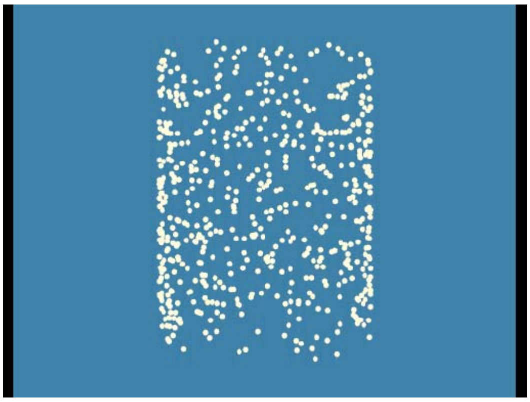
3D Motion Analysis Based on 2D Point Displacements

2D displacements of points observed on an unknown moving rigid body may provide information about

- the 3D structure of the points
- the 3D motion parameters



Rotating cylinder experiment
by S. Ullman (1981)

Cases of interest:

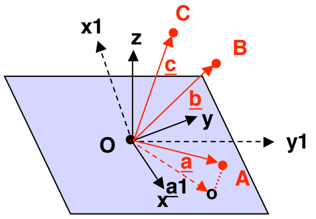
- stationary camera, moving object(s)
- moving camera, stationary object(s)
- moving camera, moving object(s)

camera motion parameters may be known

1

Structure from Motion (1)

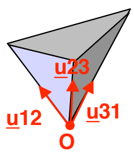
Ullman showed 1979 that the spatial structure of 4 rigidly connected non-coplanar points may be recovered from 3 orthographic projections.



projection plane Π_1

O, A, B, C	4 rigid points
a, b, c	vectors to A, B, C
Π_1, Π_2, Π_3	projection planes
xi, yi	coordinate axes of Π_i
ai, bi, ci	coordinate pairs of points A, B, C in projection plane Π_i

The problem is to determine the spatial orientations of Π_1, Π_2, Π_3 from the 9 projection coordinate pairs ai, bi, ci, i = 1, 2, 3.



The 3 projection planes intersect and form a tetrahedron.
u12, u23, u31 are unit vectors along the intersections.
The idea is to determine the uij from the observed coordinates ai, bi, ci.

O

2

Structure from Motion (2)

The projection coordinates are

$a1_x = \underline{a}^T\underline{x}1$ $a1_y = \underline{a}^T\underline{y}1$
$b1_x = \underline{b}^T\underline{x}1$ $b1_y = \underline{b}^T\underline{y}1$
$c1_x = \underline{c}^T\underline{x}1$ $c1_y = \underline{c}^T\underline{y}1$

Since each $\underline{u}ij$ lies in both planes Π_i and Π_j, it can be written as

$\underline{u}ij = \alpha_{ij}\underline{xi} + \beta_{ij}\underline{yi}$
$\underline{u}ij = \gamma_{ij}\underline{xj} + \delta_{ij}\underline{yj}$ $\Rightarrow$ $\alpha_{ij}\underline{xi} + \beta_{ij}\underline{yi} = \gamma_{ij}\underline{xj} + \delta_{ij}\underline{yj}$

Multiplying with $\underline{a}^T$, $\underline{b}^T$ and $\underline{c}^T$ we get

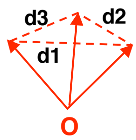
$\alpha_{ij}ai_x + \beta_{ij}ai_y = \gamma_{ij}aj_x + \delta_{ij}aj_y$
$\alpha_{ij}bi_x + \beta_{ij}bi_y = \gamma_{ij}bj_x + \delta_{ij}bj_y$
$\alpha_{ij}ci_x + \beta_{ij}ci_y = \gamma_{ij}cj_x + \delta_{ij}cj_y$

Exploiting the constraints $\alpha_{ij}^2 + \beta_{ij}^2 = 1$ and $\gamma_{ij}^2 + \delta_{ij}^2 = 1$, we can solve for α_{ij}, β_{ij}, γ_{ij}, δ_{ij}.

3

Structure from Motion (3)

From the coefficients α_{ij}, β_{ij}, γ_{ij}, δ_{ij} one can compute the distances between the 3 unit vectors $\underline{u}12$, $\underline{u}23$, $\underline{u}31$:



$d1 = \| \underline{u}23 - \underline{u}12 \| = \| (\alpha_{23} - \alpha_{12})\underline{xi} + (\beta_{23} - \beta_{12})\underline{yi} \| = (\alpha_{23} - \alpha_{12})^2 + (\beta_{23} - \beta_{12})^2$
$d2 = (\alpha_{31} - \alpha_{23})^2 + (\beta_{31} - \beta_{23})^2$
$d3 = (\alpha_{12} - \alpha_{31})^2 + (\beta_{12} - \beta_{31})^2$

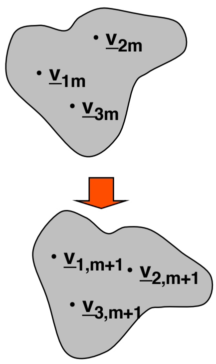
Hence the relative angles of the projection planes are determined.

The spatial positions of A, B, C relative to the projection planes (and to the origin O) can be determined by intersecting the projection rays perpendicular on the projected points $\underline{ai}$, $\underline{bi}$, $\underline{ci}$.

4

Perspective 3D Analysis of Point Displacements

- relative motion of one rigid object and one camera
- observation of P points in M views



For each point $\underline{v}_p$ in 2 consecutive images we have:

$\underline{v}_{p,m+1} = R_m \underline{v}_{pm} + \underline{t}_m$ motion equation

$\underline{v}_{pm} = \lambda_{pm} \underline{v}_{pm}{}'$ projection equation

For P points in M images we have

- 3MP unknown 3D point coordinates $\underline{v}_{pm}$
- 6(M-1) unkown motion parameters R_m and $\underline{t}_m$
- MP unknown projection parameters λ_{pm}
- 3(M-1)P motion equations
- 3MP projection equations
- 1 arbitrary scaling parameter

\# equations ≥ # unknowns => $P \geq 3 + \dfrac{2}{2M-3}$ =>

M	P
2	5
3	4
4	4
5	4

5

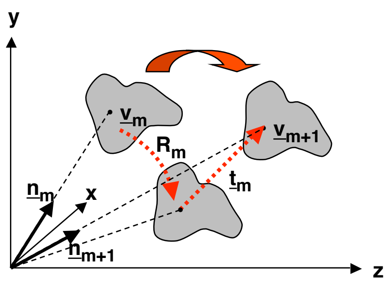
Essential Matrix

Geometrical constraints derived from 2 views of a point in motion

- motion between image m and m+1 may be decomposed into

  1) rotation R_m about origin of coordinate system (= optical center)

  2) translation $\underline{t}_m$
- observations are given by direction vectors $\underline{n}_m$ and $\underline{n}_{m+1}$ along projection rays



$R_m \underline{n}_m$, $\underline{t}_m$ and $\underline{n}_m$ are coplanar: $[\underline{t}_m \times R_m \underline{n}_m]^T \underline{n}_{m+1} = 0$

After some manipulation: $\boxed{\underline{n}_m{}^T E_m \underline{n}_{m+1} = 0 \quad E = \text{essential matrix}}$

with $E_m = \begin{bmatrix} | & | & | \\ \underline{t}_m x \underline{r}_1 & \underline{t}_m x \underline{r}_2 & \underline{t}_m x \underline{r}_3 \\ | & | & | \end{bmatrix}$ and $R_m = \begin{bmatrix} | & | & | \\ \underline{r}_1 & \underline{r}_2 & \underline{r}_3 \\ | & | & | \end{bmatrix}$

6

Solving for the Essential Matrix

$\underline{n}_m^T E_m \underline{n}_{m+1} = 0$ formally one equation for 9 unknowns e_{ij}

But: only 6 degrees of freedom
(3 rotation angles, 3 translation components)
e_{ij} can only be determined up to a scale factor

Basic solution approach:

- observe P points, alltogether in 2 views, P >> 8
- fix e_{11} arbitrarily
- solve an overconstrained system of equations for the other 8 unknown coefficients e_{ij}

E may be decomposed into S and R by Singular Value Decomposition (SVD).

E may be written as $E = S R^{-1}$ with R = rotation matrix and $S = \begin{bmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{bmatrix}$

Note: S (and therefore E) has rank 2

7

Singular Value Decomposition of E

Any m x n matrix A, m ≥ n, may be decomposed as $A = U D V^T$ where

U	has orthonormal columns	m x n
D	is non-negative diagonal	n x n
V^T	has orthonormal rows	n x n

This can be applied to E to give $E = U D V^T$ with

$R = U G V^T$ or $R = U G^T V^T$

$S = V Z V^T$

where $G = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ and $Z = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$

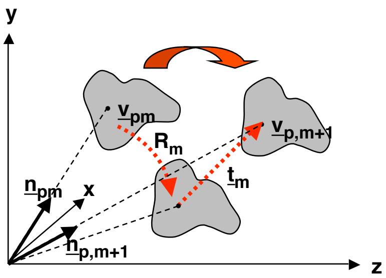
8

Nagel-Neumann Constraint

Consider 2 views of 3 points $\underline{v}_{pm}$, $p = 1 \dots 3$, $m = 1, 2$

The planes through $R_m \underline{n}_{pm}$ and $\underline{n}_{p,m+1}$ all intersect in $\underline{t}_m$

=> the normals of the planes are coplanar



Coplanarity condition for 3 vectors $\underline{a}$, $\underline{b}$, $\underline{c}$: $(\underline{a} \times \underline{b})^T \underline{c} = 0$

$$\boxed{\left([R_m \underline{n}_{1m} \times \underline{n}_{1,m+1}] \times [R_m \underline{n}_{2m} \times \underline{n}_{2,m+1}] \right)^T [R_m \underline{n}_{3m} \times \underline{n}_{3,m+1}] = 0}$$

Nonlinear equation with 3 unknown rotation parameters.
=> Observation of at least 5 points required to solve for the unknowns.

Reminder: Homogeneous Coordinates

- (N+1)-dimensional notation for points in N-dimensional Euclidean space
- allows to express projection and translation as linear operations

Normal coordinates: $\underline{v}^T = [x\ y\ z]$

Homogeneous coordinates: $\underline{v}^T = [wx\ wy\ wz\ w]$
$w \neq 0$ is arbitrary constant

Rotation and translation in homogeneous coordinates:

$\underline{v}' = A\underline{v}$ with $A = \begin{bmatrix} R & \underline{t} \\ \underline{0} & 1 \end{bmatrix}$

Projection in homogeneous coordinates:

$\underline{v}' = B\underline{v}$ with $B = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix}$

Divide the first N components by the (N+1)rst component to recover normal coordinates

From Homogeneous World Coordinates to Homogeneous Image Coordinates

x, y, z = $\underline{v}$ = scene coordinates

x_p'', y_p'' = $\underline{v}_p$ = image coordinates

$$\begin{bmatrix} wx_p'' \\ wy_p'' \\ w \end{bmatrix} = \begin{bmatrix} K R & K \underline{t} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = M \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \Rightarrow \underline{v}_p = M\,\underline{v}$$

$$K = \begin{bmatrix} fa & fb & x_{p0} \\ 0 & fc & y_{p0} \\ 0 & 0 & 1 \end{bmatrix}$$ intrinsic camera parameters ("camera calibration matrix K")

fa = scaling in x_P-axis
fc = scaling in y_P-axis
fb = slant of axes
x_{P0}, y_{P0} = "principal point" (optical center in image plane)

R, $\underline{t}$ extrinsic camera parameters

M = 3 x 4 projective matrix

Camera Calibration

Determine intrinsic and/or extrinsic camera parameters for a specific camera-scene configuration. Prior calibration may be needed

- to measure unknown objects
- to navigate as a moving observer
- to perform stereo analysis
- to compensate for camera distortions

Important cases:

1. Known scene

   Each image point corresponding with a known scene point provides an equation $\underline{v}_p = M\,\underline{v}$

2. Unknown scene

   Several views are needed, differing by rotation and/or translation

   a. Known camera motion

   b. Unknown camera motion ("camera self-calibration")

Calibration of One Camera from a Known Scene

- "known scene" = scene with prominent points, whose scene coordinates are known
- prominent points must be non-coplanar to avoid degeneracy

Projection equation $\underline{v}_p = M\,\underline{v}$ provides 2 linear equations for unknown coefficients of M:

$$x_p\,(m_{31}x + m_{32}y + m_{33}z + m_{34}) = m_{11}x + m_{12}y + m_{13}z + m_{14}$$
$$y_p\,(m_{31}x + m_{32}y + m_{33}z + m_{34}) = m_{21}x + m_{22}y + m_{23}z + m_{24}$$

Taking N points, N > 6, M can be estimated with a least-square method from an overdetermined system of 2N linear equations.

From M = [KR K$\underline{t}$] = [A $\underline{b}$], one gets K and R by Principle Component Analysis (PCA) of A and $\underline{t}$ from $\underline{t} = K^{-1}\underline{b}$.

13

Fundamental Matrix

The fundamental matrix F generalizes the essential matrix E by incorporating the intrinsic camera parameters of two (possibly different) cameras.

Essential matrix constraint for 2 views of a point:

$$\underline{n}^T\, E\, \underline{n}' = 0$$

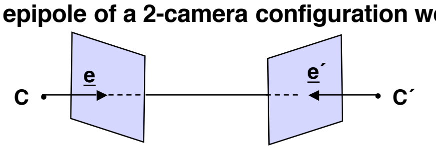
From $\underline{v}_p = K\alpha\,\underline{n}$ and $\underline{v}_p' = K'\beta\,\underline{n}'$ we get:

$$K = \begin{bmatrix} fa & fb & x_{p0} \\ 0 & fc & y_{p0} \\ 0 & 0 & 1 \end{bmatrix}$$

$$\boxed{\underline{v}_p\,(K^{-1})^T\, E\, (K')^{-1}\, \underline{v}_p' = \underline{v}_p\, F\, \underline{v}_p' = 0}$$

Note that E and hence F have rank 2.

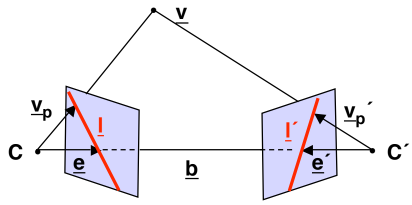
For each epipole of a 2-camera configuration we have $\underline{e}^T F = 0$ and $F\underline{e}' = 0$.



14

Epipolar Plane

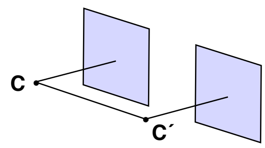
The epipolar plane is spanned by the projection rays of a point $\underline{v}$ and the baseline CC' of a stereo camera configuration.



The epipoles $\underline{e}$ and $\underline{e}'$ are the intersection points of the baseline with the image planes. The epipolar line $\underline{l}$ and $\underline{l}'$ mark the intersections of the epipolar plane in the left and right image, respectively.

> Search for corresponding points in stereo images may be restricted to the epipolar lines.

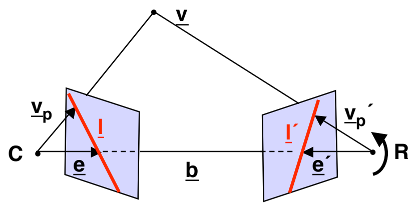
In a canonical stereo configuration (optical axes parallel and perpendicular to baseline) all epipolar lines are parallel:



15

Algebra of Epipolar Geometry

Observation $\underline{v}_p'$ can be modelled as a second observation after translation $\underline{b}$ and rotation R of the optical system.



Coplanarity of $\underline{v}_p$, $\underline{b}$ and $\underline{v}_p'$ (rotated back into coo-system at C) can be expressed as

$$\underline{v}_p [\, \underline{b} \times R\underline{v}_p' \,] = 0 = \underline{v}_p [\, \underline{b} \,] R\underline{v}_p' = \underline{v}_p\, E\, \underline{v}_p'$$

essential matrix

> A vector product $\underline{c} \times \underline{d}$ can be written in matrix form:
>
> $$\underline{c} \times \underline{d} = \begin{bmatrix} c_y d_z - c_z d_y \\ c_z d_x - c_x d_z \\ c_x d_y - c_y d_x \end{bmatrix} = \begin{bmatrix} 0 & -c_z & c_y \\ c_z & 0 & -c_x \\ -c_y & c_x & 0 \end{bmatrix} \begin{bmatrix} d_x \\ d_y \\ d_z \end{bmatrix} = [\, c \,]\, \underline{d}$$

16

Correspondence Problem Revisited

For multiple-view 3D analysis it is essential to find corresponding images of a scene point - the correspondence problem.

Difficulties:

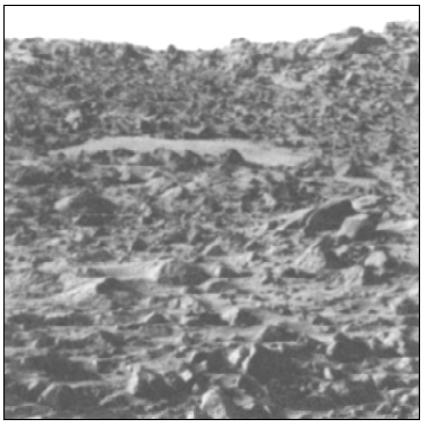
- **scene may not offer enough structure to uniquely locate points**
- **scene may offer too much structure to uniquely locate points**
- **geometric features may differ strongly between views**
- **there may be no corresponding point because of occlusion**
- **photometric features differ strongly between views**

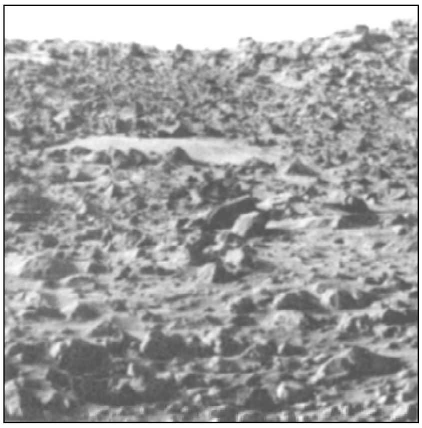
Note that difficulties apply to multiple-camera 3D analysis (e.g. binocular stereo) as well as single-camera motion analysis.

17

Correspondence Between Two Mars Images

Two images taken from two cameras of the Viking Lander I (1978). Disparities change rapidly moving from the horizon to nearby structures. (From B.K.P. Horn, Robot Vision, 1986)





18

Constraining Search for Correspondence

The ambiguity of correspondence search may be reduced by several (partly heuristic) constraints.

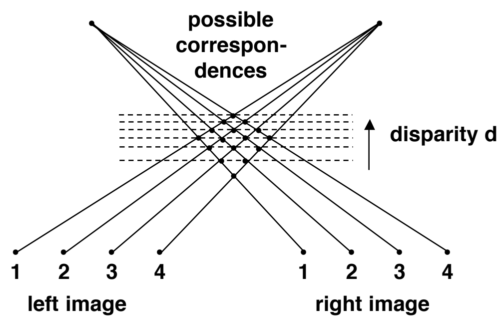
- **Epipolar constraint**
  reduces search space from 2D to 1D
- **Uniqueness constraint**
  a pixel in one image can correspond to only one pixel in another image
- **Photometric similarity constraint**
  intensities of a point in different images may differ only a little
- **Geometric similarity constraint**
  geometric features of a point in different images may differ only a little
- **Disparity smoothness constraint**
  disparity varies only slowly almost everywhere in the image
- **Physical origin constraint**
  points may correspond only if they mark the same physical location
- **Disparity limit constraint**
  in humans disparity must be smaller than a limit to fuse images
- **Ordering constraint**
  corresponding points lie in the same order on the epipolar line
- **Mutual correspondence constraint**
  correspondence search must succeed irrespective of order of images

19

Neural Stereo Computation

Neural-network inspired approach to stereo computation devised by Marr and Poggio (1981)

Exploitation of 2 constraints:
- each point in the left image corresponds only to one point in the right image
- depth varies smoothly



Relaxation procedure:
Modify correspondence values c(x, y, d) interatively until values converge.

$$c_{n+1}(x,y,d) = w_1 \sum_{S_1} c_n(x',y',d') - w_2 \sum_{S_2} c_n(x',y',d') + w_0 c_0(x,y,d)$$

S_1 = { neighbours of (x, y) with $d' = d$ }
S_2 = { neighbours of (x, y) with $|d' - d| = 1$ and $(x, y) = (x', y')$ }

20

Obtaining 3D Shape from Shading Information

Under certain conditions, a 3D surface model may be reconstructed from the greyvalue variations of a monocular image.

From "Shape from Shading",
B.K.P. Horn and M.J. Brooks (eds.),
MIT Press 1989

21

Principle of Shape from Shading

See "Shape from Shading" (B.K.P. Horn, M.J. Brooks, eds.), MIT Press 1989

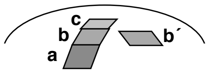
Physical surface properties, surface orientation, illumination and viewing direction determine the greyvalue of a surface patch in a sensor signal.

For a single object surface viewed in one image, greyvalue changes are mainly caused by surface orientation changes.

The reconstruction of arbitrary surface shapes is not possible because different surface orientations may give rise to identical greyvalues.

Surface shapes may be uniquely reconstructed from shading information if possible surface shapes are constrained by smoothness assumptions.

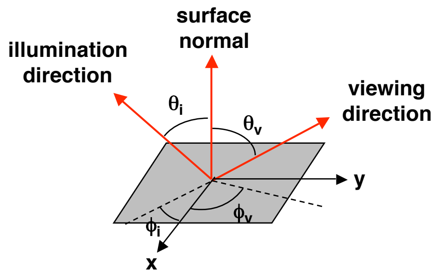
Principle of incremental procedure for surface shape reconstruction:



a: patch with known orientation
b, c: neighbouring patches with similar orientations
b´: radical different orientation may not be neighbour of a

22

Photometric Surface Properties



θ_i, θ_v polar (zenith) angles

ϕ_i, ϕ_v azimuth angles

In general, the ability of a surface to reflect light is given by the Bi-directional Reflectance Distribution Function (BRDF) r:

$$r(\theta_i, \phi_i; \theta_v, \phi_v) = \frac{\delta L(\theta_v, \phi_v)}{\delta E(\theta_i, \phi_i)}$$

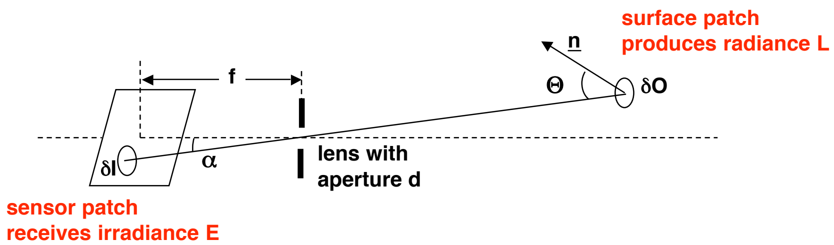
δE = irradiance of light source received by the surface patch

δL = radiance of surface patch towards viewer

For many materials the reflectance properties are rotation invariant, in this case the BRDF depends on θ_i, θ_v, ϕ, where $\phi = \phi_i - \phi_v$.

23

Irradiance of Imaging Device



irradiance = light energy falling on unit patch of imaging sensor, sensor signal is proportional to irradiance

$$E = L\frac{\pi}{4}\left(\frac{d}{f}\right)^2 \cos^4\alpha$$

sensor signal depends on span-off angle α of surface element ("vignetting")

off-center pixels in wide-angle images are darker

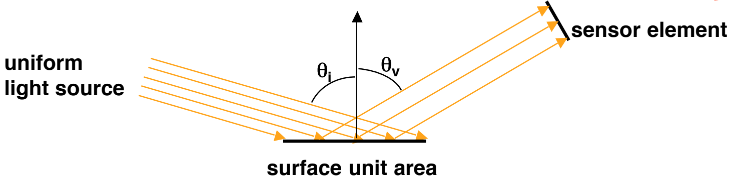
24

Lambertian Surfaces

A Lambertian surface is an ideally matte surface which looks equally bright from all viewing directions under uniform or collimated illumination. Its brightness is proportional to the cosine of the illumination angle.

- surface receives energy per unit area $\sim \cos\theta_i$
- surface reflects energy $\sim \cos\theta_v$ due to matte reflectance properties
- sensor element receives energy from surface area $\sim 1/\cos\theta_v$

} cancel out



$r_{Lambert}(\theta_i, \theta_v, \phi) = \rho(\lambda)/\pi$

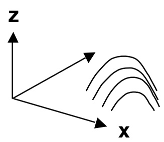
$$\rho(\lambda) = \frac{\int_\Omega L\,\partial\Omega}{E_i}$$

"albedo" = proportion of incident energy reflected back into half space Ω above surface

25

Surface Gradients

For 3D reconstruction of surfaces, it is useful to represent reflectance properties as a function of surface orientation.



z(x, y)	surface
$p = \delta z/\delta x$	x-component of surface gradient
$q = \delta z/\delta y$	y-component of surface gradient

$\begin{bmatrix}1\\0\\p\end{bmatrix}$ tangent vector in x direction $\quad$ $\begin{bmatrix}0\\1\\q\end{bmatrix}$ tangent vector in y direction $\quad$ $\begin{bmatrix}-p\\-q\\1\end{bmatrix}$ vector in surface normal direction

$$\underline{n} = \frac{1}{\sqrt{1+p^2+q^2}}\begin{bmatrix}-p\\-q\\1\end{bmatrix}$$ surface normal vector

If the z-axis is chosen to coincide with the viewer direction, we have

$$\cos\theta_v = \frac{1}{\sqrt{1+p^2+q^2}} \qquad \cos\theta_i = \frac{1+p_ip+q_iq}{\sqrt{1+p^2+q^2}\sqrt{1+p_i^2+q_i^2}} \qquad \cos\varphi = \frac{1}{\sqrt{1+p_i^2+q_i^2}}$$

The dependency of the BRDF on θ_i, θ_v and ϕ may be expressed in terms of p and q (with p_i and q_i for the light source direction).

26

Simplified Image Irradiance Equation

Assume that

- the object has uniform reflecting properties,
- the light sources are distant so that the irradiation is approximately constant and equally oriented,
- the viewer is distant so that the received radiance does not depend on the distance but only on the orientation towards the surface.

With these simplifications the sensor greyvalues depend only on the surface gradient components p and q.

$$E(x,y) = R(p(x,y),q(x,y)) = R\left(\frac{\partial z}{\partial x},\frac{\partial z}{\partial y}\right)$$

"Simplified Image Irradiance Equation"

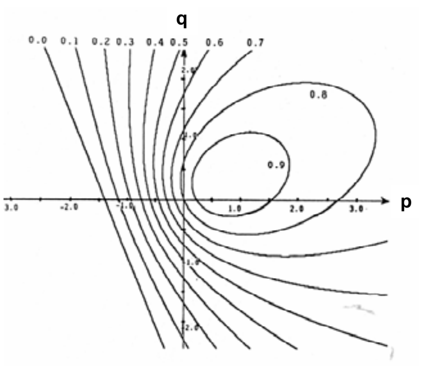
R(p, q) is the reflectance function for a particular illumination geometry. E(x, y) is the sensor greyvalue measured at (x, y). Based on this equation and a smoothness constraint, shape-from-shading methods recover surface orientations.

27

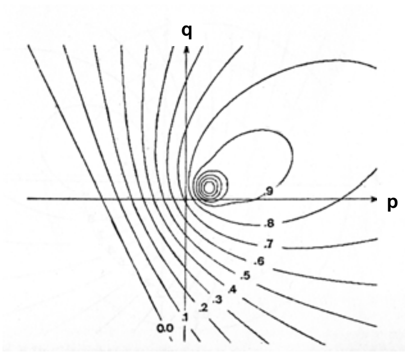
Reflectance Maps

R(p, q) may be plotted as a reflectance map with iso-brightness contours.

Reflectance map for Lambertian surface illuminated from $p_i = 0.7$ and $q_i = 0.3$



Reflectance map for matte surface with specular component



28

Characteristic Strip Method

Given a surface point (x, y, z) with known height z, orientation p and q, and second derivatives $r = z_{xx}$, $s = z_{xy} = z_{yx}$, $t = z_{yy}$, the height z+δz and orientation p+δp, q+δq in a neighbourhood x+δx, y+δy can be calculated from the image irradiance equation E(x, y) = R(p, q).

Infinitesimal change of height:

$$\delta z = p\,\delta x + q\,\delta y$$

Changes of p and q for a step δx, δy:

$$\delta p = r\,\delta x + s\,\delta y \qquad \delta q = s\,\delta x + t\,\delta y$$

Differentiation of image irradiance equation w.r.t. x and y gives

$$E_x = r\,R_p + s\,R_q \qquad E_y = s\,R_p + t\,R_q$$

Choose a step δξ in the direction of steepest surface descent ("characteristic strip"):

$$\delta x = R_p\,\delta\xi \qquad \delta y = R_q\,\delta\xi$$

For this direction the image irradiance equation can be replaced by

$$\delta x/\delta\xi = R_p \qquad \delta y/\delta\xi = R_q \qquad \delta z/\delta\xi = p\,R_p + q\,R_q \qquad \delta p/\delta\xi = E_x \qquad \delta q/\delta\xi = E_y$$

Boundary conditions and initial points may be given by
- occluding contours with surface normal perpendicular to viewing direction
- singular points with surface normal towards light source.

Shape from Shading by Global Optimization

Given a monocular image and a known image irradiance equation, surface orientations are ambiguously constrained. Disambiguation may be achieved by optimizing a global smoothness criterion.

Minimize

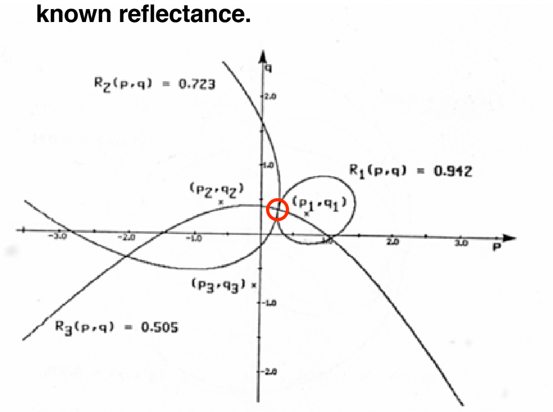
$$D(x,y) = \underbrace{\left[E(x,y) - R(p,q)\right]^2}_{\text{violation of reflectance constraint}} + \underset{\text{Lagrange multiplier}}{\lambda} \underbrace{\left[\left(\nabla^2 p\right)^2 + \left(\nabla^2 q\right)^2\right]}_{\text{violation of smoothness constraint}}$$

There exist standard techniques for solving this minimization problem iteratively. In general, the solution may not be unique.

Due to several uncertain assumptions (illumination, reflectance function, smoothness of surface) solutions may not be reliable.

Principle of Photometric Stereo

In photometric stereo, several images with different known light source orientations are used to uniquely recover 3D orientation of a surface with known reflectance.



- The reflectance maps $R_1(p, q)$, $R_2(p, q)$, $R_3(p, q)$ specify the possible surface orientations of each pixel in terms of iso-brightness contours ("isophotes").
- The intersection of the isophotes corresponding to the 3 brightness values measured for a pixel (x, y) uniquely determines the surface orientation (p(x, y), q(x, y)).

From "Shape from Shading", B.K.P. Horn and M.J. Brooks (eds.), MIT Press 1989

31

Analytical Solution for Photometric Stereo

For a Lambertian surface:

$E(x, y) = R(p, q) = \rho \cos(\theta_i) = \rho\, \underline{i}^T \underline{n}$

$\underline{i}$ = light source direction, $\underline{n}$ = surface normal, ρ = constant

If K images are taken with K different light sources $\underline{i}_k$, k = 1 ... K, there are K brightness measurements E_k for each image position (x, y):

$E_k(x, y) = \rho\, \underline{i}_k^T \underline{n}$

In matrix notation:

$\underline{E}(x, y) = \rho\, L\, \underline{n}$ where $L = \begin{bmatrix} \underline{i}_1^T \\ \vdots \\ \underline{i}_K^T \end{bmatrix}$

For K=3, L may be inverted, hence

$$\underline{n}(x,y) = \frac{L^{-1}\underline{E}(x,y)}{\left\| L^{-1}\underline{E}(x,y) \right\|}$$

In general, the pseudo-inverse must be computed:

$$\underline{n}(x,y) = \frac{\left(L^T L\right)^{-1} L^T \underline{E}(x,y)}{\left\| \left(L^T L\right)^{-1} L^T \underline{E}(x,y) \right\|}$$

32