Implementation of an automatic slice-to-slice registration tool

Pavel A. Koshevoy, Tolga Tasdizen, and Ross T. Whitaker

May 26, 2006

Abstract

This paper outlines the basic steps in the design and implementation of a feature based Transmission Electron Microscopy (TEM) image registration application and highlights some of the implementation details, such as the detection of features, feature descriptor design, robust filtering of mismatched descriptors, and transform estimation. Although the approach chosen is based on the Scale Invariant Feature Transform (SIFT) method, it is optimized for the TEM image registration.

1 Motivation

The goal of this project is to provide a fully automatic tool for slice-to-slice image registration of several hundred slices assembled from high-resolution tile images. This tool is aimed at researchers working with Transmission Electron Microscopy images. The challenges lay in the fact that each slice is arbitrarily oriented in the imaging plane, and may have been warped independently from all other slices.

2 Problem statement

Given an ordered sequence of slices (e.g. S_0, S_1,... S_n) a transform must be constructed for each adjacent slice pair that would map from the image space of slice S_i to the space of slice S_{i+1}. This task will be addressed within a feature matching framework. The problem can be partitioned into several sub-problems outlined below:

- For each slice, a gradient vector image pyramid and a Difference-of-Gaussian image pyramid must be constructed.
- The extrema points of the DoG pyramid must be determined.
- The dominant gradient vector orientation(s) in the neighborhood of each extrema point must be detected.
- A descriptor for every detected gradient vector orientation of the extrema point must be generated.
- For each pair of adjacent slices, matching descriptors must be found.
- Given the matching descriptors, a transform that best maps the extrema points from the image space of slice A into the image space of slice B must be calculated.

3 Implementation details

The specifics of the construction of the image pyramids are thoroughly covered by David G. Lowe[2] and will not be repeated here. Suffice it to say, that a pyramid is a collection of octaves, where each octave represents a reduction of image resolution by a factor of 2. Each octave is partitioned into a set of scales where each successive image is convolved with a Gaussian filter of increasing sigma value. The details of efficient implementation of this are covered by Lowe[2].

3.1 Detecting extrema points

The extrema points are the local minima and maxima points of the Difference-of-Gaussian image pyramids. Lowe[2] proposed looking for an extrema point in a $3 \times 3 \times 3$ neighborhood within a DoG pyramid. However, experimentation has shown that this technique does not yield strict extrema points that are greater than or less than all of the neighbors. Relaxing the extrema criteria to allow the extrema point to be equal to its neighbors yields a large number of adjacent extrema points. Therefore, an alternative method of extrema detection is proposed.

Let D_1 be a non-boundary image within the DoG pyramid. Let D_0 be the image preceeding D_1 in the pyramid, and D_2 the succeeding image. Assuming there are minima points within the D_0, D_1, D_2 slices of the pyramid, calculate

$$A_{min} = D_0 - D_1$$

$$B_{min} = D_2 - D_1$$

The resulting images A_{min} and B_{min} are thresholded to remap the negative values to zero. The minima point image is calculated as

$$E_{min} = A_{min} \times B_{min}$$

The maxima point image is calculated analogously.

$$A_{max} = D_1 - D_0$$

$$B_{max} = D_1 - D_2$$

Again, A_{max} and B_{max} are thresholded to remap the negative values to zero. The maxima point image is calculated as

$$E_{max} = A_{max} \times B_{max}$$

The resulting extrema point images E_{min}, E_{max} are thresholded to isolate strong maxima, and an 8-connected clustering algorithms is used to detect the peaks. For each cluster, the key point is positioned at the center-of-mass of the cluster.

3.2 Detecting descriptor orientations

The descriptor has to be rotationally invariant, therefore it is necessary to select a consistent frame of reference for sampling the neighborhood around the extrema point. The method that is currently implemented in the application follows the one described by Lowe[2]. Essentially, the neighborhood gradient orientation angles are accumulated into a 1D histogram. Each contribution is weighed by the gradient magnitude and a 2D Gaussian weighting function centered at the extrema point. The peaks of the histogram define the feature vector orientation angles.

3.3 Generating the descriptors

During experimentation, several different descriptor generators were evaluated, including 2 versions of the descriptor recommended by Lowe[2]. All of them share the following properties:

- The descriptors are based on extrema point neighborhood properties derived from the image (such as the gradient vector image, or the extrema image).
- The neighborhood is sampled within a local coordinate system based on the descriptor orientation angle.
- The radius of the sampling window has to be large enough (in pixels) to capture the neighborhood properties.

The major difference between the alternate descriptor generators and the design proposed by Lowe rests in the way the sampling window is partitioned. Lowe recommends that the descriptor consist of a 4×4 cell grid of 8-bin gradient orientation histograms, which leads to a 128 dimensional descriptor vector. The downside of this design is that it discards information that falls outside the grid. The alternative design partitions the neighborhood into a set of concentric annuli, where each annulus is partitioned into a set of cells of equal area. Each cell may hold an orientation histogram as suggested by Lowe, or some other information (such as the average extrema intensity values extracted from E_{min} and E_{max}, or dominant gradient vector angle). Unfortunately, experimentation with alternative descriptor designs has not shown performance improvement over the design proposed by Lowe. The performance was evaluated in terms of the number of known matching descriptors being correctly matched using brute force matching.

3.4 Matching descriptors

The matching process is slightly different from the one outlined by Lowe. Lowe addresses a more general computer vision problem, where detection of the same object at different scales is important. The electron transmission microscopy images are typically taken at the same scale, and undergo minor deformation on the global scale, making the scale invariant feature matching unnecessary. Therefore, for the purposes of TEM image registration, the descriptors are matched against other descriptors selected from the same octave and scale of the pyramid. In order to achieve scale invariance, all that is required is the matching of descriptors from any octave and scale of a pyramid against any other octave and scale of the other pyramid. This would, of course, increase the number of mismatches.

A brute force implementation of descriptor matching is not unreasonable for the purposes of this project. However, following in Lowes footsteps the current implementation uses an optimized kd-tree[5] with a best-bin-first nearest neighbor search algorithm[3].

3.5 Filtering out bad matches

Lowe has suggested two filtering stages for removing poorly matched descriptors.

The first stage is based on the thresholding of the ratio of Euclidian distance (in descriptor space) between the query descriptor and its closest match to the distance between the query descriptor and its second closest match. This is founded on the observation that a well matched descriptor is usually distinct enough from the second closest match that the ratio of distances would fall below 0.5, where as the ratio of distances for a mismatched descriptor and its second closest match is typically greater than 0.5.

Unfortunately, our experimental results on matching TEM images have shown that the ratio of the descriptor distance ratio between closest and second closest match is not nearly as well separated for correct matches and mismatches, therefore this property can not be used for filtering out bad matches, as it discards practically all of the correct matches as well.

The second stage proposed by Lowe is based on clustering with the Hough transform[7], which will not be covered here. Suffice it to say that in our implementation it was not as effective as the alternative method described below. The performance of the two filters was compared in terms of the ratio of the detected correct matches to the number of matches in the filtered set.

An alternative filter that appears to be extremely effective for TEM images is based on the ratio of the distance (in image space) between nearest extrema points in image S_i, to the distance between their matching points in the image S_{i+1}. This filter relies on the assumption that the scales of the images being matched are the same, which is true for the TEM images. Since the scales are the same, the distance between nearest neighbors in one image and the matching image should be nearly identical. If the ratio of the two distances deviates significantly from 1.0, it can be assumed that one of the matches is wrong. When it is determined that one of the points is mismatched, both of the matches are discarded. The downside of this filtering approach is that for every discarded mismatch, it may also be discarding a good match as well.

3.6 Estimating the transform

The remaining set of matches may still contain some mismatches, which presents a problem for a Least Squares solution. Matthew Brown[4] proposed the use of RANSAC[6] to select a set of matches that define a consistent transform.

Essentially, a few matches are selected at random to solve for the transform parameters. The number of initially selected matches depends on the number of transform parameters. For example, a 2nd order (affine) bivariate Legendre polynomial transform has 6 parameters, it therefore requires 3 distinct matches. A 4th-order bivariate Legendre polynomial transform has 20 parameters, it requires 10 distinct matches.

Once a transform has been estimated, the rest of the matches are verified as inliers or outliers. For each match point pair, the point expressed in the space of image S_i is mapped via the transform into the space of image S_{i+1}. The distance of the mapped point to its match is used to classify the match as an inlier or an outlier based on some threshold. The inliers and the original set of matches are then used to re-estimate the transform. This can be an iterative process, where at each iteration the matches are classified as inliers and outliers, until convergence or a maximum number of iterations is reached. Since the goal is to optimize the number of inliers, the process is repeated with a new set of initial random matches, and the best results are kept.

For further improvement, it is possible to sort the matches according to some metric, such as the Euclidean distance between the descriptors in the descriptor space. Then, instead of uniform sampling, importance sampling may be used to select initial matches for RANSAC.

3.7 Further refinement of the transform estimate

Given a transform estimate, it may be possible to rematch the descriptors between the two images by restricting the set of match candidates to a local neighborhood within the transform target image space.

For example, an initial set of descriptor matches may be used to estimate a low order transform (e.g. affine) between images S_i and S_{i+1}. Given the low order transform, each descriptor from image S_i is mapped into image S_{i+1}. Only the descriptors that fall within a local neighborhood of the mapped descriptor are considered for matching. This eliminates a number of potential mismatches that would be inconsistent with the affine transform. Once all the descriptors have been re-matched, RANSAC can be used again to estimate a higher order transform.

4 Results

An example of typical images that must be processed by our application is given in figure 1 on the following page. A Difference-of-Gaussian and a gradient vector pyramid of 2 octaves with 3 scales per octave was constructed for each image. The extrema of the DoG pyramid are detected: 2951 points in the left image, 2953 points in the right image. For each detected extrema point the local gradient vector neighborhood is examined to determine dominant gradient vector orientations. For each detected orientation a descriptor is constructed. This results in 4732 descriptors in the left image, and 4601 descriptors in the right image. An illustration of the detected descriptors is given in figure 2 on the next page. The descriptors are matched resulting in 4601 matches. These matches are filtered down to 459 matches – see figure 3 on page 6 for an illustration. RANSAC is used to select inliers consistent with an affine transform which results in 165 matches illustrated in figure 4 on page 6. The resulting registration is shown in figure 5 on page 7.

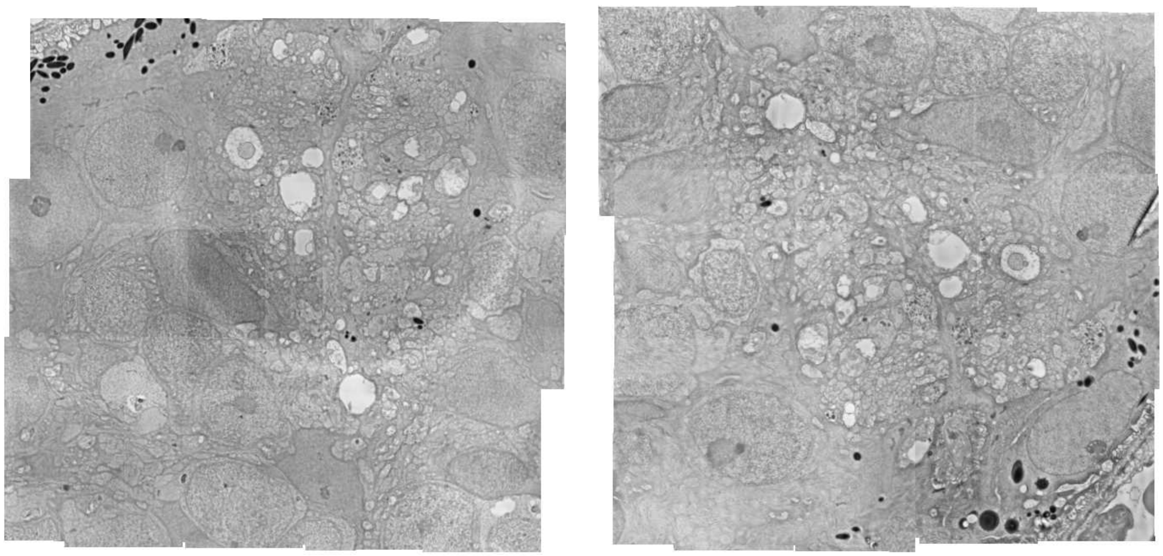
References

[1] Lindeberg, T. 1994. Scale-space theory: A basic tool for analysing structures at different scales. *Journal of Applied Statistics*, 21(2):224-270.

[2] Lowe, D.G. 2004. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*.

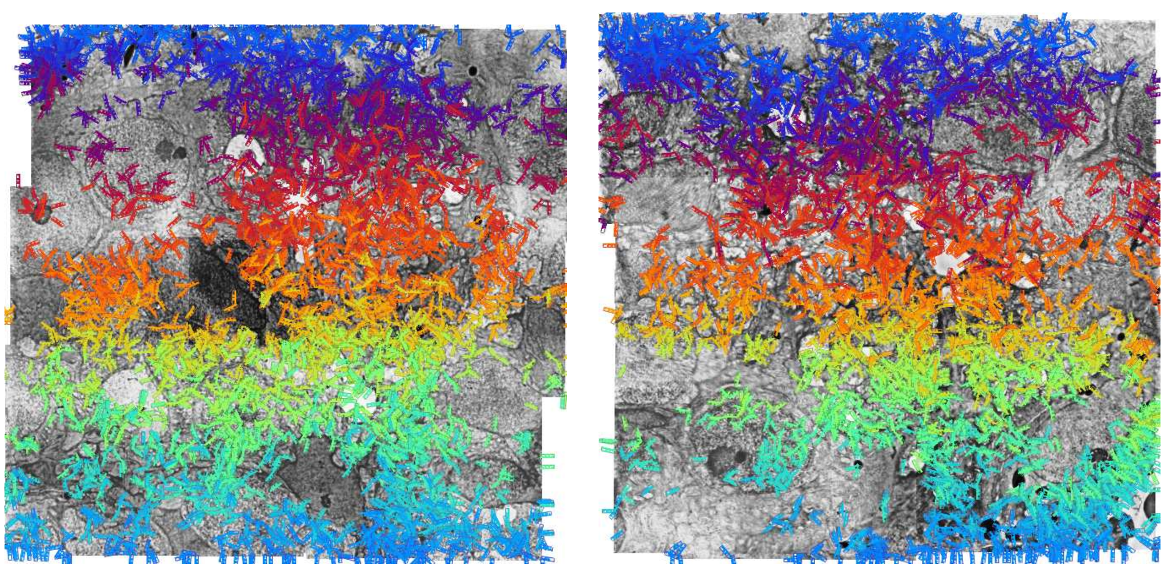
[3] Beis, J. and Lowe, D. G. 1997. Shape Indexing Using Approximate Nearest-Neighbour Search in High-Dimensional Spaces. In *Conference on Computer Vision and Pattern Recognition*, Puerto Rico, pp. 1000-1006.

Figure 1: Sample slices



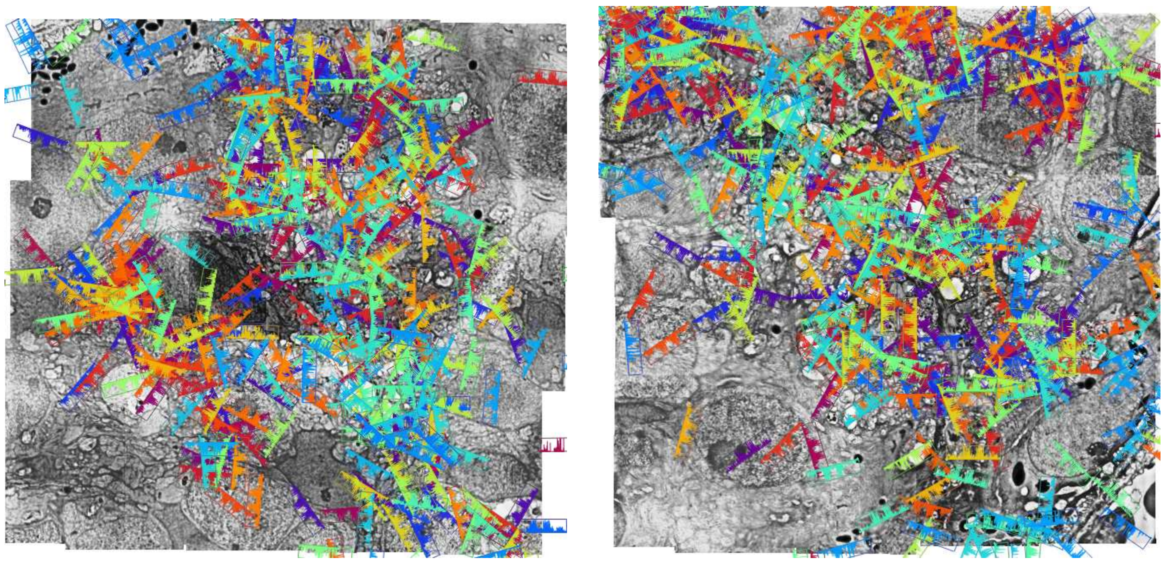
Two consecutive slices. Each slice was assembled from 12 high resolution Transmission Electron Microscopy images of a rabbit retina.

Figure 2: The descriptors



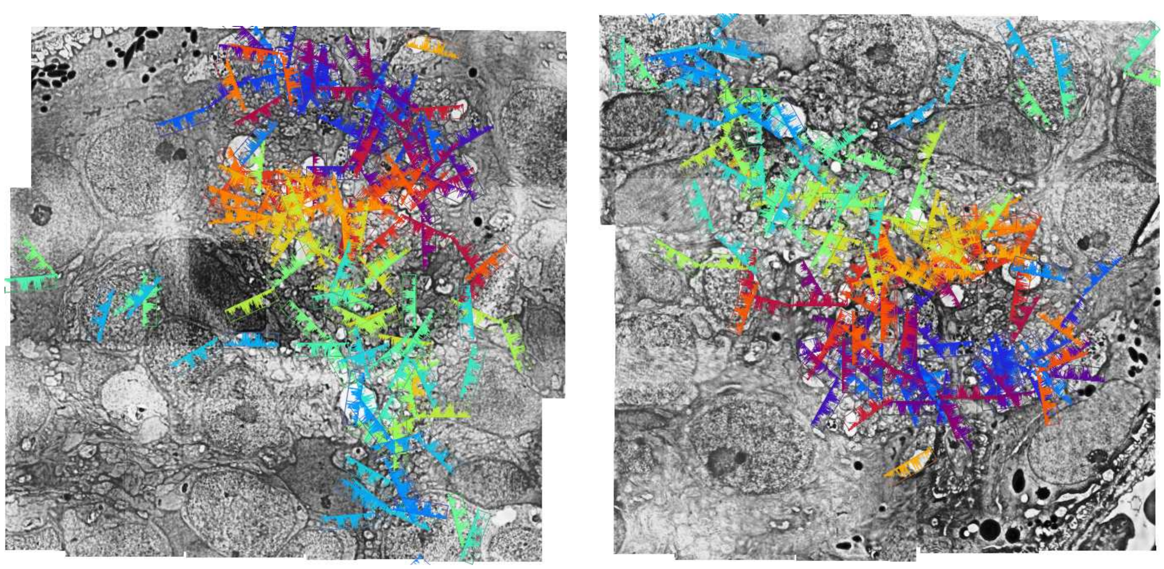
Visualization of the unmatched descriptor vectors detected in the two images: 4732 descriptors in the image on the left, 4601 – on the right.

Figure 3: The filtered descriptor matches



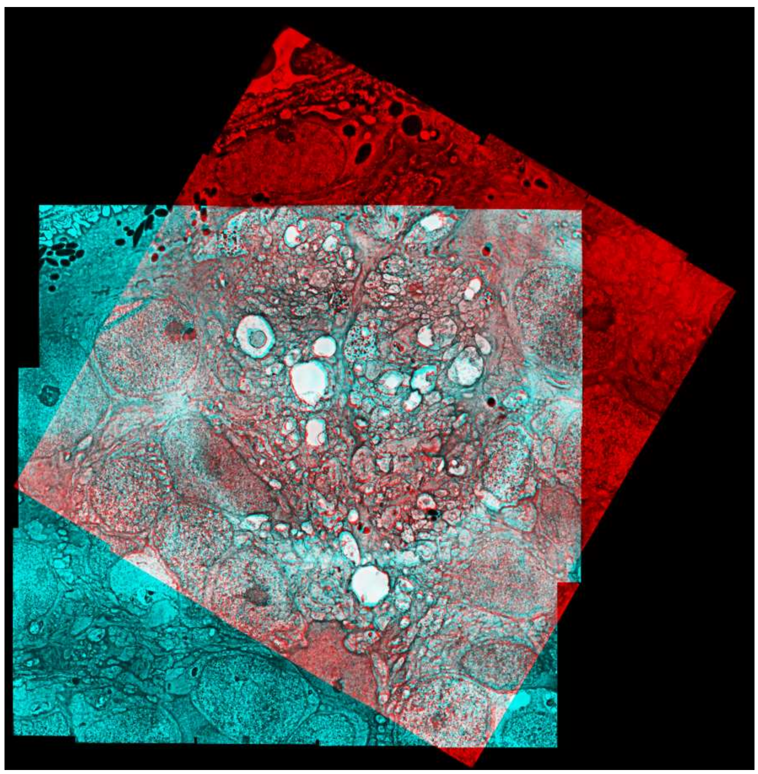
Visualization of the filtered matches – 459 out of 4601 matches remain.

Figure 4: RANSAC filtered matches



Visualization of the RANSAC filtered matches – 165 consistent matches are selected out of 459 remaining matches.

Figure 5: slice to slice registration



Visualization of the slice-to-slice registration results.

[4] Brown, M. and Lowe, D.G. 2002. Invariant Features from Interest Point Groups. In *British Machine Vision Conference*, Cardiff, Wales, pp. 656-665.

[5] Friedman, J.H., Bentley, J.L. and Finkel, R.A. 1977. An Algorithm for Finding Best Matches in Logarithmic Expected Time. *ACM Transactions on Mathematical Software*, 3(3):209-226.

[6] Fischler, M.A. and Bolles, R.C. 1981. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Communications of the ACM*, 24(6):381-395.

[7] D. H. Ballard. 1981. Generalizing the Hough transform to detect arbitrary patterns. *Pattern Recognition*, 13(2):111-122.