Unfortunately, the course book does not cover all aspects of image registration. You will find pieces of it, however.

- Matching: 6.4
- ICP: 12.2.6
- Transformations: 5.2, 11.2
- Interest points: 5.3.10, 5.3.11

Image Alignment and Stitching: A Tutorial

Image Registration Methods: A Survey

Only things from this lecture and the course book will be covered by the written examination. Ask questions or email if anything is unclear!

(Many of the algorithms can be found on Wikipedia. I highly recommend Google and Wikipedia to get alternative views on this and other topics.)

---

Registration – To Read?!

- Exercises

Registration – What is it?

- "Register" "To adjust so as to be properly aligned."
- "Fusion" "something new created by a mixture of qualities, ideas, or things" "become or cause to become bent or twisted out of shape, typically as a result of the effects of heat or dampness" "a person or thing that resembles or corresponds to another"  

\[ x_1, x_2, y_1, y_2, \phi : X \rightarrow Y, y = \phi(x) \]

Registration – When is it used?

- Art and Photography
  - Stitching photos together, panoramic images
- Astronomy
  - Stitching photos together and fusion of different wavelengths
- Robotics
  - Finding and tracking objects in a scene (2-D)
- Chemistry
  - Finding similar images of molecules
- "Word Spotting"
  - Matching and aligning words in old handwritten historical documents
  - Enables searching in large document collections without the need for manual translation
  - F. Wahlberg et al. 2011
In Word Spotting using Dynamic Time Warping:
- Text lines are transformed to 1-D signals
- DTW is optimal non-linear matching of 1-D signals

DTW finds optimal alignment of 1-D signals using dynamic programming. Upper: Before warping, lower: after warping.

Handwritten signature recognition
- Comparison of handwritten signatures (1-D)

Human motion analysis
- Temporal alignment of gesture sequences
- Speech recognition
  - Warping the spectrum of spoken words (1-D or 2-D)

Medical Imaging
- study skin changes over time
- colonscopy (1-D)
- Pre operative vs post operative
- Fusion of different modalities
  - CT-MRI
  - CT-Ultrasound
  - PET-CT
- ...
- And much more ...

The brain is not a rigid object
- "Brain shift" occurs during surgery
- Using "open MR", images can be taken in realtime ...
- But how to compare these images with preoperative CT and MRI images?
- Image-based registration is necessary!
Registration – Cues

- Edges, corners, eyes, interest points, …
- Color, texture, similarity measures...
- Fiducials and frames – and screws!
- Skin markers – less painful
- **Restricted transformation space**
- Anatomical landmarks
- Expert knowledge
- Local phase

Registration – An ill posed problem

- An ill-posed problem: Many degrees of freedom compared to available data
- In registration: If X has a total number of N pixels and Y has M pixels, then we need to find 2N coordinate values given N+M values of input data
- Thus: Finding the best transformation y = φ(x) is an ill-posed problem!
- Solution: Restrict the set of possible transformations φ

Registration – How to do it?

1) Define the set of allowed transformations φ
2) Define a useful functional to measure similarity
3) Choose a practical optimization procedure

Geometrical Transformations

- Translation (2-D, 3-D, ... N-D):
  \[ y = x + t \]

- Rigid transformation
  \[ y = Rx + t \]
  \[ R \in SO(N) \iff RR^T = I \text{ and } \det(R) = 1 \]

- Rigid transformation + mirroring
  \[ y = Rx + t \]
  \[ R \in O(N) \iff RR^T = I \text{ and } |\det(R)| = 1 \]
Geometrical Transformations

- Affine transformations
  \[ y = Ax + t \]

  \[ \varphi : X \rightarrow Y \]
  \[ y = \varphi(x) \]

- Perspective (2-D):
  \[ \begin{align*}
  x &= ax + by + c \\
  y &= dx + ey + 1
  \end{align*} \]

Images from www.panoramio.com

B-spline interpolation (2-D)
- A grid of patches, combined into a function
  - Continuous
  - Smooth
- Built by cubic polynomial basis functions

  \[ \varphi : X \rightarrow Y \]
  \[ y = \varphi(x) \]

B-spline (3-D)
- Interpolating local basis functions (3rd order polynomials)

Intensity Interpolation

- Both position and signal value is interpolated during registration
- Without proper resampling of the image values (color/intensity...), the image might look ugly and the registration may perform poor
- Nearest neighbor
- Bi- (2-D) or Trilinear (3-D) interpolation
- Bicubic or Spline based interpolation
- In Matlab: imrotate, imresize, interp2, interp3, ...
- Value interpolation enables sub-pixel precision registration

Registration – How to do it?

- 1) Define the set of allowed transformations \( \varphi \)
- 2) Define a useful functional to measure similarity
- 3) Choose a practical optimization procedure
Different kinds of image fusion

- Similar images
  - Same camera
  - Same modality
  - Same patient

- Similar but different images
  - Different brightness or contrast
  - Different lighting (e.g. day/night)
  - Different modalities (CT/MRI/Ultrasound)
  - Photo vs a drawing

Image similarity and dissimilarity

- Several options exist to measure (dis)similarity between the transformed source image $\phi(X)$ and the target $Y$:
  - Mean Squared Error (dissimilarity)
  - Correlation (similarity)
  - Mutual information (similarity)
  - These are based on pointwise comparisons between pixels in $\phi(X)$ and $Y$
  - They can be modified using preprocessing of the images: denoising, edge filters, corner filters, ... if needed.

The Joint Histogram

- The Ordinary Histogram of an image $Y$:
  - $H(k, Y) = \text{Number of pixel positions with intensity } k \text{ in an image } Y.$

- The Joint Histogram of images $Y$ and $\phi(X)$:
  - $H(k, m, \phi(X), Y) = \text{Number of pixel positions with intensity } k \text{ in image } Y \text{ and intensity } m \text{ in image } \phi(X)$

The Joint Histogram

- Here is a medical example:
  - Two SPECT scans of the same patient. The patient has moved between the scans. This introduces movement artefacts when the two images are subtracted.
  - The voxel intensities are linearly correlated

- Mean Squared Error

  - In the SPECT example, the mean squared error works fine:
    $$MSE = E((x - y)^2)$$

  - Cons:
    - Signals must have exactly the same brightness and contrast
    - Does not work for different modalities

  - Pros:
    - Simple and intuitive
    - Ensures exact intensity matching
Linear intensity changes are common in practice. MSE may not work in such cases. Correlation does.

Definition of the correlation coefficient:

\[
p_{xy} = \frac{E((X - \mu_x)(Y - \mu_y))}{\sigma_x \sigma_y}
\]

Pros:
- Corrects for intensity changes (common in practice)
- Efficient to evaluate
- Can be generalized to non-scalar signals using “Canonical Correlation Analysis” (Borga 1998)

Cons:
- Underlying assumption about Gaussian distributions
- Does not work for (very) different modalities

Note, we can also register inverted images by analysis of the squared correlation coefficient.

Cost Function for Dissimilar Images

- When data come from different modalities, there is no linear relationship between voxel intensities and a simple similarity measure MSE or correlation won’t suffice.
• The MI registration criterion states that the images are geometrically aligned when $MI(A,B)$ is maximal.
• For example, let $A$ be an MRI-scan and $B$ a SPECT-scan. If we know the MRI intensities, the uncertainty of the SPECT intensities is minimal when the scans are aligned.

**Mutual Information**

**Mutual information – an Example**

Before registration

After registration

**Mutual information: PET-MRI**

**Mutual information: CT - MRI**

**Mutual information – an Example**

**Mutual Information**

- **Pros:**
  - Very general idea
  - Works for different modalities
  - Works for non-scalar signals (e.g. RGB)
- **Cons:**
  - Can be tricky to implement
  - Slower to evaluate than MSE and correlation
  - To exactly measure MI, we need infinitely large images. We assume that images are stochastic signals from which we can draw and infinite number of samples. Thus, it may not work well for small images.
Functionals for Image similarity

- A cost functional evaluates a particular transformation \( \phi \)
  \[
  \text{Cost} = \text{image dissimilarity}(\phi(X), Y)
  \]
- The goal is to minimize the cost. A low cost means that we have a good fit.
- We can also include an additional cost to punish “bad” inappropriate transformations
  \[
  \text{Cost} = \text{image dissimilarity}(\phi(X), Y) + \text{deformation cost}(\phi)
  \]
  (Restricting the set of possible transformations is also a kind of punishment, i.e. setting the cost to infinite to all other transformations)

Registration – How to do it?

1. Define the set of allowed transformations \( \phi \)
2. Define a useful functional to measure similarity
3. Choose a practical optimization procedure

The optimization loop

- Gradient decent
  - Simplex
  - Conjugate gradient
- Stochastic optimization
  - Genetic algorithms
  - Simulated Annealing
- Hierarchical
- Iterative
- Stochastic vs Deterministic

The Fitness Landscape

- The choice of method depends on the nature of the “fitness landscape”, which depends on both the objective function and the images you try to register

The optimization loop

- Gradient decent
  - Simplex
  - Conjugate gradient
- Stochastic optimization
  - Genetic algorithms
  - Simulated Annealing
- Demons algorithm
- Morphons
- Hierarchical
- Iterative
- Stochastic vs Deterministic
Demons algorithm

- Demons algorithm, Thirion 1998
- An analogy with Maxwell’s demon
- Demons are located at “interfaces”
- Demons pull or push the model-image locally


Variants
- Different placement of demons (at boundaries or everywhere)
- Different forces (unit strength, or variable)
- Deformation (rigid, elastic, …)
- Choice of interpolation (linear, cubic, …)
- Available in ITK

Example: The Morphon algorithm

- Deformable canvas
- Local smart image operators decide where to go
- Global regularization
- Hierarchical
- Based on local phase
- Available in ITK

Interest Point Detectors

- Has a position in space
- Invariant under image degradation
- Rich content around it
- Corner detection
- Edge detection
- SIFT – Scale-Invariant Feature Transform
- Blob detectors

SIFT – scale invariant feature transform

- SIFT, detect interest points:
  1) \( L(x,y,0) = G(x,y,0)I(x,y) \), where \( G \) is a Gaussian
  2) \( D(x,y,0) = L(x,y,k0) - L(x,y,0) \)
  3) Find max/min in \( D(x,y,0) \)
  4) Interpolate subpixel positions of max/min

SIFT, detect interest points (cont.):

- 4) Eliminate low contrast responses \( D(x,y,0) < 0.03 \)
- 5) Eliminate edge responses \( r=10 \)
  \[
  H = \begin{bmatrix} D_x & D_y \\ D_y^T & (r+1) \end{bmatrix}, \quad \text{Det}(H) < r
  \]
SIFT – scale invariant feature transform

- Now compute a feature vector
- Position a 4 x 4 grid
  - at the position of each interest point
  - scaled according to the detected scale
  - oriented according to the dominant gradient direction

SIFT – scale invariant feature transform

- In each part of the grid, compute an 8 direction histogram over the gradient direction
- Compose this into a 4 x 4 x 8 = 128 value vector
- This feature vector is invariant to scale, rotation and intensity changes

SIFT features are invariant to scale, rotation and intensity changes
SIFT features are robust to noise and 3D viewpoint changes
Scale and position are given by the interest point detection step
Rotation invariance comes from choosing the strongest gradient as a reference direction
http://www.youtube.com/watch?v=pIhX52p2Iuo

SURF – Speeded Up Robust Features

1. Associate feature points in X with their closest neighbors in Y
2. Estimate a model to transform respective points in X to Y (e.g., an affine transformation)
3. Transform points in X, i.e., X = φ(X)
4. Iterate 1..3 until convergence

Iterative Closest Points – ICP

- Find pairs of closest feature points in X & Y
- Transform X closer to Y, using matching pairs
**Iterative Closest Points – ICP**

- Used to compare two point-clouds
  - Two rigid bodies
  - Two range-laser scans in robot localization, to find out ego-motion
  - Two mesh models or one mesh model + measurements during e.g. bone-surgery
  - Points from range-scanners fitted to a model of some object, e.g. a car or a building

**RANSAC**

- RANSAC – RANdom Sample Consensus
  - A robust estimation technique
    - Non-deterministic
    - Similarities with Hough- and Radon transforms
    - Robustness with respect to outliers
  - Paradigm: Hypothesize and Test

**RANSAC**

- An example. Find the translation that transforms red dots to yellow.
  - There is one outlier present
  - One corresponding pair is needed (at least) to determine the transformation

**RANSAC**

- In RANSAC we randomly select a pair and use it to estimate a transformation
  - Hypothesis
  - No match

**RANSAC**

- And continue to select pairs until we find a good transformation where many points are matching (the outliers will of course not match)
  - Hypothesis
  - Match!

**RANSAC – Algorithm**

- N, minimal number of datapoints needed to estimate a model
- K, maximum number of allowed iterations
- T, threshold for data fitting
- D, number of datapoints needed for a good fit
1. Draw a random set of N datapoints
2. Use these N datapoints to estimate a model M
3. Test if this model fits D datapoints, within a threshold T, of all datapoints
4. If it does, you have a global fit. Exit.
5. If it doesn’t, and if you have iterated fewer than K times, return to 1.

An example:

From Feature based methods for structure and motion estimation by P. H. S. Torr and A. Zisserman

• The Perspective Transformation

\[ \begin{align*}
K &= k(x) \\
(\alpha x_1 + \alpha x_2 + \alpha x_3) &= (\beta x_1 + \beta x_2 + \beta x_3) \\
(\alpha y_1 + \alpha y_2 + \alpha y_3) &= (\beta y_1 + \beta y_2 + \beta y_3)
\end{align*} \]

• Massage it a bit...

• Oh, it yields two linear equations. Neat!

\[ \begin{align*}
0 &= -\alpha y_2 x_1 + \alpha y_1 x_2 + 0 - \alpha y_3 x_3 \\
0 &= -\alpha y_2 y_1 + \alpha y_1 y_2 + 0 - \alpha y_3 y_3
\end{align*} \]

• Thus, every corresponding pair \((x_1, x_2, x_3, y_1, y_2, y_3)\) yields kills 2 unknowns

• In total we have 9 (actually 8) unknowns. Thus four point pairs are needed \((N=4)\).

3-D registration of photos
From Microsoft Live Labs and Washington University

Detects interesting feature points
The 3-D positions of feature points are determined by “bundle adjustment”
The browsing is actually not so demanding as the preprocessing...

Obviously, random sampling may be inefficient. You can sample your data in a more clever way, e.g. by matching features.
RANSAC is non-deterministic, you may not find the global optimal fit within the K allowed iterations.
With larger model complexities, N, RANSAC may need a lot of iterations.
Feature images (binary) can seldom be used directly in combination with a gradient descent search technique. (why?)

The chamfer transform can turn them into grayvalue images.

Log-polar mapping
- \( (x_c, y_c) = \text{center} \)
- \( \rho = \log \left( \sqrt{(x - x_c)^2 + (y - y_c)^2} \right) \)
- \( \theta = \tan^{-1} \left( \frac{y - y_c}{x - x_c} \right) \)

Rotation/scale \( \rightarrow \) translations!

Useful to perform rigid matching
- We still need to know the center
- Solution: We do this in the Fourier domain
- The spectrum of the image is invariant to translation

We have seen that image registration is an optimization problem. Most algorithms fit this paradigm well
- Define a similarity functional
- Restrict transformation space
- Choose an optimization method
- Feature point based algorithms are also based on optimization, but the functional is based on distances of feature points rather than pure image information
- Pre-processing makes algorithms invariant to certain image characteristics, e.g. noise. Extracting feature points could be seen as a special kind of pre-processing.

We have to compute the affine transformation matrix that maps set \( A \) to set \( B \) exactly (linear algebra)

Which functional is best to use for the registration of two very different image modalities (mutual information)

Demonstrate a RANSAC algorithm that completes a registration task on the third iteration. (Hint: Assume translation is the only transformation allowed)