Chapter 6

Automatic Multiview Image Registration

6.1 Introduction

The automatic construction of large, high-resolution multi view image registration is an active area of research in the fields of computer vision, image processing, and computer graphics. Multiview image registration can be used for many different applications. The most traditional application is the construction of large aerial and satellite photographs from collections of images, construction of virtual travel etc. This proposed Automatic feature based image registration method does not allow any user interaction and perform all registration steps automatically. Here the matching points are found automatically using local feature detector i.e. harris corner detector which find invariant features using feature descriptors as oriented patches. For estimating homography between detected features of images to be registered, Homography estimator i.e. modified RANSAC (RANdom SAmple Consensus) algorithm, and direct linear transformation algorithm is used. Here features are located at Harris corners (new improved) in discrete scale-space and oriented using a blurred local gradient. The density of features in the image is controlled using a novel adaptive non-maximal suppression algorithm, which gives a better spatial distribution of features. Feature matching are achieved using RANSAC which also uses DLT (Direct Linear Transformation) and warping is applied to achieve final registered image. This proposed algorithm can be applied for the series of images that may or may not be in the same alignment as per desired output image, thus mainly scaling, rotation and image transformation must be done to the series of images to get the proper aligned image.

6.2 Local Feature Detection

Image content is transformed into local feature coordinates that are invariant to translation, rotation, scale, and other imaging parameters [57].

Advantages of local features are
1. Locality: features are local, so robust to occlusion and clutter (no prior segmentation)

2. Distinctiveness: individual features can be matched to a large database of objects

3. Quantity: many features can be generated for even small objects

4. Efficiency: close to real-time performance

5. Extensibility: can easily be extended to wide range of differing feature types, with each adding robustness

In image flat region consists of no change in intensity in all directions, edge consists of no change along the edge direction and corner means significant changes in all directions. So in corner shifting a window in any direction should give a large change in intensity. Corners (junction of contours) are considered as more stable features over changes of viewpoints. They are the points that have high curvature and lie in the junction of different brightness regions of images. In a variety of image features, corners are not affected by illumination and have the property of rotational invariance. They are only about 0.05% in the whole pixels. Without losing image data information, extracting corners can minimize the processing data [58]. Corners in images represent a lot of important information. Extracting corners accurately is significant to image processing, which can reduce much of the calculations.

It is desirable for a corner detector to satisfy a number of criteria [59]
1. All "true corners" should be detected.
2. No "false corners" should be detected.
3. Corner points should be well localized.
4. Detector should have a high repeatability rate (good stability).
5. Detector should be robust with respect to noise.
6. Detector should be computationally efficient.

### 6.2.1 Interest Point Detection

From the literature [58,59], it has been concluded that methods based on intensity, when compared in stability and the runtime of each algorithm, harris corner detection algorithm is superior
Chapter 6 Automatic Multiview Image Registration

to other corner detection algorithm. Thus for finding the interest points, multi-scale harris corner
detector is used. In our research unlike haris corner detector, we have not consider user detected
corner points for accurate point selection and function of cornerness is also modified to give
more accuracy then harris corner response. Harris corner detection algorithm is realized by calcul-
ating each pixel’s gradient. If the absolute gradient values in two directions are both great, then
judge the pixel as a corner.

Following steps are followed for detection of corners as interest point.

1. For each input image \( I(x, y) \), first Gaussian smoothing is applied which has isotropic charac-
teristics so it is easy to filter out weak corners. Gaussian filters are a class of linear smoothing fil-
ters with the weights chosen according to the shape of a Gaussian function

Gaussian low-pass filters in two dimensions

\[
g_1(x, y) = e^{-\frac{D^2(x,y)}{2\sigma^2}} \quad (6.1)
\]

Where \( D(x,y) \) is the distance from the point \((x,y)\) to the origin of the frequency rectangle for an
MxN Image and given by

\[
D(x, y) = \left[ \left( x - \frac{M}{2} \right)^2 + \left( y - \frac{N}{2} \right)^2 \right]^{\frac{1}{2}} \quad (6.2)
\]

\( \sigma \) is the standard deviation and is a measure of spread of the Gaussian curve
so convolving \( I \) with \( g_1 \) gives Gaussian smoothed image \( I_g \).

\[
I_g(x, y) = I(x, y) * g_1(x, y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} I(s,t)g_1(x+s, y+t) \quad (6.3)
\]

Where \( a = \frac{M-1}{2}, b = \frac{N-1}{2} \)

2. For input image \( I(x,y) \), change of intensity for shift \([u,v]\) is given by
Chapter 6 Automatic Multiview Image Registration

\[ E(u, v) = \sum_{x,y} [I(x + u, y + v) - I(x, y)]^2 \]  

(6.4)

Where \( I(x+u,y+v) \) is shifted intensity and \( I(x,y) \) is intensity. For nearly constant patches, \( E(u,v) \) will be near 0. For very distinctive patches, this will be larger. Hence, we want patches where \( E(u,v) \) is large.

Now using Taylor series for 2D functions

\[
I(x + u, y + v) = I(x, y) + uI_x(x, y) + vI_y(x, y) + \frac{1}{2!}\left[ u^2I_{xx}(x, y) + v^2I_{yy}(x, y) \right] + \frac{1}{3!}\left[ u^3I_{xxx}(x, y) + u^2vI_{xxy}(x,y) + uv^2I_{xyy}(x,y) \right] \\
+ \ldots \quad \text{(Higher order terms)}
\]

First order approximation

\[
I(x + u, y + v) \approx I(x, y) + uI_x(x, y) + vI_y(x, y)
\]  

(6.5)

\[
E(u, v) = \sum_{x,y} [I(x + u, y + v) - I(x, y)]^2 \\
\approx \sum_{x,y} [I(x, y) + uI_x + vI_y - I(x, y)]^2 \\
= \sum u^2I_x^2 + 2uvI_xI_y + v^2I_y^2
\]

Rewrite as matrix equation

\[
= \sum [u \ v] \begin{bmatrix} I_x^2 & I_xI_y \\ I_xI_y & I_y^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}
\]

\[
= [u \ v] \sum \begin{bmatrix} I_x^2 & I_xI_y \\ I_xI_y & I_y^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}
\]  

(6.6)

For small shifts \([u \ v]\) we have a bilinear approximation:

\[
E(u, v) \approx [u \ v] M \begin{bmatrix} u \\ v \end{bmatrix}
\]

Where \( M \) is a 2x2 matrix computed from image derivatives and given by
\[
M = \begin{bmatrix}
I_x^2 & I_x I_y \\
I_x I_y & I_y^2
\end{bmatrix}
\] (6.7)

Where \( I_x \) and \( I_y \) are first order partial derivatives of \( I(x, y) \) in x direction and y direction respectively and given by

\[
I_x (x, y) = I(x, y - 1) - I(x, y + 1)
\] (6.8)

\[
I_y (x, y) = I(x - 1, y) - I(x + 1, y)
\] (6.9)

3. To compute \( M \), following images are computed.

\[
a(x, y) = \sum_{x, y} I_x (x, y)^2
\] (6.10)

\[
b(x, y) = \sum_{x, y} I_x (x, y) I_y (x, y)
\] (6.11)

\[
c(x, y) = \sum_{x, y} I_y (x, y)^2
\] (6.12)

These a,b,c are convolve with Gaussian LPF, given in equation (6.1) because first-order directional differentials are sensitive to noise.

\[
A(x, y) = a(x, y) * g_1(x, y)
\]

\[
B(x, y) = b(x, y) * g_1(x, y)
\]

\[
C(x, y) = c(x, y) * g_1(x, y)
\]

Let \( \lambda_1 \geq \lambda_2 \) be the two eigen values of the matrix \( M \). As \( M \) is symmetric and positive, both \( \lambda_1 \) and \( \lambda_2 \) are non-negative. Interpretation for these eigen values are given as

(i) In a uniform and homogeneous region, \( \lambda_1 = \lambda_2 = 0 \).

(ii) At the location of a step edge, \( \lambda_1 > \lambda_2 = 0 \), the corresponding eigenvector for \( \lambda_1 \) is associated with the direction that is orthogonal to the edge.

(iii) At the location of a corner, \( \lambda_1 \geq \lambda_2 > 0 \). The larger are the values of \( \lambda_1 \) and \( \lambda_2 \), the higher are the contrasts of the edges orthogonal to the directions of the corresponding eigenvectors. Thus,
the eigenvectors encode the edge directions and the eigenvalues encode the edge magnitudes. This implies that a corner should be marked at a location where the smaller eigenvalue, $\lambda_2$, is large enough.

4. In our research, we have not used the function given by Harris for measuring the corner response. We have taken the corner strength function given in [57].

We have the corner strength function given in [57]

$$F_{cr} = \frac{\det M(x, y)}{\text{trace}(x, y)}$$  \hspace{1cm} (6.13)

So in terms of eigenvalues $\lambda_1$ and $\lambda_2$

$$F_{cr} = \frac{\lambda_1 \lambda_2}{\lambda_1 + \lambda_2}$$  \hspace{1cm} (6.14)

$$F_{cr} = \frac{I_x^2 \cdot I_y^2 - I_x I_y \cdot I_x I_y}{I_x^2 + I_y^2}$$  \hspace{1cm} (6.15)

This $F_{cr}$ depends on eigenvalues of $M$ only and is large for corners.

### 6.2.2 Non Maxima Suppression

Here these corners strength are used for selecting the matching points for images to be registered. Thus the computational cost of matching is more for the number of interest points selected using equation (6.14) so it is desirable to restrict the maximum number of interest points extracted from each image. At the same time, it is important that interest points are spatially well distributed over the image, since for multi view image registration, the area of overlap between a pair of images may be small. To satisfy these requirements, we have developed a non-maximal suppression algorithm to select a fixed number of interest points for each image.

To achieve this first the corner close to image borders are removed, for that border pixels are made zero in $M$. Interest points are suppressed based on the corner strength $F_{cr}$ and only those that are a maximum in a neighborhood of $3x3$ pixels are retained, thus interest points with the highest corner strength are selected.

Figure 6.1 shows the flow diagram for finding interest points.
6.3 Extraction of Feature Descriptor

After determination of interest points, we need to extract a description of the local image structure that will support reliable and efficient matching of features across images to be registered. Given an interest point \((x, y)\) the descriptor is formed by a patch centered at \((x, y)\). We sample a \(21 \times 21\) patch of pixels around the sub-pixel location of the interest point. Now these extracted patch descriptors around detected local features descriptors are blurred by a (fairly large) Gaussian with \(\sigma=3\) pixels. So for interest point \((x,y)\) the patch is given by

\[
P = I(x-w : x+w, y-w : y+w) \quad \text{where } w=10 \text{ for patch of } 21 \times 21.
\]

If we described image descriptors by \(D\), then each column in \(D\) is one vectorized descriptor of dimension \(441(21\times21\text{ pixels})\).

6.4 Matching of Local Features

After finding the descriptors of images to be registered, the tentative matches are computed by matching descriptor. It is required to find nearest neighbors for all features from all images,
which is the well known all nearest neighbors problem. To match the descriptors, euclidean distance between descriptors are found. Suppose if \( D_a \) is descriptors of image \( I_a \) and \( D_b \) is descriptor of image \( I_b \), where \( D_a \) and \( D_b \) are of size \( M \times N_1 \) and \( M \times N_2 \) respectively (transpose becomes size of \( N_1 \times M \) and \( N_2 \times M \)), then Euclidean distance between them is given by

\[
D_E(N_1, N_2) = \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} [D_a(i,:)-D_b(j,:)]^2
\]  

(6.16)

To reduce computation load, we have taken only those descriptors which is having minimum euclidean distance. Thus \( D_E \) is shorted in terms of minimum euclidean distance. To obtain distinctive features, we use the ratio of distances to the first and the second nearest neighbour, which has been taken 1 (i.e. below 0.8) so that the best match (1st nearest neighbour) should be much closer than the next best match (the 2nd nearest neighbour). Here only those interest points are selected from both images to be registered, whose descriptors are same or having minimum euclidean distance, so as a result we can have tentative matches between Image \( I_a \) and \( I_b \).

### 6.5 Homography Estimation

A homography is a 2D projective transformation that maps points in one plane to another. In our case, the planes are images. In essence, a homography \( H \) maps 2D points (in homogeneous coordinates) according to

\[
\begin{bmatrix}
x' \\
y' \\
w'
\end{bmatrix} = \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} x \\ y \\ w \end{bmatrix}
\]  

(6.17)

\[0R\]

\[X'=HX\]  

(6.18)

Points in homogeneous coordinates are only defined up to scale so that \( X = [x, y, w] = [\alpha x, \alpha y, \alpha w] = [x/w, y/w, 1] \) , all refer to the same 2D point. Here points are normalized with \( w = 1 \) to have a unique identification of the image coordinates \( x, y \). The extra coordinate makes it easy to represent transformations with a single matrix.
The RANdom SAmple Consensus (RANSAC) algorithm proposed by Fischler and Bolles[48] is a general parameter estimation approach designed to cope with a large proportion of outliers in the input data. RANSAC is a resampling technique that generates candidate solutions by using the minimum number observations (data points) required to estimate the underlying model parameters. Unlike conventional sampling techniques that use as much of the data as possible to obtain an initial solution and then proceed to prune outliers, RANSAC uses the smallest set possible and proceeds to enlarge this set with consistent data points [62]. It is introducing a classification of the data into inliers (valid points) and outliers while estimating the optimal transformation for the inliers. For a number of iterations, a random sample of 4 correspondences is selected and a homography H is computed from those four correspondences. Each other correspondence is then classified as an inlier or outlier depending on its concurrence with H. After all the iterations are done, the iteration that contained the largest number of inliers is selected. H can then be recomputed from all of the correspondences that were considered as inliers in that iteration. In our research we have used modified RANSAC algorithm to estimate robust homography.

Following steps are followed to achieve 2D homography between images to be registered using modified RANSAC.

### 6.5.1 Normalization

To obtain 2D homography, Direct linear transform (DLT) is used. The result of the DLT algorithm for computing homographies, depends on the coordinate frame in which the points are expressed, means it is dependent on the origin and scale of the coordinate system in the image. This is a very undesirable property as it makes the algorithm quite unstable, for this reason it is desirable to normalize the input data before applying the DLT algorithm. Data normalization improves the accuracy of the results while ensuring that the algorithm will be invariant to arbitrary choices in scale and coordinate frame [63]. Each set of points are normalized so that the origin is at centroid and mean distance from origin is \( \sqrt{2} \), and thus the average point is equal to \((1,1,1)\).

For the point correspondence version of DLT, Hartley and Zisserman [64] propose a normalization step to ensure that the solution converges to the correct result. Their normalized DLT algorithm works as follows. Compute a similarity transform \( T \) that takes points \( X \), to a new set of
points $X_i'$ such that the centroid of the points $X_i'$ is the coordinate origin and their average distance from the origin is $\sqrt{2}$.

$$X_i' = T \ast X$$ \hfill (6.19)

Similarity transform $T$ is given by

$$T = \begin{bmatrix} s & 0 & t_x \\ 0 & s & t_y \\ 0 & 0 & 1 \end{bmatrix}$$ \hfill (6.20)

Suppose $X_i = (x_i, y_i, 1)^T$, then we have

$$X_i' = TX_i = \begin{bmatrix} sx_i + t_x \\ sy_i + t_y \\ 1 \end{bmatrix} = \begin{bmatrix} x_i' \\ y_i' \\ 1 \end{bmatrix}$$ \hfill (6.21)

The centroid of $x_i$ is

$$\bar{x'} = \left( \frac{\sum x'_i}{n} \right) = (0,0,1)^T$$

Where $\bar{x}$ denotes mean, therefore

$$t_x = -s \cdot \bar{x}$$
$$t_y = -s \cdot \bar{y}$$ \hfill (6.22)

and $\bar{x}$ and $\bar{y}$ are mean values of $x$ and $y$ coordinates of points.

The average distance between $X_i'$ and origin is

$$\frac{1}{n} \sum \sqrt{x_i'^2 + y_i'^2} = \frac{1}{n} \sum \sqrt{(sx_i - \bar{x})^2 + (sy_i - \bar{y})^2}$$
\[
= \frac{5}{n} \sum_i \sqrt{(x_i - \bar{x})^2 + (y_i - \bar{y})^2}
\]

\[= \sqrt{2}\]

Therefore \(s\) can be computed as

\[
s = \frac{\sqrt{2}}{\frac{1}{n} \sum_i \sqrt{(x_i - \bar{x})^2 + (y_i - \bar{y})^2}}
\]

(6.23)

### 6.5.2 RANdom SAmple Consensus (RANSAC) Algorithm

To achieve homography estimation using RANSAC, sets of four \((r = 4)\) feature correspondences are taken and homography \(H\) is computed between them using the direct linear transformation (DLT) method. We repeat this with \(n = 500\) trials and select the solution that has the maximum number of inliers.

Following section presents the detail steps to be followed.

#### 6.5.2.1 Sample Selection

It is required to select randomly, the minimum number of points to determine the model parameters. Here we have selected 4 points.

#### 6.5.2.2 Colinearity Check

These selected points are checked whether these points are colinear, (which is not checked in based algorithm given by Fischler and Bolles[62]) if so, redo the above step.

Here we have tested any three of the four points in each set is collinear or not.

Assume that \(p_1, p_2, p_3\) and \(p_4\) are selected points of an image.

Now to check whether these points are collinear or not,

\[
R = |(p_1 \times p_2) \cdot p_3| < \text{eps}
\]

(6.24)
Where symbol \( \times \) denotes the cross product and symbol \( \cdot \) denotes the dot product. And epsilon indicates largest floating number possible, or the constant epsilon or machine epsilon is defined to be the distance between 1 and the next largest floating point number. In IEEE arithmetic, we have epsilon=\(2^{-52}\).

If \( p_1 \) and \( p_2 \) are collinear then their cross product is zero and dot product with zero will results in \( R=0 \), or if cross product of \( p_1 \) and \( p_2 \) is a normal vector to plane, then dot product of this normal with \( p_3 \) is zero then \( p_3 \) also lies in the plane, hence co-linear. If selected points are homogeneous with equal scale then also \( R=0 \), means points are collinear. So if selected points are collinear then again new random points are selected.

### 6.5.2.3 Direct Linear Transformation (DLT)

To calculate 2D homography, Direct Linear Transformation (DLT) algorithm given by Hartley and Zisserman[64] is used.

DLT algorithm is used for finding linear mappings between any two data sets, given a certain number of corresponding data points between the sets. Direct linear transformation (DLT) algorithm is used to solve for \( H \). First, we establish \( N \) correspondences between correspondence points \( X_i = (x_i, y_i, w_i)^T \) and \( X'_i = (x'_i, y'_i, w_i)^T \), where \( i = 1 \ldots N \). In our case \( N=4 \), the relationship between \( X_i \) and \( X'_i \) may be expressed as a vector cross product \( X_i \times H X'_i = 0 \), relation between \( X_i \) and \( X'_i \) is given in equation (6.18). Here algorithm is developed only for a set of four 2D to 2D point correspondence.

If the \( j^{th} \) row of matrix is denoted by \( h^{JT} \), then

\[
HX_i = \begin{pmatrix} h^{JT}X_i \\ h^{2T}X_i \\ h^{3T}X_i \end{pmatrix}
\]  

(6.26)
Chapter 6 Automatic Multiview Image Registration

\[ X_i' \times HX_i = \begin{pmatrix} y_i'h^{\text{T}}X_i - w_i'h^{\text{T}}X_i \\ w_i'h^{\text{T}}X_i - x_i'h^{\text{T}}X_i \\ x_i'h^{\text{T}}X_i - y_i'h^{\text{T}}X_i \end{pmatrix} \]  \hspace{1cm} (6.27)

Since \( h^{\text{T}}X_i = X_i'h^j \) for \( j=1,2,3 \). This gives a set of three equations in the entries of \( H \), which may be written in the form

\[
\begin{bmatrix}
0^T & -w_i'X_i^T & y_i'X_i^T \\
w_i'X_i^T & 0^T & -x_i'X_i^T \\
y_i'X_i^T & x_i'X_i^T & 0^T
\end{bmatrix}
\begin{bmatrix}
h_1 \\
h_2 \\
h_3
\end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}
\hspace{1cm} (6.28)
\]

Now in our algorithm we have taken \( X_i' = (x_i', y_i', 1)^T \) and \( X_i = (x_i, y_i, 1)^T \)

\[
\begin{bmatrix}
0^T & X_i' & y_i'X_i^T \\
X_i' & 0^T & -x_i'X_i^T \\
y_i'X_i^T & x_i'X_i^T & 0^T
\end{bmatrix}
\begin{bmatrix}
h_1 \\
h_2 \\
h_3
\end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}
\hspace{1cm} (6.29)
\]

These equations have the form \( A_ih = 0 \), where \( A_i \) is a 3 \( \times \) 9 matrix and \( h \) is a matrix made up the entries of the matrix \( H \).

\[
h = \begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix}, \quad H = \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix}
\hspace{1cm} (6.30)
\]

\[
A = \begin{bmatrix}
0 & 0 & 0 & -x_i & -y_i & 0 & x_iy_i & y_iy_i & y_i' \\
x_i & y_i & 1 & 0 & 0 & 0 & -x_iy_i & -y_ix_i & -x_i' \\
x_iy_i & -y_ix_i & -y_i & x_i & y_i & x_i & x_i' & 0 & 0 \\
-x_iy_i & -y_ix_i & -y_i & x_i & y_i & x_i & x_i' & 0 & 0 \\
-x_iy_i & -y_ix_i & -y_i & x_i & y_i & x_i & x_i' & 0 & 0
\end{bmatrix}
\hspace{1cm} (6.31)
\]

So now we have to solve following equation where \( N=4 \) (no. of selected points)

\[
A_{13\times9} h_{9\times1} = 0
\hspace{1cm} (6.32)
\]
This can be solved using Singular Value Decomposition.

### 6.5.2.4 Singular Value Decomposition

Equation (6.32) can be solved using Singular Value Decomposition. The singular value decomposition (SVD) is a widely used technique to decompose a matrix into several component matrices, exposing many of the useful and interesting properties of the original matrix [64]. The decomposition of a matrix is often called a factorization. SVD is based on a theorem from linear algebra which says that a rectangular matrix $A$ can be broken down into the product of three matrices - an orthogonal matrix $U$, a diagonal matrix $S$, and the transpose of an orthogonal matrix $V$ [66].

The theorem is presented as

$$ A_{mn} = U_{mn}S_{nn}V_{nm}^{T} \quad (6.33) $$

where $U^T \cdot U = I$, $V^T \cdot V = I$, the columns of $U$ are orthonormal eigenvectors of $AA^T$, the columns of $V$ are orthonormal eigenvectors of $A^TA$, and $S$ is a diagonal matrix containing the square roots of eigenvalues from $U$ or $V$ in descending order. So for equation (6.32), we have to find eigenvector of $A$ for which eigenvalue is zero. In equation (6.33), last column of $V^T$ presents the eigenvector which is having minimum eigenvalue, as diagonal matrix $S$ containing eigenvalues in descending order.

### 6.5.2.5 Denormalization of Homography Matrix

Now homography matrix $H$ is de-normalized as it is normalized as stated in section 6.5.1. If $X_i$ and $X_i'$ are correspond selected points for Image 1 & Image 2 to be registered and $T_1$ and $T_2$ are similarity transform used for normalization of $X_i$ and $X_i'$, and $H'$ is homography matrix, which uses normalized $X_i$ and $X_i'$, then

$$ H = H'T_1 / T_2 \quad (6.34) $$
6.5.2.6 Computation of Inliers

The symmetric transfer error of a homography with respect to a set of matched points as needed by RANSAC, is evaluated by taking the euclidean distances between the selected points and transferred points achieved with homography.

For selected points $X_i$ and $X_i'$ from images to be registered, the transferred points are selected in both directions.

$$HX_i' = HX_i$$

$$HX_i = \frac{X_i'}{h}$$  \hspace{1cm} (6.35)

All the points are normalized so that the homogeneous scale parameter for all coordinates is 1.

Now distance errors are found by

$$D_E = \sum_{i=1}^{N} (X_i - HX_i)^2 + (X_i' - HX_i)^2$$  \hspace{1cm} (6.36)

The inliers are defined as those matches $(x, x')$ for which $D_E$- distance between homogeneous points $x$ and $x'$ and model points using $H$ is less than a distance threshold $t$ [47]. This is repeated many times depending on the expected ratio of inliers to outliers. The final homography estimate is with the highest inliers count. A threshold $t$ is used to insure that none of the inliers deviates from the model by more than $t$. Only those points will be taken for $|D_E| < t$ and will be considered as inliers.

6.5.2.7 Robust Homography Estimation

Homography matrix, for points $x$ and $x'$ will be selected based on maximum number of inliers (whose projections are consistent with $H$ within a given tolerance). If the fraction of the number of inliers over the total number points in the set exceeds a predefined threshold, re-estimate the model parameters using all the identified inliers and terminate, otherwise new correspondence points are found again from the images to be registered.
The number of iterations, N, is chosen high enough to ensure that the probability p (usually set to 0.99, the data with no outliers) that at least one of the sets of random samples does not include an outlier. Let u represent the probability that any selected data point is an inlier and \( v = 1 - u \) the probability of observing an outlier. N iterations of the minimum number of points denoted s are required, where

\[
N = \frac{\log(1 - p)}{\log(1 - (1 - v)^s)} \tag{6.37}
\]

In proposed algorithm we have selected the threshold such that N< trial count, then homography solution can be considered as a best fit. Trial count is started with 1 and at each time for selection of new correspondence points, it will be incremented by one.

Figure 6.2 shows the steps to be followed for computing 2D homography.
Figure 6.2  Flow chart for computing 2D homography
6.6 Image Warping and Resampling Using Bilinear Interpolation

Image warping is a transformation which maps all positions in one image plane to positions in a second plane. Image warping is in essence a transformation that changes the spatial configuration of an image as described in section 5.2.1.

In bilinear interpolation, the intensity at a point is determined from the weighted sum of intensities at four pixels closest to it. Therefore, given location \((X,Y)\) and assuming \(u\) is the integer part of \(X\) and \(v\) is the integer part of \(Y\), the intensity at \((X,Y)\) is estimated from the intensities at \((u,v)\), \((u+1,v)\), \((u,v+1)\), \((u+1,v+1)\). This resampling involves first finding the intensity at \((X,v)\) from the linear interpolation of intensities at \((u,v)\) and \((u+1,v)\). Let this intensity be \(I(X,v)\). Then, the intensity at \((X,v+1)\) is determined from the linear interpolation of intensities at \((u,v+1)\) and \((u+1,v+1)\). Let this intensity be \(I(X,v+1)\). Then, the intensity at \((X,Y)\) is computed from the linear interpolation of intensities at \((X,v)\) and \((X,v+1)\). Figure and detail explanation is given in section 5.2.3. This can be summarized as

\[
I(X,Y) = W_{u,v}I(u,v) + W_{u+1,v}I(u+1,v) + W_{u,v+1}I(u,v+1) + W_{u+1,v+1}I(u+1,v+1) \tag{6.38}
\]

Where

\[
W_{u,v} = (u+1-x)(v+1-y)
\]

\[
W_{u+1,v} = (x-u)(v+1-y)
\]

\[
W_{u,v+1} = (u+1-x)(y-v)
\]

\[
W_{u+1,v+1} = (x-u)(y-u)
\]

Here warping returns an interpolated warp of an image under homography. Warping is done on the images using the estimated homography. Here backward mapping is used so for that first it is necessary to define a registered image (in terms of index) to warp all the images onto, so second image will be taken as reference image and map this image to the origin of the registered image using the identity homography. Now first image will be registered with second using homography matrix and bilinear interpolation.

**Algorithm 6.1** Automatic Multiview Registration
**Input**: Images A and B (or more than two) of size M x N to be registered

**Output**: Registered Image

1. Apply Gaussian Smoothing for each input image I(x,y) using equation (6.1),(6.2) and (6.3)

2. Detect interest points using local feature as explained in section 6.2

3. Apply Non maxima suppression to select only spatially well distributed points over the image as explained in section (6.2.2)

4. Define Feature descriptors for selected interest points for both images as described in section (6.3)

5. Match the Local features using Euclidean distances using methods described in section (6.4)

6. Estimate Homography using modified RANSAC, Direct linear transformation(DLT) and Singular Value Decomposition (SVD) as described in section (6.5)

7. Apply Image Warping using backward mapping and interpolation techniques as stated in section 6.6

8. Registered output image.

9. Repeat steps 1 to 8 for registration of more than two images.

10. End
6.7 Results

Experimental results are carried out on different multiview images. Here no selection points are required. Following table 6.1 shows the $N$ iteration for different interest point and different probability of observing an outlier $v$. As described earlier, in this proposed algorithm we have taken $p=0.99$, that at least one of the sets of random samples does not include an outlier.

Let $u$ represent the probability that any selected data point is an inlier and $v = 1 - u$ the probability of observing an outlier. $N_{\text{inl}}$ is number inliers which satisfies the threshold criteria. $N$ iterations of the minimum number of points denoted $s$ (minimum No of points needed to fit a homography) are required, as given in equation (6.37) where

$$N = \frac{\log(1-p)}{\log(1-(1-v)^s)}, \quad w = 1 - v = 1 - \frac{N_{\text{inl}}}{m}$$  \hspace{1cm} (6.39)

Where $m$ is total number of interest points.

<table>
<thead>
<tr>
<th>Probability $w$</th>
<th>s</th>
<th>$N$</th>
<th>$N &gt;$ current iteration count</th>
<th>Further iterations required</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.95</td>
<td>2</td>
<td>2</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>2</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>3</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>3</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>3</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>3</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>4</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>4</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>4</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>4</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>4</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>4</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>5</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>5</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>
Table 6.2 Value of required iteration N for different value of s and w

<table>
<thead>
<tr>
<th>s</th>
<th>W=0.95</th>
<th>W=0.90</th>
<th>W=0.80</th>
<th>W=0.75</th>
<th>W=0.70</th>
<th>W=0.60</th>
<th>W=0.55</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>11</td>
<td>17</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>4</td>
<td>7</td>
<td>9</td>
<td>11</td>
<td>19</td>
<td>35</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>5</td>
<td>9</td>
<td>13</td>
<td>17</td>
<td>34</td>
<td>72</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>6</td>
<td>12</td>
<td>17</td>
<td>26</td>
<td>57</td>
<td>146</td>
</tr>
<tr>
<td>6</td>
<td>4</td>
<td>7</td>
<td>16</td>
<td>24</td>
<td>37</td>
<td>97</td>
<td>293</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>8</td>
<td>20</td>
<td>33</td>
<td>54</td>
<td>163</td>
<td>588</td>
</tr>
<tr>
<td>8</td>
<td>5</td>
<td>9</td>
<td>26</td>
<td>44</td>
<td>78</td>
<td>272</td>
<td>1177</td>
</tr>
</tbody>
</table>

So from table 6.1 and 6.2, we can conclude that as the probability for inliers is more, the required iterations are less. Threshold for selecting homography matrix =0.01, which gives us maximum registration accuracy in terms of standard deviation and entropy compared to already registered image taken from camera. We have tried this for more than 40 images and it gives us best result.

Figure 6.3 Input Images

Figure 6.4 (a) Extracted interest points from image a  
(b) Extracted features from image b
Figure 6.5 Inliers for Image a

Figure 6.6 Registration of two (a) and three (b) images
Figure 6.7 Input Images

(a)

(b)

Figure 6.8 Registration of two (a) and three (b) images

Figure 6.9 Input images
Figure 6.10  Registration of two (a) and three (b) images

Figure 6.11 Input images
Figure 6.12 Registered image

(a) Image a

(b) Image b

(c)

Figure 6.13 Input images (a-b) and registered image (c)
6.8 Concluding Remark

This chapter has presented a novel system for fully automatic multiview image registration. Our use of invariant local features and a probabilistic model for homography to verify image matches allows us to register image sets, and register them fully automatically without user input. The system is robust to camera zoom, orientation of the input images, and changes in illumination due to flash and exposure settings. A blending scheme ensures smooth transitions between images despite illumination differences, while preserving high frequency details.
In this proposed algorithm, after detecting the interest points, we have also implemented non-maximal suppression that better distributes features across the image so it will improve matching with less overlap. We have tested this algorithm for 40 images and success ratio is 100%. By taking many multiview images, we can significantly increase the field of view of the resultant registered images and remove the need for expensive fisheye lenses.