Lecture-8

Feature-based Registration

Steps in Feature-based Registration
- Find features
- Establish correspondences between features in two images (correlation, point correspondence)
- Fit transformation
- Apply transformation (warp)

Features
- All pixels (spatiotemporal approach)
- Corner points
- Interest points
- Straight lines
- Line intersections
- Features obtained using Gabor/Wavelet filters
- ...

Transformations
- Affine
- Projective
- Pseudo-perspective
- Rational polynomial

Good Features to Track
- Corner like features
- Moravec’s Interest Operator

Corner like features
\[ C = \frac{\sum_{\sigma} f_x^2}{\sum_{\sigma} f_x f_y} \frac{\sum_{\sigma} f_x f_y}{\sum_{\sigma} f_y^2} \]

\( Q \) is an image patch

\[ D = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \]

Eigen Values
Corners

- For perfectly uniform region $\lambda_1 = \lambda_2 = 0$
- If $Q$ contains an ideal step edge, then $\lambda_2 = 0, \lambda_1 > 0$
- If $Q$ contains a corner of black square on white background $\lambda_1 \geq \lambda_2 > 0$

Algorithm Corners

- Compute the image gradient $(f_x, f_y)$ over entire image $f$.
- For each image point $p$:
  - form the matrix $C$ over $(2N+1) \times (2N+1)$ neighborhood $Q$ of $p$;
  - compute the smallest eigenvalue of $C$;
  - if eigenvalue is above some threshold, save the coordinates of $p$ into a list $L$.

Algorithm Corners

- Sort $L$ in decreasing order of eigenvalues.
- Select the top candidate corner, and perform Non-maxima suppression
  - Scanning the sorted list top to bottom: for each current point, $p$, delete all other points on the list which belong to the neighborhood of $p$.

Results

- (a)
- (b)

Results
Moravec’s Interest Operator

Algorithm
- Compute four directional variances in horizontal, vertical, diagonal and anti-diagonal directions for each 4 by 4 window.
- If the minimum of four directional variances is a local maximum in a 12 by 12 overlapping neighborhood, then that window (point) is interesting.

\[
V_h = \sum_{i=0}^{3} \sum_{j=0}^{3} (P(x+i,y+j) - P(x+i+1,y+j))^2 \\
V_v = \sum_{i=0}^{2} \sum_{j=0}^{3} (P(x+i,y+j) - P(x+i,y+j+1))^2 \\
V_d = \sum_{i=0}^{3} \sum_{j=0}^{2} (P(x+i,y+j) - P(x+i+1,y+j+1))^2 \\
V_a = \sum_{i=0}^{3} \sum_{j=0}^{3} (P(x+i,y+j) - P(x+i-1,y+j+1))^2
\]

\[
V(x,y) = \min(V_h(x,y), V_v(x,y), V_d(x,y), V_a(x,y))
\]

\[
I(x,y) = \begin{cases} 
1 & \text{if } V(x,y) \text{ local max} \\
0 & \text{otherwise} 
\end{cases}
\]
Correlation

- Similarity/Dissimilarity Measures
  - Sum of Squares Difference (SSD)
  - Normalized Correlation
  - Mutual Correlation
  - Mutual information $H(X;Y) = H(X) - H(X|Y)$

- Use
  - Gray levels
  - Laplacian of Gaussian
  - Gradient magnitude

Block Matching

- For each 8X8 block, centered around pixel $(x,y)$ in frame $k$, $B_k$
  - Obtain 16X16 block in frame $k-1$, centered around $(x,y)$, $B_{k-1}$
  - Compute Sum of Squares Differences (SSD) between 8X8 block, $B_k$, and all possible 8X8 blocks in $B_{k-1}$
  - The 8X8 block in $B_{k-1}$ centered around $(x',y')$, which gives the least SSD is the match
  - The displacement vector (optical flow) is given by $u = x - x'$; $v = y - y'$

Sum of Squares Differences (SSD)

$$(u(x,y), v(x,y)) = \arg\min_{u,v} \sum_{i=0}^{7} \sum_{j=0}^{7} (f(x+i,y+j) - f_k(x+u+i, y+v+j))^2$$

Minimum Absolute Difference (MAD)

$$(u(x,y), v(x,y)) = \arg\min_{u,v} \sum_{i=0}^{7} \sum_{j=0}^{7} |f(x+i,y+j) - f_k(x+u+i, y+v+j)|$$
Maximum Matching Pixel Count (MPC)

\[ T(x,y,u,v) = \begin{cases} 1 & \text{if } |f_k(x,y) - f_{k-1}(x+u,y+v)| \leq t \\ 0 & \text{otherwise} \end{cases} \]

\[ (u(x,y),v(x,y)) = \arg\max_{u,v} \frac{1}{2} \sum_{i=0}^{K} \sum_{j=0}^{t} T(x+i,y+j,u,v) \]

Cross Correlation

\[ (u,v) = \arg\max_{u,v} \frac{1}{2} \sum_{i=0}^{K} \sum_{j=0}^{t} (f_k(x+i,y+j) - \mu_k)(f_{k-1}(x+i+u,y+j+v) - \mu_{k-1}) \]

Normalized Correlation

\[ (u,v) = \arg\max_{u,v} \frac{1}{\sigma_k \sigma_{k-1}} \frac{\sum_{i=0}^{K} \sum_{j=0}^{t} (f_k(x+i,y+j) - \mu_k)(f_{k-1}(x+i+u,y+j+v) - \mu_{k-1})}{\sqrt{\sum_{i=0}^{K} \sum_{j=0}^{t} (f_k(x+i,y+j) - \mu_k)^2 \sum_{i=0}^{K} \sum_{j=0}^{t} (f_{k-1}(x+i+u,y+j+v) - \mu_{k-1})^2}} \]

\( \mu_k \) and \( \mu_{k-1} \) are the means of patch-1 and patch-2 respectively.

Mutual Correlation

\[ (u,v) = \arg\max_{u,v} \frac{1}{\sigma_k \sigma_{k-1}} \frac{\sum_{i=0}^{K} \sum_{j=0}^{t} (f_k(x+i,y+j) - \mu_k)(f_{k-1}(x+i+u,y+j+v) - \mu_{k-1})}{\sqrt{\sum_{i=0}^{K} \sum_{j=0}^{t} (f_k(x+i,y+j) - \mu_k)^2 \sum_{i=0}^{K} \sum_{j=0}^{t} (f_{k-1}(x+i+u,y+j+v) - \mu_{k-1})^2}} \]

Sigma and mu are standard deviation and mean of patch-1 and patch-2 respectively.

Issues with Correlation

- Patch Size
- Search Area
- How many peaks

Spatiotemporal Models

- First order Taylor series

\[ f(x,y,t) = f(x+dx,y+dy,t+dt) \]

\[ f_{x} = \frac{\partial f}{\partial x}, f_{y} = \frac{\partial f}{\partial y}, f_{t} = \frac{\partial f}{\partial t} \]

\[ f_{x} + f_{y} + f_{t} = 0 \]

\[ \sum X^T f_{x} f_{x}^T X \delta t = - \sum X^T f_{x} f_{t} \quad \text{Correlation} \]
Bilinear and Pseudo-Perspective

\[
(\sum \Phi^T \Phi) q = -\sum f_i \Phi
\]

\[
\Phi' = \begin{bmatrix} f_x(xy,x,y,1) & f_y(xy,x,y,1) \end{bmatrix} \text{bilinear}
\]

\[
\Phi' = \begin{bmatrix} f_x(x,y,1) & f_y(x,y,1) & c_1 & c_2 \end{bmatrix}
\]

\[
c_1 = x^2 f_x + xy f_y
\]

\[
c_2 = xy f_x + y^2 f_y
\]

Correlation Complexity

- \(m \times m\) multiplications and additions
- \(2 \times m \times m\) additions and 2 divisions for two means
- \(2 \times m \times m\) multiplications and additions for variances

\[
(u,v) = \arg \max_{u,v} \sum_{x,y} f(x+i,y+j) - \mu_1 (f(x+i,y+j) - \mu_1)
\]

Spatiotemporal Complexity

- \(3 \times m \times m\) subtractions for spatiotemporal derivatives
- \((36+6) \times m \times m\) additions for generating linear system
- \(6 \times 6 \times 6\) multiplications and additions for solving 6 by 6 linear system

\[
\sum X^T f X f X^T X \delta x = -\sum X^T f_X f_X
\]

Feature-based Matching

- The input is formed by \(f_1\) and \(f_2\), two frames of an image sequence.
- Let \(Q_1, Q_2\) and \(Q'\) be three \(N \times N\) image regions.
- Let “\(d\)” be the unknown displacement vector between \(f_1\) and \(f_2\) of a feature point “\(p\)”, on which \(Q_1\) is centered.
Algorithm

- Set $d=0$, center $Q_1$ on $p_1$.
- Estimate the displacement "$d_0$" of "$p$", center of "$Q_1$", using Lucas and Kanade method. Let $d=d_0$.
- Let $Q'$ be the patch obtained by warping $Q_1$ according to "$d_0$". Compute Sum of Square (SSD) difference between new patch $Q'$ and corresponding patch $Q_2$ in frame $f_2$.
- If SSD is more than a threshold, set $Q_1=Q'$ and go to step 1, otherwise exit.

Lucas & Kanade (Least Squares)

- Optical flow eq

$$ f_{xu} + f_{yv} = -f_t $$

- Consider n by n window

$$ f_{xu}^i + f_{yv}^i = -f_{ti} $$

$$ Au = f_t $$

Shi-Tomasi-Kanade (STK) Tracker

After the first order Taylor expansion at $A=I$ and $d=0$, we can get a linear $6 \times 6$ system:

$$ T = \frac{f}{2} $$

$$ z = \begin{bmatrix} A_1 & A_2 & A_3 & d_1 & d_2 \end{bmatrix} $$

$$ f = \begin{bmatrix} x & y & A & d & \end{bmatrix} $$

$$ T = \begin{bmatrix} x & y & d & \end{bmatrix} $$

Shi-Tomasi-Kanade (STK) Tracker

- Advantages:
  - Easy to implement.
  - Works very well with a small transformation.

- Drawbacks:
  - Fail to track in presence of a large rotation.

- Reason:
  - The rotation component implied in affine model is non-linear.

Useful Links