Photometric Stereo Method and one more MRF[1]

(80 min lecture)

See Material in
Reinhard Klette: Concise Computer Vision
Springer-Verlag, London, 2014

[1]See last slide for copyright information.

Surface Reconstruction

Computer vision reconstructs and analyzes the visible world, typically defined by textured surfaces (e.g. not considering fully or partially transparent objects)

For example, three different techniques for vision-based reconstruction of 3D shapes:

1. Structured lighting as a relatively simple but accurate method (not discussed her)
2. Stereo vision is *the* 3D shape reconstruction method in computer vision (see previous lectures)
3. As an alternative technique, shading-based 3D shape understanding (in this lecture)

Photometric Stereo Method

Before: one light source (e.g. sun) and multiple static or mobile cameras

Now: one static camera and multiple static light sources (indoor)

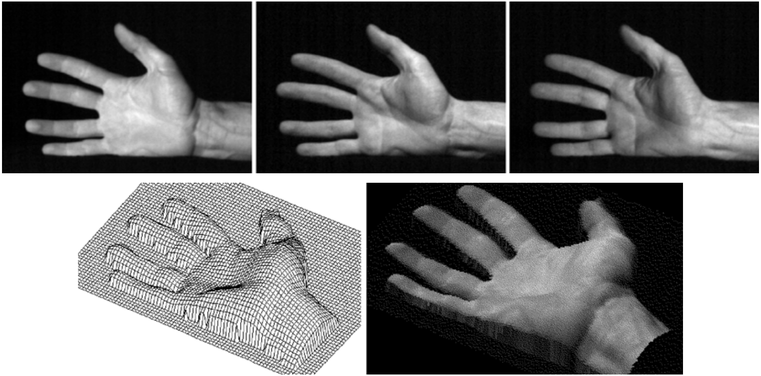
Light sources are switched on and off while taking images

Defines *photometric stereo method* (PSM), example of *shape from shading*

We describe

1. Gradient space (for modelling reflectance maps)
2. *Lambertian reflectance* (e.g. human skin is approximate Lambertian)
3. 3-light-source PSM (3PSM) for deriving surface gradients
4. How surface gradients can be mapped into a 3D shape

3PSM Example



Three input images for PSM; the hand is not allowed to move when taking these images. Reconstructed surface (in 1994) by 3PSM

Agenda

Surface Gradients and Normals

Gradient of surface $Z = f(X, Y)$ in 3D space

$$\nabla Z = \mathbf{grad}\ Z = \left[\frac{\partial Z}{\partial X}, \frac{\partial Z}{\partial Y}\right]^\top$$

Example: A plane $Z = aX + bY + c$, then gradient $[a, b]^\top$

Normal of surface $Z = f(X, Y)$

$$\mathbf{n} = \left[\frac{\partial Z}{\partial X}, \frac{\partial Z}{\partial Y}, 1\right]^\top = [a, b, 1]^\top$$

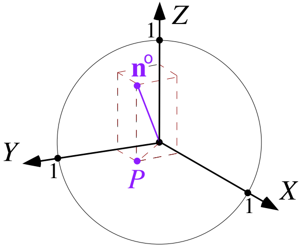
Going away from the image plane by having value $+1$ in third component

We use of a and b for denoting X- and Y-components of a normal

Unit normal

Unit normal (of length 1)

$$\mathbf{n}^\circ = [n_1, n_2, n_3]^\top = \frac{\mathbf{n}}{||\mathbf{n}||_2} = \frac{[a, b, 1]^\top}{\sqrt{a^2 + b^2 + 1}}$$



Gaussian sphere: radius 1, centered at origin O

Slant σ: angle between vector $\mathbf{n}^\circ$ and Z-axis

Tilt θ: angle between vector from O to $P = (X, Y, 0)$ and X-axis

Unit normal $\mathbf{n}^\circ$ uniquely represented by spherical coordinates (σ, θ)

Gradient Space

ab-space of gradient or normal coordinates a and b

Point (a, b) in gradient space represents gradient $[a, b]^\top$ in XYZ space

Example

Plane $Z = aX + bY + c$ in XYZ space

(a, b) in gradient space represents all planes parallel to the given plane

(i.e., for any $c \in \mathbb{R}$)

Orthogonal Normals

Given: $\mathbf{n}_1 = [a_1, b_1, 1]^\top$ in XYZ space (say: direction to a light source)

Task: characterize normal $\mathbf{n}_2 = [a_2, b_2, 1]^\top$ orthogonal to $\mathbf{n}_1$
(say: directions where surface points are "just" not illuminated anymore)

Dot product of both vectors:

$$\mathbf{n_1} \cdot \mathbf{n_2} = a_1 a_2 + b_1 b_2 + 1 = ||\mathbf{n_1}||_2 \cdot ||\mathbf{n_2}||_2 \cdot \cos \frac{\pi}{2} = 0$$

Given a_1 and b_1 define a straight line $\gamma : a_1 a_2 + b_1 b_2 + 1 = 0$ in gradient space with unknowns a_2 and b_2:

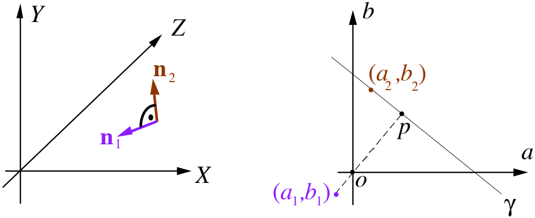
1. the line incident with origin $o = (0, 0)$ and point p is orthogonal to line γ
2. $\sqrt{a_1^2 + b_1^2} = d_2((a_1, b_1), o) = 1/d_2(p, o)$
3. p and (a_1, b_1) are in opposite quadrants

Dual Straight Line

Line γ is uniquely defined by these three properties on page before

Line γ is called the *dual straight line* to normal $\mathbf{n}_1$ or to (a_1, b_1)

Any direction $\mathbf{n}_2$ orthogonal to $\mathbf{n}_1$ is located on γ



Agenda

1. Gradient Space
2. Light Source and Lambertian Reflectance
3. 3PSM
4. Integration of Gradient Fields

Point-Light Source Assumptions

1. A light source L is identified with a single point in $\mathbb{R}^3$
2. Light source emits light uniformly in all directions; with intensity

$$E_L$$

(i.e. integral over energy distribution curve of light source L in visible spectrum of light)

3. Light source "not very close" to objects of interest (objects are "small" compared to the distance to the light source); in direction

$$\mathbf{s}_L = [a_L, b_L, 1]^\top$$

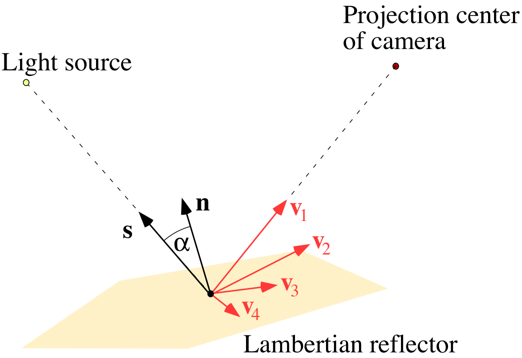
relatively to illuminated surfaces

Model is abstract: an existing light source actually

Illuminates only a limited range
It has a particular energy distribution curve $L(\lambda)$
It has a geometric shape

Illuminated and Recorded Point on a Surface

Surface normal **n**, direction **s** to light source, viewing directions $\mathbf{v}_i$



Lambertian reflector: Radiant intensity observed from an *diffusely reflecting surface* is directly proportional to the cosine of angle α (i.e. viewing direction does not matter as long as point is visible)

Recall Inner Vector Product and Albedo

Normal $\mathbf{n}_P = [a, b, 1]^\top$ at a visible and illuminated surface point P

1. Inner vector product:

$$\mathbf{s}^\top \cdot \mathbf{n}_P = ||\mathbf{s}||_2 \cdot ||\mathbf{n}_P||_2 \cdot \cos\alpha$$

   defines

$$\cos\alpha = \frac{\mathbf{s}^\top \cdot \mathbf{n}_P}{||\mathbf{s}||_2 \cdot ||\mathbf{n}_P||_2}$$

2. Emitted light at a point P is scaled by

$$\eta(P) = \rho(P) \cdot \frac{E_L}{\pi}$$

   where π is the spatial angle of a halfsphere
   $\rho(P)$ is the *albedo* (i.e. surface reflectance) at P, with $0 \leq \rho(P) \leq 1$

Lambert's Cosine Law

Emitted light at surface point P, called the *reflectance* at P:

$$
\begin{aligned}
R(P) &= \eta(P) \cdot \frac{\mathbf{s}_L^\top \cdot \mathbf{n}_P}{||\mathbf{s}_L||_2 \cdot ||\mathbf{n}_P||_2} \\
&= \eta(P) \cdot \frac{a_L a + b_L b + 1}{\sqrt{a_L^2 + b_L^2 + 1} \cdot \sqrt{a^2 + b^2 + 1^2}}
\end{aligned}
$$

Reflectance $R(P) \geq 0$ is a second-order curve in unknowns a and b

$\eta(P)$ depends on energy E_L and albedo $\rho(P)$

Reflectance Maps

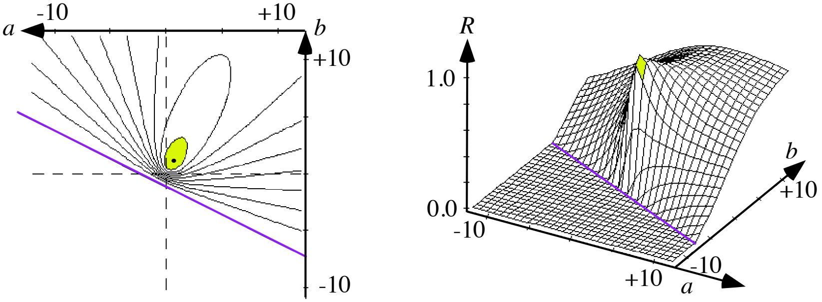
A *reflectance map* is defined in *ab* gradient space

Reflectance map assigns a reflectance value to gradient value (a, b) assuming that the given surface reflectance is uniquely defined by the gradient value (as it is the case for Lambertian reflectance)

Lambertian Reflectance Map:
Point P with albedo $\rho(P)$ and gradient (a, b)

1. $\mathbf{n}_P = \mathbf{s}_L$:
   $\alpha = 0$ and $\cos\alpha = 1$; curve degenerates into $R(P) = \eta(P)$, the maximal possible value; value $\eta(P)$ at point (a_L, b_L) in gradient space
2. $\mathbf{n}_P$ orthogonal to $\mathbf{s_L}$:
   surface point P is "just" not illuminated anymore; $\mathbf{n}_P$ is on the dual straight line to $\mathbf{s}_L$; $\alpha = \pi/2$ and $\cos\alpha = 0$; curve degenerates into $R(P) = 0$, the minimal possible value; value 0 along the straight line dual to (a_L, b_L) in gradient space

Isolines in a Lambertian Reflectance Map



Duality between straight line and direction to light source

For $R(P)$ between 0 and $\eta(P)$, the curve is either parabolic or hyperbolic

Agenda

1. Gradient Space
2. Light Source and Lambertian Reflectance
3. 3PSM
4. Integration of Gradient Fields

Recovering Surface Gradients

One static camera (on a tripod) and three different point-light sources at directions $\mathbf{s}_i$, for $i = 1, 2, 3$

Reflectance map value $R_i(P)$ uniformly scaled by constant $c > 0$ (due to distance between object and camera) and mapped into monochromatic value u_i in the image

Capture three images, turn only one light source on at a time

Point P maps at the same pixel position into three intensity values

$$u_i = R_i(P) = \frac{\eta(P)}{c} \cdot \frac{\mathbf{s}_i^\top \cdot \mathbf{n}_P}{||\mathbf{s_i}||_2 \cdot ||\mathbf{n}_P||_2}$$

defining three second-order curves in gradient space, which (ideally) intersect at (a, b), where $\mathbf{n}_P = [a, b, 1]^\top$ is the normal at point P

nPSM

n-source photometric stereo method (nPSM) attempts to recover surface normals by some kind of **practical implementation** for determining this ideal intersection at (a, b), for $n \geq 3$

Example: Intersect regions in gradient space with values $R_i \geq u_i - \varepsilon$

Three light sources at least for unique normal reconstruction

PSM also used short for 3PSM

Albedo-Independent PSM

Consider surfaces with (approximate) Lambertian reflectance

We cannot assume that albedo $\rho(P)$ is constant on a recorded surface

Coloring of surface points changes and thus also the albedo

We need to consider *albedo-independent PSM*

Example:

Human skin is approximate Lambertian but with changes in albedo

Non-Lambertian: e.g. specularities in open eyes

Derivation of an Algebraic Solution

We have three equations

$$u_i = E_i \cdot \frac{\rho(P)}{c\pi} \cdot \frac{\mathbf{s}_i^\top \cdot \mathbf{n}_P}{||\mathbf{s_i}||_2 \cdot ||\mathbf{n}_P||_2}$$

for three light sources, with $i = 1, 2, 3$

1. Multiply first equation (i.e. $i = 1$) by $u_2 \cdot ||\mathbf{s}_2||_2$
2. Multiply second equation (i.e. $i = 2$) by $-u_1 \cdot ||\mathbf{s}_1||_2$
3. Add both; this results into

$$\rho(P) \cdot \mathbf{n}_P \cdot (E_1 u_2 ||\mathbf{s}_2||_2 \cdot \mathbf{s}_1 - E_2 u_1 ||\mathbf{s}_1||_2 \cdot \mathbf{s}_2) = 0$$

   Thus: vector $\mathbf{n}_P$ is orthogonal to this difference of vectors $\mathbf{s}_1$ and $\mathbf{s}_2$

4. Assume $\rho(P) > 0$. If using Images 1 and 3, we have that

$$\mathbf{n}_P \cdot (E_1 u_3 ||\mathbf{s}_3||_2 \cdot \mathbf{s}_1 - E_3 u_1 ||\mathbf{s}_1||_2 \cdot \mathbf{s}_3) = 0$$

The Solution

$\mathbf{n}_P$ is collinear with the cross product

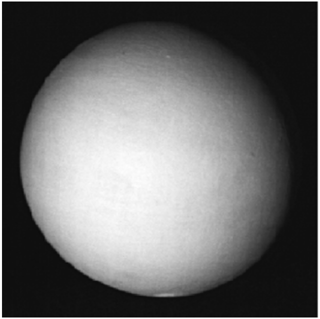
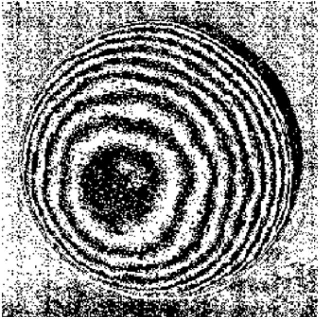
$$(E_1 u_2 ||\mathbf{s}_2||_2 \cdot \mathbf{s}_1 - E_2 u_1 ||\mathbf{s}_1||_2 \cdot \mathbf{s}_2) \times (E_1 u_3 ||\mathbf{s}_3||_2 \cdot \mathbf{s}_1 - E_3 u_1 ||\mathbf{s}_1||_2 \cdot \mathbf{s}_3)$$

This uniquely defines the unit normal $\mathbf{n}_P^\circ$ pointing away from the camera

Needed:

1. Directions to light sources
2. Relative intensities of light sources (no absolute measurements E_i)

Calibration Sphere

Calibration sphere: Uniform albedo and Lambertian reflectance

Use mean intensities in recorded calibration sphere images for estimating energy ratios between intensities E_i of the three light sources

Illustration of detected iso-intensity “lines” with expected noisiness

Direction to Light Sources by Inverse PSM

Calibration of directions $\mathbf{s}_i$ to the three light sources

We apply *inverse photometric stereo*:

Use of a Lambertian *calibration sphere* with uniform albedo

Place sphere about at location of objects (to be modelled)

1. Identify the circular border of the imaged sphere
2. Calculate surface normals (of the sphere) at more than three points P (say, at about 100) projected into pixel positions within the circle
   How?
3. Identify direction $\mathbf{s}_i$ by least-square error optimization using the 3PSM solution equations
4. We have values u_i and normals $\mathbf{n}_P$; solve for unknown direction $\mathbf{s}_i$

Albedo Recovery

For $i = 1, 2, 3$, consider equations

$$u_i = \frac{E_i}{c\pi} \cdot \rho(P) \cdot \frac{\mathbf{s}_i^\top \cdot \mathbf{n}_P}{||\mathbf{s_i}||_2 \cdot ||\mathbf{n}_P||_2}$$

We have three values u_i at p (projection of surface point P)

We have (approximate) values for unit vectors $\mathbf{s}_i^\circ$ and $\mathbf{n}_P^\circ$

We have relative intensities of the three light sources.

Only remaining unknown is $\rho(P)$

Combine first and second, first and third, and second and third image for a robust estimation of $\rho(P)$

How?

Why Albedo Recovery?

The knowledge of the albedo is of importance for light-source independent modeling of the surface of an object, defined by geometry and texture (albedo)

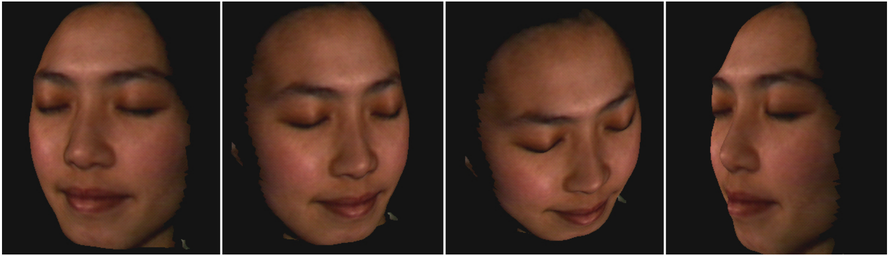
In general (if not limited to Lambertian reflectance), the albedo depends upon the wavelength of the illuminating light

As a first approximation, we may use light sources of different color, such as red, green, or blue light, to recover the related albedo values

Note that after knowing $\mathbf{s}^\circ$ and $\mathbf{n}^\circ$, we only have to change the wave length of illuminations (e.g. using transparent filters), assuming the object is not moving in between

Example: Human Faces

3PSM is of reasonable accuracy for recovering the albedo values of a human face



Face recovered by 3PSM (at University of Auckland in 2000)

Closed eyes avoid the recording of specularity

Agenda

1. Gradient Space
2. Light Source and Lambertian Reflectance
3. 3PSM
4. Integration of Gradient Fields

Discrete Gradient Field

Integration: Discrete fields of gradients into a surface

Integration is not unique even when dealing with smooth surfaces

Result only determined up to an additive constant

Ill-Posedness of Discrete Integration

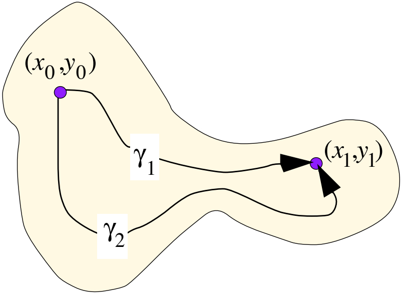
Results of PSM are discrete and erroneous surface gradient data

Surfaces often "non-smooth" (e.g. polyhedral)

Example: Camera looks onto a stair case, orthogonal to the front faces; recovered normals point straight towards camera

Densities of recovered surface normals do not correspond uniformly to local surface slopes

Integration Paths in Continuous Case



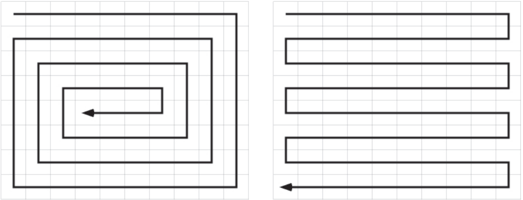
Assume:

1. A surface patch defined on a simply-connected set
2. Its explicit surface function satisfies the integrability condition

Then: local integration along different paths will lead (in the continuous case) to identical elevation results at point (x_1, y_1), after starting at (x_0, y_0) with the same initial elevation value

Local Integration Method in Discrete Case

Implement integration along a selected path



Use

1. Initial Z-value and
2. Local neighborhoods at pixel locations when updating Z-values incrementally

Task

Recover depth function Z such that

$$\begin{aligned}\frac{\partial Z}{\partial x}(x,y) &= a(x,y)\\ \frac{\partial Z}{\partial y}(x,y) &= b(x,y)\end{aligned}$$

Averaging of gradients or normals
Line connecting $(x, y+1, Z_{x,y+1})$ and $(x+1, y+1, Z_{x+1,y+1})$ is approximately perpendicular to average normal between these two points

Notation
Gradient values $a_{x,y}$ and $b_{x,y}$ at pixel locations $(x,y) \in \Omega$

Example for Local Increment Calculation

$$Z_{x+1,y+1} = Z_{x,y+1} + \frac{1}{2}\left(a_{x,y+1} + a_{x+1,y+1}\right)$$

$$Z_{x+1,y+1} = Z_{x+1,y} + \frac{1}{2}\left(b_{x+1,y} + b_{x+1,y+1}\right)$$

Adding both equations and dividing by 2

$$\begin{aligned} Z_{x+1,y+1} &= \frac{1}{2}\left(Z_{x,y+1} + Z_{x+1,y}\right) \\ &+ \frac{1}{4}\left(a_{x,y+1} + a_{x+1,y+1} + b_{x+1,y} + b_{x+1,y+1}\right) \end{aligned}$$

Two Stages of Algorithm

Total number of points on object surface is $N_{cols} \times N_{rows}$

Two arbitrary initial height values at $(1,1)$ and at (N_{cols}, N_{rows})

Two-scan algorithm: first stage starts at $(1,1)$, determines height values along x-axis and y-axis by discretizing weak integrability in terms of forward differences

$$\begin{aligned} Z_{x,1} &= Z_{x-1,1} + a_{x-1,1} \\ Z_{1,y} &= Z_{1,y-1} + b_{1,y-1} \end{aligned}$$

where $x = 2, ..., N_{cols}$ and $j = 2, ..., N_{rows}$, and scans image then vertically using the local increments defined on the last slide

Second Stage

Starts at corner (N_{cols}, N_{rows}), sets height values by

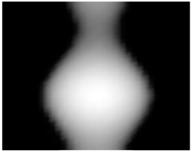
$$\begin{aligned} Z_{x-1,N_{rows}} &= Z_{x,N_{rows}} - a_{x,N_{rows}} \\ Z_{N_{cols},y-1} &= Z_{N_{cols},y} - b_{N_{cols},y} \end{aligned}$$

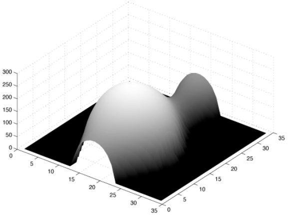
and scans the image horizontally using

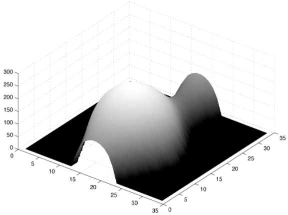
$$\begin{aligned} Z_{x-1,y-1} &= \frac{1}{2}\left(Z_{x-1,y} + Z_{x,y-1}\right) \\ &- \frac{1}{4}\left(a_{x-1,y} + a_{x,y} + b_{x,y-1} + b_{x,y}\right) \end{aligned}$$

Final step: Estimated height values are affected by the choice of the initial height values. Take average of results of both scans as final height value

Example for Two-scan Method







Original synthetic vase object

Ground truth: 3D plot of the vase object

Reconstruction result using the two-scan method

General: Local methods provide unreliable reconstructions for noisy gradient inputs since errors propagate along the scan paths

Generation of the Vase

Synthetic vase generated by using the following explicit surface equation

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

where

$$\begin{aligned} f(y) &= 0.15 - 0.1 \cdot y(6y+1)^2(y-1)^2(3y-2)^2 \\ &\text{for} \quad -0.5 \leq x \leq 0.5 \\ &\text{and} \quad 0.0 \leq y \leq 1.0 \end{aligned}$$

Global Integration

Gradient vector estimated at any $p \in \Omega$

Task: map this uniform and dense gradient vector field into a surface in 3D space which is likely to be the actual surface which caused the estimated gradient vector field

Depth values $Z(x, y)$ define labels at pixel locations (x, y)

Back to a labeling problem with error (or energy) minimization

Data term

$$E_{data}(Z) = \sum_{\Omega}[(Z_x - a)^2 + (Z_y - b)^2] + \lambda_0 \sum_{\Omega}[(Z_{xx} - a_x)^2 + (Z_{yy} - b_y)^2]$$

Smoothness term

$$E_{smooth}(Z) = \lambda_1 \sum_{\Omega}[Z_x^2 + Z_y^2] + \lambda_2 \sum_{\Omega}[Z_{xx}^2 + 2Z_{xy}^2 + Z_{yy}^2]$$

Notation

Z_x and Z_y

first-order partial derivatives of Z

a_x and b_y

first-order partial derivatives of a and b

Z_{xx}, $Z_{xy} = Z_{yx}$, and Z_{yy}

second-order partial derivatives of Z

$\lambda_0 \geq 0$ controls consistency between surface curvature and changes in available gradient data

$\lambda_1 \geq 0$ controls smoothness of surface

$\lambda_2 \geq 0$ controls smoothness of surface curvature

Total Energy, MRF Model, and Two Algorithms

Determine surface Z (i.e. the labeling function) such that total error (or total energy)

$$E_{total}(Z) = E_{data}(Z) + E_{smooth}(Z)$$

is minimized; derivatives in E_{smooth} define MRF again

Two Algorithms

Frankot-Chellappa algorithm is for $\lambda_0 = \lambda_1 = \lambda_2 = 0$, thus not using the second part of the data constraint and no smoothness constraint at all

Wei-Klette algorithm also uses second-order derivatives (curvature) and smoothness optimization

Optimization problem can be solved by using the theory of projections onto convex sets

Gradient field $(a_{x,y}, b_{x,y})$ is projected onto the nearest integrable gradient field in the least-square sense, using the Fourier transform for optimizing in the frequency domain

2D DFT of surface function $Z(x, y)$

$$\mathbf{Z}(u, v) = \frac{1}{|\Omega|} \sum_{(x,y)\in\Omega} Z(x, y) \cdot \exp\left[-i2\pi\left(\frac{xu}{N_{cols}} + \frac{yv}{N_{rows}}\right)\right]$$

Inverse transform

$$Z(x, y) = \sum_{(u,v)\in\Omega} \mathbf{Z}(u, v) \cdot \exp\left[i2\pi\left(\frac{xu}{N_{cols}} + \frac{yv}{N_{rows}}\right)\right]$$

$i = \sqrt{-1}$ and u and v represent frequencies in 2D Fourier domain

Relevant Fourier Pairs and Parseval's Theorem

$$
\begin{aligned}
Z_x(x,y) &\Leftrightarrow iu\,\mathbf{Z}(u,v) \\
Z_y(x,y) &\Leftrightarrow iv\,\mathbf{Z}(u,v) \\
Z_{xx}(x,y) &\Leftrightarrow -u^2\,\mathbf{Z}(u,v) \\
Z_{yy}(x,y) &\Leftrightarrow -v^2\,\mathbf{Z}(u,v) \\
Z_{xy}(x,y) &\Leftrightarrow -uv\,\mathbf{Z}(u,v)
\end{aligned}
$$

Define the appearance of considered derivatives of Z in frequency domain

$\mathbf{A}(u,v)$ and $\mathbf{B}(u,v)$ be Fourier transforms of gradients $A(x,y) = a_{x,y}$ and $B(x,y) = b_{x,y}$, respectively

Parseval's Theorem

$$
\frac{1}{|\Omega|}\sum_{\Omega} |Z(x,y)|^2 = \sum_{\Omega} |\mathbf{Z}(u,v)|^2
$$

Optimization in Frequency Domain

In conclusion to Parseval's Theorem: Equivalence of optimization problem in spatial domain to optimization problem in frequency domain

Minimize, where sums are for $(u, v) \in \Omega$:

$$\begin{aligned} & \sum_{\Omega} \left[(iu\mathbf{Z} - \mathbf{A})^2 + (iv\mathbf{Z} - \mathbf{B})^2 \right] \\ + \quad & \lambda_0 \sum_{\Omega} \left[\left(-u^2\mathbf{Z} - iu\mathbf{A}\right)^2 + \left(-v^2\mathbf{Z} - iv\mathbf{B}\right)^2 \right] \\ + \quad & \lambda_1 \sum_{\Omega} \left[(iu\mathbf{Z})^2 + (iv\mathbf{Z})^2 \right] \\ + \quad & \lambda_2 \sum_{\Omega} \left[\left(-u^2\mathbf{Z}\right)^2 + 2\,(-uv\mathbf{Z})^2 + \left(-v^2\mathbf{Z}\right)^2 \right] \end{aligned}$$

Start of Solution Process

Above expression expanded into

$$\begin{aligned}
&\sum_{\Omega}\left[u^2\mathbf{ZZ}^\star - iu\mathbf{ZA}^\star + iu\mathbf{Z}^\star\mathbf{A} + \mathbf{AA}^\star\right.\\
&\qquad\left. + v^2\mathbf{ZZ}^\star - iv\mathbf{ZB}^\star + iv\mathbf{Z}^\star\mathbf{B} + \mathbf{BB}^\star\right]\\
&+\lambda_0\sum_{\Omega}\left[u^4\mathbf{ZZ}^\star - iu^3\mathbf{ZA}^\star + iu^3\mathbf{Z}^\star\mathbf{A} + u^2\mathbf{AA}^\star\right.\\
&\qquad\left. + v^4\mathbf{ZZ}^\star - iv^3\mathbf{ZB}^\star + iv^3\mathbf{Z}^\star\mathbf{B} + v^2\mathbf{BB}^\star\right]\\
&+\lambda_1\sum_{\Omega}\left(u^2+v^2\right)\mathbf{ZZ}^\star\\
&+\lambda_2\sum_{\Omega}\left(u^4+2u^2v^2+v^4\right)\mathbf{ZZ}^\star
\end{aligned}$$

$\star$ denotes the complex conjugate, and sums for $(u, v) \in \Omega$

Optimization in Frequency Space

Differentiating the above expression with respect to $\mathbf{Z}^\star$ and setting the result to zero, we can deduce the necessary condition for a minimum of the cost function

For each $(u, v) \in \Omega$ we have

$$\left(u^2\mathbf{Z} + iu\mathbf{A} + v^2\mathbf{Z} + iv\mathbf{B}\right) + \lambda_0\left(u^4\mathbf{Z} + iu^3\mathbf{A} + v^4\mathbf{Z} + iv^3\mathbf{B}\right) \\ + \lambda_1\left(u^2 + v^2\right)\mathbf{Z} + \lambda_2\left(u^4 + 2u^2v^2 + v^4\right)\mathbf{Z} = 0$$

A rearrangement of this equation yields

$$\left[\lambda_0\left(u^4 + v^4\right) + (1 + \lambda_1)\left(u^2 + v^2\right) + \lambda_2\left(u^2 + v^2\right)^2\right]\mathbf{Z}(u, v) \\ + i\left(u + \lambda_0 u^3\right)\mathbf{A}(u, v) + i\left(v + \lambda_0 v^3\right)\mathbf{B}(u, v) = 0$$

Solution

Solve the above equation except for $(u, v) \neq (0, 0)$:

$$\mathbf{Z}(u, v) = \frac{-i\left(u + \lambda_0 u^3\right)\mathbf{A}(u, v) - i\left(v + \lambda_0 v^3\right)\mathbf{B}(u, v)}{\lambda_0\left(u^4 + v^4\right) + \left(1 + \lambda_1\right)\left(u^2 + v^2\right) + \lambda_2\left(u^2 + v^2\right)^2}$$

Result

This is the Fourier transform of the unknown surface function $Z(x, y)$ expressed as a function of the Fourier transforms of the given gradients $A(x, y) = a_{x,y}$ and $B(x, y) = b_{x,y}$

Algorithm Part 1: Forward Transform

input gradients $a(x, y), b(x, y)$; parameters λ_0, λ_1, and λ_2
for $(x, y) \in \Omega$ **do**
  if $(|a(x, y)| < c_{max}$ & $|b(x, y)| < c_{max})$ **then**
    A1(x,y)=a(x,y); A2(x,y)=0;
    B1(x,y)=b(x,y); B2(x,y)=0;
  else
    A1(x,y)=0; A2(x,y)=0;
    B1(x,y)=0; B2(x,y)=0;
  end if
end for
Calculate Fourier transform in place: A1(u,v), A2(u,v);
Calculate Fourier transform in place: B1(u,v), B2(u,v);

Algorithm Part 2: Optimize in Frequency Domain

for $(u, v) \in \Omega$ **do**
    if $(u \neq 0 \ \& \ v \neq 0)$ **then**
        $\Delta = \lambda_0 \left(u^4 + v^4\right) + (1 + \lambda_1) \left(u^2 + v^2\right) + \lambda_2 \left(u^2 + v^2\right)^2;$
        $H1(u, v) = [(u + \lambda_0 u^3) A2(u, v) + (v + \lambda_0 v^3) B2(u, v)]/\Delta;$
        $H2(u, v) = [-(u + \lambda_0 u^3) A1(u, v) - (v + \lambda_0 v^3) B1(u, v)]/\Delta;$
    else
        $H1(0, 0) =$ average depth; $H2(0, 0) = 0;$
    end if
end for

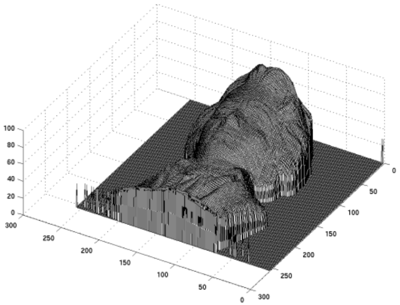
Algorithm Part 3: Backward Transform

Calculate inverse Fourier transform of H1(u,v) and H2(u,v) in place: H1(x,y), H2(x,y);
for $(x, y) \in \Omega$ **do**
    $Z(x, y) = H1(x, y)$;
end for

Example 1

Image triplet of a Beethoven statue used as input for 3PSM

Example



Left: Recovered surface using the Frankot-Chellappa algorithm

Right: Recovered surface using the Wei-Klette algorithm with $\lambda_0 = 0.5$ and $\lambda_1 = \lambda_2 = 0$

Comments

Constant $c_{\max}$ eliminates gradient estimates which define angles with the image plane close to $\pi/2$

A value such as $c_{\max} = 12$ is an option

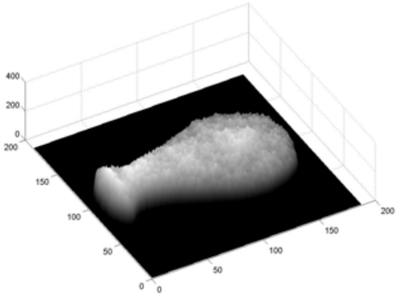
Real parts are stored in arrays A1, B1, and H1, and imaginary parts in arrays A2, B2, and H2

Average height can be estimated for the visible scene

Parameters λ_0, λ_1 and λ_2 should be chosen based on experimental evidence for the given scene

Test on Noisy Gradients

Generate discrete gradient vector field for synthetic vase and add Gaussian noise (with a mean, set to zero, and a standard deviation, set to 0.01)



Left: Frankot-Chellappa. *Right*: Wei-Klette with $\lambda_0 = 0$, $\lambda_1 = 0.1$, and $\lambda_2 = 1$

Copyright Information

This slide show was prepared by Reinhard Klette
with kind permission from Springer Science+Business Media B.V.

The slide show can be used freely for presentations.
However, *all the material* is copyrighted.

In case of citation: just cite the book, that's fine.