IMAGE REGISTRATION

General idea: Put two images of the same thing into spatial alignment. Figure out how to "superimpose" them. Image registration applications can be divided into 4 main groups according to the manner of image acquisition:

1. Different viewpoints (multiview analysis): Aim is to gain a larger 2D view or a 3D representation of the scene.
   - Remote sensing mosaicing
2. Different times (multitemporal analysis): Aim is to find and evaluate changes.
   - Remote sensing images from one year compared to ones from the next year. Land use surveys. Slightly different area covered, different orientation angle, illumination, crops, etc.
   - Medical images:
     - CT chest scans taken 6 months apart: has the tumor gone away or gotten worse? Some difficulties: Different lung sizes because breath size not the same, patient positioning not the same, etc.
     - Measurement in change of volume of bone grafts
     - Subtraction angiography
   - Security monitoring (change detection), fingerprints
3. Different sensors (multimodal analysis): Aim is to integrate information obtained from different sources.
   - Medical images depicting different characteristics (x-ray attenuation, echogenicity, radioactivity, MR signal) are often complementary from a diagnostic standpoint. For example, low resolution SPECT (single positron emission computed tomography) image showing metabolic function could be superimposed on high resolution MR image showing anatomy.
4. Scene-to-model registration: Aim is to localize the acquired image in the scene/model and/or to compare them.
   - Guidance systems: compare an image of the ground with a map or cleaned reference image
   - Manufacturing: Quality control for printed circuit boards: alignment of candidate board image with ideal image to look for defects
   - Medical: comparison of patient image with digital anatomical atlas

The aims are basically in 2 categories: comparative analysis, and composite analysis.

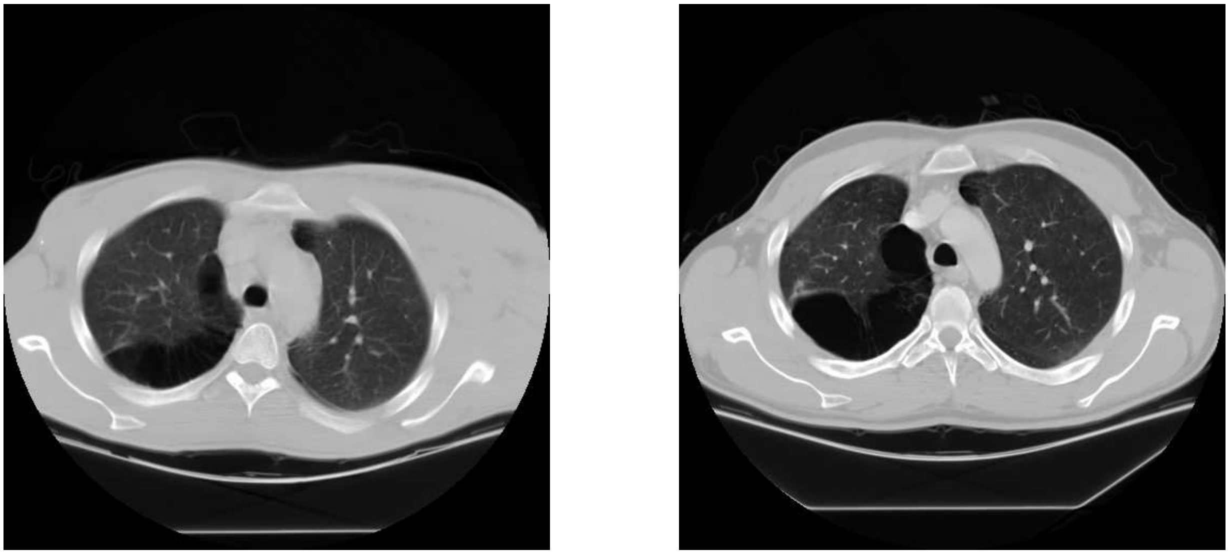
The problem of image registration is closely related to the problem of *model-based object recognition* (template matching, image detection). Instead of trying to register one entire image to another entire image, in model-based object recognition we try to register a template of the object to the image.

Why is registration hard to do?

- Machine factors: resolution, orientation, format
- Image factors: translation, rotation, scaling (magnification), warping (nonlinear distortions), gray level transformations, noise

- Object factors: objects in the 2 images may be different. Some of these differences may be what you are trying to find (e.g., tumor growth) and some differences you don't care about (e.g., patient motion, cardiac cycle, breathing cycle, positioning, etc.).

Example: CT chest scans showing a substantial change in a tumor over time.



We can divide the variations between the images into 3 types:

1. those differences which we wish the registration algorithm to account for, and thereby *remove*
2. those which are of interest to us, and so should not be accounted for, and therefore will be *exposed* after registration,
3. those which are not of interest to us, but which we are unable to remove, and which are called uncorrected distortions.

A Framework for Image Registration

Many registration methods can be viewed as different combinations of choices for four components:

1. A *feature space*, which extracts the information in the image that will be used for matching
2. A *search space*, which is the class of transformations that is capable of aligning the images
3. A *search strategy*, which decides how to choose the next transformation from this space, to be tested in the search for the optimal transformation, and
4. A *similarity metric*, which determines the relative merit for each test.

For example, suppose we wish to align 2 chest x-ray images taken of the same patient at different times. A standard approach might be to

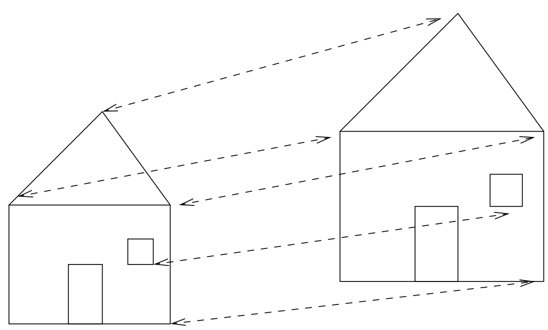
- Reduce the grayscale image to a binary image by detecting the edges. This removes extraneous information and reduces the amount of data to be evaluated.

- If it is thought that primary difference in acquisition between the two images was a small translation, the search space will be the set of a small translations.
- For each translation of the edges from the first image onto the edges of the second one, a similarity metric will be computed. The metric might be the correlation.
- If we compute the correlation for all small translations, the search strategy is exhaustive.

The images are then registered using the translation which optimizes the similarity measure. The choice of of using edges for features, translations for the search space, exhaustive search for the search strategy, and correlation for the similarity metric will influence the outcome of this registration.

Some possibilities for the feature space:

1. Pixel intensities from the entire image
2. Edge points derived from any edge detector, either thresholded or not. Sometimes when it is important to be more invariant to shape and scale, edge fragments that are joined in a Y or T are used.
3. Operator-outlined objects: e.g., objects are taken to be ellipses, and the ellipses are matched from one modality to another.
4. Principal axes, or axis of bilateral symmetry - this might be useful for medical images which are quite left-right symmetric. In some modalities, this axis of symmetry can be found easily without user interaction.
5. Reference points = landmark points = tiepoints



- Manually selected landmark points: labor intensive but more reliable.
- Extrinsic versus intrinsic
- Interpolation versus Approximation
- Global versus Local
- The reference points may not be evident in both of the images to be registered. For example, a SPECT image depicting metabolism will not have the same landmark points as an MR image depicting anatomic information, although they could be images of the same region.
- Landmark points can also be obtained as centroids of outlined objects. Since centroids are an integral calculation, they should be less sensitive to individual point differences. Computer can sometimes use segmentation to find objects by itself, and then compute centroid.

- Not all landmark points are created equal. Ideally, you want to pick a landmark that is the intersection of two lines, or the tip of a pointy region, or the centroid of a well-defined region, but you don't want to have to use the "top" of a structure as a landmark point, etc.
- Registration algorithms which use landmark points will work better if the landmark points are spread out. For some images, this is a drawback of the centroid method. Many centroids in a medical image will all be near the center of the image.

The choice of feature space determines what is matched. The similarity metric determines how matches are rated. Together the feature space and the similarity metric can ignore many types of variations which are not relevant to the proper registration. For example, in a very noisy pair of images, the registration algorithm can try to ignore the noise either by using a similarity metric that is robust to noise, or by using a feature space that eliminates the noise (e.g., feature space = high contrast edges).

Some possibilities for the search space (the set of transformations):
The transformation can be either *global* or *local*.

The most common types of transformations are rigid, affine, projective, perspective, and global polynomial.

A rigid-body transformation is composed of a combination of a rotation, a translation, and a scale change. There are four parameters: t_x, t_y, s, θ.

$$\begin{pmatrix} x_2 \\ y_2 \end{pmatrix} = \begin{pmatrix} t_x \\ t_y \end{pmatrix} + s \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x_1 \\ y_1 \end{pmatrix}$$

Since the rotation matrix R is orthogonal (the rows or columns are perpendicular to each other), the angles and lengths in the original image are preserved after the registration (under rotation). Without the addition of the translation vector, the transformation is linear.

With the translation vector, the transformation is *affine*, which means that $T(x) - T(0)$ is linear. Affine transformations are linear in the sense that they map straight lines into straight lines.

The general 2D affine transformation is

$$\begin{pmatrix} x_2 \\ y_2 \end{pmatrix} = \begin{pmatrix} a_{13} \\ a_{23} \end{pmatrix} + \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} x_1 \\ y_1 \end{pmatrix}$$

which does not have the properties associated with the orthogonal matrix. Angles and lengths are no longer preserved, but parallel lines do remain parallel. This general affine transformation can account for *shear*, (also called *skew*):

$$Shear_x = \begin{pmatrix} 1 & a \\ 0 & 1 \end{pmatrix}, \qquad Shear_y = \begin{pmatrix} 1 & 0 \\ b & 1 \end{pmatrix},$$

and also for changes in *aspect ratio*. The aspect ratio refers to the relative scale between the x and y axes. By scaling each axis independently,

$$Scale = \begin{pmatrix} s_x & 0 \\ 0 & s_y \end{pmatrix}$$

the ratio between the x and y scale is altered. An affine transformation thus consists of any sequence of rigid-body transformations, shears, and aspect ratio changes.

- There exists a gradation of registration algorithms, each one accounting for more types of transformations than the last one.
- If we could account for enough types of transformations, we could register any image exactly to any other! In that case, nothing would be learned by doing the registration except for the parameters of the best transformation.

  For example, one registration method applied to the CT images shown at the beginning caused the images to register almost "exactly," in the sense that the difference image was nearly empty, and showed no useful structure. The tumor seemed to disappear. The information about the disappearance of the tumor is contained in the parameters of the transformation, but not in a way which is useful to us. We wanted the algorithm *not* to register the part of the image with the tumor, but rather to present that area to us in the difference image.
- So an algorithm which accounts for more types of transformations is not necessarily better. One must pick the algorithm which best represents the types of transformations that one wants accounted for.

Some possibilities for the search strategy:

- Exhaustive search– only if the search space is small, or the similarity metric is very simple computationally.
- Sequential search: for example, for each window to be tested, one of the similarity measures defined below is accumulated until a threshold is exceeded. The number of points that were examined before the threshold was exceeded is recorded. The window which examined the most points is assumed to have the lowest measure.

  The sequential search can be thought of as a difference in similarity metric rather than a difference in search strategy. The metric is now, for example, find the maximum number of points that are examined before the mse exceeds a threshold T. All positions in the search window are examined, so it is still an exhaustive search. But the metric has changed from being the mse to being this max points things.

  A way to improve this type of search is to examine the points in some order (e.g., extrema first) such that the threshold is likely to be exceeded quickly in the case of mismatch.
- Hill-climbing techniques: don't lay the evaluation window in all possible locations. Try certain locations, and use these results to choose the next set of locations to try.
- Hierarchical techniques (also called coarse-fine matching): we reduce the resolution of both images using low-pass filtering followed by subsampling to produce coarse representations of the images. Then we use our similarity metric on the coarse images, and find the peak. This is then used to define a spatially restricted search area at the original fine resolution. Significant reduction in computation, but danger of false results.
- There are lots of other search strategies (dynamic programming, tree-structured methods, linear programming, generalized Hough transform, etc.)

Some possibilities for the similarity metric:

1. **TSE = total squared error or MSE = mean squared error**

$$E(m,n) = \sum_j \sum_k (F_1(j,k) - F_2(j-m, k-n))^2$$

The registration is best when the TSE or MSE is minimized.

2. **SAD or SAVD metric (Sum of absolute value of the differences):**

$$E(m,n) = \sum_j \sum_k \mid F_1(j,k) - F_2(j-m, k-n) \mid$$

(maybe with subtracting off the window means first). Similar to TSE or MSE, but computationally simpler.

3. **Cross-correlation:**

The normalized cross-correlation is defined by

$$R(m,n) = \frac{\sum_j \sum_k F_1(j,k) F_2(j-m,k-n)}{[\sum_j \sum_k [F_1(j,k)]^2]^{1/2} [\sum_j \sum_k [F_2(j-m,k-n)]^2]^{1/2}}$$

The best registration is when this cross-correlation function is at its maximum.

- The cross-correlation measure is directly related to the more intuitive squared error measure.
- A related measure, which is advantageous when an absolute measure is needed, is the correlation coefficient. This is just like the cross-correlation, except that you subtract off the means of the 2 images first. Can be useful to measure the confidence in a match.
- Generally use exhaustive search only when the allowable transformations include a small range of translation, rotations, and scale changes. Otherwise, the computational costs quickly become unmanageable.
- The measure is often computed on a reduced set of features (edge points, or just edge fragments joined in a T or a Y).
- If there is high correlation between pixels, and there is image noise, the cross-correlation function may be rather broad, and the actual peak may be difficult to discern. One way to get around this is by pre-filtering the images. This is sometimes referred to as *statistical correlation*.

  Instead of computing the cross-correlation function directly with the images F_1 and F_2 we use instead G_1 and G_2 where

  $$G_i(j,k) = [F_i(j,k) - \overline{F}_i(j,k)] * D_i(j,k).$$

  $\overline{F}_i(j,k)$ is the spatial average of $F_i(j,k)$ over the correlation window. The functions D_i are chosen based on, for example, the adjacent pixel correlation, so as to maximize the peak correlation when the images are in best register. For example, with high correlation between pixels, we could use something like

  $$D_i = \frac{1}{4} \begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix}$$

  which is a second derivative point detection operator. The output will be large in magnitude only in regions of an image for which its amplitude simultaneously changes significantly in both coordinate directions.