Fourier Analysis and Gabor Filtering for Texture Analysis and Local Reconstruction of General Shapes

Fabio Galasso and Joan Lasenby
University of Cambridge, UK
{fg257 and jl221}@cam.ac.uk

Abstract

Since the pioneering work of Gibson in 1950, Shape-From-Texture *has been considered by researchers as a hard problem, mainly due to restrictive assumptions which often limit its applicability. We assume a very general stochastic homogeneity and perspective camera model, for both deterministic and stochastic textures. A multi-scale distortion is efficiently estimated with a previously presented method based on Fourier analysis and Gabor filters. The novel 3D reconstruction method that we propose applies to general shapes, and includes non-developable and extensive surfaces. Our algorithm is accurate, robust and compares favorably to the present state of the art of Shape-From-Texture. Results show its application to non-invasively study shape changes with laid-on textures, while rendering and retexturing of cloth is suggested for future work.*

1. Introduction

Shape-From-Texture (SFT) has long been considered a difficult problem, and relevant research has often addressed the issue by using overly restrictive assumptions, therefore limiting its applicability. The algorithm that we propose here produces accurate and robust results under very general assumptions, enabling future practical usage.

In the general case of a perspective camera model, a weak assumption of stochastic homogeneity is made ([16]). Both deterministic and stochastic textures are considered. For the latter, homogeneity is equivalent to having the 'same' texture in the scene, i.e. the stochastic process generating the texture is stationary under translation.

As in the relevant literature, we approach SFT as measuring the texture distortion in an image and then reconstructing the 3D surface coordinates.

Texture distortion is based on *local spatial frequency* (LSF). In fact we use the method which we have already introduced in [6, 7], derived from [20]. The Fourier transform of the texture image is used to set minimal groups of Gabor filters, so as to efficiently recover all relevant frequencies in the texture. The method utilizes the multi-scale nature of texture by using all main LSFs.

A novel algorithm is introduced to reconstruct the 3D shape from local variations of each pair of LSFs without any further assumption (e.g. texture representing a plane or displaying a frontal point). The proposed technique takes its motivation from the work of [18], which we extend and use for comparison.

The results we show prove the accuracy and robustness of our algorithm, and a sensitivity to small changes of the textured shape which opens the way to considering SFT as a non-invasive shape analysis technique: 3D changes are studied by laying a texture on the object. Also, as the reconstruction is carried out locally, the algorithm can easily be extended to large and complex objects.

Related work. SFT was pioneered by Gibson ([10]), who refers to texture as a shape cue. It generally includes distortion estimation and surface 3D reconstruction ([12, 9, 21, 16, 11, 3, 19, 1, 14, 18]). A stochastic model includes a wider variety of textures ([12, 16, 21]) and implies local spectral measurements, via the Fourier transform ([16]) or wavelets ([11, 1]).

The method described here allows a multi-scale analysis, while most of the related work on SFT only estimates a single distortion measure (e.g.[20, 18]) or more rarely considers two preferred directions in the spectral domain (e.g. [21, 15])

As well as homogeneity, the most widely adopted method ([12, 20, 18]), texels ([14]) or isotropy ([22]) are also used, but these are seldom found in nature.

Although reconstruction of most of the objects can be done with a simpler orthographic model ([13]), we adopt a more general perspective camera model to also include planar surfaces ([2, 17] and references therein).

2. Texture Description

Here we describe how to efficiently compute all main LSFs, which characterize the distortion measure, by using

978-1-4244-3991-1/09/$25.00 ©2009 IEEE

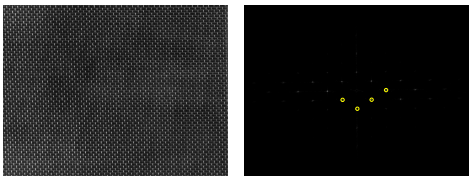


Figure 1. (**left**) Brodatz texture D6 and (**right**) its spectrum amplitude

Fourier analysis and minimal sets of Gabor functions.

2.1. Estimating the Instantaneous Frequencies

We can analyze an image $I(\mathbf{x})$ using a pass-band filter $h(\mathbf{x}, \mathbf{u})$, a function of a point $\mathbf{x} = (x, y)$ and of a central frequency $\mathbf{u} = (u, v)$, which is convolved with the image to provide the local spectrum. As first proposed in [20] and described by us in [6, 7], we choose 2D Gabor functions:

$$h(\mathbf{x}, \mathbf{u}) = \frac{1}{2\pi\gamma^2} e^{-\frac{1}{2\gamma^2}\mathbf{x}\cdot\mathbf{x}} e^{2\pi j \mathbf{x}\cdot\mathbf{u}} \ . \tag{1}$$

These functions satisfy exactly the lower bound for the measure of a frequency $\mathbf{u}$ and its position $\mathbf{x}$, given by the uncertainty principle ([5]). Furthermore the functions are geometrically similar and can therefore be easily compared.

Our goal is to estimate the instantaneous frequencies $\mathbf{u}$ of the image points. [20] and references therein show that this can be done by considering a Gabor function $h(\mathbf{x}, \mathbf{u})$, and its two first order derivatives, $h_x(\mathbf{x}, \mathbf{u})$ and $h_y(\mathbf{x}, \mathbf{u})$:

$$\begin{aligned} |\tilde{u}(\mathbf{x})| &= \frac{|h_x(\mathbf{x}, \mathbf{u}) * I(\mathbf{x})|}{2\pi |h(\mathbf{x}, \mathbf{u}) * I(\mathbf{x})|} \\ |\tilde{v}(\mathbf{x})| &= \frac{|h_y(\mathbf{x}, \mathbf{u}) * I(\mathbf{x})|}{2\pi |h(\mathbf{x}, \mathbf{u}) * I(\mathbf{x})|} \ . \end{aligned} \tag{2}$$

The estimate, $\tilde{\mathbf{u}} = (\tilde{u}, \tilde{v})$, can be assumed to be correct if the measured frequency is in the pass-band of the filter. The method implies therefore that we set the central frequencies $\mathbf{u}$ and the spatial constants γ of the Gabor functions, i.e. their centers and widths, to relevant values.

As we explained in [6, 7], this is done by setting appropriate groups of Gabor functions with the information provided by the Fourier transform of the image.

2.2. Setting the Gabor Filter Parameters

We refer to [7] for a detailed explanation of how the amplitude of the Fourier transform represents the 2D frequencies of an image. Here we consider the Brodatz texture D6 and its spectrum amplitude in figure 1. From the spectrum we deduce that the texture is mainly characterized by four LSFs, marked with small yellow circles (only half of the spectrum need be considered due to symmetry). By slanting and rotating the texture (fig. 2(a)), the four dots of figure 1 become corresponding areas in the spectrum (fig. 2(b)). Figure 2(c) represents the instantaneous frequencies of each of the image pixels for the four LSFs (yellow areas) on the spectrum (as the image is synthesized, the ground truth, i.e. the values of each LSF at each pixel, is known and marked on the spectrum). It is clear from this that non-zero values represent corresponding frequencies of the images, and that contiguous areas correspond to the same LSF. In our algorithm each area is used to set a distinct group of Gabor functions (yellow circles in fig. 2(d)), i.e. to set distinct sets of central frequencies $\mathbf{u}$ (centers of the circles) and spatial constants γ (their radii).

A significant overlapping of the filters increases the robustness of the estimation, while the use of the Fourier transform optimizes the number and positioning of filters, so as to minimize the computational expense.

Finally the method exploits the multi-scale nature of the texture, because all different-scale frequencies are estimated and used in the shape reconstruction.

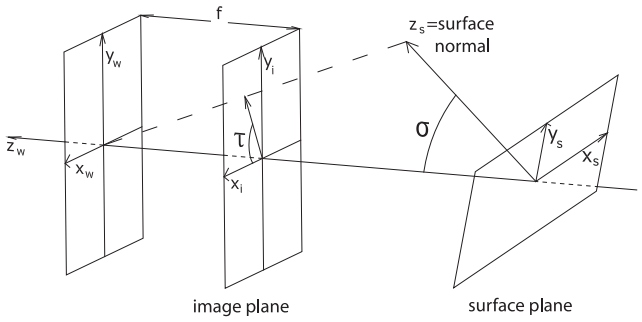


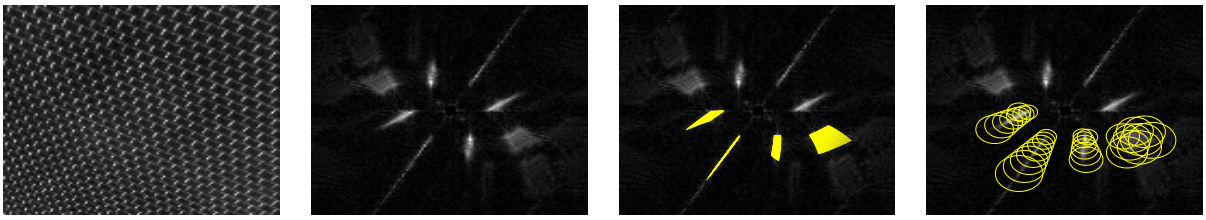
Figure 3. Viewing geometry and projection model of [20]

3. Projection of Texture

Here we describe the viewing geometry and a projection model, to provide a relationship between the surface texture and the image plane frequencies as a function of the shape.

Figure 3 illustrates the viewing geometry of [20]. $\mathbf{x}_w = (x_w, y_w, z_w)$ is the world coordinate system and $\mathbf{x}_i = (x_i, y_i)$ is the image plane coordinate system, placed at $z_w = f < 0$, with $|f|$ being the focal length. The orientation of the surface is described using the slant-tilt system: the slant σ is the angle between the surface normal and the optical axis; the tilt τ is the angle between the x_i-axis and the projection on the image plane of the surface normal. The surface is described by the coordinate system $\mathbf{x}_s = (x_s, y_s, z_s)$: the x_s-axis is aligned with the perceived tilt direction, the z_s-axis is aligned with the surface normal.

The above coordinate systems can be easily extended to general surfaces: assuming that the surface is smooth and that at any point it can be locally approximated with the



(a) (b) (c) (d)

Figure 2. (**a**) Brodatz texture D6 rendered on a plane; (**b**) its spectrum amplitude; (**c**) real LSFs (yellow areas) represented on the spectrum; (**d**) chosen Gabor functions (yellow circles) superimposed on the spectrum

corresponding tangent plane, the equations from [20] then apply to the tangent plane at the considered point.

Given the homogeneity assumption in section 1, surface texture frequencies $\mathbf{u}_s$ corresponding to the same LSF must be the same all over the surface texture. However the $\mathbf{x}_s$ coordinate system changes with the tilt-direction, as the x_s axis is aligned with it at each point. For aligning the $\mathbf{x}_s$ coordinate system we counter-rotate it by the angle $-\tau$ on the tangent plane, so as to have $\mathbf{x}_s$ oriented along the same direction in a virtual unfolded surface texture.

This system of coordinates was first introduced by Dugué and Elghadi [3], who illustrate the relation between $\mathbf{x}_i$ and $\mathbf{x}_s$ in the orthographic case. Here we generalize the orthographic relationships to the general perspective case.

The transformation between the aligned $\mathbf{x}_s$ and the $\mathbf{x}_w$ coordinate system is

$$\mathbf{x}_w = \begin{bmatrix} \cos\sigma\cos\tau & -\sin\tau & \sin\sigma\cos\tau \\ \cos\sigma\sin\tau & \cos\tau & \sin\sigma\sin\tau \\ -\sin\sigma & 0 & \cos\sigma \end{bmatrix} \begin{bmatrix} \cos\tau & \sin\tau & 0 \\ -\sin\tau & \cos\tau & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{x}_s + \begin{bmatrix} 0 \\ 0 \\ z_0 \end{bmatrix} , \tag{3}$$

where z_0 is the z_w coordinate of the point at the intersection of the tangent plane at the considered point and the optical axis. We combine equation 3 with the perspective projection equation between $\mathbf{x}_w$ and $\mathbf{x}_i$

$$\mathbf{x}_i = \frac{f}{z_w} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \mathbf{x}_w , \tag{4}$$

to obtain the relation between the image plane $\mathbf{x}_i$ and the surface texture $\mathbf{x}_s$ coordinates (note that, from here on, we will take $\mathbf{x}_s$ to be (x_s, y_s), as on the surface texture $z_s = 0$):

$$\mathbf{x}_i = \frac{f}{z_w} R(\tau)P(\sigma)R(-\tau)\mathbf{x}_s , \tag{5}$$

$$\text{where} \quad z_w = z_0 - \sin\sigma(x_s\cos\tau + y_s\sin\tau) , \tag{6}$$

$$R(\tau) = \begin{bmatrix} \cos\tau & -\sin\tau \\ \sin\tau & \cos\tau \end{bmatrix} , \quad P(\sigma) = \begin{bmatrix} \cos\sigma & 0 \\ 0 & 1 \end{bmatrix} . \tag{7}$$

$R(\tau)$ and $P(\sigma)$ are respectively responsible for rotating the system of coordinates by τ and foreshortening the distances parallel to the tilt direction by a coefficient $\cos\sigma$, z_w is the z coordinate (in the $\mathbf{x}_w$ coordinate system) of the surface texture point which projects to the image point $\mathbf{x}_i$.

As in [20], we assume that the image intensity $I(\mathbf{x}_i)$ is proportional to the surface reflectance $I_s(\mathbf{x}_s(\mathbf{x}_i))$, and that shading is negligible because it is assumed to vary slowly compared to I_s. Also, as in [20], we compute the relationship between the frequencies on the image plane $\mathbf{u}_i = (u_i, v_i)$ and on the surface texture $\mathbf{u}_s = (u_s, v_s)$ by using the transpose of the Jacobian of the vector function $\mathbf{x}_i(\mathbf{x}_s)$ in equation 5:

$$\mathbf{u}_s = J^t(\mathbf{x}_i, \mathbf{x}_s)\mathbf{u}_i \tag{8}$$

$$\text{with} \quad J^t(\mathbf{x}_i, \mathbf{x}_s) = \frac{\sin\sigma}{z_w} \begin{bmatrix} x_i\cos\tau & y_i\cos\tau \\ x_i\sin\tau & y_i\sin\tau \end{bmatrix} + \frac{f}{z_w} R(\tau)P(\sigma)R(-\tau) . \tag{9}$$

In such a coordinate system, the homogeneity assumption of section 1 implies that the surface texture frequencies $\mathbf{u}_s$ belonging to the same LSF are the same at all points of the surface. Therefore, for all frequencies $\mathbf{u}_i$, belonging to the same LSF, measured at all image points $\mathbf{x}_i$, the back-projected $\mathbf{u}_s$ are the same and their variance is zero.

It is easy to show that equation 8 coincides with the relationship between image plane and surface texture frequencies found by Massot and Hérault in [18] (equation 20), which confirms the result under a first order approximation.

4. Computing Surface Orientation

We explain here how our algorithm processes the LSFs to produce the shape of the surface texture.

Let us consider equation 8 for two different LSFs $\mathbf{u}'_i = (u'_i, v'_i)$ and $\mathbf{u}''_i = (u''_i, v''_i)$ measured at the same image point $\mathbf{x}_i$:

$$\mathbf{u}'_s = J^t(\mathbf{x}_i, \mathbf{x}_s)\mathbf{u}'_i \quad \text{and} \quad \mathbf{u}''_s = J^t(\mathbf{x}_i, \mathbf{x}_s)\mathbf{u}''_i . \tag{10}$$

Equations 10 can be combined as

$$U_s = J^t(\mathbf{x}_i, \mathbf{x}_s) U_i \tag{11}$$

$$\text{where } U_s = \begin{bmatrix} u'_s & u''_s \\ v'_s & v''_s \end{bmatrix} \text{ and } U_i = \begin{bmatrix} u'_i & u''_i \\ v'_i & v''_i \end{bmatrix}. \tag{12}$$

Here U_s contains the surface texture frequencies corresponding to the two LSFs, constant over the surface according to the homogeneity assumption, while U_i contains the image frequencies for the two LSFs, which are functions of the particular point.

By substituting in equation 11 for the value of the Jacobian as expressed in equation 9 we have

$$U_s = \frac{\sin\sigma(x_i\cos\tau + y_i\sin\tau) + f\cos\sigma}{z_0 f\cos\sigma} \left\{ \sin\sigma \begin{bmatrix} \cos\tau \\ \sin\tau \end{bmatrix} \begin{bmatrix} x_i & y_i \end{bmatrix} + fR(\tau)P(\sigma)R(-\tau) \right\} U_i \ , \tag{13}$$

where we have chosen to express U_s in terms of z_0 instead of z_w, according to equation 6, because the former is constant when we consider the first order approximation at a given point, i.e. z_0 is constant once the particular tangent plane is considered (see section 3 for details).

Let us now differentiate the left and right hand sides of equation 13 in a general direction, which we indicate as the direction of the arbitrary vector $\mathbf{s}$. As for z_0, under a first order approximation τ, σ and f are also constant in the differentiation, as they do not vary on the tangent plane. In what follows we indicate with $\partial_{\mathbf{s}} U_i$ the following:

$$\partial_{\mathbf{s}} U_i = \begin{bmatrix} \partial_{\mathbf{s}} u'_i & \partial_{\mathbf{s}} u''_i \\ \partial_{\mathbf{s}} v'_i & \partial_{\mathbf{s}} v''_i \end{bmatrix} \equiv \begin{bmatrix} \mathbf{s}\cdot\nabla u'_i & \mathbf{s}\cdot\nabla u''_i \\ \mathbf{s}\cdot\nabla v'_i & \mathbf{s}\cdot\nabla v''_i \end{bmatrix}. \tag{14}$$

Furthermore, we note that $\partial_{\mathbf{s}} U_s = 0$, as the surface texture frequencies in U_s are constant over the surface according to the homogeneity assumption and to the alignment (see section 3). Below is the result of the differentiation, i.e. the expression for local frequency variation

$$\partial_{\mathbf{s}} U_i = -\frac{\sin\sigma}{\sin\sigma(x_i\cos\tau + y_i\sin\tau) + f\cos\sigma} \left\{ (\partial_{\mathbf{s}} x_i \cos\tau + \partial_{\mathbf{s}} y_i \sin\tau) I + \begin{bmatrix} \cos\tau \\ \sin\tau \end{bmatrix} \begin{bmatrix} \partial_{\mathbf{s}} x_i & \partial_{\mathbf{s}} y_i \end{bmatrix} \right\} U_i \ . \tag{15}$$

The above equation 15 coincides with the relation (21) in [18] between the frequency measured on the image plane and its derivative, where the a_i in [18] is expressed according to our coordinate system and f_i, a single frequency in [18], is replaced by U_i, containing two LSFs.

Equation 15 relates the orientation parameters (σ, τ) at a point $\mathbf{x}_i$ to the local frequency variation, expressed by the matrix $\partial_{\mathbf{s}} U_i$. We can write this as

$$-\left(\partial_{\mathbf{s}} U_i\right) U_i^{-1} = \frac{\sin\sigma}{\sin\sigma(x_i\cos\tau + y_i\sin\tau) + f\cos\sigma} \begin{bmatrix} 2\partial_{\mathbf{s}} x_i\cos\tau + \partial_{\mathbf{s}} y_i \sin\tau & \partial_{\mathbf{s}} y_i \cos\tau \\ \partial_{\mathbf{s}} x_i \sin\tau & \partial_{\mathbf{s}} x_i \cos\tau + 2\partial_{\mathbf{s}} y_i \sin\tau \end{bmatrix}. \tag{16}$$

The sum of the eigenvalues of the right hand side of equation 16 is

$$\lambda_{r1} + \lambda_{r2} = 3\frac{\sin\sigma(\partial_{\mathbf{s}} x_i\cos\tau + \partial_{\mathbf{s}} y_i\sin\tau)}{\sin\sigma(x_i\cos\tau + y_i\sin\tau) + f\cos\sigma} \ , \tag{17}$$

while the sum of eigenvalues for the left hand side is

$$\lambda_{l1} + \lambda_{l2} = \frac{\partial_{\mathbf{s}} u'_i v''_i - \partial_{\mathbf{s}} u''_i v'_i - \partial_{\mathbf{s}} v'_i u''_i + \partial_{\mathbf{s}} v''_i u'_i}{u''_i v'_i - u'_i v''_i} = \tag{18}$$

$$-\partial_{\mathbf{s}}\left[\log\left|(u'_i v''_i - u''_i v'_i)\right|\right] = -\partial_{\mathbf{s}}\left[\log(r)\right] \ , \tag{19}$$

having set

$$r = |u'_i v''_i - u''_i v'_i| = \ \big|\ |\mathbf{u}'_i|\,|\mathbf{u}''_i|\sin(\angle\mathbf{u}''_i - \angle\mathbf{u}'_i)\ \big| \ . \tag{20}$$

Equating the sums of eigenvalues in equations 17 and 19 we get

$$-\partial_{\mathbf{s}}\left[\log(r)\right] = 3\frac{\sin\sigma(\partial_{\mathbf{s}} x_i\cos\tau + \partial_{\mathbf{s}} y_i\sin\tau)}{\sin\sigma(x_i\cos\tau + y_i\sin\tau) + f\cos\sigma} \ . \tag{21}$$

From which finally, by studying the differentiation direction $\angle\mathbf{s}$ which makes the left and right hand sides of equation 21 minima, and by equating these minima, we estimate the orientation parameters (σ, τ) at each point of the image plane:

$$\tau \ = \ \angle(\nabla r) \tag{22}$$

$$\sigma \ = \ \arctan\frac{-|\nabla(\log r)|f}{3 + |\nabla(\log r)|(x_i\cos\tau + y_i\sin\tau)} \ . \tag{23}$$

Alternatively, equation 21 can be solved by noticing that

$$3\frac{\sin\sigma(\partial_{\mathbf{s}} x_i\cos\tau + \partial_{\mathbf{s}} y_i\sin\tau)}{\sin\sigma(x_i\cos\tau + y_i\sin\tau) + f\cos\sigma} = 3\partial_{\mathbf{s}}\left[\log(|\sin\sigma(x_i\cos\tau + y_i\sin\tau) + f\cos\sigma|)\right]. \tag{24}$$

Equation 21 therefore becomes

$$-\partial_{\mathbf{s}}\left[\log(r)\right] = 3\partial_{\mathbf{s}}\left[\log|\sin\sigma(x_i\cos\tau + y_i\sin\tau) + f\cos\sigma|\right]. \tag{25}$$

By integrating equation 25 along the direction of the differentiation vector $\mathbf{s}$, studying its minima with respect to the direction of $\mathbf{s}$ and equating the minima, as done for equation 21, we get an alternative solution for (σ, τ) at each pixel, which in fact coincides with equations 22 and 23.

The same solution is also found by computing the determinants of both sides of equation 13 and then either the ratio

$\frac{\det U_i(\mathbf{x}_i)}{\det U_i(\mathbf{x}_i+\mathbf{s})}$ or the difference $\det U_i(\mathbf{x}_i+\mathbf{s})-\det U_i(\mathbf{x}_i)$, as illustrated in the technical report [8] which gives additional detail. The agreement of results confirms the validity of the calculations.

Finally, once the orientation parameters (σ, τ) are computed at each pixel with equations (22, 23), the method of [4] is used to enforce integrability and recover the actual shape of the texture, i.e. z_w.

It is of interest to note that ∇r is proportional to the density gradient $\nabla\rho$ ([9]) expressed as a function of the two LSFs $\mathbf{u}'_i = (u'_i, v'_i)$ and $\mathbf{u}''_i = (u''_i, v''_i)$:

$$\nabla\rho \propto \nabla|u'_i v''_i - u''_i v'_i| = \nabla r \ . \tag{26}$$

The density gradient was used in early SFT works ([10, 17]) in the spatial domain and is here reformulated for the spectral domain, to give more robust and applicable solutions.

Using two LSFs does not reduce the applicability of our method, because most real textures have at least two LSFs. On the other hand, our frequency estimation technique can recover all relevant LSFs, and a shape is reconstructed for each pair. When more than two LSFs are detected, the best final reconstruction is selected according to the homogeneity assumption: the measured LSFs are back-projected onto the virtually unfolded surface texture by using equation 8, where the pair with lowest variance, closest to the ideal zero value (see the homogeneity assumption in section 3), is chosen. In this sense we use the multi-scale nature of textures.

Most importantly, the method proposed here does not require any assumption about frontal points ([3], [7]) nor planar textures ([11], [20]). Our geometric equations use local frequency variations, and are therefore applicable to any textured surface of any size.

Finally, the algorithm lends itself to parallel implementation, because filters and LSFs can be processed independently and implemented separately.

5. Results

Here we demonstrate our method on the real images of a pillowcase laid on a cylinder and a rubber rug on a plane, and on a real video sequence of a textured cloth laid on a sphere-like shape, acquired with a Pulnix TM-6EX camera.

The chosen textures are simple but suffice to demonstrate the power of the 3D reconstruction algorithm. In fact this locally applies to any pair of LSF estimates and our multi-scale distortion estimation has already been shown in [6] to work on a range of deterministic and statistical textures.

We compare our reconstruction algorithm with the present state of the art, the reconstruction algorithm of [18]. Both the algorithms are here fed with the same distortion measures, i.e. the estimated LSFs.

Figures 4 and 6 show the real pictures of the pillowcase and rubber rug, their spectrum amplitude and the spectral directions at the central pixel of the two main LSFs.

Image	**Method**	$\overline{\lvert\epsilon_\tau\rvert}$	$\overline{\lvert\epsilon_\sigma\rvert}$
Pillowcase	Our method on LSFs 1,2	2.18	6.60
	Method of [18] on LSF 1	12.52	1.82
	Method of [18] on LSF 2	4.03	13.03
Rubber Rug	Our method on LSFs 1,2	2.26	2.55
	Method of [18] on LSF 1	16.52	4.73
	Method of [18] on LSF 2	3.84	7.13

Table 1. Average of absolute estimation errors for τ and σ, $\overline{|\epsilon_\tau|}$ and $\overline{|\epsilon_\sigma|}$, for the pillowcase and rubber rug sample (degrees)

Figures 5 and 7 show the ground truths and the reconstructions of the central parts of the image, obtained by applying our reconstruction method to the two estimated LSFs, and the method of [18] to the same LSFs separately. The figures and the numerical results in table 1 show more accurate results for our algorithm (note that the errors on σ and τ should be considered together).

Figure 8 shows the first frame of the real video sequence, its spectrum amplitude and the spectral directions at the central pixel of the two main LSFs.

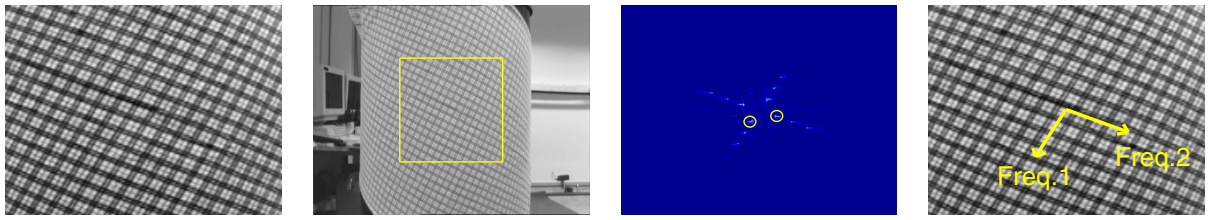
Figure 9 shows four sample frames from the video sequence, reconstructed using our method with the two LSFs, and the method of [18] with the same LSFs separately.

The reconstruction is challenging because: (i) the image quality is non-optimal and granular with reflections and changes in illumination; (ii) laying a cloth on a non-developable shape, i.e. a nearly spherical balloon, implies stretches and creases; (iii) on a sphere, the tilt is directed from the center to the edges, therefore assuming on the image pixels all values in the $(-180°,180°)$ range.

Despite these difficulties, our method achieves valid reconstructions and detects small changes in the shape from frame to frame.

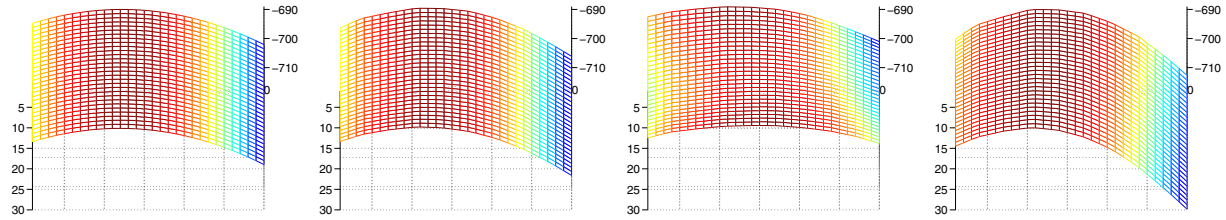
By contrast, the reconstructions with the method of [18] look heavily biased by the spectral direction of the LSF used: the shape seems to be correct at pixels where the spectral direction of the considered LSF is parallel to the tilt direction, but wrong on pixels where the two are orthogonal, with increasing errors inbetween the two cases.

Figure 10 illustrates our explanation for the failure of the reconstruction method of [18]: when a tilt direction is orthogonal to the spectral direction of an LSF, the foreshortening effect is null and one would struggle to recognize the underlying shape by only considering the distortion due to perspective convergence to vanishing points; a second LSF, parallel or at least not orthogonal to the tilt direction of the shape, creates a foreshortening effect and makes the structure pop up from the image. This explanation agrees with [13], according to which the perspective effects are generally negligible for most of the pictures of objects that we might want to consider for shape reconstruction.



(a) (b) (c) (d)

Figure 4. (**a**) 240x240 real picture of a pillow case laid on a cylinder; (**b**) 640x480 image from which (a) is extracted (indicated by the yellow square); (**c**) spectrum amplitude of (a) (the two yellow circles represent the spectral areas corresponding to the two main LSFs); (**d**) spectral directions of the two LSFs at the central pixel, superposed on (a)



(a) (b) (c) (d)

Figure 5. (**a**) Ground truth shape of the cylinder beneath the texture image in fig. 4(a), central 120x120 part. Perspective reconstructions (same central part) of shape using: (**b**) our method on the two LSFs; (**c**) the method of [18] on LSF 1; (**d**) the method of [18] on LSF 2

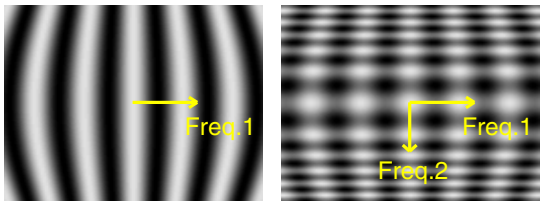


Figure 10. (**left**) Cosine shape textured with an LSF (its spectral direction represented by the yellow arrow) orthogonal to the tilt direction; (**right**) cosine shape textured with the same LSF orthogonal to the tilt direction and a second LSF directed parallel to the tilt direction (the two LSFs are represented with yellow arrows)

Our algorithm does not fail in any direction because it uses the two LSFs with distinct spectral directions, and at each pixel the contribution that each LSF gives to the estimate is weighted by the alignment of the LSF with the tilt direction, i.e. the more aligned LSF contributes more to the estimates of σ and τ. Moreover, when the tilt direction is in-between the two considered LSFs, our algorithm can count on an effective redundancy and cope with higher levels of noise. Therefore, despite noise, creases and stretches, our proposed algorithm robustly reconstructs every frame of the real video sequence.

6. Conclusions and Future Work

Under the very general assumptions of stochastic homogeneity and a perspective camera model, a SFT algorithm is presented for both deterministic and stochastic textures.

The multi-scale distortion of textures is characterized by LSFs and estimated using the Fourier transform of the image and a minimal set of Gabor filters.

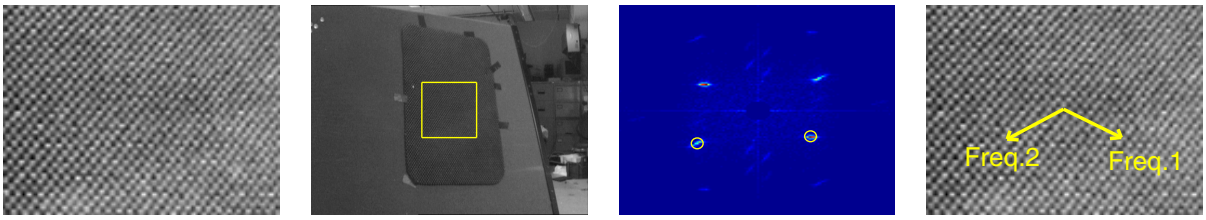
Then a novel method is presented to recover the 3D coordinates of the shape from local variations of the LSFs. The technique applies to all shapes, planar, non-planar and non-developable and, given its local nature, can be extended to complex and large surfaces.

The algorithm is accurate, robust against variations in illuminations, creases and stretches, and compares favorably with the state of the art of SFT.

The generality of assumptions and quality of results opens the way to applications such as rendering and retexturing of clothing, which we address in future work. In addition, the sensitivity shown in the real video sequence case, suggests the use of our algorithm as a non-invasive way of studying shape changes of objects with laid-on textures.

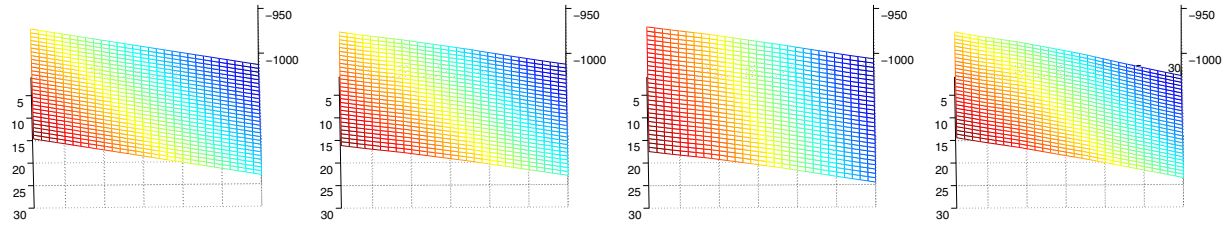
Acknowledgements

Fabio Galasso is grateful to St. John's college and Giovanni Aldobrandini for the 'A Luisa Aldobrandini' Scholarship, supporting his graduate studies. He would also wish



(a) (b) (c) (d)

Figure 6. (**a**) 128x128 real picture of a rubber rug laid on a planar surface; (**b**) 640x480 image from which (a) is extracted (indicated by the yellow square); (**c**) spectrum amplitude of (a) (the two yellow circles represent the spectral areas corresponding to the two main LSFs); (**d**) spectral directions of the two LSFs at the central pixel, superposed on (a)



(a) (b) (c) (d)

Figure 7. (**a**) Ground truth shape of the plane beneath the texture image in fig. 6, central 64x64 part. Perspective reconstructions (same central part) of shape using: (**b**) our method on the two LSFs; (**c**) the method of [18] on LSF 1; (**d**) the method of [18] on LSF 2

to credit the work of Corentin Massot, from which much of the stimulus for the investigations detailed here originated.

References

[1] M. Clerc and S. Mallat. The texture gradient equation for recovering shape from texture. *IEEE Trans. Patt. Anal. Mach. Intell.*, 24(4):536–549, 2002.

[2] A. Criminisi and A. Zisserman. Shape from texture: Homogeneity revisited. In *BMVC*, pages 82–91, 2000.

[3] A. G. Dugué and M. Elghadi. Shape from texture by local frequencies estimation. In *SCIA*, pages 533–544, Kangerlussuaq, Greenland, 1999.

[4] R. T. Frankot and R. Chellappa. A method for enforcing integrability in shape from shading algorithms. *IEEE Trans. Patt. Anal. Mach. Intell.*, 10(4):439–451, 1988.

[5] D. Gabor. Theory of communications. *J. Inst. Elec. Engrs*, 93(26):429–457, 1946.

[6] F. Galasso and J. Lasenby. Shape from texture: fast estimation of planar surface orientation via fourier analysis. In N. Rajpoot and A. Bhalerao, editors, *BMVC*, pages 660–669, Cambridge, UK, 2007.

[7] F. Galasso and J. Lasenby. Shape from texture of developable surfaces via fourier analysis. In *ISVC*, pages 702–713, Lake Tahoe, CA/NV, 2007.

[8] F. Galasso and J. Lasenby. Fourier analysis and gabor filtering for texture analysis and local reconstruction of general shapes. Technical report, CUED/F-INFENG/TR-610, September 2008.

[9] J. Gårding. Shape from texture for smooth curved surfaces in perspective projection. *JMIV*, 2:327–350, 1992.

[10] J. J. Gibson. *The Perception of the Visual World*. Houghton Mifflin, Boston, Massachusetts, 1950.

[11] W.-L. Hwang, C.-S. Lu, and P.-C. Chung. Shape from texture: Estimation of planar surface orientation through the ridge surfaces of continuous wavelet transform. *IEEE Trans. Image Processing*, 7(5):773–780, May 1998.

[12] K. Kanatani and T.-C. Chou. Shape from texture: general principle. *Artificial Intelligence*, 38(1):1–48, 1989.

[13] A. Lobay and D. A. Forsyth. Shape from texture without boundaries. *IJCV*, 67(1):71–91, 2006.

[14] A. M. Loh and R. Hartley. Shape from non-homogeneous, non-stationary, anisotropic, perspective texture. In *BMVC*, pages 69–78, 2005.

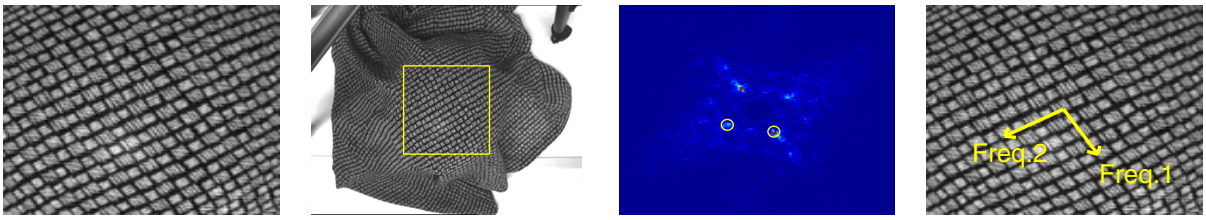
[15] A. M. Loh and P. Kovesi. Shape from texture without estimating transformations. Technical report, UWA-CSSE-05-001, July 2005.

[16] J. Malik and R. Rosenholtz. Computing local surface orientation and shape from texture for curved surfaces. *IJCV*, 23(2):149–168, 1997.

[17] C. Marinos and A. Blake. Shape from texture: The homogeneity hypothesis. In *ICCV*, pages 350–353, 1990.

[18] C. Massot and J. Hérault. Model of frequency analysis in the visual cortex and the shape from texture problem. *IJCV*, 76(2):165–182, 2008.

[19] E. Ribeiro and E. R. Hancock. Estimating the perspective pose of texture planes using spectral analysis on the unit sphere. *Pattern Recognition*, 35:2141–2163, 2002.



(a) (b) (c) (d)

Figure 8. (**a**) Frame 1 (128x128) extracted from a real video sequence of a textured cloth laid on a deflating balloon; (**b**) 400x300 whole frame 1 from which (a) is extracted (indicated by the yellow square); (**c**) spectrum amplitude of (a) (the two yellow circles represent the spectral areas corresponding to the two main LSFs); (**d**) spectral directions of the two LSFs at the central pixel, superposed on (a)

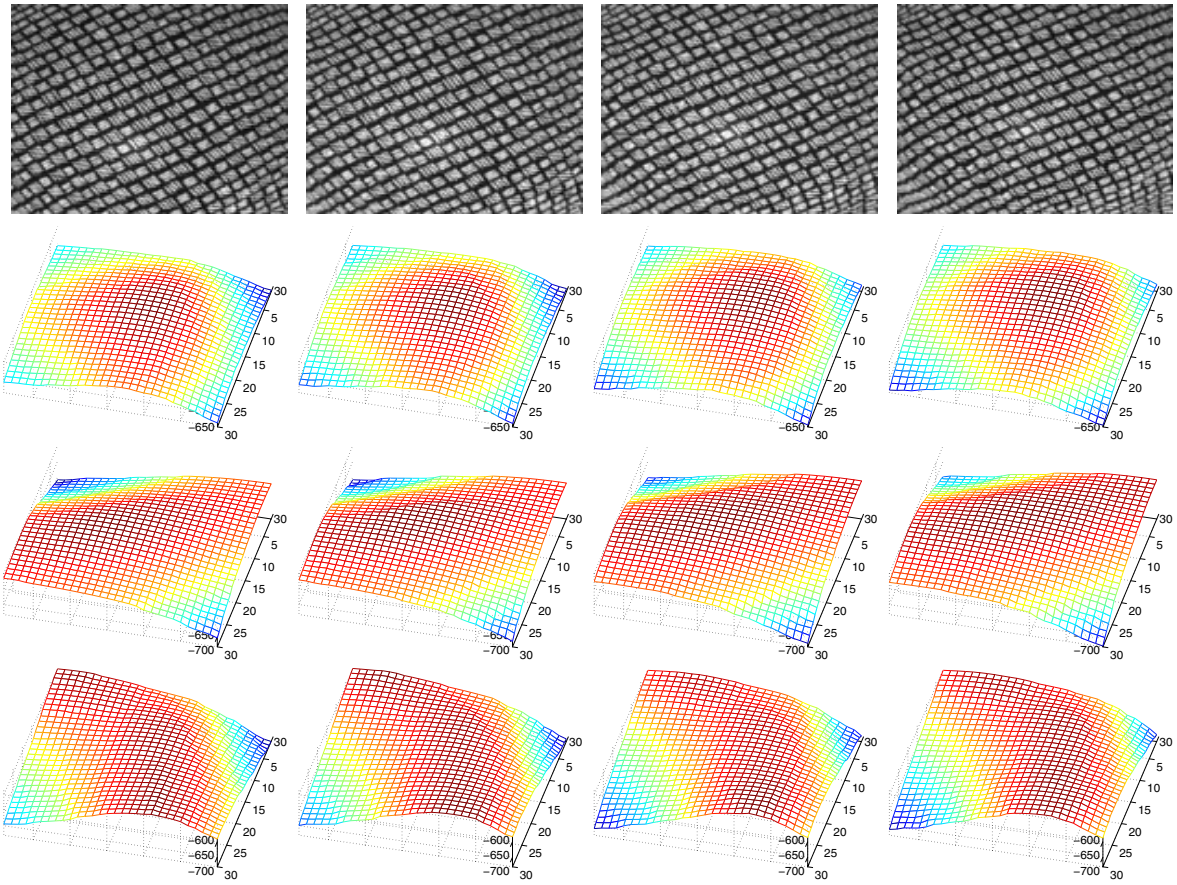


Figure 9. More frames extracted from the real video sequence of fig. 8. From the top: (**first row**) frames 1,100,200,300; (**second row**) perspective reconstructions with the proposed method using LSFs 1 and 2; (**third row**) perspective reconstructions with the method of [18] using LSF 1; (**fourth row**) perspective reconstructions with the method of [18] using LSF 2. The full video is available online

[20] B. J. Super and A. C. Bovik. Planar surface orientation from texture spatial frequencies. *Pattern Recognition*, 28(5):729–743, 1995.

[21] B. J. Super and A. C. Bovik. Shape from texture using local spectral moments. *IEEE Trans. Patt. Anal. Mach. Intell.*, 17(4):333–343, 1995.

[22] A. P. Witkin. Recovering surface shape and orientation from texture. *Artificial Intelligence*, 17:17–45, 1981.