Anandan’s Approach

Project Assignment

\[ I_x u + I_y v = -I_t \]

\[ u = a_1 x + a_2 y + b_1 \]

\[ v = a_3 x + a_4 y + b_2 \]
Anandan’s Approach

Project Assignment

\[ \sum \frac{X^T \Delta I \Delta I^T X}{\text{all pixels}} = - \sum \frac{X^T \Delta I}{\text{all pixels}} \]

\[ \begin{bmatrix}
    x_1^2 I_x^2 & xy_1 I_x I_y & x_1 I_y & x_1 x_1 I_x & x_1 y_1 I_x & x_1 y_1 I_y \\
    xy_1 I_x I_y & y_1^2 I_y^2 & y_1 I_y & y_1 x_1 I_y & y_1 y_1 I_y & y_1 y_1 I_y \\
    x_1 y_1 I_x & y_1 x_1 I_y & I_y & x_1 x_1 I_y & x_1 y_1 I_y & x_1 y_1 I_y \\
    x_1 x_1 I_x & x_1 y_1 I_y & x_1 I_y & I_x & x_1 x_1 I_x & x_1 y_1 I_x \\
    x_1 x_1 I_x & x_1 y_1 I_y & x_1 I_y & x_1 x_1 I_x & I_y & x_1 x_1 I_y \\
    x_1 y_1 I_x & y_1 x_1 I_y & I_y & x_1 y_1 I_x & x_1 x_1 I_y & I_x \\
\end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} = \begin{bmatrix} -x_1 I_x \\ -y_1 I_y \\ -I_y I_y \\ -I_x I_x \end{bmatrix} \]

HINTS

- Normalize the image size between [-0.7, +0.7] in both directions
  - To provide equal weighting for all the pixels
- Compute the image derivatives in both images using the following filter pairs
  - Multiply the derivatives with your normalization factor

\begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix}, \begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix}, \begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix}

Derivative in \( x \) Derivative in \( y \) Derivative in \( t \)

Alper Yilmaz, Fall 2005 UCF
Anandan’s Approach
Project Assignment

HINTS

- Pyramid implementation
  1. Start from the smallest resolution
  2. Iteratively compute affine parameters (5 iteration)
  3. Go to a bigger scale
    - Keep A matrix as it is multiply B vector by 2
  4. Repeat steps 2 and 3 until all the pyramid levels are visited

\[
A = A_2 A_1 \quad B = A_2 B_1 + B_2
\]
Stereo Constraints

Given \( p \) in left image, where can the corresponding point \( p' \) in right image be?

Epipolar Geometry

- \( X_1 \), \( Y_1 \), \( Z_1 \) - Image plane
- \( O_1 \), \( Y_2 \), \( Z_2 \) - Epipole
- \( M \) - Epipolar Line
- \( X_2 \) - Right image coordinates

Alper Yilmaz, Fall 2005 UCF
Basic Stereo

Recap
Stereo Approaches

- Token Based Stereo
  - Find tokens (Corners, etc.) in both images
  - Find correspondences between tokens
  - Compute depth surface using disparity
- Computing Correspondence
  - Marr-Poggio (you don’t need to know)
  - Correlation based stereo methods
  - Barnard’s algorithm

\[ Z = \frac{fT}{x_1 - x_2} \]

\[ \]
Recap
Correlation Based Stereo Methods

- Find corresponding tokens using correlation

\[
SSD = \sum \sum (I_{\text{left}} - I_{\text{right}})^2
\]

\[
AD = \sum \sum |I_{\text{left}} - I_{\text{right}}|
\]

\[
CC = \sum \sum I_{\text{left}} I_{\text{right}}
\]

- Given correspondences compute depth

\[
\frac{\sum \sum (I_{\text{left}} - I_{\text{right}})}{\sqrt{\sum \sum I_{\text{left}} I_{\text{right}}}}
\]

\[
1 \leq \sum \sum + \sum \sum = AD
\]

\[
nC = \frac{1}{64\sigma_I^2\sigma_r^2} \sum \sum (I_{\text{left}} - \mu_{\text{left}})(I_{\text{right}} - \mu_{\text{right}})
\]

Recap
Barnard’s Stereo Method

- Define an energy based on intensity similarity

\[
E = \sum \sum \left| I_{\text{left}}(x+i, y+j) - I_{\text{right}}(x+i + D_t(x, y), y+j + D_r(x, y)) \right| + A \left| \nabla D(x, y) \right|
\]

\[
\nabla D(x, y) = \sum \sum \left| D(x+i, y+j) - D(x, y) \right|
\]

- Minimize energy using simulated annealing
Fundamental Matrix

- Defines relation between two image planes of two cameras

\[ P_l = R_l (P_l - T_l) \]

Basics

- Co-planarity condition
  - 3 vectors on a plane
    \[ A^T (B \times C) = 0 \]
  - Let's define for epipolar plane
    \[ (P_l - T_l)^T (T \times P_l) = 0 \]
    \[ (R_l^T P_l)^T (T \times P_l) = 0 \]
    \[ P_l^T R_s P_l \neq 0 \]
Essential Matrix

- Related to camera extrinsic parameters
  - Rotation and translation
    \[ P_r^T RSP_i = 0 \]
    \[ E = RS \]
- Captures relation between to camera coordinates

How do we relate image coordinates?

- From camera coordinates to image coordinates.
  - Intrinsic camera parameters (in homogenous coordinates)
    \[ x_r^T M_r^T RSM_i^{-1} x_i = 0 \]
    \[ E = RS \]
    \[ F = M_r^T RSM_i^{-1} \]
    \[ F = M_r^T EM_i^{-1} \]  
    Fundamental matrix
Fundamental Matrix

- Relates left image coordinates to right image coordinates.
- Includes all camera parameters.
- 3x3 matrix with 7 degrees of freedom
  - 4 for epipoles \((x,y)\) and 3 for point projection.
- Rank 2 (due to rank deficient \(S\))

\[
\begin{bmatrix}
x & y & 1 \\
\end{bmatrix}F
\begin{bmatrix}
x' \\
y' \\
1 \\
\end{bmatrix} = 0
\]

Computing Fundamental Matrix

- Eight point algorithm
  - Most common approach
  - Requires 8 corresponding points in both images.
- Write unknown fundamental matrix parameters \((f_1..f_6)\) into vector and the rest into equation (observation) matrix \(O\).
  - \(O.f = 0\)
- Compute least squares solution.
Computing Fundamental Matrix

- Impose rank constraint using SVD
- Constrain solution of $f'$ to $|f'|=1$