Image alignment
A look into the past

A look into the past

- Leningrad during the blockade

http://komen-dant.livejournal.com/345684.html
Bing streetside images

Image alignment: Applications

Panorama stitching

Recognition of object instances
Image alignment: Challenges

Small degree of overlap

Intensity changes

Occlusion, clutter
Two families of approaches:

- **Direct (pixel-based) alignment**
  - Search for alignment where most pixels agree

- **Feature-based alignment**
  - Search for alignment where *extracted features* agree
  - Can be verified using pixel-based alignment
Alignment as fitting

• Previous lectures: fitting a model to features in one image

Find model $M$ that minimizes

$$\sum_i \text{residual}(x_i, M)$$
Alignment as fitting

Previous lectures: fitting a model to features in one image

- Alignment: fitting a model to a transformation between pairs of features (*matches*) in two images

Find model \( M \) that minimizes

\[
\sum_i \text{residual}(x_i, M)
\]

Find transformation \( T \) that minimizes

\[
\sum_i \text{residual}(T(x_i), x'_i)
\]
2D transformation models

- **Similarity**
  (translation, scale, rotation)

- **Affine**

- **Projective**
  (homography)
Let’s start with affine transformations

- Simple fitting procedure (linear least squares)
- Approximates viewpoint changes for roughly planar objects and roughly orthographic cameras
- Can be used to initialize fitting for more complex models
Fitting an affine transformation

- Assume we know the correspondences, how do we get the transformation?

\[
\begin{bmatrix}
    x'_i \\
    y'_i
\end{bmatrix} = \begin{bmatrix}
    m_1 & m_2 \\
    m_3 & m_4
\end{bmatrix} \begin{bmatrix}
    x_i \\
    y_i
\end{bmatrix} + \begin{bmatrix}
    t_1 \\
    t_2
\end{bmatrix}
\]

\[
\begin{bmatrix}
    m_1 \\
    m_2 \\
    m_3 \\
    m_4 \\
    t_1 \\
    t_2
\end{bmatrix} = \begin{bmatrix}
    m_1 \\
    m_2 \\
    m_3 \\
    m_4 \\
    t_1 \\
    t_2
\end{bmatrix}
\]
Fitting an affine transformation

\[
\begin{bmatrix}
\vdots \\
x_i & y_i & 0 & 0 & 1 & 0 \\
0 & 0 & x_i & y_i & 0 & 1 \\
\vdots \\
\end{bmatrix}
\begin{bmatrix}
m_1 \\
m_2 \\
m_3 \\
m_4 \\
t_1 \\
t_2 \\
\end{bmatrix}
= \begin{bmatrix}
\vdots \\
x_i' \\
y_i' \\
\vdots \\
\end{bmatrix}
\]

- Linear system with six unknowns
- Each match gives us two linearly independent equations: need at least three to solve for the transformation parameters
Fitting a plane projective transformation

- **Homography**: plane projective transformation (transformation taking a quad to another arbitrary quad)
Homography

- The transformation between two views of a planar surface

- The transformation between images from two cameras that share the same center
Application: Panorama stitching

Source: Hartley & Zisserman
Fitting a homography

- Recall: homogeneous coordinates

\[(x, y) \Rightarrow \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}\]

Converting to homogeneous image coordinates

\[
\begin{bmatrix} x \\ y \\ w \end{bmatrix} \Rightarrow (x/w, y/w)
\]

Converting from homogeneous image coordinates
Fitting a homography

- Recall: homogeneous coordinates

\[(x, y) \Rightarrow \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad \Rightarrow \begin{bmatrix} x \\ y \\ w \end{bmatrix} \Rightarrow (x/w, y/w)\]

Converting to homogeneous image coordinates

Converting from homogeneous image coordinates

- Equation for homography:

\[
\lambda \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}
\]
Fitting a homography

- Equation for homography:

\[
\lambda \begin{bmatrix}
x'_i \\
y'_i \\ 1
\end{bmatrix} = \begin{bmatrix}
h_{11} & h_{12} & h_{13} \\
h_{21} & h_{22} & h_{23} \\
h_{31} & h_{32} & h_{33}
\end{bmatrix} \begin{bmatrix}
x_i \\
y_i \\ 1
\end{bmatrix}
\]

\[
\lambda x'_i = H x_i
\]

\[
x'_i \times H x_i = 0
\]

\[
\begin{bmatrix}
x'_i \\
y'_i \\ 1
\end{bmatrix} \times \begin{bmatrix}
h_1^T x_i \\
h_2^T x_i \\
h_3^T x_i
\end{bmatrix} = \begin{bmatrix}
y'_i \ h_3^T x_i - h_2^T x_i \\
h_1^T x_i - x'_i \ h_3^T x_i \\
x'_i \ h_2^T x_i - y'_i \ h_1^T x_i
\end{bmatrix}
\]

\[
\begin{bmatrix}
0^T & -x_i^T & y'_i x_i^T \\
x_i^T & 0^T & -x'_i x_i^T \\
-y'_i x_i^T & x'_i x_i^T & 0^T
\end{bmatrix}\begin{pmatrix}
h_1 \\
h_2 \\
h_3
\end{pmatrix} = 0
\]

3 equations, only 2 linearly independent
H has 8 degrees of freedom (9 parameters, but scale is arbitrary)

One match gives us two linearly independent equations

Four matches needed for a minimal solution (null space of 8x9 matrix)

More than four: homogeneous least squares
Example

1st view

2nd view

2nd view warped by the planar homography between two views
Rotation Only - Calibrated Case

- Calibrated Two views related by rotation only
  \[ \lambda_2 x_2 = R \lambda_1 x_1 \quad \hat{x}_2 R x_1 = 0 \]
- Mapping to a reference view – rotation can be estimated
- Mapping to a cylindrical surface