Chapter 1

On Depth Recovery from Gradient Vector Fields

Tiangong Wei and Reinhard Klette
CITR, Department of Computer Science, The University of Auckland
Tamaki Campus, Auckland, New Zealand

Depth recovery from gradient vector fields is required when reconstructing a surface (in three-dimensional space) from its gradients. Such a reconstruction task results, for example, for techniques in computer vision aiming at calculating surface normals (such as shape from shading, photometric stereo, shape from texture, shape from contours and so on). Surprisingly, discrete integration has not been studied very intensively so far. This chapter presents three classes of methods for solving problems of depth recovery from gradient vector fields: a two-scan method, a Fourier-transform based method, and a wavelet-transform based method. These methods extend previously known techniques, and related proofs are given in a short but concise form.

The two-scan method consists of two different scans through a given gradient vector field. The final surface height values can be determined by averaging these two scans. Fourier-transform based methods are noniterative so that boundary conditions are not needed, and their robustness to noisy gradient estimates can be improved by choosing associated weighting parameters. The wavelet-transform based method overcomes the disadvantage of the Fourier-transform based method, which implicitly require that a surface height function is periodic. Experimental results using synthetic and real images are also presented.

1.1. Introduction

Discrete integration maps a dense but discrete gradient vector field into a surface representation, normally identified as "height" or "depth". The authors studied discrete integration in the context of computer vision. Here, this way of surface recovery may be part of techniques such as shape from shading (SFS), photometric stereo, shape from texture, or shape from contours. The SFS problem is to reconstruct the 3D shape of an object from a single 2D image of the object using shading and or lighting models for surface normal calculation. Algorithms for solving the SFS problem (see, for example, [9, 11, 14, 20, 23]) consist typically of two steps: the first step is to obtain the estimates of surface gradients or surface normals for a discrete set of visible points on the object surface (i.e., discrete gradient vector fields), and the second step is to recover the surface height from the estimated sur-

face orientation. This second step results in the problem of depth recovery from gradient vector fields.

The problem of depth recovery from gradients also arises when applying the photometric stereo method (PSM) [22]. PSM is to recover the 3D shape of an object from more than one image taken at the same attitude but for varying illumination. PSM allows an approximate solution for surface normals [10, 13, 22]. A subsequent integration step (i.e., depth recovery from gradients) is again required to convert estimated surface normals into an estimate of the surface shape.

Shape from texture is another area that leads to the depth recovery from gradients problem. Smith et al. [18, 19] proposed a technique for the recovery of a surface texture relief. The recovered texture relief has an useful potential for both visualization and numerical assessments of surface roughness, or for obtaining other parameters such as peak height or peak count. The technique utilizes three or more images to determine a dense gradient field at first. This gradient field is then integrated to obtain the surface texture relief. On the other hand, texture gradient is related to surface shape parameters (orientation, curvature). Shape recovery is made possible by measuring the texture gradient in the image.

The problem of depth recovery from gradients also results when inferring surface shape from a Gauss map or surface shape from the Hessian matrix [5], where the task is the estimation of an unknown surface height from a set of measurements of the gradients of some surface function. Therefore, it turns out that the entire shape reconstruction process can often be decomposed into two independent steps: gradient computation and gradient integration.

As mentioned above, several active fields of research related to computer vision produce gradient values for a discrete set of visible points on object surfaces. In order to achieve the relative height or depth values of the surface, these surface gradients have to be integrated by using gradient integration techniques.

Only a few numerical methods for the depth recovery from gradients problem have been developed. These methods have been classified traditionally into two categories: *local integration methods* [3, 7, 12, 16, 24] and *global integration methods* [6, 8, 21].

So far, the problem of depth recovery from gradients has not been studied often. Just for illustration, recent papers such as [2, 17] still apply algorithms for discrete integration as proposed about 20 years ago. In this chapter, we present theory, algorithms, and experiments for three classes of methods (different o those two categories mentioned before) for depth recovery from gradients.

The structure of the chapter is as follows. Section 1.2 analyzes the integrability of vector fields, and gives a brief review of related numerical methods for depth recovery from gradients. Section 1.3 discusses a two-scan method. Section 1.4 deals with the Fourier-transform based method for depth recovery from gradients. Section 1.5 presents a wavelet-transform based method. Section 1.6 presents experimental results using synthetic and real images. Section 1.7 concludes the chapter.

1.2. Depth Recovery from Gradient Vector Fields

It is a nontrivial problem of computing a surface height function $Z(x, y)$ from an estimated surface gradient field $(p(x, y), q(x, y))$. First, given a vector field (p, q), it may not correspond to a gradient field of any surface height function $Z(x, y)$ at all. Second, different surface height functions will have the same gradients (for example, looking orthogonally onto a stair case may produce the same gradient field as looking onto a plane). Therefore, the problem of depth recovery from gradients is ill-posed. It is only reasonable to determine the surface height function up to an additive constant, (i.e., the *relative surface height*). Notice that the relative surface height map is sometimes sufficient to recognize or inspect an object (e.g., for surface planarity tests in industrial surface inspection); and the relative surface height map may be transformed into absolute values if height values are available for proper scaling.

1.2.1. *Integrability of Vector Fields*

Generally, a given gradient field $p(x, y), q(x, y)$ may not correspond to any surface height function at all. In order of $p(x, y), q(x, y)$ to be the gradients of a surface height function, the given gradient field must be integrable. A vector field $(p(x, y), q(x, y))$ over a simply connected domain Ω is *integrable* if there exists some surface height function $Z(x, y) \in C^1(\Omega)$ such that it satisfies the *weak integrability condition*

$$Z_x(x, y) = p(x, y) \tag{1.1}$$

$$Z_y(x, y) = q(x, y) \tag{1.2}$$

for all $(x, y) \in \Omega$, where the subscripts denote partial derivatives. In other words, a gradient vector field is integrable if it is the gradient field of some surface height function.

Given a vector field $(p(x, y), q(x, y))$ over a simply connected domain Ω. Integrability can be characterized for two slightly different cases:

(i) Assume that components $p(x, y)$ and $q(x, y)$ are continuously differentiable; then the vector field $(p(x, y), q(x, y))$ is integrable if and only if

$$p_y(x, y) = q_x(x, y) \tag{1.3}$$

for all points in Ω; in other words, the surface height function $Z(x, y) \in C^2(\Omega)$ satisfies the *strong integrability condition*

$$Z_{xy}(x, y) = Z_{yx}(x, y) \tag{1.4}$$

(ii) Only assume that the components $p(x, y)$ and $q(x, y)$ are continuous; then the vector field is integrable if and only if

$$\oint_\gamma p(x, y)dx + q(x, y)dy = 0$$

for any closed curve γ in Ω; in other words, the surface height function $Z(x, y) \in C^1(\Omega)$ satisfies the *partial integrability condition*

$$\oint_\gamma Z_x(x, y)dx + Z_y(x, y)dy = 0$$

If the components of a vector field are continuously differentiable, then it is easy to determine whether or not it is integrable or nonintegrable by using the strong integrability condition.

1.2.2. *Local and Global Integration Methods*

In local integration methods [3, 7, 12, 24], an arbitrary initial height value Z_0 is preset for a starting point (x_0, y_0) somewhere in the image of the surface. Then, the relative heights at every point (x, y), which are consistent with this arbitrarily given height value Z_0, will be calculated according to a local approximation rule. Therefore, local integration approaches strongly depend on data accuracy (to avoid error propagation).

To compute surface height $Z(x, y)$ from the estimated surface gradient fields $(p(x, y), q(x, y))$, global integration techniques [6, 9] are based on minimizing the quadratic error functional (cost functional) between ideal and given gradient values:

$$W = \iint_\Omega [|Z_x - p|^2 + |Z_y - q|^2]dxdy \tag{1.5}$$

The above functional is invariant when a constant value is added to the surface height $Z(x, y)$. This expresses the fact that depth recovery from gradients can only reconstruct a surface height up to a constant.

Generally speaking, there are three possible methods to solve this optimization problem: variational approaches, direct discretization methods, and expansion methods. The variational approach [9] results in an Euler-Lagrange equation as the necessary condition for a minimum. Then there is a need to solve this Poisson equation. In order to solve the minimization problem (1.5) numerically, the continuous functional W is converted into a discrete problem directly by a discretization method. The disadvantage of variational approaches or discretization methods is the requirement of boundary conditions. But boundary information is not easy to obtain when dealing with depth recovery from gradients.

The surface height is expressed as a linear combination of a set of basis functions in expansion methods such as proposed and studied in [6] (and used for formulating a *Frankot-Chellappa algorithm* in [13]). Nevertheless, the errors of this algorithm are high for imperfect estimates of surface gradients, or noisy gradient vector fields [12]. Also, the algorithm is very sensitive to abrupt changes in orientation. In this chapter, we will focus on expanding the surface height function using the Fourier basis functions and third-order Daubechies' scaling basis functions.

1.3. Two-Scan Method

This section presents a local method for depth recovery from gradients. Suppose that the surface normals at four grid points $\{(i,j),(i+1,j),(i,j+1),(i+1,j+1)\}$ are represented by the following surface gradients:

$$\begin{aligned}\mathbf{n}_{i,j} &= (p_{i,j}, q_{i,j}, -1)^T \\ \mathbf{n}_{i+1,j} &= (p_{i+1,j}, q_{i+1,j}, -1)^T \\ \mathbf{n}_{i,j+1} &= (p_{i,j+1}, q_{i,j+1}, -1)^T \\ \mathbf{n}_{i+1,j+1} &= (p_{i+1,j+1}, q_{i+1,j+1}, -1)^T\end{aligned}$$

Consider grid points $(i, j+1)$ and $(i+1, j+1)$; since the line connecting points $(i, j+1, Z_{i,j+1})$ and $(i+1, j+1, Z_{i+1,j+1})$ is approximately perpendicular to the average normal between these two points, the dot product of the slope of this line and the average normal is equal to zero. This gives

$$Z_{i+1,j+1} = Z_{i,j+1} + \frac{1}{2}\left(p_{i,j+1} + p_{i+1,j+1}\right)$$

Similarly, we obtain the following regressive relation for grid points $(i+1, j)$ and $(i+1, j+1)$:

$$Z_{i+1,j+1} = Z_{i+1,j} + \frac{1}{2}\left(q_{i+1,j} + q_{i+1,j+1}\right)$$

Adding above two recursions together, and dividing the result by 2 gives

$$\begin{aligned}Z_{i+1,j+1} = &\frac{1}{2}\left(Z_{i,j+1} + Z_{i+1,j}\right) \\ &+ \frac{1}{4}\left(p_{i,j+1} + p_{i+1,j+1} + q_{i+1,j} + q_{i+1,j+1}\right)\end{aligned} \tag{1.6}$$

Suppose further that the total number of points on the object surface be $N \times N$. If two arbitrary initial height values are preset at grid points $(1,1)$ and (N,N), then the two-scan algorithm consists of two stages; the first stage starts at the left-most, bottom-most corner of the given gradient field, and determines the height values along x-axis and y-axis by discretizing (1.1) in terms of the forward differences

$$Z_{i,1} = Z_{i-1,1} + p_{i-1,1} \tag{1.7}$$

$$Z_{1,j} = Z_{1,j-1} + q_{1,j-1} \tag{1.8}$$

where $i = 2, ..., N, j = 2, ..., N$. Then scan the image vertically using (1.6). The second stage starts at the right-top corner of the given gradient field and sets the height values by

$$Z_{i-1,N} = Z_{i,N} - p_{i,N} \tag{1.9}$$

$$Z_{N,j-1} = Z_{N,j} - q_{N,j} \tag{1.10}$$

Then scan the image horizontally using the following recursive equation

$$Z_{i-1,j-1} = \frac{1}{2}\left(Z_{i-1,j} + Z_{i,j-1}\right) - \frac{1}{4}\left(p_{i-1,j} + p_{i,j} + q_{i,j-1} + q_{i,j}\right)$$

Since the estimated height values may be affected by the choice of the initial height value, we take an average of the two scan values for the surface height.

1.4. Fourier-Transform Based Methods

[6] suggested a solution for the SFS problem to enforce the weak integrability condition (1.1) by using the theory of projections onto convex sets. Their method is to project the given (possibly, non-integrable gradient field) onto the nearest integrable gradient field in the least-square sense. In order to improve the accuracy and robustness, and to strengthen the relation between the surface height function and the given gradient field, we introduce two new constraints as follows:

$$Z_{xx}(x,y) = p_x(x,y)$$
$$Z_{yy}(x,y) = q_y(x,y)$$

The two new constraints model the behavior of a change rate in second-order derivatives between the variables. Therefore, the changes of surface height will be more regular. Having the new constraints, we consider the following energy functional

$$\begin{aligned} W = & \iint_\Omega \left[|Z_x - p|^2 + |Z_y - q|^2\right] dxdy \\ & + \lambda \iint_\Omega \left[|Z_{xx} - p_x|^2 + |Z_{yy} - q_y|^2\right] dxdy \\ & + \mu_1 \iint_\Omega \left(|Z_x|^2 + |Z_y|^2\right) dxdy \\ & + \mu_2 \iint_\Omega \left(|Z_{xx}|^2 + 2|Z_{xy}|^2 + |Z_{yy}|^2\right) dxdy \end{aligned} \tag{1.11}$$

where non-negative parameters λ, μ_1, and μ_2 establish a trade-off between those constraints (i.e., they are used to adjust the weighting between the constraints). This cost function reflects the relations among $Z(x,y)$, $p(x,y)$, and $q(x,y)$ more effectively, and makes the best use of the information provided by the given gradient field because it not only constraints the tangent line of the surface, but also constraints its concavity and convexity. The following objective is to solve for the unknown $Z(x,y)$ subject to an optimization process which minimizes the cost function W.

To solve this minimization problem (1.11), Fourier-transform techniques can be applied. The two-dimensional Fourier transform of the surface function $Z(x,y)$ is defined by

$$Z_F(u,v) = \iint_\Omega Z(x,y) e^{-j(ux+vy)} dxdy \tag{1.12}$$

and the inverse Fourier transform is defined by

$$Z(x,y) = \frac{1}{2\pi} \iint_\Omega Z_F(u,v) e^{j(ux+vy)} dudv \tag{1.13}$$

where $j = \sqrt{-1}$ is the imaginary unit, and u and v represent the two-dimensional frequencies in the Fourier domain. The following differentiation properties can be obtained easily:

$$\begin{aligned}
Z_x(x,y) &\leftrightarrow juZ_F(u,v)\\
Z_y(x,y) &\leftrightarrow jvZ_F(u,v)\\
Z_{xx}(x,y) &\leftrightarrow -u^2Z_F(u,v)\\
Z_{yy}(x,y) &\leftrightarrow -v^2Z_F(u,v)\\
Z_{xy}(x,y) &\leftrightarrow -uvZ_F(u,v)
\end{aligned}$$

where the sign $\leftrightarrow$ means that the Fourier transform of the function on the left-hand side is equal to the one on the right-hand side.

Let $P(u,v)$ and $Q(u,v)$ be the Fourier transforms of the given gradients $p(x,y)$ and $q(x,y)$, respectively. Taking the Fourier transform in the functional (1.11), and using the differentiation properties of the Fourier transform and the following Parseval's formula

$$\iint_\Omega |Z(x,y)|^2 dxdy = \frac{1}{2\pi}\iint_\Omega |Z_F(u,v)|^2 dudv \tag{1.14}$$

we obtain that

$$\begin{aligned}
&\frac{1}{2\pi}\iint_\Omega \left[|juZ_F - P|^2 + |jvZ_F - Q|^2\right] dudv\\
+\ &\frac{\lambda}{2\pi}\iint_\Omega \left[\left|-u^2Z_F - juP\right|^2 + \left|-v^2Z_F - jvQ\right|^2\right] dudv\\
+\ &\frac{\mu_1}{2\pi}\iint_\Omega \left[|juZ_F|^2 + |jvZ_F|^2\right] dudv\\
+\ &\frac{\mu_2}{2\pi}\iint_\Omega \left[\left|-u^2Z_F\right|^2 + 2\left|-uvZ_F\right|^2 + \left|-v^2Z_F\right|^2\right] dudv\\
&\rightarrow minimum
\end{aligned}$$

where $Z_F = Z_F(u,v)$, $P = P(u,v)$, and $Q = Q(u,v)$.The left-hand side of the above expression can be expanded into

$$\begin{aligned}
&\frac{1}{2\pi}\iint_\Omega \left[u^2Z_FZ_F^* - juZ_FP^* + juZ_F^*P + PP^*\right.\\
&\left.+v^2Z_FZ_F^* - jvZ_FQ^* + jvZ_F^*Q + QQ^*\right] dudv\\
&+\frac{\lambda}{2\pi}\iint_\Omega \left[u^4Z_FZ_F^* - ju^3Z_FP^* + ju^3Z_F^*P + u^2PP^*\right.\\
&\left.+v^4Z_FZ_F^* - jv^3Z_FQ^* + jv^3Z_F^*Q + v^2QQ^*\right] dudv\\
&+\frac{\mu_1}{2\pi}\iint_\Omega \left(u^2+v^2\right)Z_FZ_F^* dudv\\
&+\frac{\mu_2}{2\pi}\iint_\Omega \left(u^4+2u^2v^2+v^4\right)Z_FZ_F^* dudv
\end{aligned}$$

where the asterisk $*$ denotes the complex conjugate. Differentiating the above expression with respect to Z_F^* and setting the result to zero, we can deduce the necessary condition for a minimum of the cost function (1.11) as follows:

$$\begin{aligned}&\left(u^2 Z_F + juP + v^2 Z_F + jvQ\right) + \lambda\left(u^4 Z_F + ju^3 P + v^4 Z_F + jv^3 Q\right)\\ &+ \mu_1\left(u^2+v^2\right) Z_F + \mu_2\left(u^4 + 2u^2v^2 + v^4\right) Z_F = 0\end{aligned}$$

A rearrangement of this equation then yields

$$\begin{aligned}&\left[\lambda\left(u^4+v^4\right) + (1+\mu_1)\left(u^2+v^2\right) + \mu_2\left(u^2+v^2\right)^2\right] Z_F(u,v)\\ &+j\left(u+\lambda u^3\right) P(u,v) + j\left(v+\lambda v^3\right) Q(u,v) = 0\end{aligned}$$

Solving the above equation except for $(u,v) \neq (0,0)$, we obtain that

$$Z_F(u,v) = \frac{-j\left(u+\lambda u^3\right) P(u,v) - j\left(v+\lambda v^3\right) Q(u,v)}{\lambda\left(u^4+v^4\right) + (1+\mu_1)\left(u^2+v^2\right) + \mu_2\left(u^2+v^2\right)^2} \tag{1.15}$$

Therefore, a Fourier transform of the unknown surface height $Z(x,y)$ is expressed as a function of Fourier transforms of given gradients $p(x,y)$ and $q(x,y)$. This Fourier-transform based method can be summarized as follows:

Theorem 1.1. *Given a gradient field $(p(x,y), q(x,y))$; the corresponding surface height function $Z(x,y)$ can be computed by taking the inverse Fourier transform of $Z_F(u,v)$ in (1.15), where $Z_F(u,v)$, $P(u,v)$, and $Q(u,v)$, respectively, are Fourier transforms of $Z(x,y)$, $p(x,y)$, and $q(x,y)$.*

Our algorithm (see Algorithm 1) specifies the implementation details for this Fourier-transform based method. The constant pq_{max} eliminates gradient estimates which define angles with the image plane close to 90°, and a value such as $pq_{max} = 12$ is an option. Real parts are stored in arrays P1, Q1, and H1, and imaginary parts in arrays P2, Q2, and H2. The initialization in line 19 can be by an estimated value for the average height of the visible scene. Parameters λ, μ_1 and μ_2 should be chosen based on experimental evidence for the given scene.

1.5. Wavelet-Transform Based Method

Wavelet theory has proved to be a powerful tool, and has begun to play a serious role in a broad range of applications, including numerical analysis, pattern recognition, signal and image processing. The wavelet transform is a generalization of the Fourier transform. Wavelets have advantages over traditional Fourier methods in analyzing physical situations where the function contains discontinuities and sharp spikes. In order to take the advantages of the wavelet transform, we present a wavelets based method for depth recovery from gradients.

Algorithm 1 Fourier-Transform Based Method

input gradients $p(x,y), q(x,y)$; parameters λ, μ_1, and μ_2
for $0 \le x, y \le N-1$ **do**
  if $(|p(x,y)| < pq_{max}$ & $|q(x,y)| < pq_{max})$ **then**
    P1(x,y)=p(x,y); P2(x,y)=0;
    Q1(x,y)=q(x,y); Q2(x,y)=0;
  else
    P1(x,y)=0; P2(x,y)=0;
    Q1(x,y)=0; Q2(x,y)=0;
  end if
end for
Calculate Fourier transform in place: P1(u,v), P2(u,v);
Calculate Fourier transform in place: Q1(u,v), Q2(u,v);
for $0 \le u, v \le N-1$ **do**
  if $(u \neq 0$ & $v \neq 0)$ **then**
    $\Delta = \lambda\left(u^4+v^4\right) + (1+\mu_1)\left(u^2+v^2\right) + \mu_2\left(u^2+v^2\right)^2$;
    $H1(u,v) = [(u+\lambda u^3)P2(u,v) + (v+\lambda v^3)Q2(u,v)]/\Delta$;
    $H2(u,v) = [-(u+\lambda u^3)P1(u,v) - (v+\lambda v^3)Q1(u,v)]/\Delta$;
  else
    $H1(0,0)$ = average height; $H2(0,0) = 0$;
  end if
end for
Calculate inverse Fourier transform of H1(u,v) and H2(u,v) in place: H1(x,y), H2(x,y);
for $0 \le x, y \le N-1$ **do**
  $Z(x,y) = H1(x,y)$;
end for

1.5.1. *Daubechies Wavelet Basis*

Let $\phi(x)$ and $\psi(x)$ are the Daubechies *scaling function* and *wavelet*, respectively. They both are implicitly defined by the following, two-scale relation [4]:

$$\phi(x) = \sum_{k\in\mathbb{Z}} a_k \phi(2x-k) \tag{1.16}$$

and the equation

$$\psi(x) = \sum_{k\in\mathbb{Z}} (-1)^k a_{1-k} \phi(2x-k) \tag{1.17}$$

where $\mathbb{Z} = \{\cdots, -1, 0, 1, \cdots\}$, and a_k are called the Daubechies wavelet filter coefficients.

Connection coefficients (see, for example, Beylkin [1], Mallat [15]) play an important role in representing the relation between the scaling function and differential

operators. For $k \in \mathbb{Z}$, the connection coefficients with Mth order *vanishing moments* (that is, $\int_{-\infty}^{+\infty} x^k \phi(x)dx = 0$, for $0 \leq k \leq M$) are defined by

$$\Gamma_k^0 = \int \phi(x)\phi(x-k)dx \tag{1.18}$$

$$\Gamma_k^1 = \int \phi^{(x)}(x)\phi(x-k)dx \tag{1.19}$$

$$\Gamma_k^2 = \int \phi^{(x)}(x)\phi^{(x)}(x-k)dx \tag{1.20}$$

Then we have the following properties:

(i) $\Gamma_0^1 = 0$,
(ii) for the scaling function $\phi(x)$, which has Mth order vanishing moments, $\Gamma_k^1 = \Gamma_k^2 = 0$, $k \notin [-2M+2, 2M-2]$, and
(iii) $\Gamma_k^0 = \begin{cases} 1, & k = 0, \\ 0, & \text{otherwise.} \end{cases}$

The connection coefficients for Daubechies' wavelet with 3rd order vanishing moments are shown in Table 1.5.1:

Table 1.1. Connection coefficients for $N = 3$.

Γ_k^1	Γ_k^2
Γ_{-4}^1 =0.00034246575342	Γ_{-4}^2 = -0.00535714285714
Γ_{-3}^1 =0.01461187214612	Γ_{-3}^2 = -0.11428571428571
Γ_{-2}^1 =-0.14520547945206	Γ_{-2}^2 = 0.87619047619052
Γ_{-1}^1 =0.74520547945206	Γ_{-1}^2 = -3.39047619047638
Γ_0^1 =0.0	Γ_0^2 = 5.26785714285743
Γ_1^1 =-0.74520547945206	Γ_1^2 = -3.39047619047638
Γ_2^1 =0.14520547945206	Γ_2^2 = 0.87619047619052
Γ_3^1 =-0.01461187214612	Γ_3^2 = -0.11428571428571
Γ_4^1 =-0.00034246575342	Γ_4^2 = -0.00535714285714

1.5.2. *Iteration Formula for Wavelet-Transform Based Method*

In order to discretize the functional (1.5), the tensor product of the third-order Daubechies' scaling functions is used to span the solution space. The surface height is described as a linear combination of a set of scaling basis functions. After discretization, the problem of depth recovery from gradients becomes a discrete minimization problem. To solve the minimization problem, a perturbation method will be used. The surface height is finally decided after finding the weight coefficients.

We assume that the size of the domain of the surface $Z(x, y)$ equals $N \times N$. Suppose further that the surface $Z(x, y)$ is represented by a linear combination of

a set of third-order Daubechies' scaling basis functions in the following format:

$$Z(x,y) = \sum_{m=0}^{N-1}\sum_{n=0}^{N-1} z_{m,n}\phi_{m,n}(x,y) \tag{1.21}$$

where $z_{m,n}$ are the weight coefficients, $\phi_{m,n}(x,y)$ are the tensor products of the third-order Daubechies scaling functions, with

$$\phi_{m,n}(x,y) = \phi(x-m)\phi(y-n) \tag{1.22}$$

For the known gradient values $p(x,y)$ and $q(x,y)$, we assume that

$$p(x,y) = \sum_{m=0}^{N-1}\sum_{n=0}^{N-1} p_{m,n}\phi_{m,n}(x,y) \tag{1.23}$$

$$q(x,y) = \sum_{m=0}^{N-1}\sum_{n=0}^{N-1} q_{m,n}\phi_{m,n}(x,y) \tag{1.24}$$

where the weight coefficients $p_{m,n}$ and $q_{m,n}$ can be determined by

$$p_{m,n} = \iint p(x,y)\phi_{m,n}(x,y)dxdy \tag{1.25}$$

$$q_{m,n} = \iint q(x,y)\phi_{m,n}(x,y)dxdy \tag{1.26}$$

Substituting (1.21), (1.23) and (1.24) into (1.5), we have that

$$\begin{aligned} W &= \iint \left[\left(\sum_{m,n=0}^{N-1} z_{m,n}\phi^{(x)}_{m,n}(x,y) - \sum_{m,n=0}^{N-1} p_{m,n}\phi_{m,n}(x,y)\right)^2\right] dxdy \\ &+ \iint \left[\left(\sum_{m,n=0}^{N-1} z_{m,n}\phi^{(y)}_{m,n}(x,y) - \sum_{m,n=0}^{N-1} q_{m,n}\phi_{m,n}(x,y)\right)^2\right] dxdy \\ &= W_1 + W_2, \end{aligned} \tag{1.27}$$

where

$$\phi^{(x)}_{m,n}(x,y) = \frac{\partial\phi_{m,n}(x,y)}{\partial x}, \phi^{(y)}_{m,n}(x,y) = \frac{\partial\phi_{m,n}(x,y)}{\partial y}$$

In order to derive the iterative scheme for computing the surface height function $Z(x,y)$, let $\Delta z_{i,j}$ represent the updates of $z_{i,j}$ in the iterative equation, and $z'_{i,j}$ be the value after the update. Then we have that

$$z'_{i,j} = z_{i,j} + \Delta z_{i,j} \tag{1.28}$$

Substituting $z'_{i,j}$ into W_1, W_1 will be changed by ΔW_1, that is,

$$\begin{aligned}
W'_1 &= W_1 + \Delta W_1 \\
&= \iint \left[\left(\sum_{m,n=0}^{N-1} z_{m,n}\phi^{(x)}_{m,n}(x,y) - \sum_{m,n=0}^{N-1} p_{m,n}\phi_{m,n}(x,y) \right) + \Delta z_{i,j}\phi^{(x)}_{i,j}(x,y) \right]^2 dxdy \\
&= W_1 + 2\Delta z_{i,j} \sum_{m,n=0}^{N-1} z_{m,n} \iint \phi^{(x)}_{m,n}(x,y)\phi^{(x)}_{i,j}(x,y)dxdy \\
&\quad -2\Delta z_{i,j} \sum_{m,n=0}^{N-1} p_{m,n} \iint \phi_{m,n}(x,y)\phi^{(x)}_{i,j}(x,y)dxdy \\
&\quad +\Delta z^2_{i,j} \iint \phi^{(x)}_{i,j}(x,y)\phi^{(x)}_{i,j}(x,y)dxdy
\end{aligned} \tag{1.29}$$

By using the tensor product (1.22) and the definitions of the connection coefficients (1.18), (1.19) and (1.20) yields

$$\begin{aligned}
&\iint \phi^{(x)}_{m,n}(x,y)\phi^{(x)}_{i,j}(x,y)dxdy \\
&= \iint \phi^{x}(x-m,y-n)\phi^{(x)}(x-i,y-j)dxdy \\
&= \iint \phi^{x}(x-m)\phi(y-n)\phi^{(x)}(x-i)\phi(y-j)dxdy \\
&= \int \phi^{x}(x-m)\phi^{(x)}(x-i)dx \int \phi(y-n)\phi(y-j)dy \\
&= \int \phi^{x}(x)\phi^{(x)}(x-i+m)dx \int \phi(y)\phi(y-j+n)dy \\
&= \Gamma^2_{i-m}\Gamma^0_{j-n}
\end{aligned} \tag{1.30}$$

Using the same way, we obtain that

$$\iint \phi_{m,n}(x,y)\phi^{(x)}_{i,j}(x,y)dxdy = \Gamma^1_{i-m}\Gamma^0_{j-n} \tag{1.31}$$

$$\iint \phi^{(x)}_{i,j}(x,y)\phi^{(x)}_{i,j}(x,y)dxdy = \Gamma^2_0 \tag{1.32}$$

Substituting (1.30), (1.31) and (1.32) into (1.29) gives

$$\begin{aligned}
W'_1 = W_1 &+ 2\Delta z_{i,j} \sum_{m,n=0}^{N-1} z_{m,n}\Gamma^2_{i-m}\Gamma^0_{j-n} \\
&- 2\Delta z_{i,j} \sum_{m,n=0}^{N-1} p_{m,n}\Gamma^1_{i-m}\Gamma^0_{j-n} + \Delta z^2_{i,j}\Gamma^2_0
\end{aligned} \tag{1.33}$$

Using the same derivation, we have that

$$\begin{aligned} W_2' &= W_2 + \Delta W_2 \\ &= W_2 + 2\Delta z_{i,j} \sum_{m,n=0}^{N-1} z_{m,n} \Gamma_{i-m}^0 \Gamma_{j-n}^2 \\ &\quad -2\Delta z_{i,j} \sum_{m,n=0}^{N-1} q_{m,n} \Gamma_{i-m}^0 \Gamma_{j-n}^1 + \Delta z_{i,j}^2 \Gamma_0^2 \end{aligned} \tag{1.34}$$

Substituting (1.33) and (1.34) into (1.27), it is shown that the energy change is given by

$$\begin{aligned} \Delta W &= \Delta W_1 + \Delta W_2 \\ &= 2\Delta z_{i,j} \sum_{m,n=0}^{N-1} z_{m,n} \left(\Gamma_{i-m}^2 \Gamma_{j-n}^0 + \Gamma_{i-m}^0 \Gamma_{j-n}^2 \right) \\ &\quad -2\Delta z_{i,j} \sum_{m,n=0}^{N-1} p_{m,n} \Gamma_{i-m}^1 \Gamma_{j-n}^0 - 2\Delta z_{i,j} \sum_{m,n=0}^{N-1} q_{m,n} \Gamma_{i-m}^0 \Gamma_{j-n}^1 + 2\Delta z_{i,j}^2 \Gamma_0^2 \end{aligned}$$

In order to make the cost function decrease as fast as possible, ΔW must be maximized. From $\partial \Delta W / \partial \Delta z_{i,j} = 0$, we have that

$$\Delta z_{i,j} = \frac{1}{2\Gamma_0^2} \sum_{k=-2N+2}^{2N-2} \left[\left(p_{i-k,j} + q_{i,j-k} \right) \Gamma_k^1 - \left(z_{i-k,j} + z_{i,j-k} \right) \Gamma_k^2 \right] \tag{1.35}$$

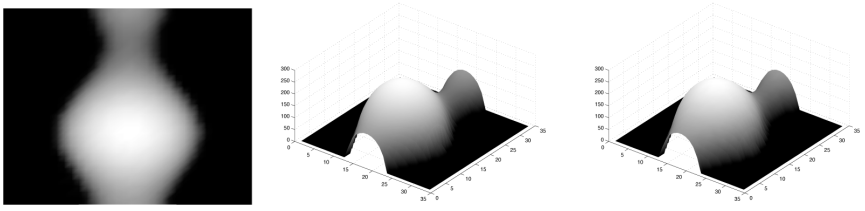
Substituting (1.35) into (1.28) leads to the following iterative scheme:

$$z_{i,j}^{t+1} = z_{i,j}^t + \Delta z_{i,j} \tag{1.36}$$

where t is the iteration index. By taking zero as the initial values, we can iteratively solve the depth recovery from gradients problem using the iterative scheme (1.36).

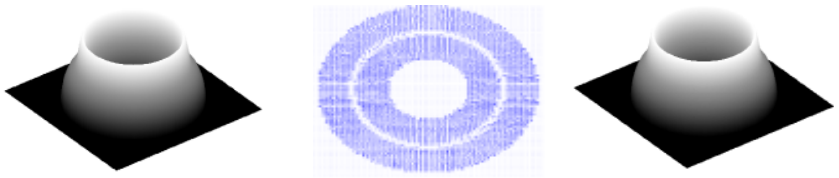
1.6. Experimental Results

To investigate the performance of the algorithms described in the previous sections, we have done several computer simulations on both synthetic and real images.



(a) (b) (c)

Fig. 1.1. Results of a synthetic vase object. (a) Original image. (b) 3D plot of the vase object. (c) Reconstruction result using the proposed two-scan method.



(a) (b) (c)

Fig. 1.2. Results for a torus object. (a) Original surface. (b) Gradient vector fields. (c) Reconstructed surface using wavelet-transform based method.

1.6.1. *Test on Noiseless Gradients*

The two-scan method was tested on a synthetic vase image, which is generated mathematically by the following explicit surface equation:

$$Z(x, y) = \sqrt{f^2(y) - x^2}$$

where

$$f(y) = 0.15 - 0.1y(6y + 1)^2(y - 1)^2(3y - 2)^2,$$
$$-0.5 \leq x \leq 0.5, \quad 0.0 \leq y \leq 1.0$$

The image of this synthetic vase object is shown on the left of Figure 1.1. The 3D plot of the reconstructed surface using the proposed two-scan algorithm is shown in the middle of Figure 1.1. By comparing the 3D plots of the true surface (middle) and the reconstructed surface (right), we can see that they look very similar to each other.

The wavelet-transform based method was applied to a torus image. The original torus image is illustrated on the left of Figure 1.2. The gradient fields of the torus surface is shown in the middle of Figure 1.2. The reconstructed surface height from this gradient fields by the proposed wavelet based method is shown on the

Fig. 1.3. Image triplet of a Beethoven statue.

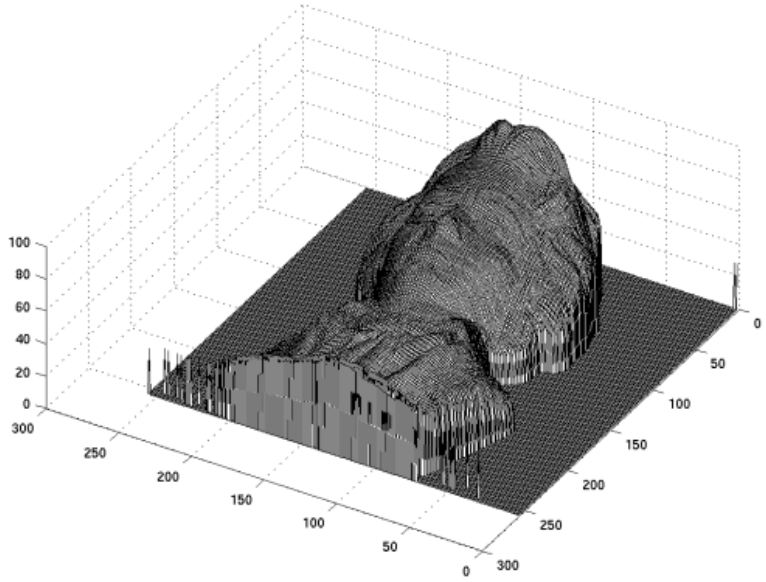


Fig. 1.4. Recovered surface using the Frankot-Chellappa method.

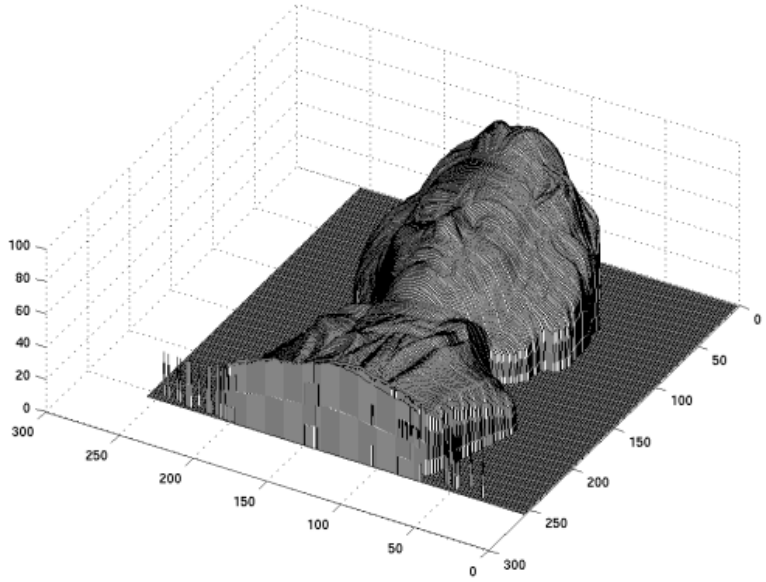


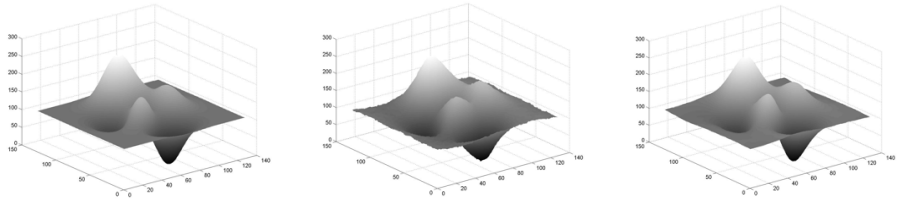
Fig. 1.5. Recovered surface using our Fourier-transform based method, with $\lambda = 0.5$, and $\mu_1 = \mu_2 = 0$.

right of Figure 1.2. It can be seen that the shape of the torus object is correctly reconstructed.

Figure 1.3 shows three captured images of a Beethoven plaster statue, using a static camera but different light sources. The gradients were generated using the albedo-independent photometric stereo method with three light sources (3S PSM) as specified in [13]. Figure 1.5 illustrates both recovered surfaces. The left-hand surface was calculated using the Frankot-Chellappa algorithm [6] as specified in [13], and the right-hand surface was calculated using our Fourier-transform based method with $\lambda = 0.5$, $\mu_1 = 0$, and $\mu_2 = 0$. By comparing the reconstructed surfaces shown in Figure 1.5, we see that the Fourier-transform based method (with the specified parameters) improves the recovered shape.

1.6.2. *Test on Noisy Gradients*

Generally speaking, local methods may provide an unreliable reconstruction, since the errors can propagate along the scan paths. Therefore, we only test the proposed Fourier-transform based method for noisy gradients. This method was implemented with one synthetic image (peaks) and one real image (vase). The discrete gradient vector fields were generated using an SFS algorithm proposed in [23]. The Gaussian noise (with a mean, set to zero, and a standard deviation, set to 0.01) was subsequently added to the generated gradient field in order to test the sensitivity to noise.

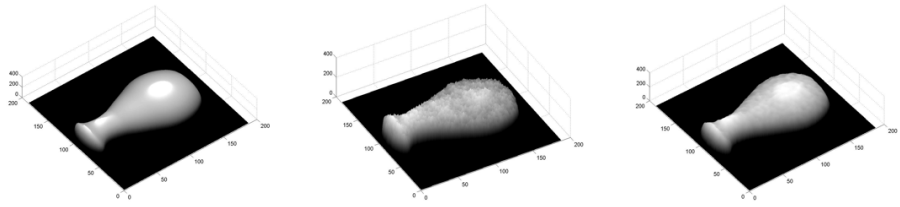


(a) (b) (c)

Fig. 1.6. Results of a synthetic image. (a) Original surface. (b) Reconstructed surface using the Frankot-Chellappa algorithm. (c) Reconstructed surface using our Fourier-transform based method with $\lambda = 0, \mu_1 = 0.1$ and $\mu_2 = 1$.

Figure 1.6 and Figure 1.7 show the reconstructed surfaces when the parameters are given by some specific values. Figure 1.6 shows the original 3D height plot of a synthetic peaks surface (left), the 3D plot of reconstructed surfaces using the Frankot-Chellappa algorithm (middle), and the 3D plot of the reconstructed surface using our Fourier-transform based method with $\lambda = 0$, $\mu_1 = 0.1$, and $\mu_2 = 1$. By comparing the true heights, we can see that the noise is reduced.

Figure 1.7 shows the 3D plot of the original vase surface (left). The reconstructed surfaces with $\lambda = 0$, $\mu_1 = \mu_2 = 0$ and $\mu_1 = 0.1, \mu_2 = 10$ are shown in the middle Figure 1.7 and on the right of Figure 1.7.



(a) (b) (c)

Fig. 1.7. Results of a vase object. (a) Original surface. (b) Reconstructed surface using the Frankot-Chellappa algorithm. (c) Reconstructed surface our Fourier-transform based method with $\lambda = 0, \mu_1 = 0.1, \mu_2 = 10$.

1.7. Conclusion

This chapter proposed three classes of methods for solving the problem of depth recovery from gradient vector fields, based on previous work by the authors (see, for example, [21]). The derivation details of these approaches are given. The derivation process of the two-scan method is very simple. The wavelet-transform based method is derived by representing the surface height as a linear combination of third-order Daubechies' scaling basis functions. This method converts the depth recovery from gradients problem to one of solving an iterative equation. Hence, the method can be easily implemented. The mathematics is somewhat more complicated, but the fact that fewer iterations are required is the major advantage of the wavelet-transform based method. The Fourier-transform based method has some distinct advantages. The surface of the object is constructed in one pass utilizing all of the given gradient estimates, and the robustness of the Fourier-transform based method to noisy gradient fields can be improved by choosing associated weighting parameters. The choice of parameters heavily affects the surface reconstructed from gradients. Therefore, the criterion for the choice of the parameters and the relation between the parameters and noise should be a future topic of research. Generally speaking, a constrained minimization problem can be formulated for an optimal choice of parameters. This is a problem which many researchers have been trying to solve for several years, and so far there is no systematic way derived for choosing parameters.

References

[1] G. Beylkin. On the representation of operators in bases of compactly supported wavelets. *SIAM J. Numerical Analysis*, **29**:1716–1740, 1992.

[2] M. Castelán, A. Robles-Kelly and E. R. Hancock. A coupled statistical model for face shape recovery from brightness images. *IEEE Trans. Image Processing*, **16**:1139–1151, 2007.

[3] N. E. Coleman, Jr. and R. Jain. Obtaining 3-dimensional shape of textured and

specular surfaces using four-source photometry. *Computer Vision Graphics Image Processing*, **18**:439–451, 1982.

[4] I. Daubechies. Orthonormal bases of compactly supported wavelets. *Commun. Pure Appl. Mathematics*, **41**:909–996, 1988.

[5] J. Fan and L. B. Wolff. Surface curvature and shape reconstruction from unknown multiple illumination and integrability. *Computer Vision Image Understanding*, **65**:347–359, 1997.

[6] R. T. Frankot and R. Chellappa. A method for enforcing integrability in shape from shading algorithms. *IEEE Trans. Pattern Analysis Machine Intelligence*, **10**:439–451, 1988.

[7] G. Healey and R. Jain. Depth recovery from surface normals. In Proc. *Int. Conf. Pattern Recognition*, Volume **2**, pages 894–896, 1984.

[8] B. K. P. Horn. Height and gradient from shading. *Int. J. Computer Vision*, **5**:37–75, 1990.

[9] B. K. P. Horn and M. J. Brooks. The variational approach to shape from shading. *Computer Vision Graphics Image Processing*, **33**:174–208, 1986.

[10] K. Ikeuchi. Determining surface orientations of specular surfaces by using the photometric stereo method. *IEEE Trans. Pattern Analysis Machine Intelligence*, **3**:661–669, 1981.

[11] K. Ikeuchi and B. K. P. Horn. Numerical shape from shading and occluding boundaries. *Artificial Intelligence*, **17**:141–184, 1981.

[12] R. Klette and K. Schlüns. Height data from gradient fields. In Proc. *Machine Vision Applications Architectures Systems Integration*, SPIE **2908**, pages 204–215, 1996.

[13] R. Klette, A. Koschan, and K. Schlüns. *Computer Vision – Räumliche Information aus digitalen Bildern*. Vieweg, Braunschweig, 1996.

[14] C.-H. Lee and A. Rosenfeld. Improved methods of estimating shape from shading using the light source coordinate system. *Artificial Intelligence*, **26**:125–143, 1985.

[15] S. G. Mallat. *A Wavelet Tour of Signal Processing*. Academic Press, 1999.

[16] A. Robles-Kelly and E. R. Hancock. A graph-spectral method for surface height recovery. *Pattern Recognition*, **38**:1167–1186, 2005.

[17] A. Robles-Kelly and E. R. Hancock. Shape-from-shading using the heat equation. *IEEE Trans. Image Processing*, **16**:7–21, 2007.

[18] M. L. Smith. The analysis of surface texture using photometric stereo acquisition and gradient space domain mapping. *Image Vision Computing*, **17**:1009–1019, 1999.

[19] M. L. Smith, T. Hill, and G. Smith. Surface texture analysis based upon the visually acquired perturbation of surface normals. *Image Vision Computing*, **15**:949–955, 1997.

[20] W. A. P. Smith and E. R. Hancock. Recovering facial shape using a statistical model of surface normal direction. *IEEE Trans. Pattern Analysis Machine Intelligence*, **28**:1914–1930, 2006.

[21] T. Wei and R. Klette. Depth recovery from noisy gradient vector fields using regularization. In Proc. *Computer Analysis Images Patterns*, pages 116-123, LNCS 2756, Springer, Berlin, 2003.

[22] R. J. Woodham. Photometric method for determining surface orientation from multiple images. *Optical Engineering*, **19**:139–144, 1980.

[23] P. L. Worthington and E. R. Hancock. New constraints on data-closeness and needle map consistency for shape-from-shading. *IEEE Trans. Pattern Analysis Machine Intelligence*, **21**:1250–1267, 1999.

[24] Z. Wu and L. Li. A line-integration based method for depth recovery from surface normals. *Computer Vision Graphics Image Processing*, **43**:53–66, 1988.