E1 216 COMPUTER VISION

LECTURE 04: APPLICATIONS OF RADIOMETRY

Venu Madhav Govindu
Department of Electrical Engineering
Indian Institute of Science, Bengaluru

2026

Image Formation

Last class we looked at the **radiometry** of image formation
In this lecture we shall look at applications of radiometric principles

Applications

We shall look at

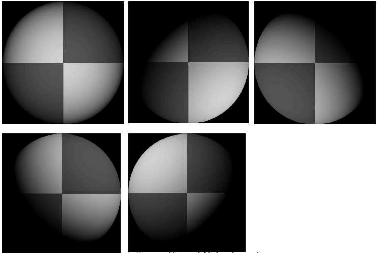
- **Photometric Stereo** : 3D Shape using multiple light sources
- Will not discuss **Illumination Cones**
    - Under assumptions images of objects lie in low-dim space
    - Can explain many observations
    - Build models for recognition (independent of lighting)
- Will not discuss **Shape from Shading** (3D from single image)
- Major recent development: **Neural Radiance Fields (NeRF)**

Applications : Photometric Stereo

- Assume a Lambertian model
- Fix camera and object position
- Vary illumination using point source
- Developed by Woodham

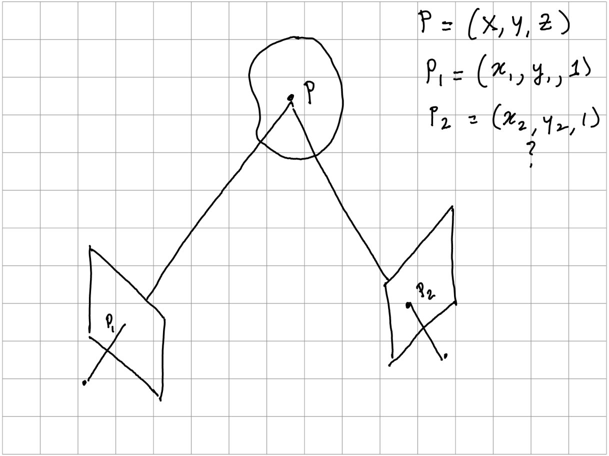
$$
\begin{aligned}
B(P) &= \rho(P)\boldsymbol{N}(P).\boldsymbol{V}_1 \\
I(x,y) &= kB(x,y) \\
&= k\rho(x,y)\boldsymbol{N}(x,y).\boldsymbol{V}_1 \\
&= \boldsymbol{g}(x,y).\boldsymbol{V}_1
\end{aligned}
$$

Applications : Photometric Stereo

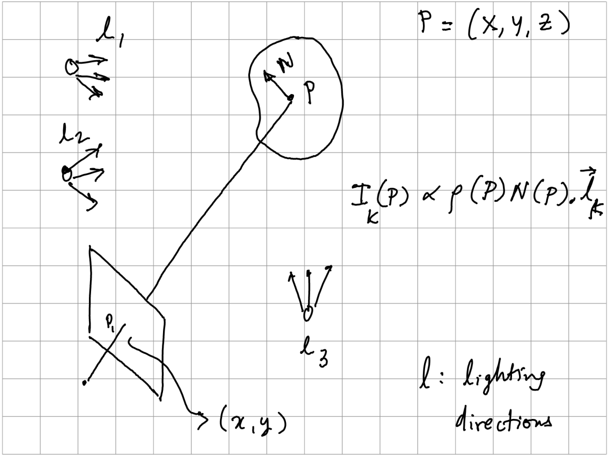


$$
\begin{aligned}
\mathcal{V} &= \begin{pmatrix} \boldsymbol{V_1}^T \\ \boldsymbol{V_2}^T \\ \vdots \\ \boldsymbol{V_n}^T \end{pmatrix} \\
i(x,y) &= \{I_1(x,y), \cdots, I_n(x,y)\}^T \\
\boldsymbol{i}(x,y) &= \mathcal{V}\boldsymbol{g}(x,y)
\end{aligned}
$$

Triangulation



Photometric Stereo

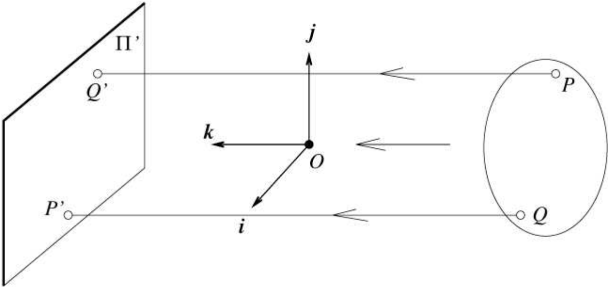


Photometric Stereo

Strong Assumptions

- Assumptions used in original method
- Orthographic Camera
- Lambertian surface
- Lighting is
  - point source
  - at infinity
- Calibrated Light Sources (known direction)
- Newer methods overcome/relax some or all assumptions
- Depth + Photometric + Learning

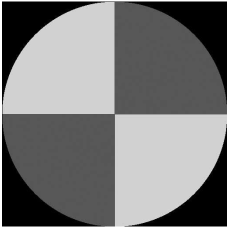
Camera Model : Orthographic Projection

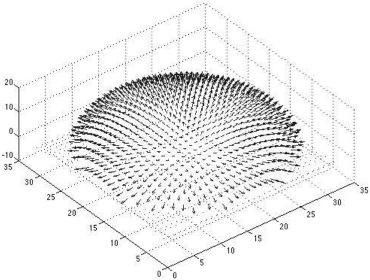


Orthographic Projection Model

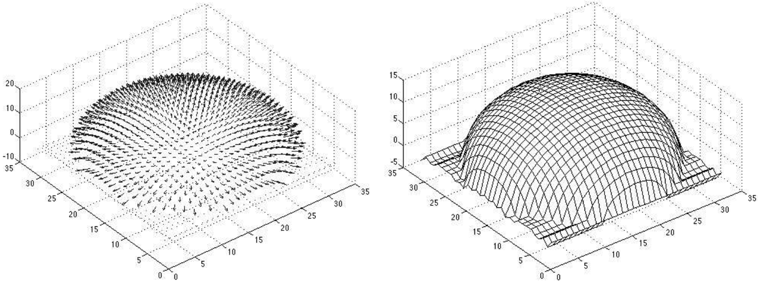
- Projects object points onto plane along direction of optical axis
- Also known as parallel projection
- Drastic approximation $\boldsymbol{x} = \boldsymbol{X}$! (not $\frac{f\boldsymbol{X}}{\boldsymbol{Z}}$)
- Useful as linear methods are applicable

Applications : Photometric Stereo





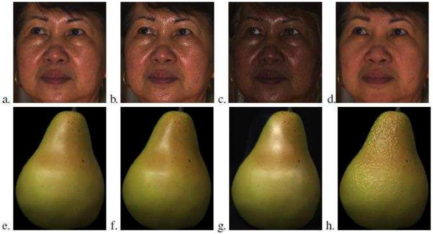
- Take care of shadows
- Albedo can be measured since $||\boldsymbol{N}|| = 1$
- $\boldsymbol{N}(x, y) = \frac{1}{||\boldsymbol{g}(x,y)||} \boldsymbol{g}(x, y)$
- Offers a check for correctness



Height map recovered by careful integration of normal field

Specular Surfaces

- Traditional methods have treated specularities as a problem
- Specularities are much harder to model correctly
- Model a surface as specular + diffuse component
- New methods try to separate out specular and diffuse component
- We shall not study specularities in this course



Can manipulate specular and diffuse components independently

Radiometry

- General radiometry is very complex
- Many simplifying assumptions can be made
- Radiometric ideas applied to both
  - shape reconstruction
  - recognition
- Classical approach to shape is **shape from shading**
- Newer approaches combine depth and photometric measurements
- Recent developments using deep learning
- Developments in rendering: Neural Radiance Fields (NeRF)

In the next few slides, we'll do a bit of elementary linear algebra and least squares estimation

Linear Estimation

- Linear models are
  - simple
  - powerful
  - easy to interpret
- Spectral representation (eigen decomposition)
- Many efficient solvers available
- Exploit sparsity for large-scale systems

$$A_{m,n} = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \cdots & a_{m,n} \end{pmatrix}$$

Row Space

$$A_{m,n} = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \cdots & a_{m,n} \end{pmatrix}$$

Column Space

Vector Space Representations

- $\boldsymbol{Ax} = \lambda \boldsymbol{x}$ (column space, image/range space)
- $\boldsymbol{A}^T \boldsymbol{x} = \lambda' \boldsymbol{x}$ (row space)
- SVD: Unified spectral representation

Linear Least Squares

- $\boldsymbol{Ax} = \mathbf{0}$
- Rank of $\boldsymbol{A}$?
- Scale of $\boldsymbol{x}$? Usually fix $||\boldsymbol{x}|| = 1$
- Rank of $\boldsymbol{A} + \epsilon \boldsymbol{N}$?
- No exact fit for noisy obs
- Best solution: **least squares** (why?)
- Minimise $C = ||\boldsymbol{Ax}||^2$
- $||\boldsymbol{Ax}||^2 = (\boldsymbol{Ax})^T(\boldsymbol{Ax})$
- Assume $\boldsymbol{x}$ is an eigen vector, what is C?
- Why should solution $\boldsymbol{x}$ be an eigen vector?
- Properties of eigen vectors of $\boldsymbol{A}^T\boldsymbol{A}$?
- Good exercise to prove $\boldsymbol{x}$ is smallest eigen vector
- **Hint:** Write $\boldsymbol{x} = \Sigma_i \alpha_i \boldsymbol{v}_i$ (for eigen vectors $\boldsymbol{v}$)

Linear Least Squares

$$\begin{aligned} \boldsymbol{x} &= \arg\min_{x} ||\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}||^2 \\ &= \arg\min_{x} (\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b})^T (\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}) \end{aligned}$$

Taking derivative wrt $\boldsymbol{x}$ etc.

$$\boldsymbol{x} = (\boldsymbol{A}^T \boldsymbol{A})^{-1} \boldsymbol{A}^T \boldsymbol{b} \text{ (check)}$$

- $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$
- Existence of solutions?
- Least squares for over-determined soe
- Minimise $C = ||\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}||^2$
- Scale of $\boldsymbol{x}$?
- $\boldsymbol{A}^\dagger = (\boldsymbol{A}^T \boldsymbol{A})^{-1} \boldsymbol{A}^T$ (pseudo-inverse)
- When $\boldsymbol{A}$ is square? rank-deficient?
- **Warning:** Never solve directly! Why?

Singular Value Decomposition

- Powerful result used extensively in many areas
- Generalises spectral theorem to non-square matrices
- Provides a good interpretation of matrix properties
- Very useful approach to solving linear problems in vision
- Will repeatedly encounter the SVD in many approaches
- Stable algorithms exists, eg. matlab svd()

Singular Value Decomposition

For an $m \times n$ matrix $\boldsymbol{A}$, there exists a factorisation

$$\boldsymbol{A} = \boldsymbol{U}\Sigma\boldsymbol{V}^{\star}$$

- $\boldsymbol{U}$ and $\boldsymbol{V}$ are unitary of size $m \times m$ and $n \times n$ resp.
- $\boldsymbol{V}^{\star}$ denotes the Hermitian of $\boldsymbol{V}$
- Unitary matrices are such that $\boldsymbol{U}\boldsymbol{U}^{\star} = \boldsymbol{U}^{\star}\boldsymbol{U} = \boldsymbol{I}$
- SVD is a generalisation of "eigendecomposition"
- $\boldsymbol{U}$ and $\boldsymbol{V}$ denote the eigenvectors of $\boldsymbol{A}\boldsymbol{A}^{\star}$ and $\boldsymbol{A}^{\star}\boldsymbol{A}$
- Σ is a diagonal matrix containing the singular values of $\boldsymbol{A}$
- Decomposition such that singular values are ordered

$$\Sigma = diag(\sigma_1, \sigma_2, \cdots, \sigma_k), \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_k$$

Low Rank Approximation

$$\begin{aligned} \boldsymbol{A} &= \boldsymbol{U}\Sigma\boldsymbol{V}^T \\ &= \sum_i \sigma_i u_i v_i^T \end{aligned}$$

Eckart-Young-Mirsky Theorem

- Say we want $\boldsymbol{A}_k$ s. t. $rank(\boldsymbol{A}_k) = k < rank(\boldsymbol{A})$
- How do we pick $\boldsymbol{A}_k$?
- $\boldsymbol{A}_k$ closest to $\boldsymbol{A}$
- Minimise $||\boldsymbol{A} - \boldsymbol{A}_k||_F$
- Solution: $\boldsymbol{A}_k = \sum_{i=1}^{k} \sigma_i u_i v_i^T$
- Very commonly used approach
- Each $u_i v_i^T$ is rank-1
- Like peeling an onion
- $\boldsymbol{A} = \sigma_1 u_1 v_1^T + (\sigma_2 u_2 v_2^T + (\sigma_3 u_3 v_3^T + \cdots))$

Low Rank Approximation

$$\begin{aligned} \boldsymbol{A} &= \boldsymbol{U}\Sigma\boldsymbol{V}^T \\ &= \sum_i \sigma_i u_i v_i^T \end{aligned}$$

Eckart-Young-Mirsky Theorem

- Say we want $\boldsymbol{A}_k$ s. t. $rank(\boldsymbol{A}_k) = k < rank(\boldsymbol{A})$
- How do we pick $\boldsymbol{A}_k$?
- $\boldsymbol{A}_k$ closest to $\boldsymbol{A}$
- Minimise $||\boldsymbol{A} - \boldsymbol{A}_k||_F$
- Solution: $\boldsymbol{A}_k = \sum_{i=1}^{k} \sigma_i u_i v_i^T$
- Very commonly used approach
- Each $u_i v_i^T$ is rank-1
- Like peeling an onion
- $\boldsymbol{A} = \sigma_1 u_1 v_1^T + (\sigma_2 u_2 v_2^T + (\sigma_3 u_3 v_3^T + \cdots))$

Low Rank Approximation

$$\begin{aligned} \boldsymbol{A} &= \boldsymbol{U}\Sigma\boldsymbol{V}^T \\ &= \sum_i \sigma_i u_i v_i^T \end{aligned}$$

Eckart-Young-Mirsky Theorem

- Say we want $\boldsymbol{A}_k$ s. t. $rank(\boldsymbol{A}_k) = k < rank(\boldsymbol{A})$
- How do we pick $\boldsymbol{A}_k$?
- $\boldsymbol{A}_k$ closest to $\boldsymbol{A}$
- Minimise $||\boldsymbol{A} - \boldsymbol{A}_k||_F$
- Solution: $\boldsymbol{A}_k = \sum_{i=1}^{k} \sigma_i u_i v_i^T$
- Very commonly used approach
- Each $u_i v_i^T$ is rank-1
- Like peeling an onion
- $\boldsymbol{A} = \sigma_1 u_1 v_1^T + (\sigma_2 u_2 v_2^T + (\sigma_3 u_3 v_3^T + \cdots))$

Low Rank Approximation

$$\begin{aligned} \boldsymbol{A} &= \boldsymbol{U}\Sigma\boldsymbol{V}^T \\ &= \sum_i \sigma_i u_i v_i^T \end{aligned}$$

Eckart-Young-Mirsky Theorem

- Say we want $\boldsymbol{A}_k$ s. t. $rank(\boldsymbol{A}_k) = k < rank(\boldsymbol{A})$
- How do we pick $\boldsymbol{A}_k$?
- $\boldsymbol{A}_k$ closest to $\boldsymbol{A}$
- Minimise $||\boldsymbol{A} - \boldsymbol{A}_k||_F$
- Solution: $\boldsymbol{A}_k = \sum_{i=1}^{k} \sigma_i u_i v_i^T$
- Very commonly used approach
- Each $u_i v_i^T$ is rank-1
- Like peeling an onion
- $\boldsymbol{A} = \sigma_1 u_1 v_1^T + (\sigma_2 u_2 v_2^T + (\sigma_3 u_3 v_3^T + \cdots))$

Low Rank Approximation

$$\begin{aligned} \boldsymbol{A} &= \boldsymbol{U}\Sigma\boldsymbol{V}^T \\ &= \sum_i \sigma_i u_i v_i^T \end{aligned}$$

Eckart-Young-Mirsky Theorem

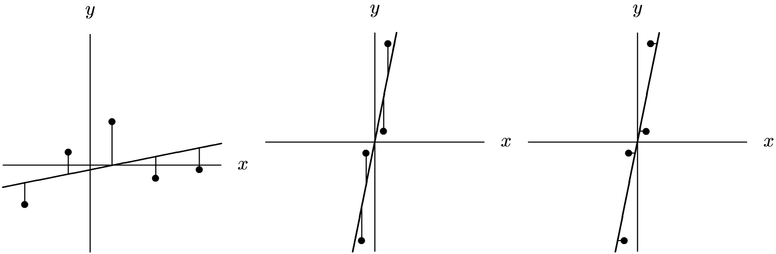
- Say we want $\boldsymbol{A}_k$ s. t. $rank(\boldsymbol{A}_k) = k < rank(\boldsymbol{A})$
- How do we pick $\boldsymbol{A}_k$?
- $\boldsymbol{A}_k$ closest to $\boldsymbol{A}$
- Minimise $||\boldsymbol{A} - \boldsymbol{A}_k||_F$
- Solution: $\boldsymbol{A}_k = \sum_{i=1}^{k} \sigma_i u_i v_i^T$
- Very commonly used approach
- Each $u_i v_i^T$ is rank-1
- Like peeling an onion
- $\boldsymbol{A} = \sigma_1 u_1 v_1^T + (\sigma_2 u_2 v_2^T + (\sigma_3 u_3 v_3^T + \cdots))$

Low Rank Approximation

$$\begin{aligned} \boldsymbol{A} &= \boldsymbol{U}\Sigma\boldsymbol{V}^T \\ &= \sum_i \sigma_i u_i v_i^T \end{aligned}$$

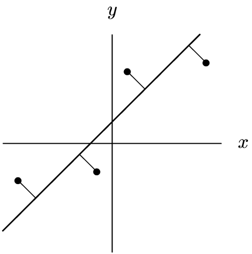
Eckart-Young-Mirsky Theorem

- Say we want $\boldsymbol{A}_k$ s. t. $rank(\boldsymbol{A}_k) = k < rank(\boldsymbol{A})$
- How do we pick $\boldsymbol{A}_k$?
- $\boldsymbol{A}_k$ closest to $\boldsymbol{A}$
- Minimise $||\boldsymbol{A} - \boldsymbol{A}_k||_F$
- Solution: $\boldsymbol{A}_k = \sum_{i=1}^{k} \sigma_i u_i v_i^T$
- Very commonly used approach
- Each $u_i v_i^T$ is rank-1
- Like peeling an onion
- $\boldsymbol{A} = \sigma_1 u_1 v_1^T + (\sigma_2 u_2 v_2^T + (\sigma_3 u_3 v_3^T + \cdots))$

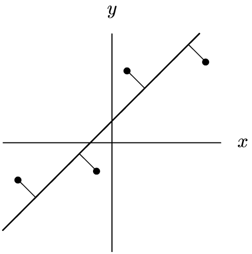


Algebraic Line Fitting

- $y_i = mx_i + c$
- Cost to minimise : $\sum_i (y_i - mx_i - c)^2$
- Familiar slope-intercept form
- Lacks geometric invariance
- Can be highly unstable



- Error measured perp. to fitted line
- Error is geometrically invariant to choice of co-ordinates
- Cost to minimise : $\frac{\sum_i (y_i - m x_i - c)^2}{1+m^2}$
- Easier to consider form : $a y_i + b x_i + c = 0$



Cost :

$\sum_i (ay_i + bx_i + c)^2 \Rightarrow \sum_i \{(y_i, x_i, 1).(a, b, c)\}^2$

Consider form :

$$\underbrace{\begin{pmatrix} y_1 & x_1 & 1 \\ y_2 & x_2 & 1 \\ & \cdots & \\ y_N & x_N & 1 \end{pmatrix}}_{\boldsymbol{M}} \overbrace{\begin{bmatrix} a \\ b \\ c \end{bmatrix}}^{\boldsymbol{p}} = 0$$

- Problem equivalent to $\boldsymbol{Mp} = \mathbf{0}$
- Boils down to $\min_{\boldsymbol{p}} ||\boldsymbol{Mp}||^2$
- Determine null-space using SVD
- Solution is rotationally invariant

Least Squares Formulations

$$\boldsymbol{Ax} = \boldsymbol{b}$$

Ordinary Least Squares

$$\begin{aligned} \boldsymbol{Ax} &= \boldsymbol{b} + \Delta\boldsymbol{b} \\ \arg\min_{\boldsymbol{x}} & \ ||\boldsymbol{Ax} - \boldsymbol{b}||^2 \end{aligned}$$

- Assume error only in rhs $\boldsymbol{b}$
- $\boldsymbol{x} = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{b}$
- Familiar pseudo-inverse form
- Biased, low variance

Total Least Squares

$$\begin{aligned} (\boldsymbol{A} + \Delta\boldsymbol{A})\boldsymbol{x} &= \boldsymbol{b} + \Delta\boldsymbol{b} \\ \Rightarrow [\boldsymbol{A}| - \boldsymbol{b}] \begin{bmatrix} \boldsymbol{x} \\ \hline 1 \end{bmatrix} &= 0 \end{aligned}$$

- Error in both lhs and rhs
- Solve using the ‘homogenous’ form
- Solution provided by SVD
- Unbiased, high variance

Condition number of a matrix

Vector Norm

- Consider vector norm $||\boldsymbol{x}||$
- Triangle inequality: $||\boldsymbol{x} + \boldsymbol{y}|| \leq ||\boldsymbol{x}|| + ||\boldsymbol{y}||$
- What do these norms mean?
  - ℓ_2
  - ℓ_1
  - ℓ_∞
  - $\ell_p (0 < p < 1)$

Condition number of a matrix

Matrix Norm

- Induce a matrix norm for $\boldsymbol{A}$
- $||\boldsymbol{A}|| = \sup_{||\boldsymbol{x}||\neq 0} \frac{||\boldsymbol{A}\boldsymbol{x}||}{||\boldsymbol{x}||}$
- Is $||\boldsymbol{A} + \boldsymbol{B}|| < ||\boldsymbol{A}|| + ||\boldsymbol{B}||$?
- For *some* matrix norms $||\boldsymbol{A}\boldsymbol{B}|| < ||\boldsymbol{A}||||\boldsymbol{B}||$
- Also elementwise norms etc. (Frobenius)

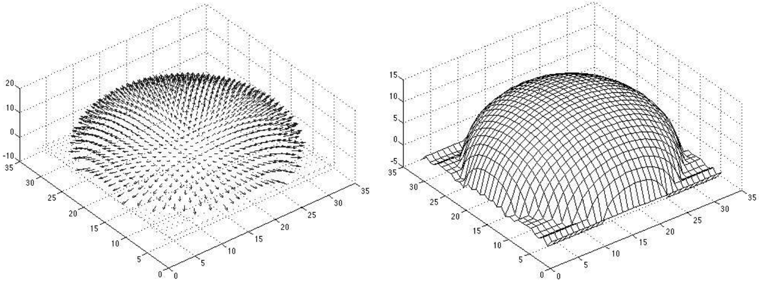
Condition number of a matrix

Condition number

- Condition number $\kappa(\boldsymbol{A})$ measures "sensitivity"
- How do quantities change when we perturb measurements?
- Consider $\boldsymbol{A}^{-1}$ and $\boldsymbol{Ax} = \boldsymbol{b}$
- Finite-precision computations
- Unstable solutions for large $\kappa(\boldsymbol{A})$
- $\kappa(\boldsymbol{A})$ for singular matrices is ∞
- For ℓ_2 norm $\kappa(\boldsymbol{A}) = \frac{\sigma_{max}(\boldsymbol{A})}{\sigma_{min}(\boldsymbol{A})}$
- Lesson: Algebraic rank is not sufficient for stable computation

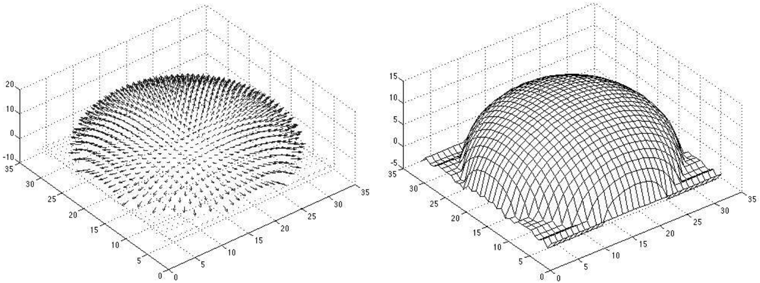
The following slides sketch out methods for integrating normals to get surfaces.
We will not be covering this material this semester.
You are not required to know this material, but I've left it here if anyone is interested.

Applications : Photometric Stereo



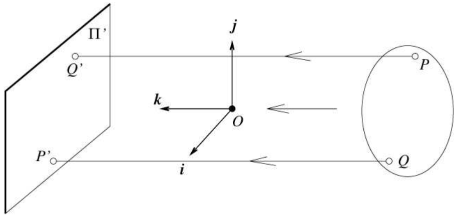
Height map recovered by careful integration of normal field

Applications : Photometric Stereo



- Height map recovered by careful integration of normal field
- Assumes orthographic projection

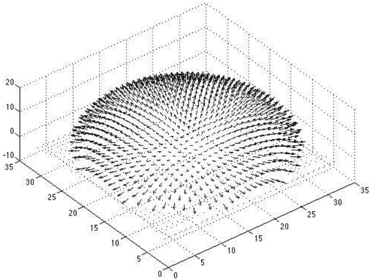
Camera Model : Orthographic Projection

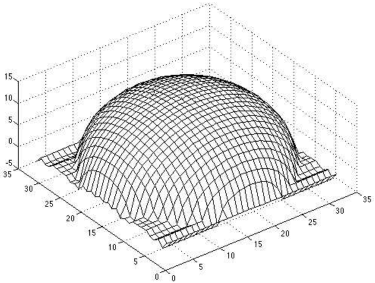


Orthographic Projection Model

- Projects object points onto plane along direction of optical axis
- Also known as parallel projection
- Drastic approximation $\boldsymbol{x} = \boldsymbol{X}$!
- Useful as linear methods are applicable

3D Reconstruction from Normals





Two approaches

- Discrete integration
- *Global* solution

Discrete Integration

Reconstruct $Z(x, y) \in \mathbb{R}^3$ given gradient field

$$\begin{aligned} \boldsymbol{K}(\boldsymbol{p}) &: \mathbb{R}^2 \to \mathbb{R}^2 \\ \boldsymbol{p} &= (x, y) \end{aligned}$$

$$\begin{aligned} \boldsymbol{K}(\boldsymbol{p}) &= grad(Z(x, y)) \\ &= (p(x, y), q(x, y)) = \left(\frac{\partial Z}{\partial x}, \frac{\partial Z}{\partial y} \right) \end{aligned}$$

Solve using integration

Integrate

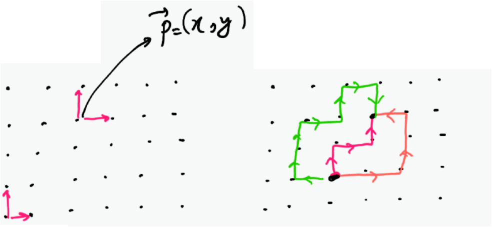
- locally?
- globally?

Map normals to gradient field : (p, q)

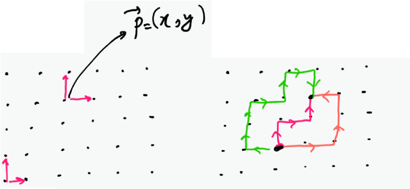
$$\begin{aligned} curl &: \frac{\partial p}{\partial y} - \frac{\partial q}{\partial x} \\ div &: \frac{\partial p}{\partial x} + \frac{\partial q}{\partial y} \end{aligned}$$

General solution

$$\min_Z \int \int \left((Z_x - p)^2 + (Z_y - q)^2 \right) dxdy$$



- Start integration from origin
- Which path should we choose?
- Under noise, path are not *consistent*
- Local integration leads to problems



- Consistent paths
- All loops should integrate to zero
- Vector field is *integrable*

Requirement

For continuous partial derivatives
If
$$\frac{\partial K_x}{\partial y} = \frac{\partial K_y}{\partial x}$$
then Z satisfies integrability

Equivalent to
$$\frac{\partial^2 Z}{\partial x \partial y} = \frac{\partial^2 Z}{\partial y \partial x}$$

$$\frac{\partial^2 Z}{\partial x \partial y} = \frac{\partial^2 Z}{\partial y \partial x}$$

$$Z = \mathcal{F}^{-1}\left(-j\frac{u\mathcal{F}(p) + v\mathcal{F}(q)}{u^2 + v^2}\right)$$

Integrability

- Noisy gradients do not satisfy requirement
- Solution: Project gradients onto "integrable" basis
- Frankot-Chellappa algorithm (1988)
- We will not study the proof
- Modern interpretations use other compact bases (wavelets)