance flow field over the surface. Each texture has a great number of variations in light direction (both azimuthal and elevation) and we average over the entire collection.

The average error for the three estimators are: Gradient based: 13.3°, Hessian based: 11.4°, and Combined estimate: 17.2°. Thus the combined estimate cannot be expected to perform better than the gradient or the Hessian by themselves in the real world. This error amplification is due to the multiplicative factors of -3 and 4 evident in Eq. 2.24. Any error stemming from non-lambertian, higher-relief, oblique viewing etc. will be amplified. Also, errors due to image noise, camera transfer function (from irradiance to image intensity) will all give rise to deviations that are amplified by the combined measure. Compared to the rendered surfaces, the real-world textures most certainly deviates more from the model, giving rise to more unstable behavior. A hint of this error amplification is also seen by the results on the rendered textures when strong shadows occur. In the second graph of Fig. 2.4, one can just make out small erratic deviations from the predictions for the gradient and the Hessian, yet these small erratic deviations are made very noticeable when they reemerge in the combined estimate.

2.8   Conclusions

The theory of illuminance flow estimation has been successfully generalized to the case of small amounts of surface anisotropy. This has been done by introducing structure tensor analysis of the surface (not just of the image, as was done previously). The surface structure tensors are not observable, yet their output (which are measures of surface geometry) will affect the output of the image tensors; the form of this dependence is described by our theory. The theory is based on a simple approximation of the imaging process of the surface (Eq. 2.7), valid under the assumptions of a low relief isotropic Lambertian surface of uniform albedo, viewed normally in a light field[52] of one dominant directional component (*e.g.* a collimated beam).

A new measure, μ_C, which combines the image tensor outputs (Eq. 2.24), has been devised based on the theory. The predicted output has been compared to results from computer rendered Gaussian surfaces, where it has been

found to be stable for small deviations from the assumptions of low-relief, and isotropy (as predicted), especially when the light elevation is oblique (say close to 45° or so). Results on real-world textures show the behavior for larger deviations from the model where the combined estimate performs worse than the gradient or the Hessian by themselves. This is because of the error amplification due to the multiplicative factors of -3 and 4 (as seen in Eq. 2.24). The combined estimate is a non-linear transform of the original vectors (seen as complex numbers, we take them to the power of -3 and 4 to change their angle) that propagates and amplifies the error. More stable estimates based on our theory can be derived, and is a worthy subject of future work.

Inter-reflections are not factored into the approximation due to the low-relief assumption. For the statistics we investigate here, under the Lambertian assumption, the inter-reflections should have only a minor effect. This is especially true if the surface has low albedo, that is if some amount of light is absorbed upon each reflection (most natural surfaces have a rather low albedo, with a reported average of 0.2[31]). Even with unit albedo and high relief, however, there is no indication that incorporating more reflections will change the anisotropy of the resulting image. For isotropic Lambertian surfaces one can show[24] that inter-reflections will not affect the output of the tensors, and thus (in theory) not affect the output of the tensors.

In real scenes one expects surfaces to have non-uniform albedo, non-Lambertian reflectances and to be viewed from an arbitrary direction; therefore these cases will be discussed briefly.

If the albedo is non-uniform, we are faced with an additional 2D signal: an albedo map $a(\boldsymbol{r})$. The results of our findings will still hold if we relax the non-uniform albedo assumption to the case of $a(\boldsymbol{r})$ being uncorrelated with the surface $h(\boldsymbol{r})$, and the scale of the contents being sufficiently different, *i.e.* it boils down to a scale selection question: outer scale defining areas of interest, inner scale defining size of derivative filters.

For non-Lambertian surfaces, we consider the Bidirectional Reflectance Distribution Function (BRDF))[56] denoted: $B(\boldsymbol{v}, \boldsymbol{n}, \boldsymbol{i})$. Here, $\boldsymbol{v}$ and $\boldsymbol{i}$ indicate the viewing and light directions and $\boldsymbol{n}$ the local normal. A Lambertian BRDF is constant whereas a perfectly reflective surface (to which the present analysis is not intended) has a BRDF equalling a weighted δ - function. The local radiance due to a collimated beam is $I^{(B)}(x, y) = B(\boldsymbol{v}, \boldsymbol{n}(x, y), \boldsymbol{i})I(x, y)$, where I is the irradiance of the surface given by Eq. 2.4. So, for a non-constant BRDF the structure of the surface radiance will be similar to that

of a Lambertian surface if $\|\nabla_n B(\boldsymbol{v}, \boldsymbol{n}, \boldsymbol{i})\| \leq B(\boldsymbol{v}, \boldsymbol{n}, \boldsymbol{i})$. Thus, the Lambertian assumption can be relaxed to a smooth BRDF, which applies to most natural materials.

The issue of non-normal views is important in applications and has been investigated by Pont and Koenderink[11]. They showed that, in theory, the illuminance orientation estimates can be corrected for oblique views using surface attitude estimates, up to a viewing angle of 55 degrees. For viewing angles larger than 55 degrees there are no unique solutions. For the case where the surface is anisotropic and non-obliquely viewed one could model camera deviations from the normal position as an affine transformation of the (low relief) surface, that is, approximating the rotation as foreshortening along an axis. Similarly, surface anisotropy could be introduced as an affine transformation of an originally isotropic surface. Thus, we could model the non-normal viewing of an anisotropic surface as the composition of two affine transformations (which is itself affine) applied to an isotropic surface. Therefore, we expect that the combined effects of surface anisotropy and obliqueness will affect the illuminance flow estimates in a very systematic way. However, it cannot be corrected for due to the fact that the individual effects on the radiance structure are very similar (or even the same in the low relief approximation). This will be a subject of our future work.

To conclude; we have described the second-order statistics of anisotropic surfaces viewed under directed illumination. We have shown theoretically and experimentally that the anisotropy of a surface can be corrected for, for small amounts of anisotropy, using a combination of the Hessian-based and gradient-based structure tensors. Finally, we discussed physical arguments for the generalization of our results far beyond our rather restrictive assumptions.

Illuminance Flow over Anisotropic Surfaces with Arbitrary Viewpoint

abstract

The theory of illuminance flow estimation by structure tensors is generalized for oblique viewing of anisotropic texture. An added benefit is that the theory predicts the behavior of unsupervised illuminant tilt estimators. The previous theory is refined with general matrix formulations and compacted by exploiting general properties of the structure tensor. Theoretical predictions based on the revised theory are presented, and compared with experimental results on rendered images. The predicted curves are shown to conform well to the expectations when the deviation from normal viewing and surface anisotropy are not large.

3.1   Introduction

The light field in which objects are situated is important in machine vision. Often slight changes in illumination produce large changes in an object's appearance, especially with regard to finer scale 3D corrugations across the object's surface (so-called 3D or surface texture). In this chapter, we investigate illuminated anisotropic surfaces viewed from arbitrary directions, and try to infer the so-called illuminance flow. Image illuminance flow is a 2D vector field in the image-plane that results from projecting the light vector first into the tangential plane of the surface, and then into the image-plane. Fig. 3.1 shows ground-truth from a sphere.

This work is in line with the study by Pont and Koenderink[10] in which they presented a theory for analyzing the illumination orientation of corrugated surfaces based on the structure tensor (2nd moment matrix)[4]. Generalizations of this theory to both oblique viewing[11] and to anisotropic surfaces[12] have been done, but do not account for the effects occurring simultaneously (which is the focus of the current chapter). In addition, we will refine previous and current theory by use of general matrix formulations and achieve compact expressions that are equivalent to the eigensystem of the tensor. The derivation of the theory will be done in a more intuitive fashion with greatly improved compactness and insight.

When the light field has a dominant directional component (collimated beams being the extreme case) anisotropy will be imposed in the image of the surface. In some cases this can be used to estimate the illuminance flow. It has been shown[10] that using the structure tensor of either the Hessian or the gradient yields good estimates for many real-world surfaces. However, formally, this only applies under a set of assumptions, namely: isotropic surface, normal view, Lambertian reflectance, uniform albedo, low relief surface and a light field of dominant directional component. All assumptions are important in their own right. Some work has been done on extending the theory to account for non-uniform albedo[33]. In this chapter, we focus on the isotropy and normal viewing assumptions simultaneously, which, as far as we know, have been given attention only when considered separately [11][12]. Experimental results have been presented for real world surfaces[10]; the structure tensors yielded promising results, with orientation estimates within a few degrees of the veridical orientation. However, for obliquely viewed, anisotropic surfaces, the structure tensors will yield deviations from the true orientation of the illuminance flow. The theory presented
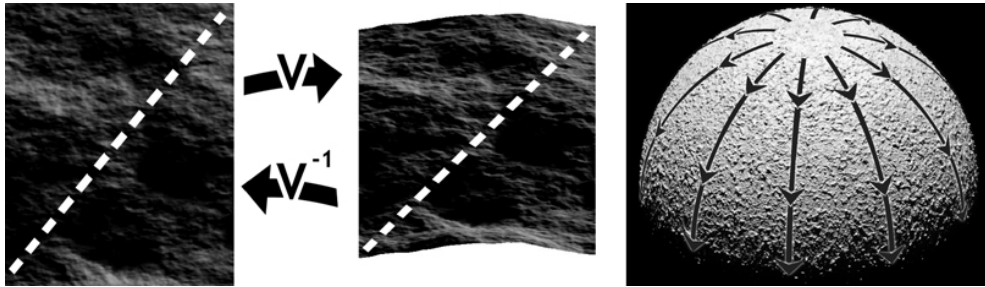
Figure 3.1: Left: illustration of the modeling of oblique viewing by affine transform V. Dotted line illustrates illuminance flow. Right: illuminance flow across the surface of a textured sphere illuminated by a distant point source. Flow lines are the projection of the light-vector into the local tangential plane of the surface.

here predicts these deviations, which are shown to conform well to results of experiments on rendered surfaces.

A connected topic is that of illuminant tilt estimation[35][9][38]. The premise for tilt estimation is traditionally a surface texture normally viewed, where the illuminant azimuthal angle (tilt) relative to the camera frame is to be estimated. Illuminance flow is relative to the tangential frames of the object (e.g. the sphere of Fig. 3.1). Illuminant tilt and illuminance flow are essentially different, except in the case where the surface texture is normally viewed, and illuminant tilt is estimated in a local fashion.

Interestingly, all local and unsupervised illuminant tilt estimators that we are aware of, correlate strongly with the structure tensor. They can be seen as identical to the structure tensor in theory, and differences between them are due to implementation parameters (discussed in[12]). Therefore, as a side-effect, the theory presented in this chapter will also predict how local and unsupervised illuminant tilt estimators will behave as anisotropy is introduced into the texture and/or the texture is tilted with respect to the camera.

3.2   Theoretical background

We will use the nabla operator $\nabla_r\{h(\boldsymbol{r})\} = \{h_x(x,y), h_y(x,y)\}^{\mathrm{T}}$, where the subscript indicates the vector variable (here $\boldsymbol{r} = \{x,y\}$) subject to differentiation. The Hessian is $H_r = \nabla\nabla_r^{\mathrm{T}}\{h(\boldsymbol{r})\}$.

We will use the notation $\langle h(\boldsymbol{r})\rangle$ to denote averaging over the function

$h(\boldsymbol{r})$. If $h(\boldsymbol{r})$ is vector-valued or a matrix, the averaging is applied on each element independently. $|A|$ and $\operatorname{tr} A$ will be used to denote the determinant and the trace of the matrix A. Diagonal and rotation matrices will be denoted $D_{a,b}$ and R_μ respectively.

Gaussian Surfaces and Structure Tensors

Henceforth $h(\boldsymbol{r})$ will denote an isotropic surface height function, modeled as a 2D, zero-mean, differentiable, stationary, Gaussian stochastic process (this family of surfaces has several convenient properties[50][51].

The Brownian fractal surface is an example of a Gaussian surface that is not differentiable and thus, strictly speaking, does not fit the model. However, the theory presented here is applicable to near Brownian surfaces, which is in line with many other works[30][31].

The surface covariance function of the isotropic surface, $\rho_h(\boldsymbol{r}) = \operatorname{Cov}(h(\boldsymbol{0}), h(\boldsymbol{r}))$, is rotationally symmetric, i.e. there exists some 1D function $f(t)$ such that

$$\rho(\boldsymbol{r}) = \sigma_h^2 f(\boldsymbol{r}^{\mathrm{T}} \boldsymbol{r})$$

where $f(0) = 1$ and $\sigma_h^2 = \langle h^2 \rangle$ is the point-wise variance of the surface. By definition, $\rho_h(\boldsymbol{r})$ is constant within its ensemble. Now, let a linearly transformed surface be denoted $h(A\boldsymbol{r})$, for a 2×2 matrix A, that is real-valued and has a positive determinant (i.e. $A \in GL_2^+(\mathbb{R})$). It is easy to show that the covariance function of $h(A\boldsymbol{r})$ is:

$$\rho(\boldsymbol{r}) = \sigma_h^2 |A| f(\boldsymbol{r}^{\mathrm{T}} A^{\mathrm{T}} A \boldsymbol{r}) = \sigma_h^2 |G_h|^{1/2} f(\boldsymbol{r}^{\mathrm{T}} G_h \boldsymbol{r}) \tag{3.1}$$

where $G_h = A^{\mathrm{T}} A$ is the gradient-based structure tensor[4], a matrix holding the information about anisotropy amount and direction. The structure tensor is known by many names, such as the 2nd moment matrix or the directionality tensor. The structure tensor is given by:

$$G_h = \left\langle \begin{array}{cc} h_x h_x & h_x h_y \\ h_x h_y & h_y h_y \end{array} \right\rangle$$

It is convenient to consider an auto-covariance function of the type in Eq. 3.1, and especially a Gaussian-shaped covariance function: $\rho_h(\boldsymbol{r}) = \sigma_h^2 \exp\left(-\boldsymbol{r} G_h \boldsymbol{r}^{\mathrm{T}}\right)$. As long as the elements of the tensor are defined, the theory is applicable.

The eigenvalues/eigenvectors of G_h yield the amount and orientation of anisotropy of the surface. We denote the maximum and minimum eigenvalues and their corresponding eigenvectors as $\lambda_{G_h}^{max}$, $\lambda_{G_h}^{min}$, $\boldsymbol{v}_{G_h}^{max}$ and $\boldsymbol{v}_{G_h}^{min}$. The surface confidence is a measure of anisotropy defined as:

$$\xi_h = \frac{\lambda_{G_h}^{max} - \lambda_{G_h}^{min}}{\lambda_{G_h}^{max} + \lambda_{G_h}^{min}} \in [0, 1)$$

$\xi_h = 0$ occurs for a completely isotropic surface, whereas $\xi_h = 1$ indicates degeneracy (the surface contains only parallel isolines). The orientation angle of anisotropy is given by the highest eigenvalue eigenvector and denoted:

$$\mu_h = \angle \boldsymbol{v}_{G_h}^{max} \in [0°, 180°)$$

In surface science and tribology, it is common to use either the surface height standard deviation ($\sigma_h = \langle h^2 \rangle^{1/2} = \rho(\boldsymbol{0})^{1/2}$) or the root mean square slope ($\sigma_{\nabla h} = \langle |\nabla h|^2 \rangle^{1/2} = \nabla^2 \rho(\boldsymbol{0})$) as surface roughness parameters. Denote with $\sigma_h(G_h)$ and $\sigma_{\nabla h}(G_h)$ the roughness parameters for a varying G_h. The isotropic case occurs when the structure tensor is the identity matrix ($G_h = \mathcal{I}$), thus $\sigma_h(\mathcal{I}) = \sigma_h$. It is easily shown that

$$
\begin{aligned}
\sigma_h^2(G_h) &= \sigma_h^2 |G_h|^{1/2} \\
\sigma_{\nabla h}^2(G_h) &= \sigma_{\nabla h}^2 |G_h|^{1/2} \operatorname{tr} G_h / 2
\end{aligned}
\tag{3.2}
$$

Thus, introducing directionality in a surface by G_h, will preserve the roughness measure σ_h only if $|G_h| = 1$. The roughness measure $\sigma_{\nabla h}$, on the other hand, will remain constant if $|G_h|^{1/2} \operatorname{tr} G_h / 2 = 1$. The only valid matrix that can fix both roughness parameters simultaneously is the trivial $G_h = \mathcal{I}$. Assuming a constant roughness of the surface, the structure tensor is given uniquely by ξ_h and μ_h by:

$$G_h = (1 - \xi_h^2)^{-a} R_{\mu_h}^{\mathrm{T}} D_{1-\xi_h, 1+\xi_h} R_{\mu_h} \tag{3.3}$$

where a is dependent on which roughness parameter is fixed ($a = {}^1\!/_2$ to constrain σ_h and $a = {}^1\!/_4$ for $\sigma_{\nabla h}$). Eq. 3.3 is verified by the trace and the determinant of the expression (these give the change in roughness by Eq. 2).

3.3   Illuminated Gaussian surfaces

In this section we will first reformulate previous theory about illuminance flow estimation (as presented in chapter 2) into a more compact and intuitive form, based on matrix formulations and the spectral theorem (Eq. 3.7).

We will then proceed to introduce both anisotropy and oblique viewing as affine transforms, finally arriving at a closed form approximation of how the directionality of the image stems from illumination, surface anisotropy and oblique viewing (Eq. 3.13).

Isotropic Normally Viewed surfaces

The radiance by collimated beams (one point source at infinity) is described by a single vector:

$$\boldsymbol{\ell}_{3D} = \{\cos\theta\cos\phi, \cos\theta\sin\phi, \sin\theta\}$$

where θ and ϕ are the illuminant incidence and azimuthal angles in the camera frame respectively. For the problem of illuminant tilt estimation, ϕ is to be estimated from an image of a surface texture.

The assumption of collimated beams is convenient for the derivation, but not critical since extended sources can be exactly accounted for by equivalent collimated sources if vignetting[1] is neglected. The following theory holds also for light-fields where a dominant directional component is present; the $\boldsymbol{\ell}_{3D}$ vector can be replaced by this dominant direction.

The normal of a surface can be written:

$$\boldsymbol{n}(h_x, h_y) = \frac{\{h_x(\boldsymbol{r}), h_y(\boldsymbol{r}), -1\}^{\mathrm{T}}}{\left(h_x^2(\boldsymbol{r}) + h_y^2(\boldsymbol{r}) + 1\right)^{1/2}}$$

The irradiance of the surface is $q(h_x, h_y) = \boldsymbol{\ell}_{3D}^{\mathrm{T}} \cdot \boldsymbol{n}(h_x, h_y)$. Under the assumptions listed in the introduction, the irradiance will be proportional to the image of the surface (assuming a linear camera transfer).

Irradiance is obviously not linear with respect to height scaling of the surface. We let ε be a height scaling factor, then the irradiance will be $q(\varepsilon h_x, \varepsilon h_y)$. Because we are interested in low-relief texture, we use an approximation through a Taylor series expansion around $\varepsilon = 0$. By ignoring the zeroth order term, and including first and second orders, the approximation becomes:

---

[1]Vignetting is the effect of the light source being partially obstructed, and the surface patch will not receive the full amount of possible irradiance. If the light source is a collimated source, then vignetting is equal to shadowing. If the light source is diffuse, then vignetting will result in variations of surface radiance that is often confused as being shading.

$$q(\varepsilon h_x, \varepsilon h_y) = \varepsilon\, \boldsymbol{g}^{\mathrm{T}} \boldsymbol{\ell} \cos\theta + \frac{\varepsilon^2}{2} \boldsymbol{g}^{\mathrm{T}} \boldsymbol{g} \sin\theta + \mathrm{O}(\varepsilon^3) \qquad (3.4)$$

where $\boldsymbol{g} = \nabla_r h(\boldsymbol{r})$, and $\boldsymbol{\ell}$ is the normalized 2D tangential component of $\boldsymbol{\ell}_{\mathrm{3D}}$ in the xy-plane ($\boldsymbol{\ell} = \{\cos\phi, \sin\phi\}^{\mathrm{T}}$). This approximation is a separation into so-called first and second order shading. First order shading depends on the tangential component of the incident light and behaves as a directional derivative of the surface height profile. Second order shading depends only on the normal component of the incident light and behaves as the square magnitude of the gradient of the height. As the incidence angle θ becomes lower, first order shading will dominate, whereas for θ closer to 90°, second order shading will dominate. Simultaneously, as θ goes to zero, cast shadows will emerge, which this model does not take into account. In line with previous work[10][11][12], for purposes of directionality estimation, we will model the image as first order shading, i.e. as a directional derivative of the height profile in the direction of $\boldsymbol{\ell}$, which is thus valid only for oblique lighting:

$$I(\boldsymbol{r}) = \nabla_r^T \left\{ h(\boldsymbol{r}) \right\} \boldsymbol{\ell} \qquad (3.5)$$

The surface texture plane (object tangent-plane) will not be aligned with the camera plane in general, which means that $\boldsymbol{\ell}$ (which is described in the camera frame) is not illuminance flow. A 2×2 matrix V encodes the foreshortening transform due to oblique viewing, as illustrated in Fig 3.1. The illuminance flow vector $\hat{\boldsymbol{\ell}}$ is then given by the relation $\hat{\boldsymbol{\ell}} = V\boldsymbol{\ell}$. We will refer to the same entities of the image $I(\boldsymbol{r})$ as we did of the surface, but will append I as subscript (e.g. the auto correlation function ρ_I, and the structure tensor G_I).

The gradient of the image is given by:

$$\nabla_r I(\boldsymbol{r}) = \nabla \nabla_r^T \left\{ h(\boldsymbol{r}) \right\} \boldsymbol{\ell} = H_r \boldsymbol{\ell}$$

The structure tensor of the image is given by

$$G_I = \left\langle \nabla_r \{I\} \nabla_r^{\mathrm{T}} \{I\} \right\rangle = \left\langle H_r \boldsymbol{\ell} \boldsymbol{\ell}^{\mathrm{T}} H_r \right\rangle \qquad (3.6)$$

From earlier results, we know that in this case (normally viewed, isotropic surface), the structure tensor yields the orientation of illumination[10], as its greatest eigenvector (i.e. $\boldsymbol{v}_{G_I}^{max} = \boldsymbol{\ell}$), with a ratio of eigenvalues: $\lambda_{G_I}^{max} = 3\lambda_{G_I}^{min}$. This relation was both derived analytically, and verified experimentally in[10]. The same relation can be shown to follow directly

from Eq. 3.6 assuming H_r is the Hessian of an isotropic surface (note that this says nothing about a specific H_r for a given $\boldsymbol{r}$, but only about the distribution of H_r viewed stochastically). By virtue of the spectral theorem $\left(G = \sum \lambda_i \boldsymbol{v}_i \boldsymbol{v}_i^T\right)$, and the fact that the tensor is positive definite ($(\lambda_{G_I}^{max}, \lambda_{G_I}^{min}) > 0$) and symmetric ($\boldsymbol{v}_{G_I}^{max} \perp \boldsymbol{v}_{G_I}^{min}$), we can write:

$$G_I \propto \left(3\boldsymbol{\ell}\boldsymbol{\ell}^{\mathrm{T}} + \boldsymbol{\ell}_\perp \boldsymbol{\ell}_\perp^T\right) \tag{3.7}$$

where $\boldsymbol{\ell}_\perp = R_{\pm 90°}\boldsymbol{\ell}$. If we expand Eq. 3.7, we get the form originally presented[10]:

$$G_I \propto \begin{pmatrix} 2 + \cos 2\phi & \sin 2\phi \\ \sin 2\phi & 2 - \cos 2\phi \end{pmatrix}, \tag{3.8}$$

Arriving at Eq. 3.7 we have an alternative way of deriving the previous theory and have a form that is more open to generalizations to anisotropic surfaces and oblique viewing, as we will show.

Anisotropic Obliquely Viewed surfaces

We will approximate anisotropy and oblique viewing by affine transforms on the surface and the image respectively. This will only be valid under small deviation of the camera from normal view (in addition to the assumptions of the introduction). Let the image of an isotropic surface, transformed by an invertible matrix V be denoted $I^V(\boldsymbol{r}) = I(V\boldsymbol{r}) = \nabla_{Vr}^T \{h(V\boldsymbol{r})\} \boldsymbol{\ell}$, for which the gradient will be:

$$\nabla_r I^V(\boldsymbol{r}) = V^{\mathrm{T}} H_{Vr} \boldsymbol{\ell}$$

The term H_{Vr} is the Hessian of an isotropic surface at transformed coordinates $\boldsymbol{k} = V\boldsymbol{r} = \{u, v\}^{\mathrm{T}}$, i.e. :

$$H_{Vr} = H_k = \begin{pmatrix} \frac{\partial^2 h(u,v)}{\partial u^2} & \frac{\partial^2 h(u,v)}{\partial u \partial v} \\ \frac{\partial^2 h(u,v)}{\partial u \partial v} & \frac{\partial^2 h(u,v)}{\partial v^2} \end{pmatrix},$$

The structure tensor becomes:

$$G_I^V = V^{\mathrm{T}} \left\langle H_{Vr} \boldsymbol{\ell}\boldsymbol{\ell}^{\mathrm{T}} H_{Vr} \right\rangle V \tag{3.9}$$

The averaging is taken over $\boldsymbol{r}$, and in Eq. 3.9, the expressions are a function of $V\boldsymbol{r}$. However, because the averaging is done for the entire domain $\mathbb{R}^2$ this will not matter. For practical applications, when the structure tensor

is taken over apertures (averaging windows) in an image, this becomes an important issue connected to scale selection. For now, we assume an infinite surface and can, analogous to Eq. 3.7, write:

$$G_I^V = V^{\mathrm{T}} G_I V = V^{\mathrm{T}} \left(3\boldsymbol{\ell}\boldsymbol{\ell}^{\mathrm{T}} + \boldsymbol{\ell}_\perp \boldsymbol{\ell}_\perp^{T} \right) V \tag{3.10}$$

Note that $\boldsymbol{\ell} = \{\cos\phi, \sin\phi\}^{\mathrm{T}}$ is still with respect to the camera frame. The transformation to the plane of the texture (the tangential frame of the object) is contained in V (as illustrated by Fig. 3.1) and will be applied shortly.

When introducing anisotropy in the surface by an affine transform ($h(A\boldsymbol{r})$, for isotropic surface $h(\boldsymbol{r})$ as outlined in section 3.2), we get:

$$\nabla_r^T \{h(A\boldsymbol{r})\} = \nabla_{Ar}^T \{h(A\boldsymbol{r})\} A$$

$$\nabla \nabla_r^T \{h(A\boldsymbol{r})\} = A^{\mathrm{T}} H_{Ar} A$$

Let the image of an anisotropic surface be denoted I^A, and we have

$$\nabla_r I^A(\boldsymbol{r}) = \nabla \nabla_r^T \{h(A\boldsymbol{r})\} \boldsymbol{\ell} = A^{\mathrm{T}} H_{Ar} A \boldsymbol{\ell}$$

The structure tensor becomes:

$$G_I^A = A^{\mathrm{T}} \left\langle H_{Ar}(A\boldsymbol{\ell})(A\boldsymbol{\ell})^{\mathrm{T}} H_{Ar} \right\rangle A \tag{3.11}$$

Analogous to Eq. 3.7, we write:

$$G_I^A = A^{\mathrm{T}} \left(3\tilde{\boldsymbol{\ell}}\tilde{\boldsymbol{\ell}}^{\mathrm{T}} + \tilde{\boldsymbol{\ell}}_\perp \tilde{\boldsymbol{\ell}}_\perp^{T} \right) A$$

where $\tilde{\boldsymbol{\ell}} = A\boldsymbol{\ell}$ and $\tilde{\boldsymbol{\ell}}_\perp = R_{\pm 90°} A\boldsymbol{\ell}$. Since $A^{\mathrm{T}} A = G_h$ and $A^{\mathrm{T}} R_{\pm 90°} A = |G_h|^{1/2} R_{\pm 90°}$, we get[2]:

$$G_I^A = 3 G_h \boldsymbol{\ell}\boldsymbol{\ell}^{\mathrm{T}} G_h + |G_h| \boldsymbol{\ell}_\perp \boldsymbol{\ell}_\perp^{T} \tag{3.12}$$

We combine equations 3.10 and 3.12 to have anisotropy and oblique viewing simultaneously:

$$G_I^{(A,V)} = V^{\mathrm{T}} \left(3 G_h \boldsymbol{\ell}\boldsymbol{\ell}^{\mathrm{T}} G_h + |G_h| \boldsymbol{\ell}_\perp \boldsymbol{\ell}_\perp^{T} \right) V \tag{3.13}$$

---

[2]The relation $A^{\mathrm{T}} R_{\pm 90°} A = |G_h|^{1/2} R_{\pm 90°}$ is easily derived by putting $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ and $R_{\pm 90°} = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$ and performing the calculation.

For compact scalar predictions, we choose to limit the matrix V to be given by only one degree of freedom: ω, the amount of oblique viewing (the angular deviation from normal position of the camera). The direction of oblique viewing (axis in the image along which the foreshortening will occur) will be fixed along the y-axis. V will then be given by $V = D_{1,sec(\omega)}$. Similarly, we limit G_h to be uniquely given by ξ_h and μ_h by Eq. 3.3. We keep the r.m.s. slope ($\sigma_{\nabla h}$) constant by putting $a = {}^1\!/_4$ in Eq. 3.3.

We can now form a scalar-valued function $\mu_I(\xi_h, \mu_h, \omega, \phi)$ as the angle of the most significant eigenvector. This describes the directionality of the image, given some direction of illumination with respect to the camera frame (ϕ), amount of oblique viewing (ω), surface anisotropy amount (ξ_h) and anisotropy axis direction (μ_h). Now, the eigenvector angle of any 2×2 symmetric matrix G_I (with elements g_{11}, g_{22} and $g_{12} = g_{21}$) is given by ${}^1\!/_2 \arg[g_{11} - g_{22} + \mathrm{i}\, 2g_{12}]$. This, applied to the structure tensor, will yield:

$$
\begin{aligned}
\mu_I(\xi, \mu, \omega, \phi) \;=\; & \\
& \frac{1}{2} \arg \big[3\xi^2 \cos(4\mu - 2\phi) \left(\sec^2 \omega + 1 \right) - \ldots \\
& \left(\left(\xi^2 + 2 \right) \cos 2\phi - 6\xi \cos 2\mu \right) \left(\sec^2 \omega + 1 \right) + \ldots \\
& 2\xi^2 - 2 \left(\xi^2 + 2 \right) \sec^2 \omega + 4 + \ldots \\
& \mathrm{i}\, 6\xi \sec \omega \left(\xi \sin(4\mu - 2\phi) - 2 \sin 2\mu \right) + \ldots \\
& \mathrm{i}\, 2 \left(\xi^2 + 2 \right) \sec \omega \sin 2\phi \big]
\end{aligned}
\tag{3.14}
$$

This prediction is based in the camera frame and thus describes how unsupervised illuminant tilt estimators[9][35][38] will behave as surface anisotropy and oblique viewing is introduced. The illuminance flow vector $\hat{\boldsymbol{\ell}}$ is given by the relation $\hat{\boldsymbol{\ell}} = V\boldsymbol{\ell}$. Replacing $\boldsymbol{\ell} \to V^{-1}\hat{\boldsymbol{\ell}}$ in Eq. 3.13 will therefore provide predictions based on the illuminance flow. The corresponding direction of illuminance flow is given by $\hat{\phi} = \mathrm{atan}[\cos \omega \tan \phi]$. For illuminance flow prediction we use $\mu_I(\xi, \mu, \omega, \hat{\phi} = \mathrm{atan}[\cos \omega \tan \phi])$. Some predicted curves with respect to $\hat{\phi}$ are found in Fig. 3.2 which is further explained in section 3.5.

3.4 Experiments

We experimented on rendered surfaces to test the theoretical predictions. Images(400×400 pixels) were generated as the orthographic projection of
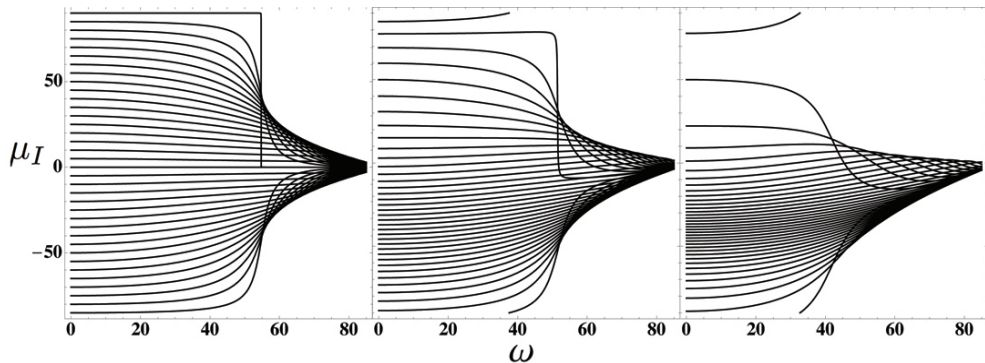
Figure 3.2: Predictions based on the theory. Each curve corresponds to a specific illuminance flow direction ($\hat{\phi} = -90, -85, ..., 90$) . Left: ($\xi_h = 0$), Center: ($\xi_h = 0.2$), Right: ($\xi_h = 0.4$).

the radiance map of a Gaussian surface illuminated by a collimated beam of light. Cast shadows were calculated in full.

The surfaces were generated randomly in the Fourier domain by using a Gaussian shaped power spectrum and pairing it with a random phase spectrum of independently generated values from a uniform distribution $\in [0, 360°)$. The power spectrum of the surface (with 2D frequency coordinates $\boldsymbol{u}$) is then given uniquely by G_h as: $\exp(\boldsymbol{u}^{\mathrm{T}} G_h^{-1} \boldsymbol{u})$.

This way of generating the surfaces ensures constancy of the estimated covariance function $\rho_h(\boldsymbol{r})$ and of the height probability distribution function. As is the case for all signals generated in the discrete Fourier domain, the surface will be defined on a topological torus. This topology is desirable for our purposes as there will be no concern about the surface boundaries (there are none!). Every aspect of the rendering, surface generation and estimation of statistics was done without violating the topology, *i.e.* $(x, y) \rightarrow \{x \bmod width, y \bmod height\}$ when the surface is represented as a finite 2D array. When viewing the surface obliquely, copies of the same surface was patched seamlessly, such that no artificial edges appear.

We varied the amount of anisotropy (ξ as 0, 0.2 and 0.4), the amount of oblique viewing (ω as 0°, 5°, ..., 65°) and illuminance flow direction ($\hat{\phi}$ as 0°, 20°, ..., 360°), and estimated the structure tensor (μ_I) from the resulting image. We also varied the illuminant incidence angle in three possible values (θ as 70°, 45° and 20°). We kept fixed the direction of the surface anisotropy axis ($\mu_h = 45°$) and the surface r.m.s. slope roughness ($\sigma_{\nabla h} = 0.45$).
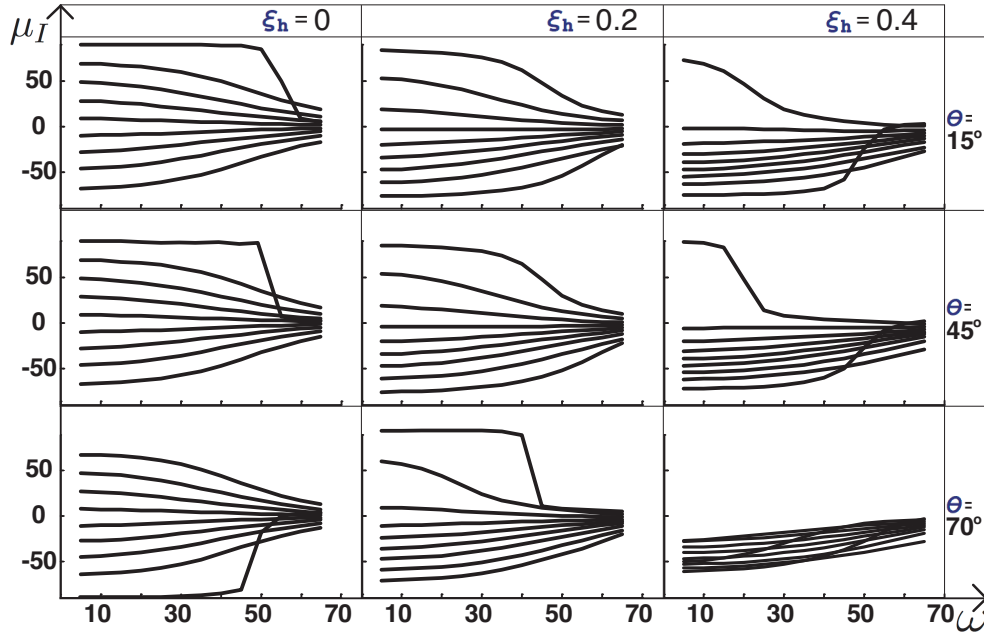
Figure 3.3: Results on renderings. Each row is the output for different illuminant incidence angle. The rows are all predicted by the theoretical curves of Fig. 3.2. Curves are for ($\hat{\phi} = -80, -60, ..., 80$).

3.5   Results

Typical output of the renderings can be seen in figure 3.4. Comparing Fig. 3.3 with Fig. 3.2 reveals, as expected, a strong resemblance. Especially when the angle of the incident light is oblique (at 45°, the middle row of Fig. 3.3) and when the anisotropy ξ_h is small (left column of Fig. 3.3). One characteristic that was noted on in previous works[10][11] is the critical behavior that starts at $\omega \approx 50°$, where-after no unique solution can be expected (while influenced by oblique viewing before that, each curve is non overlapping).

One can clearly see that all curves correlate with the ground-truth illuminance flow at low amounts of oblique viewing ($\omega \to 0$) and anisotropy ($\xi_h \to 0$). For the normal viewing and isotropic surface case, the only source of directionality in the image comes from the direction of illumination. As the two other sources of directionality (oblique viewing and surface anisotropy) are introduced the resulting directionality of the image will be changed drastically. As these new sources of image directionality become more acute, they will tend to dominate the output of the image structure tensor in the systematic way predicted by our theory, and the structure tensor will correlate
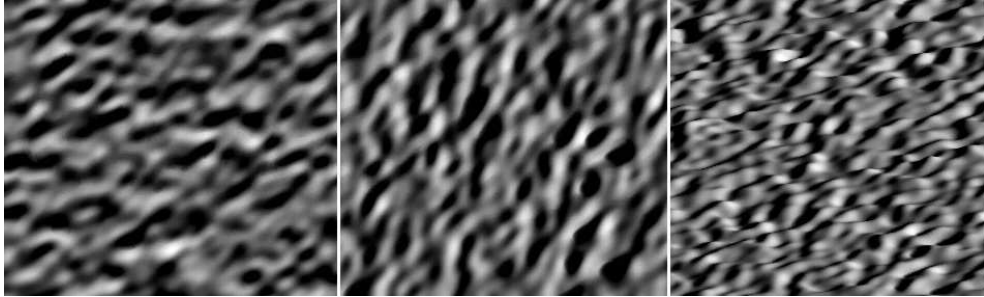
Figure 3.4: A rendered Gaussian surface used in the experiments with $\xi = 0.4$ and $\theta = 15$. Left: $\hat{\phi} = 90°$. Middle: $\hat{\phi} = -10°$. Right: $\hat{\phi} = -10°$ and viewed from another direction.

less with the illuminant direction.

Also expected were the less accurate predictions due to non-oblique incidence light (the top and bottom rows of Fig. 3.3). When the incidence angle(θ) of light becomes closer to 90° (normal direction) illuminance flow becomes undefined (as is evident from Fig. 3.1, where the flow lines all converge at the point where the illuminance flow direction is ill-defined). For lower values of the incident light angle shadows affect the outcome of the tensor, as they give rise to new edges in the image. Shadow edges are non-local and more noisy with respect to changes in $\hat{\phi}$. Interestingly, shadows did not have an especially big impact on the estimates. As evident by Fig. 3.4, the images used for the top row of Fig. 3.3 are well into the shadowing regime.

3.6   Conclusions

We have extended the theory of illuminance flow estimation to the case of obliquely viewed anisotropic surfaces. The theory has been revised by using intuitive and compact matrix formulations (Eq. 3.13). Experiments on rendered surfaces have shown that the theory is useful for predicting the output of the tensor for obliquely viewed anisotropic surfaces. This is true under the assumptions of Lambertian reflectance, uniform albedo and low relief.

Regarding the assumption a Lambertian reflectance function, it can easily be relaxed. As is discussed in[12], the predictions are still valid for reflectance functions that are not strictly Lambertian. The less Lambertian the reflectance, the more crucial is the assumption of low-relief, and vice-versa. The most crucial assumption for this theory is that of uniform albedo, not low-relief or Lambertian reflectance (although, of course, violating them to

the extreme will invalidate the theory).

As was expected, we found that obliquely incident light is essential for the predictions to hold true. When the incident light is close to the normal direction, it makes no sense to model the image as a directional derivative, and illuminance flow is not well defined. As shown in Eq. 3.4 the second order approximation of the irradiance of the surface provides a good model for the imaging process (for our purposes). The second order shading was mentioned, but not explicitly used in the theory thereafter. When the surface is isotropic one can omit second order shading from the model. However, when the surface is anisotropic, directionality will stem from second order shading as well as from first order. This kind of shading correlates strongly with the magnitude squared of the derivative of the surface. When dealing with isotropic texture (obliquely viewed or not) second order shading will not display any directionality. It makes sense, therefore, to go back to the formula, and incorporate into the theory a term dependent on second order shading. This is straightforward to achieve. The model of the image is given by Eq. 3.4, and (putting $\varepsilon = 1$) we would have the form ($\boldsymbol{g}$ is the surface gradient):

$$I(\boldsymbol{r}) = \boldsymbol{g}^{\mathrm{T}}(\boldsymbol{r}) \left(\boldsymbol{\ell} \cos\theta + \frac{\boldsymbol{g}(\boldsymbol{r})}{2} \sin\theta \right)$$

When the incident illuminant is on the horizon ($\theta = 0$) this is identical to the modeling of Eq. 3.5. When some general θ is used, the modeling differs in that averaging is not done solely over the tangential part of the incident light, but also, indistinguishably, over the gradient of the surface. This will have as a general effect that the confidence of the estimates will be lower. For the isotropic surface case, as the incidence angle approaches normal direction($\theta \to 90°$), larger areas of averaging are required for a reliable estimate. For the anisotropic case the situation is quite different. Not only do we require larger averaging areas as $\theta \to 90°$, but the gradient of the surface introduces directionality from the surface itself. We can see this behaviour in the experimental results, as the lower row of Fig. 3.3 follows the general theoretical predictions quite well, albeit for an anisotropy amount which is higher then the actual. The revision of the theory by introducing second order shading would be paramount to an increase in the effect of anisotropy from the surface, as a function of illuminant incident angle. This is exactly what we have observed in the experimental results.

Illuminance Flow Estimation by Regression

abstract

We investigate the estimation of illuminance flow using Histograms of Oriented Gradient features (HOGs). In a regression setting, we found for both ridge regression and support vector machines, that the optimal solution shows close resemblance to the gradient based structure tensor (also known as the second moment matrix). Theoretical results are presented showing in detail how the structure tensor and the HOGs are connected. This relation will benefit computer vision tasks such as affine invariant texture/object matching using HOGs. Several properties of HOGs are presented, among others, how many bins are required for a directionality measure, and how to estimate HOGs through spatial averaging that requires no "binning".

4.1 Introduction

In this chapter, meso-scale stochastic variation across an object's surface is not considered part of the shape, but is treated as 3D texture. This texture makes it possible to estimate (image) illuminance flow, an axial (bi-directional) flow field in the image that results from projecting the light vector first into the objects tangential plane, and then into the image plane. Vectors $(a, b)^T$ and $(-a, -b)^T$ describe the same flow at a given position. The flow field in the image is described by local angles $\phi \in [0, 180°)$. Illuminance flow can be used as a shape cue but in this chapter we focus solely on its estimation from images.

This work is in line with the study by Pont and Koenderink[10] where a theory for analyzing the illumination orientation from 3D texture was presented. Generalizations to oblique viewing[11], anisotropic surfaces[12] and non-uniform albedo[33] have been made.

In this chapter, we investigate real-world rough objects viewed from an arbitrary direction, and using standard regression methods, estimate the illuminance flow over their surfaces (see Fig 4.1). We focus on a contemporary and in fashion low-level feature: the Histograms of Oriented Gradients (HOGs)[41][42]. The HOGs are the low-level features of the keypoints in the Scale Invariant Feature Transform (SIFT)[42]. They perform well for human pose recognition from video[41] without the scale optimization and keypoint detection of the SIFTs. We will use the HOGs in a low-level and local fashion as a way of measuring directionality at a position in an image.

Directionality can be defined in several ways, one way (several ones will be discussed) is by the structure tensor (2nd moment matrix), which can be seen as three coarse descriptors of the distribution of the gradient. Experimental results on real world surfaces show that the structure tensor yields promising results[10] for estimating the illuminance flow, with estimates within a few degrees of the veridical orientation (in normal viewing). The issue of learning an optimal estimator based on observations with ground-truth from arbitrary viewpoints has never been posed, which is the focus of the current chapter.

Several local and unsupervised illuminant estimators have been suggested[29][35][36][9][38]. Iterative non-local algorithms (such as [39]), are not considered, neither are supervised algorithms for a finite set of textures (such as [8]). These algorithms are either identical to or correlate strongly with the gradient based structure tensor. The premise is a surface texture normally viewed, where a local estimate of the illuminant tilt is to be es-
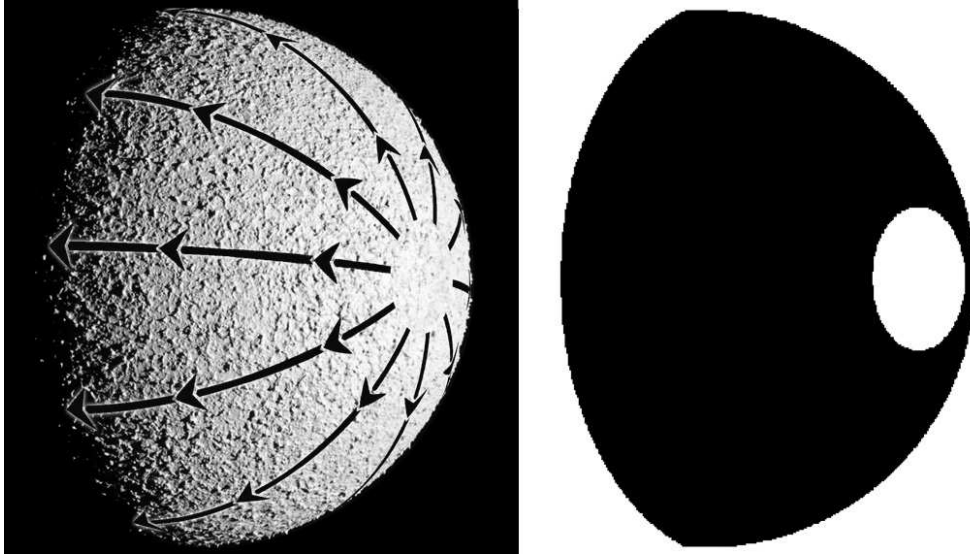
Figure 4.1: Left: Example photograph from the database used with flow direction superimposed. A textured sphere is illuminated from the right (tilt $= 0°$, slant $= 50°$). Black arrows illustrates illuminance flow (note that both directions along the black lines are valid flow lines). Right: Binary image indicating where the flow is well defined

timated. In this setting, the illuminant tilt is the same as the illuminance flow, but this is not true for arbitrary viewing of the texture, as illustrated in Fig. 4.1. The tilt is relative to the camera frame, while the illuminance flow angle is relative to the tangential frame of the object, and can change locally within the image, even for collimated beams (point source at infinity).

In this chapter, we focus on estimating the illuminance flow of the image of convex objects with rough surface texture. We use the HOG features in a regression setting, where we try both linear ridge regression and the support vector machine (SVM). To connect the results to previous work we show how the HOGs can be used to achieve a similar measure to that of the tensor (but also how this measure differs). This will enable us to show further properties of the HOGs, including how they are connected to the tensor, how many bins are required to encode a directionality measure, a different algorithm without binning to calculate the HOGs, and show how affine invariant approaches affect them.

4.2 Theoretical background

The position of the light source relative to the camera is given by the tilt and the slant angles. The tilt is the angle the light vector takes as it is projected in the camera plane (tilt $= 0°$ for light from the right, $90°$ from above) and the slant is the angle relative to the viewing direction of the camera, (slant $= 0°$ if the light comes from the direction of the camera, $90°$ if it is perpendicular to the normal of the camera plane).

The theory of illuminance flow estimation has been based on the structure tensor (second moment matrix), which in normal viewing for isotropic, uniformly Lambertian, low relief surface textures will give the flow[10]. The structure tensor (2nd moment matrix), is defined as

$$G = E\left[\nabla I \nabla^T I\right] = \left(\begin{array}{cc} E\left[I_x I_x\right] & E\left[I_x I_y\right] \\ E\left[I_x I_y\right] & E\left[I_y I_y\right] \end{array} \right)$$

where I_x is the partial derivative of the image and $E[\cdot]$ indicates expected value. The highest eigenvector will yield the directionality of the image (this constitutes one of many ways of defining directionality). G contains the second moment description of the stochastic 2D variable ∇I as well as the second moments of the image power-spectrum[4].

Measuring directionality can also be done by building a histogram of oriented gradients (HOG). For each gradient in an image a "bin" is increased in value. The angle of the gradient determines which bin, and the magnitude how much is added to it. A histogram is a crude form of non-parametric density estimation. A generalization is the Parzen window method[43] where many positions (bins) in an angular vicinity are updated (this is often called kernel-based estimation of the histogram). Assuming that such estimation is performed, there is no complication from grouping involved, i.e. one can freely choose a large number of bins based on a small number of data. Of course, the density estimation will be less reliable as data amount decreases.

The HOG can be made invariant to the sign of the gradient. The bins will then only need to cover the orientational (axial) interval $[0°, 180°)$. We will refer to the invariant version as the orientational HOG and to the regular HOG as directional. Discrete periodic sequences of the HOGs are denoted $\bar{f}_d(\frac{n2\pi}{N})$ and $\bar{f}_o(\frac{n\pi}{N})$ (for directional and orientational HOGs respectively with N bin values). These are samples of slightly different density functions, that are both related to the bivariate probability density function (pdf) $f(\boldsymbol{x})$ for the gradient ∇I.

We will use a probabilistic approach to analyze both the HOGs and G, and note on similarities with power-spectrum moments. The main drawback with frequency based analysis is the enforced toroidal topology and dependency on smooth convex window functions. Instead, we will assume that the pdf for the gradient, $f(\boldsymbol{x})$, always exists (the results generalize to distributions using the Lebesgue measure). There are three coarse descriptors of $f(\boldsymbol{x})$ found in G, namely the second order moments, e.g. $E\left[I_x I_y\right] = \iint xy f(x,y)\, dx\, dy$. The HOGs are similarly coarse descriptors for $f(\boldsymbol{x})$ as they estimate samples of the densities:

$$\begin{aligned}
f_d(\theta) &= \int_0^\infty r^2 f(r\cos\theta, r\sin\theta)\, dr \\
f_o(\theta) &= f_d(\theta) + f_d(\theta + \pi)
\end{aligned}$$

f_d has period 2π and f_o has period π, and are the population versions of $\bar{f}_d$ and $\bar{f}_o$. One could implement a HOG algorithm in two steps. First, estimate $f(\boldsymbol{x})$ (denoted $\bar{f}(\boldsymbol{x})$) by e.g. a 2D Parzens window technique or a 2D histogram. HOGs are built by collapsing $\bar{f}(\boldsymbol{x})$ into a 1D discrete signal, by weighted summing in the radial direction.

4.3 Directionality by complex change of variables

Consider the following complex expected values $\rho_\gamma(k)$, with corresponding estimations $\bar{\rho}_\gamma(k)$:

$$\begin{aligned}
\rho_\gamma(k) &= E\left[|\nabla I|^\gamma \exp(-\mathrm{i}k\operatorname{atan}(\nabla I))\right] \\
\bar{\rho}_\gamma(k) &= \frac{1}{N}\sum_{n=1}^{N} \frac{(I_x(\boldsymbol{x}_n) - \mathrm{i}I_y(\boldsymbol{x}_n))^k}{(I_x^2(\boldsymbol{x}_n) + I_y^2(\boldsymbol{x}_n))^{\frac{k-\gamma}{2}}}
\end{aligned}\tag{4.1}$$

for $\gamma \in \mathbb{R}^+$ and $k \in \mathbb{Z}$, where $\mathrm{i} = \sqrt{-1}$. We can normalize it by $\hat{\rho}_\gamma(k) = \rho_\gamma(k)/\rho_\gamma(0)$ so that $|\hat{\rho}_\gamma(k)| \in [0,1]$. The $\rho_\gamma(2)$ for different γ are different measures of directionality. $|\hat{\rho}_\gamma(2)| = 1$ always occurs for images consisting entirely of isolines in the $\frac{\angle\hat{\rho}_\gamma(2)}{2}$ orientation. When estimating $\hat{\rho}_\gamma$ by $\bar{\rho}_\gamma(k)/\bar{\rho}_\gamma(0)$, we can say that we are performing k^{th} order voting with a γ-correction term. This is similar to the theory of Bigun and Granlund[4], where the differential operator $(D_x + \mathrm{i}D_y)$ and its powers are analyzed. Powers of $(D_x + \mathrm{i}D_y)$ include higher order derivatives, which in turn correspond

to higher orders of complex moments of the power-spectrum (not of $f(\boldsymbol{x})$). We use normalized powers of $(I_x + \mathrm{i}I_y)$ which use only first derivatives. The power-spectrum and $f(\boldsymbol{x})$ are widely different entities.

A special case which connects to the Bigun-Granlund theory is $\rho_2(2) = E\left[(I_x - \mathrm{i}I_y)^2\right]$ and $\rho_2(0) = E\left[I_x^2 + I_y^2\right]$. They encode G completely:

$$
\begin{aligned}
\rho_2(2) &= (\lambda_{max} - \lambda_{min})\exp(-\mathrm{i}2\mathrm{atan}(\boldsymbol{v}_{max})) \\
\rho_2(0) &= \lambda_{max} + \lambda_{min}
\end{aligned}
$$

where λ_{max} and $\boldsymbol{v}_{max}$ are the highest eigenvalue and corresponding eigenvector of G.

Another special case is $\rho_0(k) = E\left[\exp(-\mathrm{i}k\,\mathrm{atan}(\nabla I))\right]$. This corresponds to the so-called characteristic function[44] of the circular variable: $\mathrm{atan}(\nabla I)$. The characteristic function is equivalent to a Fourier transform of the pdf of $\mathrm{atan}(\nabla I)$. Thus, $|\rho_0(2)|$ is a fit of the second harmonic to the pdf of $\mathrm{atan}(\nabla I)$, and $\frac{\angle\rho_0(2)}{2}$ is the orientation (the phase on the unit circle) of the second harmonic. For $\gamma = 0$, the magnitude of the gradient is ignored, which is one extreme way of measuring directionality.

A third special case is that of $\gamma = 1$, which is strongly connected to the HOGs, as we shall see. In general, for all γ, the change of variable formula[44] gives the relation:

$$
\rho_\gamma(k) = \int\!\!\!\int_{-\infty}^{\infty} |\boldsymbol{x}|^\gamma \exp(-\mathrm{i}k\,\mathrm{atan}(\boldsymbol{x}))f(\boldsymbol{x})\,d\boldsymbol{x} =
$$

$$
\int_{-\pi}^{\pi} \exp(-\mathrm{i}k\theta) \int_0^{\infty} r^{\gamma+1} f(r\cos\theta, r\sin\theta)\,dr\,d\theta
$$

$$
\rho_1(k) = \int_{-\pi}^{\pi} \exp(-\mathrm{i}k\theta)f_d(\theta)\,d\theta \tag{4.2}
$$

$$
\rho_1(2k) = \int_0^{\pi} \exp(-\mathrm{i}2k\theta)f_o(\theta)\,d\theta \tag{4.3}
$$

Eq. 4.3 is found by evaluating Eq. 4.2 for $k \to 2k$ as the sum of two integrals, one over $[-\pi, 0]$, the other over $[0, \pi]$, and then using $\exp(-\mathrm{i}2k\pi) =$

1 and $f_o(\theta) = f_d(\theta) + f_d(\theta \pm \pi)$. Eq. 4.2 and 4.3 yield Fourier series coefficients for f_d and f_o:

$$f_d(\theta) = \frac{1}{2\pi} \sum_{k=-\infty}^{\infty} \rho_1(k) \exp(ik\theta)$$

$$f_o(\theta) = \frac{1}{\pi} \sum_{k=-\infty}^{\infty} \rho_1(2k) \exp(i2k\theta)$$

If the population versions f_d, f_o, ρ_1 are replaced with the sample versions $\bar{f}_d$, $\bar{f}_o$ and $\bar{\rho}_1$, then Eq. 4.2 and 4.3 will turn into discrete Fourier transforms. For the orientational HOG we have:

$$\bar{\rho}_1(2k) = \sum_{n=0}^{N-1} \exp\left(-i2k\frac{n\pi}{N}\right) \bar{f}_o\left(\frac{n\pi}{N}\right) \tag{4.4}$$

Some properties of the HOGs that emerge from these observations are:

1) If a directionality measure needs to be explicitly calculated using HOGs, then a best matching sinusoidal (the second harmonic approximation of the directional HOGs, or equivalently, the first harmonic approximation of the orientational HOGs) yields the desired measure. Other methods, such as using functions not strictly sinusoidal or methods to measure the bi-modality of a circular function, can be devised, but the harmonic will yield the measure that is closest possible to G (assuming no other information of $f(\boldsymbol{x})$ is available).

2) The minimum number of bins required to yield such a directionality measure is given by the Nyquist-Shannon sampling theorem (the sampling frequency is $\frac{\#bins}{2\pi}$). For the orientational HOGs, we require 3 bins, and for the directional HOGs, 5 bins.

3) The directionality inherent in the HOGs is strongly correlated with that of G. They differ in γ-correction only. The structure tensor has $\gamma = 2$, while the HOGs have $\gamma = 1$. Algorithmically speaking, in G higher magnitude gradients are weighted more then in the HOGs. If the magnitudes of the gradients would be fixed to one ($f(\boldsymbol{x})$ is nonzero only on a circle), then the directionality of the HOGs and G would be identical.

4) An alternative to calculating the HOGs is to calculate $\bar{\rho}_1(k)$, and then to Fourier transform it. This approach avoids the grouping procedure (the 'binning') inherent in the conventional histogram approach. K elements of the sequence ($k \in [0, K-1]$) yields $2K - 1$ bin values (samples in $\bar{f}_d$).

For estimating $\bar{f}_o(\theta)$ the sequence $\bar{\rho}_1(2k)$ is used in the same way. This is equivalent to using a wrapped sinc function as a Parzen window[43]. The equivalent to a Gaussian Parzen window can be achieved by multiplying $\bar{\rho}_1(k)$ with a Gaussian (because multiplication in Fourier gives convolution and because a Gaussian function transforms back to a Gaussian).

5) If images are affine normalized using G, as is proposed in several works[45], then there is little or no discriminant information available in $\bar{\rho}_1(0)$ and $\bar{\rho}_1(2)$. There are a total of three degrees of freedom in $\rho_1(0)$ and $\rho_1(2)$ (real and complex valued resp.) that correspond closely to $\rho_2(0)$ and $\rho_2(2)$ (that encode G). If one uses the HOGs as low level features, it might be prudent to use $\bar{\rho}_1(0)$ and $\bar{\rho}_1(2)$ for affine normalization, instead of G. However, if HOGs are estimated on smaller regions within a larger affine normalized region, then $\bar{\rho}_1(0)$ and $\bar{\rho}_1(2)$ can still hold valuable information. Also note that HOGs are usually normalized to unit mean which corresponds to enforcing $\bar{\rho}_1(0) = 1$ regardless of affine normalization.

We emphasize that we are not suggesting a new set of low-level features here, but rather we suggest an analysis of the existing ones (HOG) that makes the connection to the structure tensor readily available, and sheds some light onto what the HOGs actually do in terms of non-parametric density estimation.

4.4   Axial regression

We now turn our attention to the specific topic of illuminance flow estimation using the HOGs as low-level features. We tried two standard approaches, first a linear model with ridge regression, then a support vector machine where several kernels were considered. Because illuminance flow is an axial (orientational) property, we used the orientational HOG.

Linear Model

We first phrased the problem in a linear setting as $y = \boldsymbol{f}^T \boldsymbol{w}$, where y is the illuminance flow at a point, $\boldsymbol{f} = \{\bar{f}_o(\frac{0\pi}{N}), \bar{f}_o(\frac{1\pi}{N}), ..., \bar{f}_o(\frac{(N-1)\pi}{N})\}^T$ is the feature vector, and $\boldsymbol{w} = \{w_0, w_1, ..., w_{N-1}\}^T$ is the weight vector. Collecting all feature vectors in matrix F, and all ground-truths in vector $\boldsymbol{y}$, ridge regression is a regularized version of LSE minimization, resulting in the pseudo-inverse: $\boldsymbol{w} = (F^T F + c_{lin}\mathcal{I})^{-1}F^T \boldsymbol{y}$, where $\mathcal{I}$ denotes the identity matrix, and c_{lin} is the ridge parameter for the regression. This corresponds to minimizing the

objective function:

$$E_{lin} = c_{lin}||\boldsymbol{w}||^2 + \frac{1}{2}||F\boldsymbol{w} - \boldsymbol{y}||^2$$

Using the illuminance flow angle ϕ directly to represent the flow is not suitable for regression because of the angular discontinuity (for axial data, 0° is equivalent with 180°). Instead, one can choose to regress towards $\cos(2\phi)$ and $\sin(2\phi)$ separately, arriving at 2 weight vectors independently (when using the models for predictions, one needs to divide the output angle by two). This can be eloquently phrased as one single regression, by using complex numbers where $y = \exp(\mathrm{i}2\phi)$:

$$
\begin{aligned}
y &= \boldsymbol{f}^T \boldsymbol{w} = \\
\exp(\mathrm{i}2\phi) &= \sum_{n=0}^{N-1} \bar{f}_o(\frac{n\pi}{N})w_n
\end{aligned}
\tag{4.5}
$$

where $w_n \in \mathbb{C}$. The definition of the pseudo inverse allows for complex numbers (replacing $\boldsymbol{w}^T$ with conjugate transpose $\boldsymbol{w}^*$). We are minimizing one single consistent error E_{lin} (we have $||\boldsymbol{w}||^2 = \boldsymbol{w}^*\boldsymbol{w}$). That the regression on $\cos(2\phi)$ and $\sin(2\phi)$ is done separately does not matter for the outcome, nor does the coordinate frame we choose for ϕ.

In this model, if $w_n = \exp\left(-\mathrm{i}2\pi\frac{n}{N}\right)$ then, following Eq. 4.4, $y = \bar{\rho}_1(2)$ which correlates with the structure tensor (it differs only in γ-correction, and is the closest possible to the tensor we can get using only the HOG).

Support Vector Model

The Support Vector Machine (SVM) was originally suggested by Vapnik[46] for classification and regression[47]. The SVM fits the linear function $y = \boldsymbol{f}^T\boldsymbol{w} + b$ to the data by solving the following convex optimization problem:

$$
\begin{aligned}
\text{minimize} \quad & E_{svm} = \frac{1}{2}||\boldsymbol{w}||^2 + c_{svm}\sum_{\forall k}\zeta_k \\
\text{subject to} \quad & \begin{cases} |(\boldsymbol{f}_k^T.\boldsymbol{w}) + b - y_k| & \leq \epsilon + \zeta_k \\ \zeta_k & \geq 0 \end{cases}
\end{aligned}
\tag{4.6}
$$

The constants c_{svm} and ϵ are for tweaking the regression. The SVM solves this by optimizing a dual formulation, arrived at through forming

the Lagrangian of E_{svm}. New variables are introduced, $\boldsymbol{\alpha} = \{\alpha_k\}$, where $\boldsymbol{w} = \sum_{k=1}^{K} \alpha_k \boldsymbol{f}_k$, and thus $y = b + \sum_{k=1}^{K} \alpha_k \boldsymbol{f}_k^T \boldsymbol{f}$. With a new objective function and constraints with respect to $\boldsymbol{\alpha}$, the dual problem gives solutions to the originial (primal) problem[46]. The set of training feature vectors that contribute to the output is small because $\boldsymbol{\alpha}$ is usually sparse (these are the support vectors of the machine).

Generalizing to non-linear functions is done by providing a substitute scalar product (the kernel), $\kappa(\boldsymbol{f}_1, \boldsymbol{f}_2)$. Each possible kernel corresponds to performing transformations of the data before the regression, such that a linear regression in the transformed space corresponds to non-linear regression in the original one[48]. If the objective function makes use of the kernel instead of scalar products, then the output of the machine will be $y = b + \sum_{k=1}^{K} \alpha_k \kappa(\boldsymbol{f}_k, \boldsymbol{f})$.

The linear SVM is very similar to ridge regression, but the objective function we minimize is essentially different. E_{lin} and E_{svm} have the same smoothness term $||\boldsymbol{w}||^2$, which is independent of the data, they differ in how they penalize data deviation from the model. Any deviations that is within the ϵ bound is not penalized in the SVM regression, and is penalized linearly for the amount above that threshold, with slope equal to the parameter c_{svm}. Ridge regression minimizes a squared error, with no ϵ insensitive region. However, squaring also entails penalizing lower deviations less then higher ones, so the two methods should be expected to yield similar output. This argument holds despite the fact that ridge regression and SVMs require widely different algorithms (matrix inverse vs. iterative search). Both are convex optimization problems, where a unique global optimum is guaranteed. The power of SVMs lies in their ability to introduce nonlinearities. For this reason, we compared SVM regression with several popular kernels with the result we achieved with the ridge regression.

We phrased the illuminance flow problem in a similar fashion as with the ridge regression. We regress towards $\cos(2\phi)$ and $\sin(2\phi)$ separately, and will get two SVMs that are combined to make predictions (output angle is divided by two). In the ridge regression setting, this can be phrased as one single regression using complex numbers, with a consistent error. We are not aware of any method to provide the same property for SVMs. However, because the ridge regression and the linear SVM are similar, we argue that the linear SVM has a roughly consistent error, but the same cannot be said for any kernel.

Implementation and Data

For data, we used a set of 28 images of the same textured sphere photographed under controlled laboratory conditions. The lighting was approximately collimated beams (one point source, far from the sphere), where the position of the light source was varied in 14 slant directions $(10°, 20°, ..., 140°)$. For each slant direction, the sphere was photographed with lighting from the right (tilt direction 0°) and the left (tilt direction 180°). Furthermore, the images were rotated to simulate more tilt directions, in total 16 directions $(0°, 22.5°, ..., 337.5°)$ for every slant angle.

The photographs are 8 bit gray depth, 600 by 600 pixels. The sphere was carefully positioned such that the center of the image corresponds to the center of the sphere. The sphere is roughly 0.5 meters in diameter, and was positioned 2 meters away from the camera. With a maximum variation in visible height profile of 0.25 meters, we can assume that the camera has an orthogonal projection as far as the sphere is considered. The height profile is a sphere and with collimated beams of known direction, we calculated the groundtruth illuminance flow angles ϕ for every valid position in the image. Positions outside of the silhouette, in the shadow of the sphere or where the light hits the surface very close to its normal direction were not considered, as these positions do not have illuminace flow well defined (illustrated in Fig 4.1). This amounted to over 260.000 data points in 32 dimensional feature space, with associated ground-truth illuminance flow $\phi \in [0°, 180°)$.

Orientational HOGs were calculated on square, 8 by 8 pixel wide cells in the image. We used 8 bins, representing the angular intervals of $\{[0°, 22.5°), ..., [157.5°, 180°)\}$. At the vertex of each cell are the positions to be described by the features. For each position, the HOG bin values of all 4 cells sharing that vertex are collected into a feature vector of 32 elements. These are normalized to unit mean. Each feature vector (for each position considered) get values from 4 HOGs, so the outer scale can be said to be 16 by 16 pixel wide blocks. Each HOG contributes to 4 blocks, so there is overlap between the feature vectors.

The software for calculating HOGs was Bill Triggs' [41] implementation, which uses two point derivative filters. We also implemented the structure tensor algorithm with the same derivative filters and size of blocks as with the HOGs. Linear interpolation is used in the binning algorithm of the HOGs to improve accuracy, which approximately corresponds to a triangular Parzen window[43]. Ridge regression was performed with the statistics toolbox in

the Matlab environment. For the SVM, we used the "SVM^{light} v. 6.01" software implementation[49]. We tried 4 different kernels: linear, polynomial, sigmoidal (tanh) and radial basis function (rbf).

4.5   Results

Interestingly, all models (ridge and SVM) yielded near identical results to the $\bar{\rho}_1(2)$ measure, which is visually indistinguishable from the structure tensor on these images. Typical output is illustrated in Fig. 4.2 together with ground-truth. Performance is evaluated through the average angular deviation (E_{ad}) of the estimate to the ground-truth illuminance flow angle (averaged over all images in the database, over all valid positions). All the regressed models had an E_{ad} lower then 16.5°, with the best results for the (second order) polynomial kernel at 16.43° and the worst being the rbf kernel at 16.48° (further tweaking of kernel parameters and selecting different training sets will change this ordering). The left panel of fig. 4.3 illustrates the performance across the image (different positions on the sphere). The corresponding illustrations for the other models are not shown, because they are all visually indistinguishable.

We also implemented the regular structure tensor, with equivalent derivative filters and averaging areas as with the HOGs (two point derivative filters and 16 by 16 square blocks). The structure tensor yields $E_{ad} = 16.7°$. If the weights of the linear model are fixed to those corresponding to the structure tensor, then the same error occurs (difference is smaller then 0.05°). The small reduction in performance when going from the regressed models to the pure structure tensor, is explained by the directional bias in the estimation (square blocks, and 2 point filters). The regressed models (both SVMs and linear) have compensated for directional bias in the HOG features.

Ridge regression: Inspection of the regressed weights in Fig. 4.4 verifies that $w_n \approx C \exp\left(-\mathrm{i}2\pi \frac{n}{N}\right)$, which corresponds to the $\bar{\rho}_1(2)$ measure. Fig. 4.4 depicts the weights with their angle divided by two (thus w_n occupy half a circle) which is the equivalent to dividing the output of the model by two (which is required for an estimate of the illuminance flow angle). Incorporating more features in the training set made the weights converge towards $w_n \approx C \exp\left(-\mathrm{i}2\pi \frac{n}{N}\right)$ (closer to the circles of Fig. 4.4). Changing the parameter c_{lin} changes the magnitude of the regressed weights (C above) but not their directional component, except for when c_{lin} is close to zero. For $c_{lin} = 0$ the regression is pure LSE minimization which will be (for this
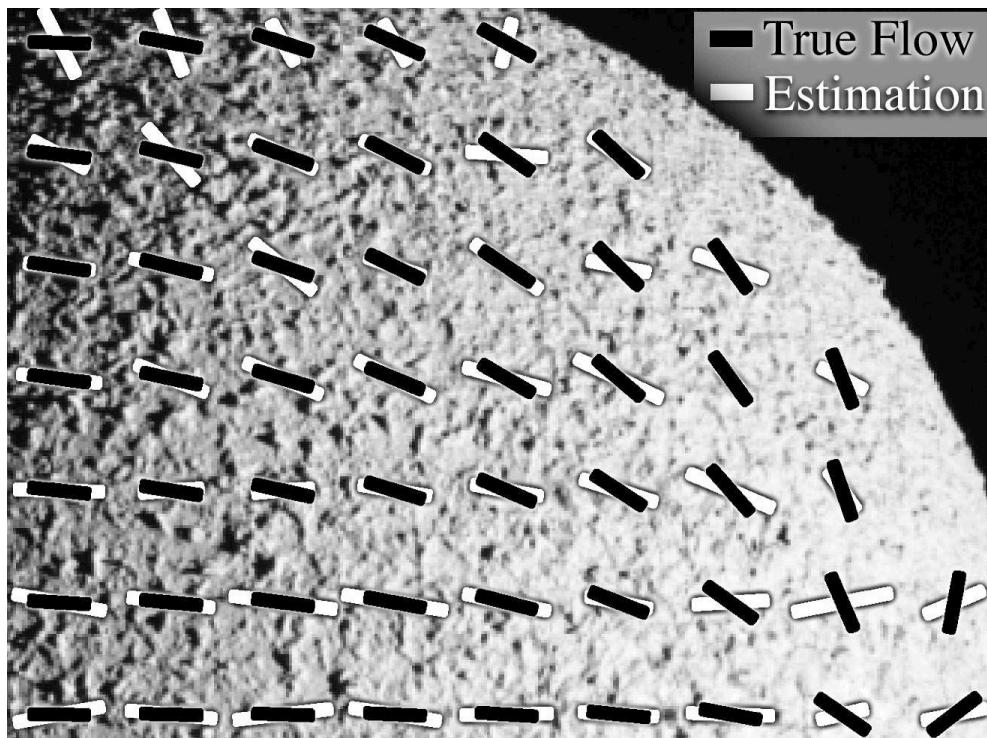
Figure 4.2: Typical output of the regressed models. Samples are taken of every third valid x and y coordinate (1/9 of the valid positions shown). Estimation is by the linear ridge predictor, but is visually indistinguishable from any other model regressed. Note that the positions in the lower right corner occupy a domain where illuminance flow is not well defined, as illustrated by Fig 4.1 (incoming light is near to being parallel to the surface normal)
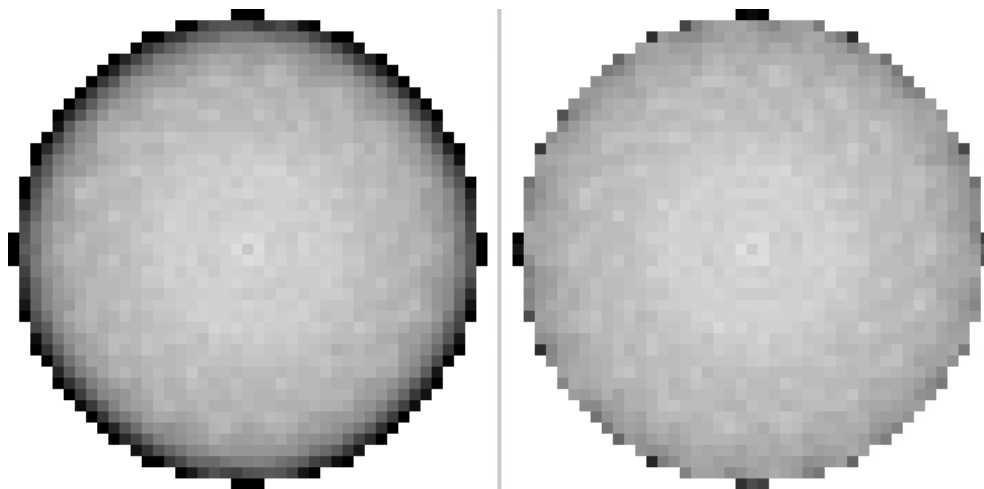
Figure 4.3: Average angular deviation of the prediction to ground-truth, averaged over all images. White: no deviation, black: deviation > 45°, gray: linear scale in [0,45]. Left panel: Result achieved with HOG features (approx. same image for all models). Maximum average deviation ≈ 79°. Right panel, equivalent image for when the normal is appended to the feature vector, for the rbf kernel SVM

problem) unstable and prone to over-fitting. We note that the results show some directional bias, especially in the directions ($\pm 45°$ and $\pm 135°$). These are weighted slightly less then the horizontal and vertical directions. This is because two point derivative filters and square regions are used in the HOG calculations which gives rise to some directional bias. If derivatives of Gaussians are used as filters and roughly circular cells implemented, then this effect disappears. Smoothing the image before the filtering reduces but does not eliminate the directional effects of two point derivative filters.

SVM models: The output is almost identical to the structure tensor, and visually indistinguishable from the ridge regression for all kernels used. We used a training set of 12.000 randomly selected features for training, and the remaining features for verification. Training is on a mere 4% of the data but took nearly a day to complete per kernel. The output of the SVM models is indistinguishable from that of the ridge regression, for both training and verification set. All the regressed predictors had very similar performance on the training and the verification set, with a difference in E_{ad} smaller then 2°. Using different kernels yields different result only for much smaller subsets of the training data, but as the training set becomes bigger they all converge towards structure-tensor-like behavior.
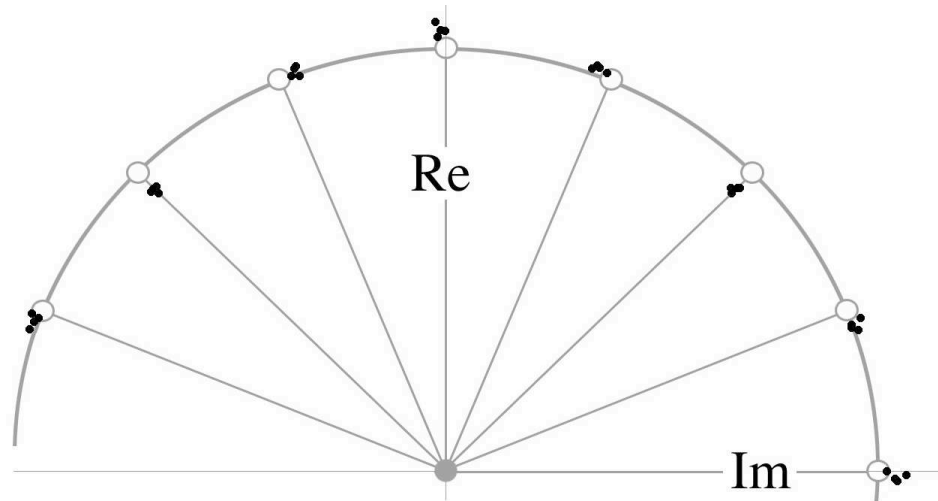
Figure 4.4: Weights of the linear model from ridge regression. Black dots: weights, gray circles: centered around the positions of the $\bar{\rho}_1(2)$ predictor. Weights have their angular part divided by two, so that each gray radial line corresponds to the center of the bin of the HOG. There are four HOGs used for each position in the image thus four weights for each direction

It seems that the HOG features are not indicative of illuminance flow beyond what is in $\rho_1(2)$. Unfortunately, with our SVM modeling, we are not minimizing a consistent error which weakens this conclusion. To illustrate this issue, if we change the frame in which the illuminance flow angle is described (the direction of the zero axis) then the results of the SVM regression change. For small training sets, a noticeable change in output is evident (this is not so with the ridge regression which is totally invariant to the frame used). As the training set becomes bigger the output becomes less dependent on the frame (in the training set used for the final results, no directional bias can be seen).

Accounting for Oblique Viewing

A major source of error from the estimators comes from oblique viewing of the texture, i.e. where image patches are taken closer to the silhouette of the object. This is clearly evident in the left panel of fig. 4.3 where E_{ad} actually goes above $45°$ when taken close to the silhouette (random guesses results in $E_{ad} = 45°$ for axial data). We appended the local normal of the sphere (relative to the camera frame) to the feature vector, to see if the regression would improve. We note that it is unfeasible to assume that the normal is

available in real-world applications. We also encoded the normals differently by e.g. projecting them into the camera plane (yielding 2D vectors with magnitude less then one), and by doubling the angle of the projected vector (which is equivalent to considering it as an axis rather than a vector). We also used the normal in a preprocessing scheme, where we performed an affine transform on the image region before the HOG features were calculated.

The affine transform aims at minimizing the effects of oblique viewing of the texture. We performed first a contraction of the patch equivalent to the foreshortening but in the orthogonal direction of the foreshortening. After that, a uniform up-sampling of the texture patch was performed. We note that modeling oblique viewing as an affine transform is an approximation to begin with, and that we loose fine scale information doing the normalization.

All the models gained performance with the affine normalization but they regressed nonetheless to the same structure-tensor-like behavior. The best performance was with the polynomial kernel ($E_{ad} = 11.9°$) and the worst was the sigmoidal kernel ($E_{ad} = 12.1°$), a difference we accredit to variability in training set selection and kernel parameters.

Ridge regresssion: We achieved no improvement with the appended normals. Inspection of the regressed weights showed that the ones corresponding to the normals equalled zero, independently of how the normals where encoded. The linear model is not powerful enough to make use of this kind of information.

SVM models: In contrast with the ridge regression, the SVM has the capability to make use of the normals. When appending them to the feature vectors, an improvement was noticed that was dependent on which kernel was used. The best improvement was achieved with the rbf kernel ($E_{ad} = 12.8°$) with the projected normal coded with double angle. The second order polynomial kernel has $E_{ad} = 13.4°$ and the sigmoidal has $E_{ad} = 15.7°$. The performance over different positions in the image of the rbf kernel SVM is illustrated in the right panel of fig. 4.3. All the other results follow the same pattern: an improvement close to the silhouette, but the closer to the center of the image (normal viewing) the closer the models agree with the structure tensor algorithm. The rbf kernel SVM performs very much like the affine normalization scheme.

4.6 Discussion

This chapter has 2 major contributions; 1) deriving theoretical properties of the Histogram of Oriented Gradients (HOGs), and 2) estimate illuminance flow through regression on the HOG features.

HOG Properties

Our theory uses spatial averaging over a set of non-linear mappings of the gradient (Eq. 4.1). We have shown how the resulting sequence is equivalent to a Fourier series expansion of the HOG features, where the second harmonic is strongly correlated with the eigenvector of the structure tensor (2nd moment matrix). The only difference between the second harmonic of the HOG and the structure tensor is a γ- correction of the gradients in the corresponding spatial averaging. In affine invariant texture and object matching the structure tensor is often used in a normalizing procedure, and our theory predicts how this will affect the HOGs. It also shows how many bins are needed of the HOG to calculate a similar measure as the structure tensor, as well as an alternative way of calculating HOGs, without binning.

The structure tensor is not the only way of achieving affine normalization. It entails a γ-correction of two in our spatial averaging. Better results might be achieved if a directionality measure is used that is consistent with the low-level features (HOGs), that involves a γ of one (i.e. enforcing the second harmonic to have zero energy). Further investigation into this will be a subject of future work.

One could naturally ask whether the gradient mappings might yield even more efficient features than the HOGs. This is an interesting topic as well, but beyond the scope of this chapter. We have not suggested a new set of features, but rather, an analysis of the existing ones (HOG) that makes the connection to the structure tensor readily available, and explains what the HOGs actually do in terms of non-parametric density estimation.

Illuminance Flow Regression

Regarding the experimental results of this chapter, we found that in a regression setting, using the HOGs as low-level features, the second harmonic of the HOGs was (approximately) found as the optimal estimator of illuminance flow. Because this is the closest possible to the structure tensor that can be achieved using only HOGs, as well as visually indistinguishable on

the images used, we conclude that the structure tensor is near optimal for the images in this study.

A natural question is whether the results generalize to arbitrary texture, which could be composed of different fine-scale surface geometry and any form of local variation in surface reflectance. We have in earlier work discussed the applicability of the structure tensor for deviations from the uniform plaster type of texture[12]. Essentially, as long as the height profile has sufficiently low average height and is reasonably smooth, then a less Lambertian reflectance will not affect the outcome of the structure tensor. This is especially true if the light source is more elongated then a point source. For flat spatially varying albedo texture (on which the vast majority of computer vision theory is based) illuminance flow is not observable. In the case where both fine-scale surface texture(say plaster) and flat albedo texture (say a flat painted pattern) are present simultaneously, things become more troublesome. Oblique viewing and anisotropy for flat texture can be modeled as one single affine transform (a single fact that makes popular computer vision).

We tried both linear ridge regression and Support Vector Machines(SVM) with several kernels. We were unable to find any significant improvement in performance using the more powerful SVM. This is an indication that for the estimation of illuminance flow there is no more useful information in HOGs other then what is in the second harmonic. Essentially, we found that the gradient structure tensor is the optimal estimator in our setting (albeit with a lower γ-correction than what is usually suggested).

This conclusion is weakened by the fact that our SVM regression is not minimizing a consistent error. We could not find a way to do the regression on both the x and the y component of the ground-truth simultaneously as was done with the ridge regression. We contended with doing 2 separate SVM regressions, one for each component of the illuminance flow vector.

We formulated the linear ridge regression through complex numbers, which easily shows that the error is independent of the particular frame we use to describe ground truth angles. If, similarly, the SVM framework could be generalized to deal with complex numbers, then it should be able to use SVMs with a consistent error for axial regression problems of this kind. Alternatively, the two SVM regressions could be performed simultaneously, with an additional constraint that the combined output should be on the unit circle (which should still be a convex optimization problem), thus coupling the models. This will be a worthwhile subject for future work.

Local Shape from Illuminance Flow

abstract

Illuminance flow is a flow field in the image produced by projecting the light vector into the local tangential plane of a visible surface. The flow is observable from objects with rough 3D surface texture. In this chapter we investigate the differential invariants of this flow field and show how shape can be inferred locally at a point in an image. In addition, the global position of the light source can be partly inferred as well, at least for the experimental surfaces we consider. A simple algorithm for estimating the second order local shape (the Hessian) is derived and shown experimentally to perform quite well on simulations of Gaussian surfaces (2D Gaussian processes) where we assume the ground-truth flow is given.

5.1 Introduction

In this chapter, meso-scale stochastic variation across an object's surface is not considered part of the shape, but is treated as 3D texture. This texture makes it possible to estimate (image) illuminance flow, an axial (bidirectional) flow field in the image that results from projecting the light vector first into the objects tangential plane, and then into the image plane. Vectors $(a, b)^T$ and $(-a, -b)^T$ describe the same flow at a given position. Fig. 5.1 shows ground-truth from a sphere. Illuminance flow can be used as a local as well as a global cue for the shape of objects. This chapter focuses solely on the local approach, but global approaches will be briefly discussed. This work is in line with the study by Pont and Koenderink[10] where a theory for analyzing and predicting the illumination orientation from 3D texture was presented. Generalizations to oblique viewing[11] of anisotropic surfaces[12] and non-uniform albedo[33] have been made. This chapter takes the subject further by exploring how illuminance flow can actually be used for shape inference. Shape from illuminance flow is a novel topic, and can be said to be somewhat in-between shape-from-shading and shape-from-texture, but fundamentally independent of both. Our notion of texture deviates significantly from most other works (especially in computer vision) in that it is given a richer definition.

Computer vision approaches to 3D reconstruction commonly model objects as consisting of shape (the surface of the object) and texture (the variation of surface albedo forming a pattern on the surface). While this definition is very useful for cases such as a box of cereals, a race car with commercial printing or a wall with graffiti, it is less useful for objects such as the stem of a tree, a boulder, a uniform colored sculpture or the sphere with plastered texture in Fig. 5.1. We adapt a more general notion of texture, which is more in line with the intuitive human notion of it. While surface variation in albedo (painted pattern on the surface say) is one part of what makes up texture, another part is meso-scale surface variation. We model objects as consisting of surface variation on two seperate scales; the meso scale and the macro scale. The macro scale constitutes the shape of the object (a perfect sphere in the example of Fig. 5.1) and the meso scale is the scale of the texture. The shape we are interested in describing using illuminance flow is on the macro scale. On this scale we can assume that object surfaces are smooth differentiable manifolds (the texture on the meso scale might not be). We will talk of the tangent plane of the macro scale surface, and simply
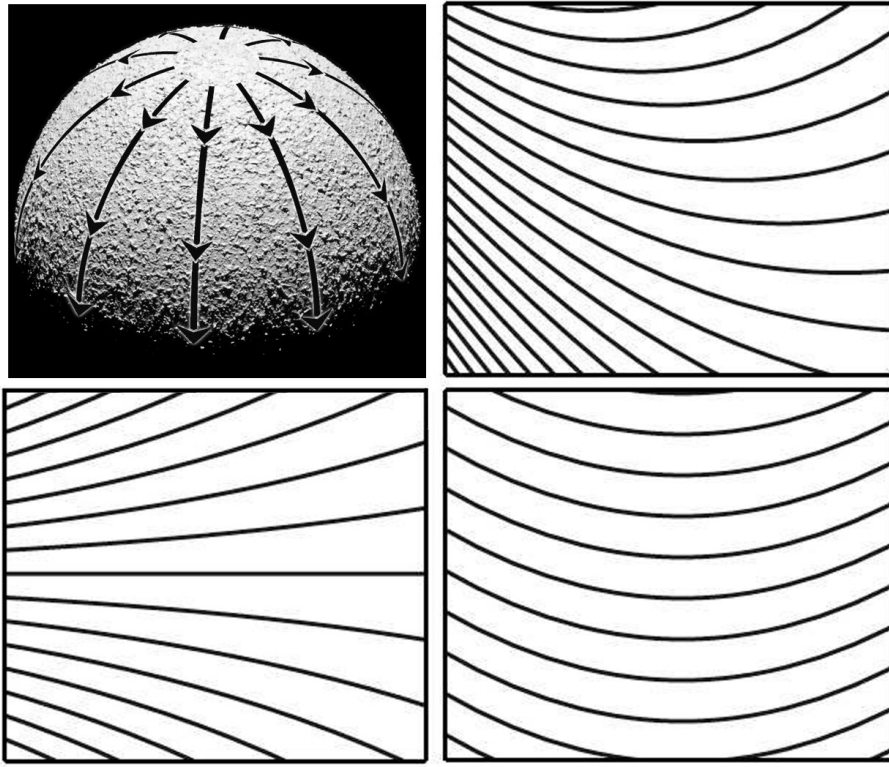
Figure 5.1: Examples of illuminance flow patterns. Top left: Flow over a real world object (sphere). Top right: A flow pattern with non-zero swerve ($\hat{g}_1$) and non-zero splay ($\hat{g}_2$). Bottom left: Pattern with non-zero splay only. Bottom right: Pattern with non-zero swerve only.

mean the tangent plane as it would be if the meso scale texture variation is not considered.

The texture is not part of the shape, but we want to exploit its presence for inferring the shape, which is as far as the analogy goes for illuminance flow and shape from texture. In traditional shape from texture (as with the majority of computer vision state of the art) only flat texture is used, where oblique viewing can be modeled as an affine transform[45], and where light direction has a very simple relation to the appearance of the texture. For 3D texture, things are very different. On the meso scale, the texture is a shading and shadowing pattern of a low relief height function. While flat Lambertian texture only depends on the elevation angle of light relative to tangent plane (giving a uniform intensity change), 3D texture has a critical dependence on the azimuthal angle of the light direction relative to the (macro scale)

tangent plane. It is precisely this dependence that allows us to observe the flow.

5.2 Theoretical background

The position of the light source relative to the camera is given by:

$$\boldsymbol{\ell} = \{\ell_1, \ell_2, \ell_3\}^{\mathrm{T}} = \{\cos\theta\cos\phi, \cos\theta\sin\phi, \sin\theta\}^{\mathrm{T}}$$

where θ is the illuminant elevation angle (the slant) and ϕ is the azimuthal angle (the tilt). The light vector $\boldsymbol{\ell}$ is *not* given in the tangential frame of the surface, but in the frame of the camera. There is no spatial dependence, it is constant throughout the scene.

We will start by working in a coordinate system where the camera is situated along the z-axis. The camera is modeled as having orthographic projection. We assume a point ($\boldsymbol{x} = \boldsymbol{0}$) on the surface of the object that does not suffer from self occlusion. The macro scale surface $h(\boldsymbol{x})$ is approximated (locally) as a 2D function by:

$$h(\boldsymbol{x}) = \boldsymbol{x}^{\mathrm{T}}\nabla h + \frac{1}{2}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{H}\boldsymbol{x} \tag{5.1}$$

where $\nabla h = \begin{pmatrix} h_{10} \\ h_{01} \end{pmatrix}$ and $\boldsymbol{H} = \begin{pmatrix} h_{20} & h_{11} \\ h_{11} & h_{02} \end{pmatrix}$ are the gradient and the Hessian at $\boldsymbol{x} = \boldsymbol{0}$. The non-unit normal of the surface is $\boldsymbol{n}_0(\boldsymbol{x}) = \{-\frac{\partial h(\boldsymbol{x})}{\partial x}, -\frac{\partial h(\boldsymbol{x})}{\partial y}, 1\}^{\mathrm{T}}$ and $\boldsymbol{n} = \boldsymbol{n}_0/|\boldsymbol{n}_0|$.

Assuming a Lambertian uniform albedo surface and that the point is not in shadow, we model the shading of the surface as:

$$I(\boldsymbol{x}) = \boldsymbol{\ell}^{\mathrm{T}}\boldsymbol{n}$$

This shading is on the macro scale regardless of whether meso scale texture is present. Over a given window, the macro scale shading is approximated by averaging all meso-scale intensities including points of the texture in shadow. We do not consider a 3D texture on any finer scale than the meso-scale. If one uses, for example, a Brownian fractal surface[30], where the surface variations exist on all finer scales (as well as coarser ones) then we note that the surface will not be Lambertian in general[13]. The only assumption that we make on finer scales than meso is that whatever they may be they yield a Lambertian reflectance.

The shading is by definition strictly positive and bounded since we assume no shadow in the macro scale context ($I(\boldsymbol{x}) \in (0, 1]$). On the meso scale, there is no problem with shadows within the texture.

The shading, as it has been defined here, is strictly speaking not observable (we do not know the camera transfer function or the intensity of the light, both of which are not treated here). As observable we will instead consider the contrast gradient, $\nabla c(\boldsymbol{0})$, which is based on the local contrast:

$$c(\boldsymbol{x}) = \frac{I(\boldsymbol{x})}{I(\boldsymbol{0})}$$

For a second observable, we will consider the illuminance flow. To arrive at the flow, we first define the twice projected light vector:

$$\tilde{\boldsymbol{f}}(\boldsymbol{x}) = \boldsymbol{P}(\boldsymbol{\ell} - (\boldsymbol{\ell}^{\mathrm{T}}\boldsymbol{n})\boldsymbol{n})$$

where the 2-by-3 matrix $\boldsymbol{P}$ is the projection into the camera frame. In our case $\boldsymbol{P}$ will simply discard the third dimension (the camera is along the z-axis): $\boldsymbol{P} = (\boldsymbol{e}_1 \vdots \boldsymbol{e}_2)^{\mathrm{T}}$. As with the contrast, we assume a point not in shadow on the macro scale, but shadow in the meso scale texture poses no problems. Illuminance flow is defined as the normalized projected light vector, with a sign ambiguity:

$$\boldsymbol{f}(\boldsymbol{x}) = s\frac{\tilde{\boldsymbol{f}}(\boldsymbol{x})}{|\tilde{\boldsymbol{f}}(\boldsymbol{x})|} \tag{5.2}$$

where $s \in \{-1, 1\}$ represents the sign ambiguity which allows us to treat illuminance flow as an orientational (axial) rather then a directional flow field. The reason for doing so is because of the sign ambiguity in shading patterns. Consider any Lambertian rough surface texture distributed on a plane viewed frontally. Roughly the same image will result if simultaneously the tilt light angle is rotated one half revolution ($\phi \to \phi \pm 180°$) while the texture height is inverted ($h_{meso}(\boldsymbol{x}) \to -h_{meso}(\boldsymbol{x})$). The sign ambiguity can sometimes be resolved using inter-reflections or shadows, but these effects are difficult to use in general, and are most often negligible when dealing with a low relief surface texture.

In[12] we discuss how to estimate illuminance flow from images of texture. It is not as straightforward to observe the illuminance flow as it is to observe the contrast but the estimation of the flow is not the focus of this chapter. We are interested in describing the structure of the flow field, deriving differential invariants and relating them to the shape of the surface on the macro scale.

Because of the bas-relief ambiguity[55], estimating first order shape (i.e. the normal) is not reliable using illuminance flow and/or contrast. We focus on the second order shape parameters (the elements of $\boldsymbol{H}$: h_{20}, h_{11} and h_{02}), and especially as they are expressed in a local frame given by the flow vectors. We will give attention to both the illuminance flow as well as the contrast gradient, and will show a fundamental connection between them. We start off with illuminance flow and its differential invariants.

5.3   Illuminance flow

We will use a standard approach from differential geometry, the method of moving frames[40], in order to derive two natural differential invariants and relate them to the local shape of the surface. According to the moving frame method, we let $\boldsymbol{f}(\boldsymbol{x})$ define an orthogonal frame locally by $\{\hat{\boldsymbol{e}}_1 = \boldsymbol{f}(\boldsymbol{x}), \hat{\boldsymbol{e}}_2 = \mathrm{R}\boldsymbol{f}(\boldsymbol{x})\}$ where R is a 90° counter-clockwise rotation. Independent differential invariants are then expressed in an intuitive way as proportional to the angular speed of rotation of the frame as it is moved along its basis vectors. We can derive two such invariants, which we name the *swerve* ($\hat{g}_1$) and the *splay* ($\hat{g}_2$) due to the intuitive characteristics that they measure, as illustrated by Fig. 5.1.

Having started with the geometric intuition of the invariants, we will now proceed with the process of deriving algebraic expressions for them. We will base this on the local, second order description we have for $\boldsymbol{f}(\boldsymbol{x})$ following equations 5.1 and 5.2. The frame vectors are given by putting $\boldsymbol{x} = \boldsymbol{0}$ which yields:

$$\boldsymbol{f}(\boldsymbol{0}) = \hat{\boldsymbol{e}}_1 = s\frac{\boldsymbol{u}}{|\boldsymbol{u}|}$$

where

$$\boldsymbol{u} = \mathrm{M}_1\boldsymbol{\ell}$$

where M_1 is a 2 by 3 matrix containing only first order shape information:

$$\mathrm{M}_1 = \begin{pmatrix} 1 + h_{01}^2 & -h_{01}h_{10} & h_{10} \\ -h_{10}h_{01} & 1 + h_{10}^2 & h_{01} \end{pmatrix}$$

The differential structure of the flow is contained in the set of partial derivatives. We first differentiate $\boldsymbol{f}(\boldsymbol{x})$ and then put $\boldsymbol{x} = \boldsymbol{0}$. Because it is a 2D field ($\boldsymbol{f}(\boldsymbol{x}) = \{f_1(\boldsymbol{x}), f_2(\boldsymbol{x})\}^{\mathrm{T}}$), we have 4 such derivatives in total, or

equivalently one pair of gradients that together describes how the frame will change given some infinitesimal motion:

$$\boldsymbol{J} = \left(\nabla f_1(\boldsymbol{0}) \;\vdots\; \nabla f_2(\boldsymbol{0}) \right)^{\mathrm{T}}$$

The angular speed of rotation of the frame as it moves in the direction of some unit vector $\boldsymbol{k}$ is given by $\left(\boldsymbol{J}^{\mathrm{T}} \hat{\boldsymbol{e}}_2 \right)^{\mathrm{T}} \boldsymbol{k}$. It is natural to define the flow gradient as $\boldsymbol{g} = \boldsymbol{J}^{\mathrm{T}} \hat{\boldsymbol{e}}_2 = \{g_1, g_2\}^{\mathrm{T}}$, and the swerve($\hat{g}_1$) and the splay ($\hat{g}_2$) are given by:

$$\hat{g}_i = \boldsymbol{g}^{\mathrm{T}} \hat{\boldsymbol{e}}_i \tag{5.3}$$

The algebraic expression for $\boldsymbol{g}$ comes to:

$$\boldsymbol{g} = \boldsymbol{H} \frac{\boldsymbol{v}}{|\boldsymbol{u}|^2}$$

where

$$\boldsymbol{v} = \left(\begin{array}{c} \boldsymbol{\ell}^{\mathrm{T}} \mathrm{M}_2 \boldsymbol{\ell} \\ \boldsymbol{\ell}^{\mathrm{T}} \mathrm{M}_3 \boldsymbol{\ell} \end{array} \right)$$

where M_2 and M_3 are 3 by 3 matrices given uniquely by first order shape information:

$$\mathrm{M}_2 = \left(\begin{array}{ccc} -h_{01} - h_{01}^3 & 0 & 0 \\ 2h_{10}(1 + h_{01}^2) & h_{01}(1 - h_{10}^2) & 0 \\ 0 & h_{01}^2 + h_{10}^2 - 1 & -h_{01} \end{array} \right)$$

$$\mathrm{M}_3 = \left(\begin{array}{ccc} h_{10}(h_{01}^2 - 1) & 0 & 0 \\ 2h_{01}(-1 - h_{10}^2) & h_{10}^3 + h_{10} & 0 \\ 1 - h_{10}^2 - h_{01}^2 & 0 & h_{10} \end{array} \right)$$

We can write the expression for $\boldsymbol{g}$ in terms of the light vector ($\boldsymbol{\ell}$ which is in the camera frame, considered constant), first order shape (M_1, M_2, M_3) and second order shape ($\boldsymbol{H}$):

$$\boldsymbol{g}^{\mathrm{T}} = \frac{1}{\boldsymbol{\ell}^{\mathrm{T}} \mathrm{M}_1^{\mathrm{T}} \mathrm{M}_1 \boldsymbol{\ell}} \left(\begin{array}{c} \boldsymbol{\ell}^{\mathrm{T}} \mathrm{M}_2 \boldsymbol{\ell} \\ \boldsymbol{\ell}^{\mathrm{T}} \mathrm{M}_3 \boldsymbol{\ell} \end{array} \right) \boldsymbol{H}$$

It is also beneficial to normalize the vector $\boldsymbol{v}$. This formulates $\boldsymbol{g}$ as a unit vector applied to the Hessian, multiplied with a non-negative scalar:

$$\boldsymbol{g}^{\mathrm{T}} = \frac{|\boldsymbol{v}|}{|\boldsymbol{u}|^2} \left(\frac{\boldsymbol{v}}{|\boldsymbol{v}|} \boldsymbol{H} \right)$$

For the invariants, we have:

$$\hat{g}_i(\nabla h, \boldsymbol{\ell}) = \frac{|\boldsymbol{v}|}{|\boldsymbol{u}|^2}\left(\frac{\boldsymbol{v}}{|\boldsymbol{v}|}\boldsymbol{H}\ \hat{\boldsymbol{e}}_i\right) \tag{5.4}$$

Now, for any two unit vectors $\boldsymbol{k}_1$ and $\boldsymbol{k}_2$, a second order directional derivative of the smooth 2D function $h(\boldsymbol{x})$ (the surface) is achieved through $\boldsymbol{k}_1^T\boldsymbol{H}\ \boldsymbol{k}_2$. This is equivalent to differentiation first in the $\boldsymbol{k}_1$ direction then $\boldsymbol{k}_2$, regardless of the relationship between $\boldsymbol{k}_1$ and $\boldsymbol{k}_2$ (orthogonality is not required). Thus, the invariants of the flow field $\hat{g}_1$ and $\hat{g}_2$ are directly proportional to second order derivatives of the surface and the factor $|\boldsymbol{v}|/|\boldsymbol{u}|^2$ is positive. When the differentiation is done in the moving frame (i.e. along the vectors $\hat{\boldsymbol{e}}_1$ and $\hat{\boldsymbol{e}}_2$) we use the notation $\hat{h}_{ij}$. Thus, for example, $\hat{h}_{11} = \hat{\boldsymbol{e}}_1^T\boldsymbol{H}\ \hat{\boldsymbol{e}}_2 = \hat{\boldsymbol{e}}_2^T\boldsymbol{H}\ \hat{\boldsymbol{e}}_1$.

We now consider the special case when the point is viewed fronto-parallel, i.e. when $\nabla h = \boldsymbol{0}$. The invariants become simpler (θ is the globally constant slant angle of light in the camera frame):

$$\left(\begin{array}{c}\hat{g}_1(\boldsymbol{0}, \boldsymbol{\ell}) \\ \hat{g}_2(\boldsymbol{0}, \boldsymbol{\ell})\end{array}\right) = s\tan\theta\left(\begin{array}{c}\hat{h}_{11} \\ \hat{h}_{02}\end{array}\right) \tag{5.5}$$

This is because

$$\text{as}\ \ \nabla h \to \boldsymbol{0},$$
$$\begin{cases}\frac{|\boldsymbol{v}|}{|\boldsymbol{u}|^2} & \to & \tan\theta \\ \frac{\boldsymbol{v}}{|\boldsymbol{v}|} & \to & \frac{\boldsymbol{u}}{|\boldsymbol{u}|} = s\ \hat{\boldsymbol{e}}_2\end{cases}$$

5.4   Contrast gradient

The algebraic expressions for the contrast gradient and the shading gradient come to:

$$\begin{aligned}\nabla I(\boldsymbol{0}) &= \frac{1}{(|\nabla h|^2 + 1)^{3/2}}\boldsymbol{u}^{\mathrm{T}}\ \boldsymbol{H} \\ &= \frac{s|\boldsymbol{u}|}{(|\nabla h|^2 + 1)^{3/2}}\hat{\boldsymbol{e}}_1^{\mathrm{T}}\ \boldsymbol{H} \\ \nabla c(\boldsymbol{0}) &= \frac{\nabla I(\boldsymbol{0})}{I(\boldsymbol{0})}\end{aligned}$$

This can be seen as the flow vector ($\hat{\boldsymbol{e}}_1 = \boldsymbol{f}(\boldsymbol{0})$) applied to the Hessian, and scaled with a non-negative value. It makes sense to describe the gradients in the moving frame ($\hat{\boldsymbol{e}}_1$, $\hat{\boldsymbol{e}}_2$):

$$\left(\begin{array}{c} \hat{c}_1(\nabla h, \boldsymbol{\ell}) \\ \hat{c}_2(\nabla h, \boldsymbol{\ell}) \end{array} \right) = \frac{|\boldsymbol{u}|}{I(\boldsymbol{0})(|\nabla h|^2 + 1)^{3/2}} \left(\begin{array}{c} \hat{h}_{20} \\ \hat{h}_{11} \end{array} \right) \tag{5.6}$$

Like the invariants of the flow, the contrast gradient is proportional to second order derivatives of the surface. Unlike the flow, however, we do not need to assume fronto-parallelity for the directions of differentiation to be of an orthogonal frame. The constant of proportionality is different from that of the flow (they are, however, always non-negative).

As with the flow, when we assume fronto-parallelity ($\nabla h = \boldsymbol{0}$) the contrast gradient assumes a simpler form:

$$\left(\begin{array}{c} \hat{c}_1(\boldsymbol{0}, \boldsymbol{\ell}) \\ \hat{c}_2(\boldsymbol{0}, \boldsymbol{\ell}) \end{array} \right) = -\frac{s}{\tan \theta} \left(\begin{array}{c} \hat{h}_{20} \\ \hat{h}_{11} \end{array} \right) \tag{5.7}$$

5.5 Local shape from illuminance flow

If we assume a point with fronto-parallelity, and if we know the global light slant angle (θ), then using the illuminance flow and the shading gradient together, one can solve for the second order shape of the point (the Hessian, in the coordinates of the moving frame):

$$\left\{ \begin{array}{rcl} \hat{h}_{02} & = & s \; \hat{g}_2 / \tan \theta \\ \hat{h}_{20} & = & -s \; \hat{c}_1 \tan \theta \\ \hat{h}_{11} & = & -s \; \hat{c}_2 \tan \theta \\ \hat{h}_{11} & = & s \; \hat{g}_1 / \tan \theta \end{array} \right. \tag{5.8}$$

For Eqs. 5.8 to be useful for shape inference, θ must be known, and sufficiently different from $0°$ and $90°$ ($\tan \theta$ must be neither 0 nor ∞). For the fronto-parallel case, these degenerate cases occur exactly when illuminance flow is not defined. The sign ambiguity s has carried through to the estimates of shape. Notice that it is the same s for all the estimated parameters, i.e. the estimated Hessian as a whole has a sign ambiguity.

Estimation of the global illuminant slant

Regarding the light slant angle (θ), if $\hat{h}_{11} \neq 0$ then we have:

$$\theta = \tan^{-1} \sqrt{-\hat{g}_1 / \hat{c}_2} \tag{5.9}$$

There are two singular cases for which $\hat{h}_{11} = 0$ and θ cannot be calculated in this manner: *a)* the shape is rotationally invariant (the Hessian is a scaled identity matrix, the principal curvatures are equal), and *b)* the moving frame $\{\hat{e}_1, \hat{e}_2\}$ is aligned with the frame of the principal curvatures of the shape. Especially in the presence of noise, the closer to the singular cases the less stable the estimate of θ.

For a given object, there are always points that are fronto-parallel to the camera. However, it is impossible to determine where these points are (without limiting constraints or more information) and it is not certain that illuminance flow is defined there (no 3D texture, or the points could be in shadow). Without knowing which points are fronto-parallel, one approach is to take all possible points, and form an average estimate as

$$\theta_{estim} = \tan^{-1} \left\langle \sqrt{-\hat{g}_1/\hat{c}_2} \right\rangle \tag{5.10}$$

where averaging $\langle \cdot \rangle$ is over all valid positions. We will investigate experimentally how the estimates of Eq. 5.8 and 5.10 will perform when assuming ground-truth illuminance flow is given.

From now on, we will assume that θ is also given (but will investigate through simulation the validity of its estimation by Eq. 5.10). The relations of Eq. 5.8 holds only for the fronto-parallel case, but we will show experimentally that reasonable results are achieved on average.

5.6   Experiments

Experiments were conducted on random surfaces. These were generated as the simulation of 2D stochastic Gaussian processes. The surfaces were generated in the Fourier domain as a fixed power spectrum with a randomly generated phase spectrum. The power spectrum is shaped as an isotropic Gaussian function ($|H(\boldsymbol{\omega})| = \exp(-\sigma^2 \boldsymbol{\omega}^{\mathrm{T}} \boldsymbol{\omega})$). The choice of the Gaussian power spectrum provides us with surfaces with energy in one single scale (σ). This is in contrast with the brownian (fractal) surface, whose power spectrum follows a power law[30]. The Brownian surface has information on all scales due to its fractal, self repeating statistics, and is not appropriate to model the shape at macro scale. The Brownian surface is among the most popular models for surface texture, even though it is not differentiable. Our power spectrum, being Gaussian, has infinite support in the Fourier domain just like the Brownian surface, but unlike the Brownian process our surface

is differentiable.

For our experiment, we assume that illuminance flow is correctly observed where it is defined. We also assume we know the sign of the flow, so that we do not need to concern ourselves with the sign ambiguity ($s = 1$). There are two cases when illuminance flow is not defined: *a)* the surface is in shadow ($\boldsymbol{n}^{\mathrm{T}}\boldsymbol{\ell} < 0$) and *b)* the illumination is orthogonal to the tangent plane ($\boldsymbol{n}^{\mathrm{T}}\boldsymbol{\ell} = 1$ and the light vector projects to zero). We ignore points in shadow and only consider points sufficiently far from orthogonal illumination by $\boldsymbol{n}^{\mathrm{T}}\boldsymbol{\ell} < 0.97$. It is relatively easy to detect points where $\boldsymbol{n}^{\mathrm{T}}\boldsymbol{\ell} = 1$; they are points where the image intensity attains a local maximum and where flow-lines converge. One can thus avoid these points as well as a small radius around them.

We calculated the contrast and shading gradients and the illuminance flow using Eq. 5.3, based on the Jacobian of the flow field. Using Eq. 5.8 the Hessian was estimated, and an error between ground-truth and estimation gathered. Likewise, θ was estimated using Eq. 5.10 and compared to the groundtruth.

The roughness of the surface is measured by the RMS slope ($\tau = \langle |\nabla h|^2 \rangle^{1/2}$). In the experiment we vary the roughness from 0.05 to 1 in increments of 0.05, which is regulated by uniformly multiplying the height map with a value after it has been generated through the Fourier transform. The second variable is the illuminant slant θ which we vary from $10°$ to $80°$ in increments of $10^c irc$.

The ground-truth Hessian with respect to the local moving frame is calculated in full. The elements ($\hat{h}_{02}$, $\hat{h}_{20}$ and $\hat{h}_{11}$) are normally distributed, with zero mean. For the roughest surfaces ($\tau = 1$) the standard deviation for $\hat{h}_{11}$ is approximately 0.0115 and for $\hat{h}_{02}$ and $\hat{h}_{20}$ it is 0.02. The standard deviation is proportional to τ.

Typical surfaces in the experiments are illustrated in the top rows of Fig. 5.2 and Fig. 5.3, where we show the surfaces also in different viewpoints for the benefit of the reader. We adapt a frontal view of the surfaces when performing the experiments, and we assume the ground truth flow is given.
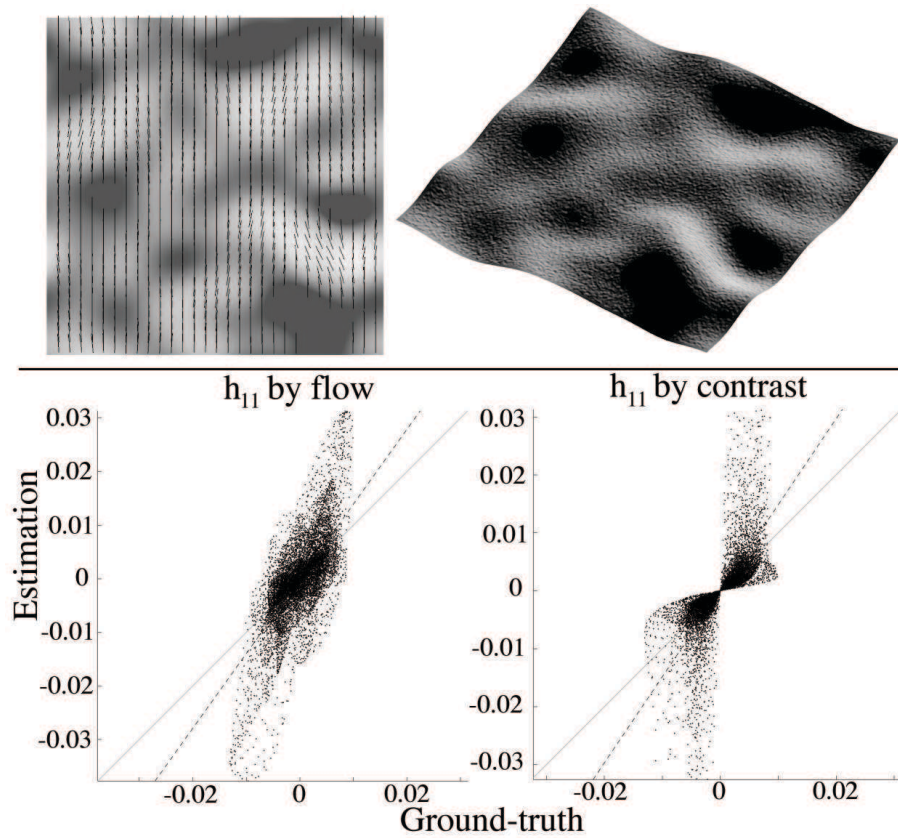
Figure 5.2: Illustration of a low roughness surface($\tau = 0.15$) with low illumination ($\theta = 10°$), and the estimation of its local shape. Top left: The viewing condition of the experiment. The surface is rendered with shading and the illuminance flow at sparsely sampled points are illustrated by black lines. Top right: An oblique view of the same surface, which is rendered with surface texture. Bottom row: Scatter plots of the estimations vs. ground-truth for the $\hat{h}_{11}$ shape parameter, estimated through the flow(left plot) and the contrast(right plot). The dotted line is the best fitting line in the data, the grey line is the line with slope one for reference.
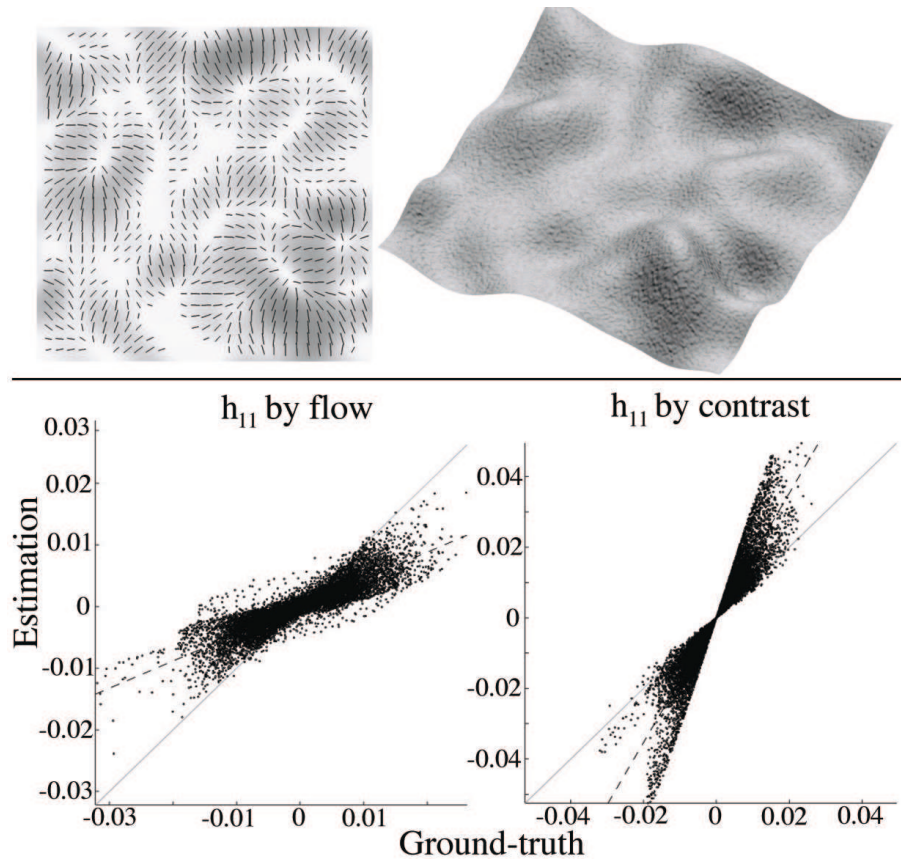
Figure 5.3: Same type of illustration as that of Fig. 5.2, but for a higher surface relief ($\tau = 0.85$), and close to fronto-parallel illumination ($\theta = 80°$).

5.7 Results

We found that the estimates of Eqs. 5.8 correlate quite well with the ground truth shape of the surface, as is evident from Fig. 5.4 and Fig. 5.5. Furthermore, the estimation of the global illuminant slant angle (θ) is found to behave very stable, although with systematic error. It is monotonic with respect to the groundtruth, as is evident from figure 5.6.

The estimates based on the flow ($\hat{h}_{02}$ and $\hat{h}_{11}$) behaved nearly identical, as did the estimates based on the contrast gradient ($\hat{h}_{02}$ and $\hat{h}_{11}$). We will therefore only display the two parallel estimations of $\hat{h}_{11}$.

While the correlation is high for most settings, this misses to pick up on a phenomenon that occurs when the slant angle (θ) is closer to the singular cases of 0° and 90°. Because our estimates are derived for the fronto parallel case only, it cannot handle these cases, which is readily seen in the propor-
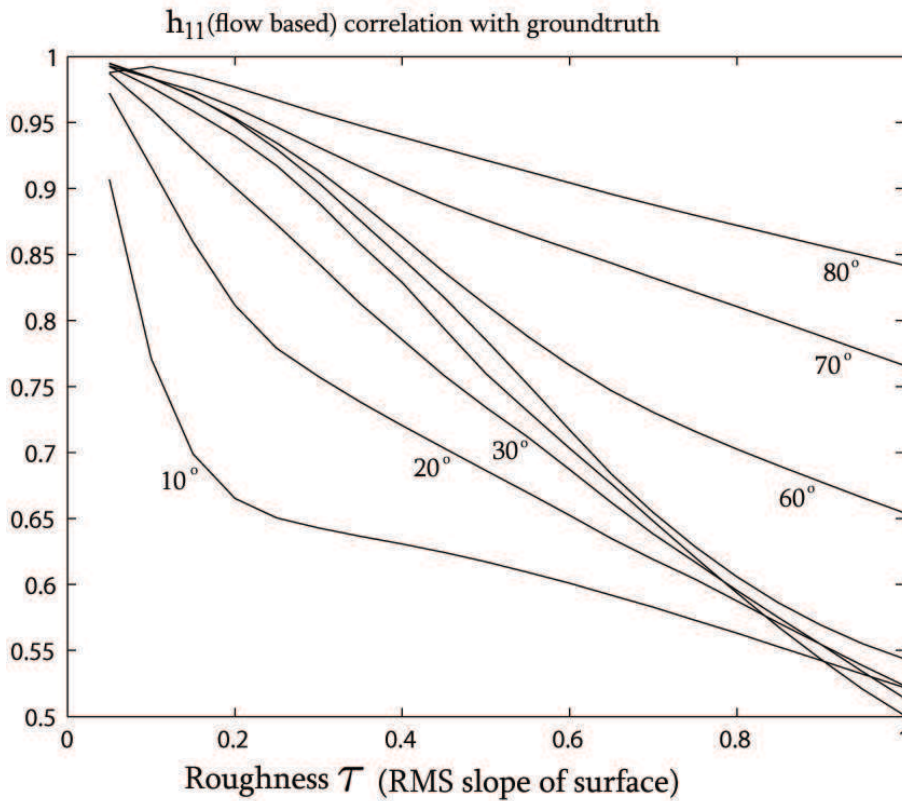
Figure 5.4: Correlation of the illuminance flow based $\hat{h}_{11}$ estimate with ground-truth as a function of the roughness τ. The curves differ in illuminant slant (θ), the values of which are indicated in the figure. The parameter $\hat{h}_{02}$, which is also estimated by the flow follows nearly identical curves and is therefore omitted.

tionality constants of Eqs. 5.8. For $\theta = 80°$, even though our results show the highest of correlation in this case, the proportionality constant of Eqs. 5.8 is not accurate. When plotting the data together with the groundtruth as done in Fig. 5.2 and Fig. 5.3 we note that the slope of the best fitting line (dotted line) is not unity. The best fitting line is very close to a slope of one if θ is kept in the interval of $[25°, 65°]$.

From the scatter plots, we note a stability at zero for the contrast gradient, but not for the flow measures. This is because of the different nature of the flow field and the contrast gradient, as outlined by Eqs. 5.4 and 5.6. The key is that the contrast gradient is proportional to the orthogonal second order derivative even for the non fronto-parallel case. The gradient of the flow field, on the other hand, is not proportional to an orthogonal second
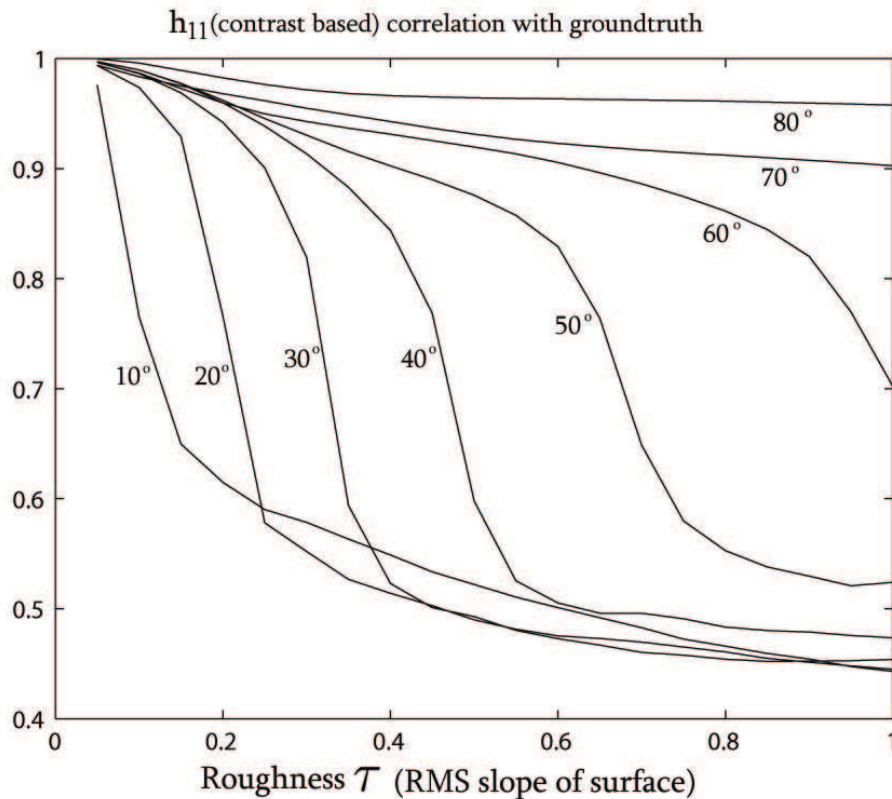
Figure 5.5: Correlation of the contrast based $\hat{h}_{11}$ estimate with ground-truthas a function of the roughness τ. The curves differ in illuminant slant (θ), the values of which are indicated in the figure. The parameter $\hat{h}_{20}$, which is also estimated by the contrast follows nearly identical curves and is therefore omitted.

order derivative (the first direction of differentiation is dependent on ∇h). For the estimates of $\hat{h}_{02}$ and $\hat{h}_{11}$, when based on the contrast, they should be exactly zero when the ground-truth is zero, while this is not predicted to happen for the illuminance flow invariants (that estimate $\hat{h}_{20}$ and $\hat{h}_{11}$).

The contrast gradient has a high sensitivity to small shading values due to the division with $I(\mathbf{0})$, evident in Eq. 5.6. If we are close to a shadow (on the macro scale), or more generally, if we are in grazing illumination, then the contrast gradient approaches infinity. This explains the large amount of outliers for the contrast based estimates in the lower right panel of figure 5.2 (The outliers are all coming from points in grazing illumination, or close to shadows). The contrast based estimates perform considerably better when the illumination is not grazing, as evident in the lower right panel of figure
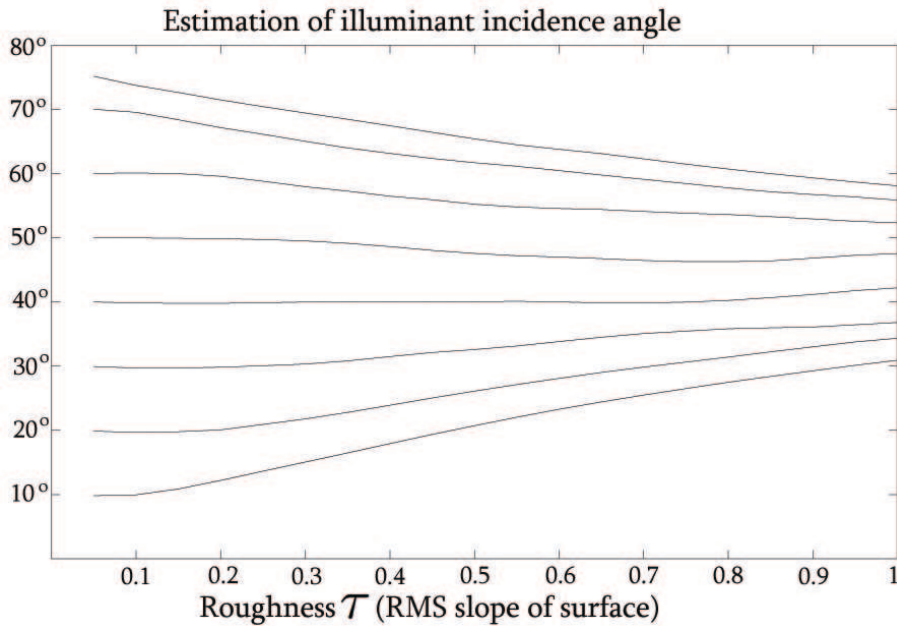
Figure 5.6: Estimation of illuminant slant(θ) as a function of surface roughness(τ). Curves differ in the ground-truth illuminant slant (θ), which are (from top to bottom) linearly spaced from 80° to 10°.

5.3. The flow based estimates do not suffer from this drawback, but instead perform worse at close to orthogonally illuminated parts of the image.

5.8 Conclusions

As our experiments have shown, the rather simple algorithm for estimating second order local shape (i.e. the Hessian) in Eq. 5.8 will work reasonably well. The algorithm of Eq. 5.10, that estimates the global slant angle of the light source, was shown to especially behave stable and quite accurate for surfaces of lower roughness.

The method for estimating the illuminant slant angle was derived for fronto-parallel points (were $\nabla h = \mathbf{0}$) and yielded a very simple relation (Eq. 5.9). Because we do not know which points are fronto-parallel we suggested to simply do averaging over all valid points (as Eq. 5.10 illustrates). If instead we truly have the illuminant slant angle (θ) as a given, then we can use Eq. 5.9 as a way to define a set in the image satisfying

$$\hat{c}_2 \tan(\theta)^2 + \hat{g}_1 = 0$$

The set of fronto-parallel points are a subset of all such points, (assuming that illuminance flow is defined on them). How useful this observation is, remains to be seen, and will be a worthwhile subject for future work.

For all the techniques used in this chapter, observed illuminance flow is needed. The fact that shape inference can be done at all in this strictly local fashion is quite fascinating in itself. New information is made available by observing illuminance flow, and shape from this quantity is a novel topic indeed. The work on shape from illuminance flow is still in its early stages. For a future, powerful algorithm for shape inference we envision much more than this purely local approach. Considerations of the global structure of the flow field and contextual information (such as the contours of a segmented object) should be taken advantage of.

In order to arrive at a fully practical algorithm, several issues remain to be addressed. Importantly, we assume a separation into macro-scale and meso-scale surface variation, where the meso-scale variation adds only to the surface texture of the object. In real world applications, this separation is not always straightforward. This is an issue of scale selection. The human visual system is very efficient at doing this, and adapts the scales readily depending on what is given attention to. In the same way, an algorithm for shape from illuminance flow needs to set the scales, which decides what is texture and what is shape to be inferred.

An important perceptual ambiguity in vision is the so-called bas-relief ambiguity[55]. Essentially, this states that it is difficult to distinguish between the shape of a surface $h(\boldsymbol{x})$ and a transformed version: $ah(\boldsymbol{x})+bx+cy$. This three parameter transform will leave the cast shadows over the surface identical to that of the original surface, if the light source similarly undergoes a transformation. Thus, in general, it is very difficult to distinguish shapes that differ only by a height scaling and an added plane. If we assume that illuminance flow is observable over the shape, then the height scaling parameter (a above) of this transform is disambiguated, while the addition of the tilted plane is not. Illuminance flow may still have the power to disambiguate the full bas-relief ambiguity, if we consider the global structure of the flow, which still remains to be investigated. If one adds a non-frontal plane to the surfaces we have used in the experiments, the global structure of the flow clearly changes, even though the average results of using Eq. 5.8 does not change much. The set of fronto-parallel points change as the plane is added, which is why the use of the condition $\hat{c}_2 \tan(\theta)^2 + \hat{g}_1 = 0$, as mentioned above, might prove useful in future work.

References

[1] R. Dawkins, "The God delusion", Bantam Press UK (2006).

[2] S. M. Karlsson, J. Bigun, "Multiscale complex moments of the local power spectrum", JOSA A **24**, 618–625 (2007).

[3] J. Bigun, T. Bigun, K. Nilsson, "Recognition by symmetry derivatives and the generalized structure tensor", PAMI **26**, 1590–1605 (2004).

[4] J. Bigun, G.H. Granlund, "Optimal orientation detection of linear symmetry", ICCV, 433–438 (1987).

[5] M. Chantler, "Photex photometric image database," Online resource http://www.macs.hw.ac.uk/texturelab/resources/databases/Photex/index.htm (October, 2007).

[6] O. Drbohlav, M. Chantler, "Illumination-invariant texture classification using single training images", intern. workshop on texture analysis and synthesis, 31–36 (2005).

[7] M. Chantler, "Why illuminant direction is fundamental to texture analysis", Vision, Image and Signal Processing **142**, 199-206 (1995).

[8] M. Chantler, M. Petrou, A. Penirsche, M. Schmid, G. McGunnigle, "Classifying surface texture while simultaneously estimating illumination direction", IJCV **62**, 83–96 (2005).

[9] M. Chantler, G. Delguste, "lluminant-tilt estimation from images of isotropic texture", Vision, Image and Signal Processing **144**, 213 – 219 (1997).

[10] J.J. Koenderink, S.C. Pont, "Irradiation direction from texture," JOSA A **20**, 1875–1882 (2003).

[11] S.C. Pont, J.J. Koenderink, "Irradiation orientation from obliquely viewed texture," DSSCV workshop, 205–210 (2005).

[12] S. M. Karlsson, S.C. Pont, J.J. Koenderink, "Illuminance flow over anisotropic surfaces," JOSA A **25**, 282–291 (2008).

[13] S. K. Nayar, M. Oren, "Visual appearance of matte surfaces", Science **267**,1153 – 1156 (1995).

[14] R. Gregory, "Knowledge in perception and illusion", Phil. Trans. R. Soc. Lond. B **352**. 1121–1128 (1997).

[15] J. P. Sartre, "La Nausee", Editions Gallimard (1938).

[16] J. Wu, "Rotation invariant classification of 3D surface texture using photometric stereo", Ph.D. Thesis, Heriot-Watt University (2003).

[17] E.W. Hobson, "The theory of spherical and ellipsoidal harmonics", Chelsea Publishing Co. (1955).

[18] J. Schlag, "Fast embossing effects on raster image data", Academic Press, Cambridge (1994).

[19] J. F. Blinn, "Simulation of wrinkled surfaces", SIGGRAPH **12**, 286–292 (1978).

[20] R. C. Dubes, A. K. Jain, "Random field models in image analysis", J. Appl. Stat. **16**, 131–164 (1989).

[21] J. Dong, "Three-dimensional surface texture synthesis", Ph.D. Thesis, Heriot-Watt University (2003).

[22] J. J. Koenderink, A. J. van Doorn, K. J. Dana, S. Nayar, "Bidirectional reflection distribution function of thoroughly pitted surfaces", IJCV **31**, 129–144 (1999).

[23] B. M. ter Haar Romeny, L. Florack, J. J. Koenderink, M. A. Viergever, "Scale-space: Its natural operators and differential invariants", IPMI, 239–255 (1991).

[24] J.J. Koenderink, A.J. van Doorn, "Geometrical modes as a general method to treat diffuse interreflections in radiometry", JOSA A **73**, 843–850 (1983).

[25] J.A. Ogilvy, "Wave scattering from rough surfaces", Reports on Progress in Physics **50**, 1553–1608 (1987).

[26] D. Mahmoud-Ghoneim et al.,"Three dimensional texture analysis in MRI: A preliminary evaluation in gliomas", JMRI **21**, 983–987 (2003).

[27] C. E. Rasmussen, C. K. I. Williams, "Gaussian processes for machine learning", The MIT Press (2006).

[28] A. Pentland, "Fractal-based description of natural scenes", PAMI **6**, 661-764 (1984).

[29] A. Pentland, "Finding the illuminant direction," JOSA **72**,448–455 (1982).

[30] P. Kube, A. Pentland, "On the imaging of fractal surfaces," PAMI **10**, 704–707 (1988).

[31] A. Pentland, "The visual inference of shape: Computation from local features,", Ph.D. Thesis. (MIT, 1982).

[32] S. Louw, A.M.L. Kappers, J.J. Koenderink, "Haptic detection thresholds of Gaussian profiles over the whole range of spatial scales", Experimental Brain Research **132**, 369–374 (2000).

[33] M. Varma, A. Zisserman, "Estimating illumination direction from textured images," CVPR, 179–186 (2004).

[34] R. Gohar, Gohar, H. Rahnejat. "Fundamentals of tribology", Imperial College Press (2008).

[35] D. Knill, "Estimating illuminant direction and degree of surface relief", JOSA A **7**, 759–775 (1990).

[36] Q. Zheng, R. Chellappa, "Estimation of illuminant direction, albedo and shape from shading" , Physics-Based Vision: Shape Recovery, 39–61, (1992).

[37] Y. Zhang, Y.H. Yang, "Illuminant direction determination for multiple light sources," in *Proc. of the CVPR*, (IEEE, 2000), pp. 269–276.

[38] X. Llado, A. Oliver, M. Petrou, J. Freixenet, J. Marti "Simultaneous surface texture classification and illumination tilt angle prediction" Brit. Mach. Vis. Conf. (2003), Norwich.

[39] M. Brooks, B. Horn, "Shape and source from shading", Proc. of 9th Int. Joint Conference on Artificial Intelligence, . 932–936 (1985).

[40] E. Cartan, "La theorie des groupes finis et continus et la geometrie differentielle traitees par la methode du repere mobile". Gauthier-Villars, Paris (1937).

[41] N. Dalal, B. Triggs, "Histograms of oriented gradients for human detection," CVPR, 886–893 (2005).

[42] D. G. Lowe, "Object recognition from local scale-invariant features," In *Proc. of ICCV*, (IEEE, 1999) pp. 1150–1157.

[43] E. Parzen,"On estimation of a probability density function and mode," Ann. Math. Stat. **33**, 1065–1076 (1962).

[44] K. V. Mardia, P. E. Jupp, "Directional statistics," Wiley Series (2000).

[45] K. Mikolajczyk et al., "A comparison of affine region detectors," IJCV **65**, 43–72 (2005).

[46] V. N. Vapnik, "The nature of statistical learning theory," (Springer, 1995).

[47] V. N. Vapnik, S. Golowich, A. Smola, "Support vector method for multivariate density estimation," Advances in Neural Information Processing Systems **12**, 659–665, MIT Press (1999).

[48] M. Aizerman, E. Braverman, L. Rozonoer, "Theoretical foundations of the potential function method in pattern recognition learning," Automation and Remote Control **25**, 821–837 (1964).

[49] T. Joachims, "Making large-Scale SVM learning practical," Advances in kernel methods - support vector learning, (MIT-Press, 1999). Software available online at "http://svmlight.joachims.org/" (June, 2008).

[50] M. Berry, V. Hannay, "Umbilic points on Gaussian random surfaces," J. Phys. A: Math. Gen. **10**, 1809–1821 (1977).

[51] M. S. Longuet-Higgins, "The statistical analysis of a random moving surface," Phil. Trans. R. Soc. A **249**, 321–64 (1956).

[52] A. Gershun, "The light field" translated by P. Moon and G. Timoshenko, J. Math. Phys. **18** 51–151 (1939).

[53] D.J. Kriegman, P.N. Belhumeur, "What is the set of images of an object under all possible illumination conditions?" IJCV **28**, 245 – 60 (1998).

[54] W. Li , C. Wang , D. Xu, B. Luo, Z. Chen, "A study on illumination invariant face recognition methods based on multiple eigenspaces". LNCS **3497**, 131–136 (2005).

[55] P.N. Belhumeur, D. Kriegman, A. Yuille, "The bas-relief ambiguity," IJCV **35**, 33–44 (1999).

[56] F. E. Nicodemus, J.C. Richmond, J.J. Hsia, I.W. Ginsberg and T. Limperis, "Geometrical considerations and nomenclature for reflectance", National Bureau of Standards (1977).

[57] D. Marr, "Vision: A computational investigation into the human representation and processing of visual information", Freeman NY (1982).

Summary

This thesis investigates the concept of illuminance flow, which is a novel topic to the area of computer vision and image analysis. Illuminance flow is a 2D flow field in the image produced by projecting the light vector into the local tangential plane of every visible point of a smooth object. The main application of illuminance flow is to monocular shape inference. Specifically, given a single image of an object, what is the shape of the visible surface of the object? In this sense, shape from illuminance flow is similar to shape from texture and shape from shading, but fundamentally independent of both. Illuminance flow is only observable from objects with rough 3D surface texture. Therefore, much of the work has been on 3D texture, and its special dependence on light direction, which is shown to be very different from "regular" 2D texture. Three fourths of the thesis work is focused on the task of estimating the flow, especially how this can be achieved with the help of the so-called structure tensors. One fourth of the work deals with the approach of "shape from illuminance flow", a subject which is still in its infancy.

To make the material more accessible, the introduction (chapter 1) addresses underlying topics in alternative and more elaborate ways. The introduction outlines the general scientific view points, modeling and problems. It has as purpose to fit the work and the approach into a broader context, and it is hoped that a less technically versed audience will find it interesting and transparent. The main body of work of this thesis consists of the four journal publications that has resulted from the work done on the topic of illuminance flow. These constitute chapters 2-5 that are presented in chronological order.

Chapter 2 investigates the use of the structure tensor to estimate illuminance flow when the surface meso-scale variation is anisotropic (has a fair amount of directionality) and the view is fronto-parallel. The structure tensor measures directionality within the image, and for an anisotropic texture, there is one additional source of image directionality. The theory of illuminance flow estimation by structure tensors is expanded upon, by introducing the surface based structure tensor. How will the observable, image-based tensor behave given some directionality as defined by the hidden structure tensor of the surface? The Hessian-based structure tensor is used to augment the estimate of the gradient based estimate in a very simple (yet non-linear) scheme, to improve on the estimate of illuminance flow. This chapter contains the results of rather extensive experiments on both computer simulations (renderings of Gaussian surfaces) as well as real-world textures. The theory is shown to conform well for reasonable amounts of anisotropy.

An important property of all unsupervised illuminant tilt estimators in the literature is that they are all strongly correlated to the gradient based structure tensor. We elaborate on this dependence, and explain how that makes our theory quite general.

Chapter 3 generalizes the theory of illuminance flow estimation by structure tensors for both anisotropic texture *and* oblique viewing. One might be tempted to handle anisotropy and oblique viewing in the same way (one combined affine transform of an isotropic template pattern, which is reasonable when modeling 2D texture), but this would be fundamentally wrong for 3D texture. The theory of chapter 2 is refined in chapter 3 by using matrix formulations and further compacted by exploiting general properties of the structure tensor. Theoretical predictions based on the revised theory are presented, and compared with experimental results on rendered images, and shown to conform well. The results of this chapter are also applicable to other areas of computer vision. The structure tensor is used as the state-of-the-art in affine invariant texture matching (where it is renamed to the second moment matrix). Our theory (of both chapters 2 and 3) predicts how the tensor will behave when there is 3D texture present.

Chapter 4 is more practical and takes an engineering approach to the problem of illuminance flow estimation. We investigate the estimation of illuminance flow using standard, state-of-the-art computer vision techniques. We choose a set of low-level features (Histograms of Oriented Gradient features (HOGs)) and in a regression setting we see how well a predictor for supervised flow estimation can be achieved. Support vector machines, with

different kernels, as well as a regular linear ridge regressor are used. We found that the optimal solution shows close resemblance to the gradient based structure tensor, for all the regressors used.

The chapter also presents new theoretical results arrived at during this work, showing in detail how the structure tensor and the HOGs are connected. This relation will benefit affine invariant texture/object matching and object tracking using HOGs. HOGs are treated (as they should always be) as crude (yet fast) non-parametric density estimation.

Chapter 5 addresses the issue of shape from illuminance flow. It is the first attempt at using the flow for recovering the shape of the underlying surface. The chapter focuses on strictly local shape inference, and shows how the differential invariants of the flow field can be used for this purpose. In addition, the global position of the light source can be partly inferred as well, at least for the experimental surfaces we consider. A simple algorithm for estimating the second order local shape (the Hessian) is derived and shown experimentally to perform quite well on simulations of Gaussian surfaces. Being local, based only on the differential invariants of the flow, and totally unsupervised one could look at this as a bottom-up approach. More global and contextual algorithms (i.e. introducing top-down information) can quite easily be construed, and hopefully will in future research. The properties and relations derived in chapter 5 will then no doubt be useful.

Samenvatting

In dit proefschrift wordt onderzoek beschreven naar het concept "illuminance flow", een nieuw onderwerp voor het gebied van computer vision en beeld-analyse. Illuminance flow is een 2D flow veld in het beeld, verkregen door de lichtvector te projecteren op het lokale tangent vlak van ieder zichtbaar punt van een glad object. De belangrijkste toepassing van illuminance flow is monoculaire vorm-schatting. Specifiek, gegeven een enkel beeld van een object, wat is de vorm van het zichtbare oppervlak van het object? In deze zin lijkt vorm uit illuminance flow op vorm uit textuur en vorm uit shading, maar is fundamenteel onafhankelijk van beide. Illuminance flow is alleen waarneembaar op objecten met ruwe 3D oppervlakte-textuur. Daarom is veel van het werk gericht op 3D textuur en zijn speciale afhankelijkheid van licht-richting, die zeer verschillend is van "reguliere" 2D textuur. Driekwart van het proefschrift is gefocusseerd op het schatten van de flow, speciaal hoe dit bereikt kan worden met behulp van zogenaamde structuur-tensoren. Een kwart van het werk betreft de benadering van "vorm uit illuminance flow", een onderwerp dat nog in het beginstadium verkeert.

Om het materiaal toegankelijker te maken, beschrijft de introductie (hoofdstuk 1) diverse concepten van de achtergrond op alternatieve en meer uitgebreide manieren. De introductie schetst de algemene wetenschappelijke standpunten, modellering en problemen. Het heeft tot doel het werk en de benadering in een bredere context te plaatsen en voor een minder technisch bedreven publiek interessant en transparant te maken. Het voornaamste deel van dit proefschrift bestaat uit de vier tijdschrift-publicaties die zijn geresul-

teerd uit het gedane werk op het onderwerp illuminance flow. Deze vormen hoofdstukken 2–5, die zijn gepresenteerd in chronologische volgorde.

Hoofdstuk 2 onderzoekt het gebruik van de structuur-tensor om illuminance flow te schatten, wanneer de oppervlak meso-schaal variatie anisotroop is (met een flinke gerichtheid) en de kijkrichting fronto-parallel. De structuur-tensor meet de gerichtheid in het beeld en voor anisotrope textuur is er een additionele bron van beeld-gerichtheid. De theorie van illuminance flow schatting door structuur-tensoren is uitgebreid door de introductie van de oppervlakte-gebaseerde structuur-tensor. Hoe zal de waarneembare, beeld-gebaseerde tensor zich gedragen gegeven enige gerichtheid zoals gedefinieerd door de niet-waarneembare structuur-tensor van het oppervlak? De Hessiaan-gebaseerde structuur-tensor wordt gebruikt om de gradient-gebaseerde schatting van illuminance flow te verbeteren volgens een zeer simpel (maar niet-lineair) schema. Dit hoofdstuk bevat de resultaten van zeer uitgebreide experimenten op zowel computer simulaties (weergaven van Gaussische oppervlakken) als echte texturen. Er wordt aangetoond dat de theorie de resultaten goed beschrijft voor redelijke waarden van anisotropie.

Hoofdstuk 3 generaliseert de theorie voor illuminance flow schatting door structuur-tensoren voor zowel anisotrope textuur als schuine kijkrichtingen. Men zou in de verleiding kunnen komen om anisotropie en schuin kijken op dezelfde manier te behandelen (een gecombineerde affine transformatie van een isotroop voorbeeld patroon, wat redelijk is voor het modelleren van 2D textuur), maar dit zou fundamenteel verkeerd zijn voor 3D textuur. De theorie van hoofdstuk 2 is verfijnd in hoofdstuk 3 door matrix formuleringen te gebruiken en nog compacter gemaakt door de algemene eigenschappen van de structuur-tensor te gebruiken. Theoretische voorspellingen gebaseerd op de aangepaste theorie worden gepresenteerd en vergeleken met experimentele resultaten van beeldweergaven, en komen goed overeen. De resultaten van dit hoofdstuk zijn ook toepasbaar in andere gebieden van computer vision. De structuur-tensor wordt gebruikt als state-of-the-art in affine invariante textuur matching (waar het de tweede moment matrix wordt genoemd). Onze theorie (van hoofdstukken 2 en 3) voorspelt hoe de tensor zich zal gedragen wanneer 3D textuur aanwezig is.

Hoofdstuk 4 is meer toegepast en volgt een engineering benadering naar illuminance flow schatting. We onderzoeken de schatting van illuminance flow door standaard, state-of-the-art computer vision technieken te gebruiken. We kiezen een verzameling laag-niveau eigenschappen (Histograms of Oriented Gradient eigenschappen (HOGs)) en gaan via regressie na hoe

goed een voorspeller voor supervised flow schatting kan worden bereikt. Zowel support vector machines, met verschillende kernels, als reguliere lineaire richel regressie-methodes werden gebruikt. We vonden dat de optimale oplossing een goede gelijkenis laat zien met de gradient gebaseerde structuur-tensor, voor alle gebruikte regressie-methodes.

Het hoofdstuk presenteert ook nieuwe theoretische resultaten die in detail laten zien hoe de structuur-tensor en de HOGs aan elkaar gerelateerd zijn. Deze relatie zal baat hebben voor affine invariante textuur/object matching en object tracking met gebruikmaking van HOGs. HOGs zijn behandeld (zoals ze altijd zouden moeten zijn) als ruwe (maar snelle) niet-parametrische dichtheids-schatting.

Hoofdstuk 5 gaat over vorm uit illuminance flow. Het is de eerste poging om de flow te gebruiken om de vorm van het onderliggende oppervlak te schatten. Het hoofdstuk focusseert op strikt lokale vorm-schatting en laat zien hoe de differentiele invarianten van het flow veld gebruikt kunnen worden voor dit doel. Daarnaast kan de globale positie van de lichtbron ook gedeeltelijk geschat worden, tenminste voor de experimentele oppervlakken die wij beschouwen. Een simpel algorithme voor het schatten van tweede orde lokale vorm (de Hessiaan) is afgeleid en experimenteel bewezen behoorlijk goed te werken op simulaties van Gaussische oppervlakken. Aangezien deze benadering lokaal, alleen gebaseerd op de differentiele invarianten van de flow, en volstrekt unsupervised is, kan deze beschouwd worden als een bottom-up benadering. Meer globale en contextuele algorithmes (bijvoorbeeld die top-down informatie introduceren) kunnen makkelijk geconstrueerd worden, wat hopelijk in de toekomst zal gebeuren. De eigenschappen en relaties die in hoofdstuk 5 zijn afgeleid zullen dan zonder twijfel nuttig zijn.

Publications

- S. M. Karlsson, S.C. Pont, J.J. Koenderink, "Illuminance Flow over Anisotropic Surfaces". Journal of the Optical Society of America A **25**, 282-291 (2008).

- S.M. Karlsson, S.C. Pont, J.J. Koenderink, "Illuminance flow over anisotropic surfaces with arbitrary viewpoint". Journal of the Optical Society of America A **26**, 1250-1255 (2009).

- S.M. Karlsson, S.C. Pont, J.J. Koenderink,"Estimation of Illuminance Flow over Anisotropic Surfaces for Arbitrary Viewpoints". In: Frontiers in Optics 2008 on CD-ROM, presentation nr FMC4, Optical Society of America, Washington, DC (2008).

- S.M. Karlsson, S.C. Pont, J.J. Koenderink, A. Zisserman, "Illuminance Flow Estimation by Regression". Submitted to International Journal of Computer Vision (August, 2009).

- S.M. Karlsson, S.C. Pont, J.J. Koenderink, "Local shape from Illuminance Flow". Submitted to Journal of Mathematical Imaging and Vision (September, 2009).

Acknowledgments

The thesis is dedicated to Oscar van Hoof, who also deserves special thanks. Oscar, a close friend, passed away during the writing of this thesis. No less than six figures contain illuminated spheres that he photographed with enthusiasm (Figures 1.2, 2.1, 3.1, 4.1, 4.2, 5.1). The database of imaged spheres that Oscar constructed was used as training data for the regressors of Chapter 4. More importantly, our discussions not only improved the quality of this work, but changed my frame of mind. He is greatly missed.

Unmeasurable gratitude go out to my supervisors Jan Koenderink and Sylvia Pont for their wonderful support and the fruitful and interesting discussions. It goes without saying that this work would have been impossible without their insightful guidance and valuable contributions. Thank you for the good times, in the past and those still to come.

Thanks are also due to the wonderful people of the physics of man lab who are too numerous to mention. Thanks guys.

Curriculum Vitae

Stefan Karlsson was born in Stafsinge, Sweden, on the 3rd of September 1978. He studied at Halmstad University, where he got a masters degree in Computer Systems Engineering with specialization in image analysis in 2005. As a student, he pursued research in vision, real time systems and image analysis.

His bachelors thesis work was to construct a testbed platform for fingerprint biometrics on a PDA. His master thesis work investigated a set of low level features, *Complex Moments of the Power Spectrum*, their interpretation in terms of structure tensors and application to 2D texture segmentation. After his Masters degree, he continued to work as part of a research internship at the Intelligent Systems laboratory with prof. Josef Bigun, in Halmstad.

In October, 2005, he got an appointment as a PhD student at physics of man group at Utrecht University, under the supervision of Jan Koenderink and Sylvia Pont.