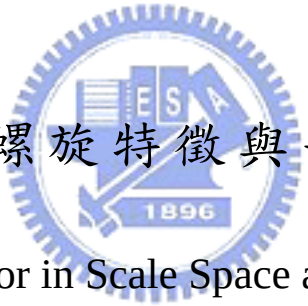
國立交通大學

資訊科學與工程研究所

碩 士 論 文



尺度空間之螺旋特徵與平面影像對位

Spiral Descriptor in Scale Space and Planar Image Registration

研 究 生：林開印

指導教授：陳永昇　教授

中 華 民 國　九 十 七　年 八 月

尺度空間之螺旋特徵與平面影像對位
Spiral Descriptor in Scale Space and Planar Image Registration

研 究 生：林開印　　Student：Kai-Ying Lin

指導教授：陳永昇　　Advisor：Yong-Sheng Chen

國 立 交 通 大 學

資 訊 科 學 與 工 程 研 究 所

碩 士 論 文

A Thesis

Submitted to Institute of Computer Science and Engineering

College of Computer Science

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

in

Computer Science

August 2008

Hsinchu, Taiwan, Republic of China

中華民國九十七年八月

摘 要

在電腦視覺的領域中，影像對位 (image registration) 是一個基礎且重要的問題。其衍生出的議題及應用相當廣泛，包括：立體匹配 (stereo matching)、三維結構重建 (3D structure reconstruction)、物體識別 (object recognition)、移動追蹤 (motion tracking) 等。在本論文中，我們著重於平面影像對位 (planar image registration)。利用創新的影像特徵描述 (image descriptor)：螺旋型特徵 (spiral descriptor)，進行影像匹配 (image matching) 以及影像對應 (image correspondences) 的修正。平面影像的校正以及最佳化也在本論文中一並提出。

此影像對位系統主要包含兩項技術，影像對應的偵測或選取以及單應性矩陣 (homography matrix) 的最佳化。對於影像之對應，我們利用可靠的影像對應法，例如：SIFT，自動取得，但其無法適用於所有情況。所以對於較困難的情況，我們使用人工點選對應點的方式，之後再利用所提出的螺旋型特徵進行修正。此螺旋型的特徵點位置位於尺度空間中 (scale space)，此特徵描述由螺旋型之外觀建立，以達到縮放以及位移不變性 (invariant)。使用動態規劃 (dynamic programming) 進行特徵匹配 (descriptor matching) 適合於旋轉不變性。對於平面影像對位，我們提出一個創新的方法以提升精確度。首先，我們利用自動偵測影像對應或人工點選對應點並且使用螺旋型特徵進行自動修正的方法來求得單應性矩陣。此矩陣可分解成其參數，並且利用非線性的最佳化方法進行調整。最後，最佳之單應性矩陣可以提供高精確度的平面影像對位。

我們所提出的螺旋型特徵可以自動且穩定的進行影像匹配。對於較困難的情況，我們使用人工點選對應點並且自動修正的方式以獲得影像對應。此平面影像對位系統不僅提升了對位精確度，並且提供方便使用者進行平面影像對位的方法。

誌 謝

在這邊我要先感謝陳永昇老師對我的指導，在這兩年於研究上持續給予我方向，使我得以完成這篇碩士論文。相信在研究的日子裡所獲得的經驗及精神，於未來的路上也是受用無窮的。實驗室夥伴們，謝謝你們陪伴我度過這校園修課與做研究的生活。安全監控小組的各位學長學弟，謝謝你們一起參與了我兩年的研究所生涯，提供了許多好的點子與想法，在彼此討論的過程中使我獲益良多。最後要感謝我的家人、女友以及朋友們，因為一直有你們在背後我給予我有形與無形的強大力量，使我得取得碩士學位，謝謝你們！

Spiral Descriptor in Scale Space and Planar Image Registration

A thesis presented

by

Kai-Ying Lin

to

Institute of Computer Science and Engineering

College of Computer Science

in partial fulfillment of the requirements

for the degree of

Master

in the subject of

Computer Science

National Chiao Tung University

Hsinchu, Taiwan

2008

Spiral Descriptor in Scale Space and Planar Image Registration

Abstract

Image registration is a fundamental problem in computer vision, and it also has been used to many research issues including stereo matching, 3D structure reconstruction, object recognition, and motion tracking. In this thesis, we focus on planar image registration. A novel image feature descriptor: spiral descriptor is proposed for image matching and correspondences refinement. The planar image registration and its optimization method is also proposed in this thesis.

The image registration system involves two major techniques, image correspondence detection/selection and homography matrix optimization. For image correspondences, we obtain them automatically using reliable image matching methods like SIFT, but it is impossible for all cases. Therefore, we manually select pairs of corresponding points and refine using proposed spiral descriptor for the hard cases of image matching. The spiral feature points are localized in scale space and the descriptors are built along spiral-shape profile, which can achieve scaling and translation. The dynamic programming technique is used to match spiral descriptors and it is suitable for rotation invariant. For planar image registration, we propose a novel method to promote the registration accuracy. First, we estimate the homography matrix by either detecting the image correspondences automatically or selecting image corresponding points manually and refining using proposed spiral descriptor. The initial homography matrix is decomposed into its parameters and the non-linear optimization process adjusts these parameters using iterative process. Finally, the optimal homography can produce high registration accuracy for planar images.

The proposed spiral descriptor can match images automatically and robustly. For the hard cases of image matching, we select the correspondences manually and refine the positions automatically. The proposed planar image registration system not only promote the registration accuracy but also provides convenience process for user doing image registration.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Image registration is the process of aligning two or more images of the same scene taken at different times, from different viewpoints, and/or by different sensors. It is a fundamental problem in computer vision, and it also has been used to many research issues including stereo matching, 3D structure reconstruction, object recognition, and motion tracking. In general, it is consisted of four major steps.

Feature detection Finding the distinctive objects like corners, edges, contours and regions is the goal of this step. They are likely manually selected or better automatically. Moreover, they further can be denoted by their representatives, like control points or feature points.

Feature matching The correspondences between images are established in this step. Various feature descriptors and similarity measures are used for finding the relationship between features.

Transformation model estimation The parameters of mapping functions for aligning images are estimated in this step. They are computed by the established correspondences, and represent the relationship between images.

Image transformation One image is overlaid on another image by the established functions of transform models. The interpolation technique may apply to those coordinates of non-integer value.

In recent years, SIFT (Scale Invariant Feature Transform) has been proven to be the most reliable method for image matching [21]. In this thesis, we also propose a registration method for planar scene image. In next section, we review the SIFT and its extension methods.

1.2 Related Works of Scale Invariant Feature Transform

SIFT (Scale Invariant Feature Transform) and its applications are proposed by Lowe [16, 17, 18]. As its name, it transforms image data into scale-invariant coordinates relative

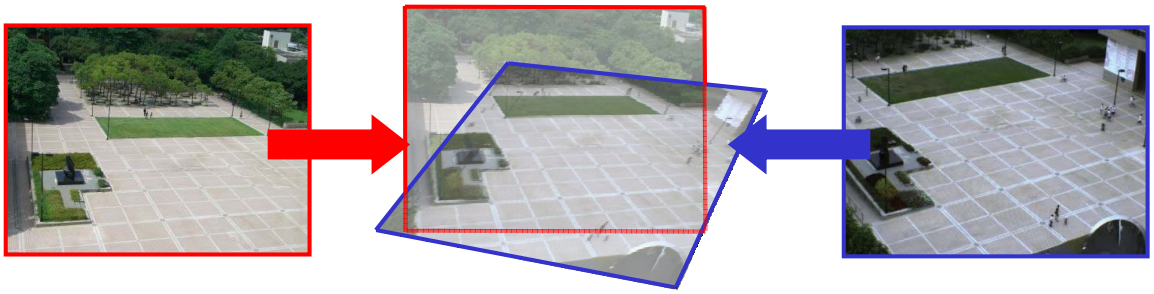


Figure 1.1: **Image registration is the process of overlaying two or more images.**

to local features. SIFT consisted of the following major stages.

1.Scale-space extrema detection

Lindeberg proves that the normalization of Laplacian with factor σ^2 is required for true scale invariance [15]. Mikolajczyk also found that the maxima and minima of scale normalized Laplacian of Gaussian, $\sigma^2\nabla^2 G$, can produce the most stable features [20]. For above reasons, this stage searches over all scales and image locations, and is implemented efficiently by using a difference of Gaussian function to approximate $\sigma^2\nabla^2 G$ (Figure 1.2). Therefore, this detection first builds the Gaussian pyramid by apply Gaussian kernel $G(x, y, \sigma)$ to image $I(x, y)$, and the scale space can represented as

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \tag{1.1}$$

where $*$ denotes the convolution operation. The difference of Gaussian pyramid is also represented as

$$D(x, y, \sigma) = [G(x, y, k\sigma) - G(x, y, \sigma)] * I(x, y) \tag{1.2}$$

where k is a parameter used to decide how close the adjacency octave are in Figure 1.2. If there are s images of an octave, the k is often selected as $2^{\frac{1}{s}}$. The extrema is last located by searching for the minimum and maximum over all scales and locations (Figure 1.3).

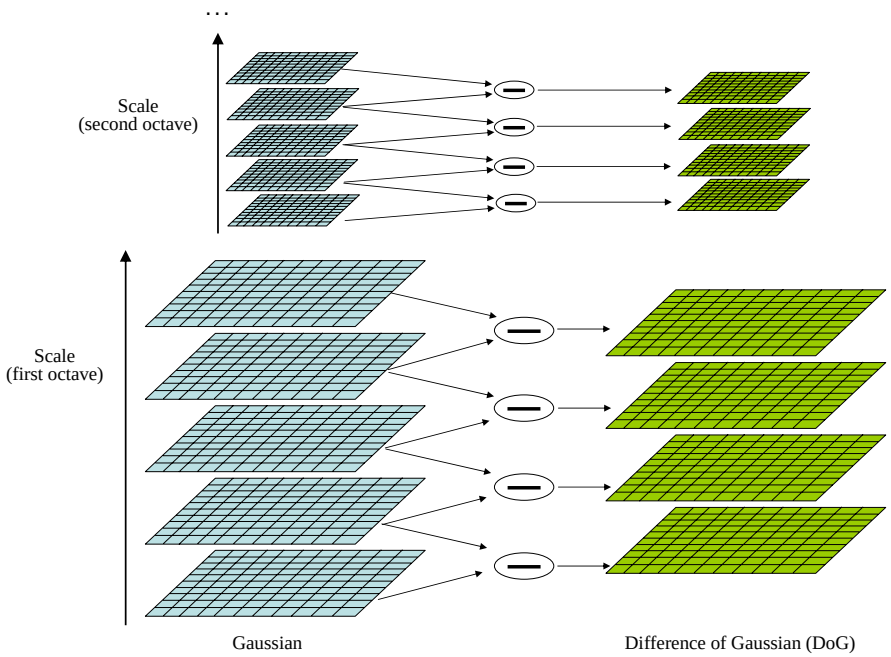


Figure 1.2: **Gaussian scale space and Difference of Gaussian scale space.** For each octave of space, the set of Gaussian images shown on the left is produced by repeatedly convolved the initial image with Gaussians. The difference of Gaussian images shown on the right is produced by subtracting the adjacent Gaussian images. The last image of each octave is down-sampled by a factor of 2 to prodce next octave.

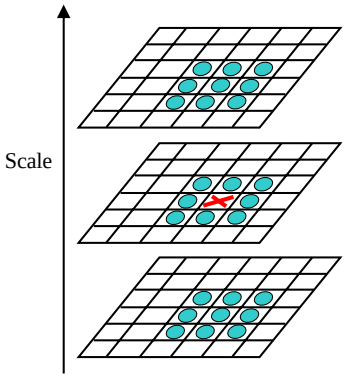


Figure 1.3: **Extrema of Difference of Gaussian space.** The extrema (maximum or minimum) of the difference of Gaussian images are detected by comparing a pixel (marked with X) to its 26 neighbors in 3×3 regions at the current and adjacent scales (marked with circles).

2.Keypoint localization

In this stage, all the detected extrema points are verify for stability.

$$\hat{\mathbf{p}} = -\frac{\partial^2 D^{-1}}{\partial \mathbf{p}^2}\frac{\partial D}{\partial \mathbf{p}} \tag{1.3}$$

For each point $\mathbf{p} = (x, y, \sigma)$, find the real location as sub-pixel by computing the Taylor expansion of $\hat{\mathbf{p}}$ in equation (1.3).

$$D(\mathbf{x}) = D + \frac{\partial D^T}{\partial \mathbf{x}}\mathbf{x} + \frac{1}{2}\mathbf{x}^{\mathbf{T}}\frac{\partial^2 D}{\partial \mathbf{x}^{\mathbf{2}}}\mathbf{x} \tag{1.4}$$

Substitute $\hat{\mathbf{p}}$ into equation (1.4) to rejected the point with low contrast. In Lowe's experiments, the point is rejected if the value is small than 0.03.

$$\mathbf{H} = \begin{bmatrix} D_{xx} D_{xy} \\ D_{yx} D_{yy} \end{bmatrix} \tag{1.5}$$

$$\frac{\mathrm{Tr}(\mathbf{H})^2}{\mathrm{Det}(\mathbf{H})} < \frac{(r+1)^2}{r} \tag{1.6}$$

If the point is not satisfy the equation (1.6) with threshold $r = 10$ (also in Lowe's experiments), it should be an edge point and have to be removed. Where $(\mathbf{H})$ is the Hessian matrix. $\mathrm{Tr}(.)$ and $\mathrm{Det}(.)$ are trace and determinate operations.

3.Orientation assignment

In this stage, the orientation of each keypoint is assigned by estimating the major orientation of a block centered at this keypoints. This major orientation is generated by voting. For each point in the window, estimate its gradient and apply a Gaussian window as weighting function. After finding the major orientation, the gradient orientation of each point in the window is rotated relative to the major orientation.

4.Keypoint descriptor

The keypoint descriptor is computed as Figure 1.4. The image gradient magnitudes and orientations are sampled around the keypoint location. They are weighted by a Gaussian

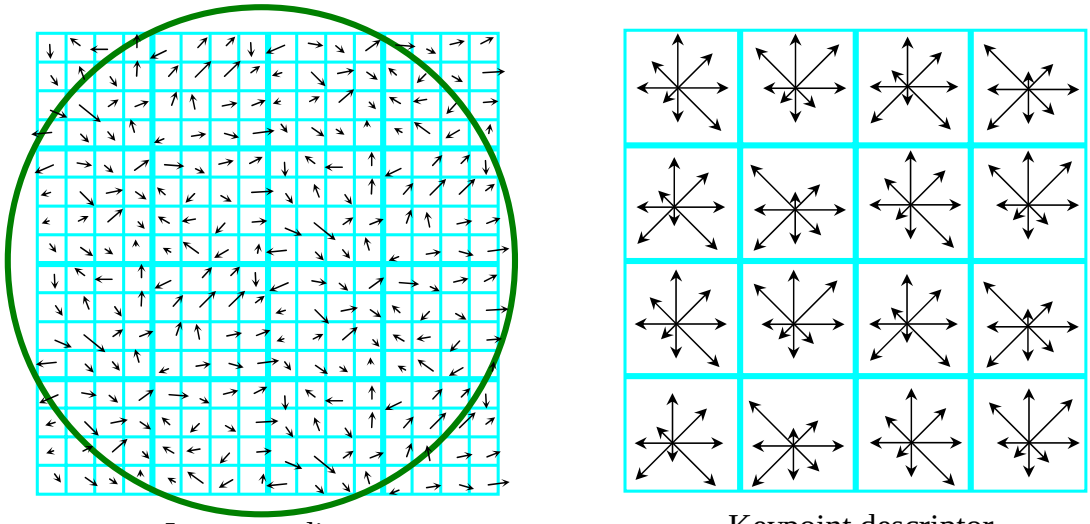


Figure 1.4: **Image gradients and its histogram.** The gradient magnitude and orientation at each image sample point in a region around the keypoint location as shown on the left. They are weighted by a Gaussian window, indicated by the overlaid circle. The orientation histograms are accumulated from these samples, and summarized over 4×4 subregions as shown on the right. The length of each arrow is corresponding to the sum of the gradient magnitudes near that direction within the region.

window, and the orientation histograms are accumulated from these samples, and summarized over 4×4 subregions. The descriptor is formed from a vector containing the values of all the orientation histogram entries, corresponding to the lengths of the arrows on the right side of Figure 1.4. Lowe use $4 \times 4 \times 8 = 128$ element feature vector for each keypoint to achieve the best experiment results.

Image Matching using SIFT

Lowe also uses SIFT to find the correspondence between two images. First, he extracts SIFT features (128 dimension vector) for images, reference image and target image. For each feature, It finds the distance ratio for itself to the closest nearest neighbor and second-closest neighbor. If the ratio is small than a threshold, the closest neighbor and this feature

point are matched.

$$\frac{d_0}{d_1} < r \tag{1.7}$$

Where d_0 and d_1 are the distances between this feature and the closest, second-closest neighbor respectively. r is the threshold represent the strictest and the loosest criteria for feature matching from 0 to 1.

Extensions of SIFT

There are many extensions for SIFT, we introduce some of them as following.

PCA-SIFT Principle Component Analysis (PCA) [11] is a standard technique for dimensionality reduction and has been applied to a broad class of computer vision problems, including feature selection [6], object recognition [22] and face recognition [27]. PCA-SIFT replace the fourth stage (Keypoint descriptor) from SIFT. It is created by concatenating the horizontal and vertical gradient maps for the 41×41 patch centered at the keypoint. Thus, the descriptor have $2 \times 39 \times 39 = 3042$ elements (due to the boundary condition). After that, apply PCA to reduce the dimension from 3042 to 20 [12].

GLOH Gradient Location-Orientation Histogram [21] instead of using the 4×4 grid location bin of SIFT (Figure 1.4), it apply log-polar location grid to construct its descriptor. The log-polar has 3 bins in radial direction and 8 in angular direction (results in 17 location bins), and the gradient orientations are quantized in 16 bins. This gives a 272 bin histogram, and the size of this descriptor is reduced with PCA from 272 to 128.

CSIFT Colored-SIFT [1] follows the same strategy of SIFT in building their descriptors. Instead of using gray gradients, they use the gradients of the color invariants model [8]. For images of the same conditions, CSIFT guarantees the performance is more robust than SIFT with respect to photometric changes.

1.3 Thesis Scope

In this thesis, we propose a novel image feature descriptor, spiral descriptor, and a planar image registration and optimization method. The planar images, target image and source image are registered using planar homography estimated by detecting/selection the corresponding points between images. These correspondences can obtain automatically using reliable image matching methods like SIFT, but it is impossible for all cases. For the cases failed, we manually select pairs of corresponding points and refine using proposed spiral descriptor.

The spiral descriptor is invariant to scaling, rotation and translation. Feature points are localized in scale space and descriptors are built along spiral-shape profile. We use dynamic programming to align and measure the similarity between descriptors. It is a novel approach for feature matching and useful for image matching.

For planar image registration, we propose a novel method to promote the registration accuracy. First, we estimate the homography matrix by either detecting the image correspondences automatically or selecting image corresponding points manually and refining using proposed spiral descriptor. The initial homography matrix is decomposed into is parameters and the optimization process adjusts these parameters using iterative process. Finally, the optimal homography can produce high registration accuracy for planar images.

1.4 Thesis Organization

In chapter 2, we introduce the proposed spiral descriptor and dynamic programming technique. In chapter 3, we introduce the planar image registration method, including how to estimate, decompose and optimize the homography matrix. In chapter 4, we do some experiments to verify the proposed spiral descriptor, the objective function and the optimization process. Finally, a conclusion for this thesis is given in the last chapter.

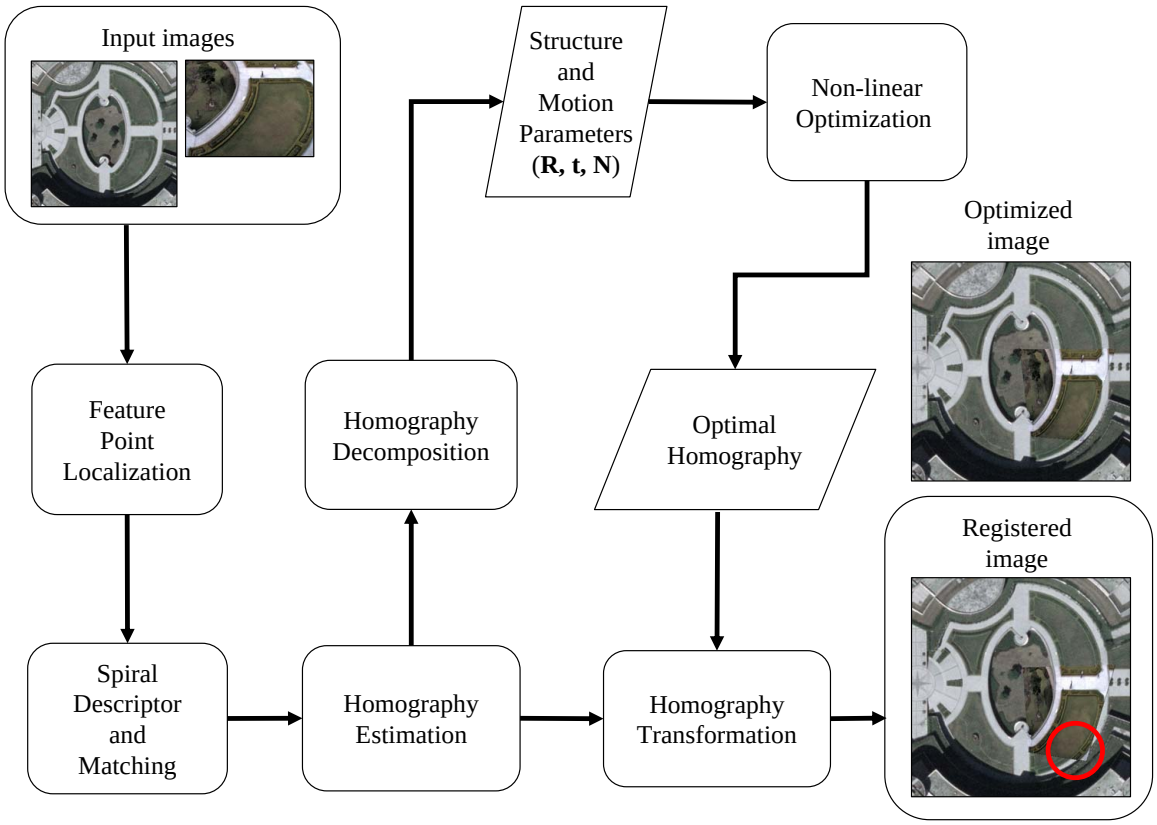


Figure 1.5: **System flowchart for planar image registration.**


E
S
A
1896

Chapter 2

Spiral Descriptor in Scale Space

In this chapter, we propose a novel image descriptor ”Spiral Descriptor” which is invariant to scaling, rotation and translation. First, we extract the reliable feature points from image pyramids. After locating the features, the proposed descriptor is built along spiral-shape profile in proper scale level of image. Finally, for each spiral feature in one image, we find the proper one in another image using proposed matching method based on dynamic programming.

2.1 Feature Point Localization

A good feature point should be invariant to different lighting and geometric imaging conditions. In our application, the feature points can be localized manually or automatically.

Manual selection User selects the feature points directly from the image. Because of that human vision system can easily find the reliable feature points. (like corners, line endings, line intersections, or other distinctive points)

Reliable feature point detection It has been prove that the extrema of Laplacian pyramid can produce the most stable image features compared to a range of other possible image functions, such as the gradient, Hessian, or Harris corner function. [21, 3] SIFT detect feature points based on Difference of Gaussian (DoG), which is an approximation of Laplacian pyramid, and we also apply this process to detect the feature points.

2.2 Spiral Descriptor

Once the feature points are localized, we construct the spiral descriptor for each feature point represented as a vector built by two processes, scale level selection and spiral profile extraction.

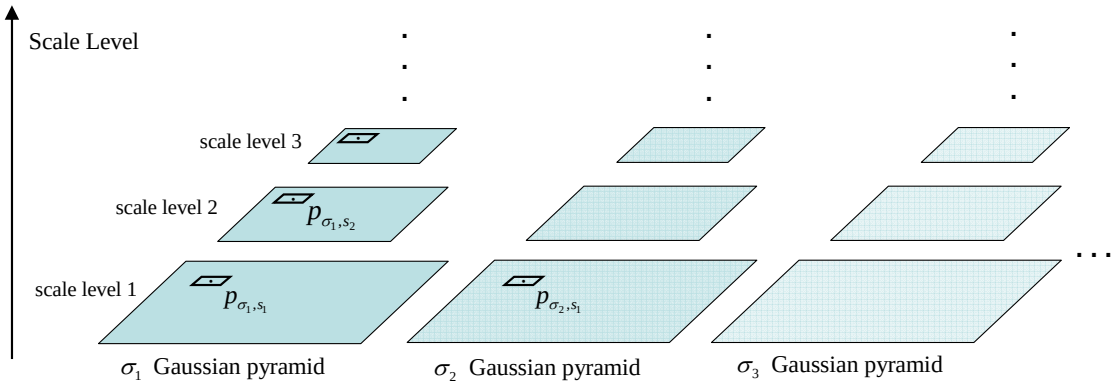


Figure 2.1: **Gaussian scale space.** p_{σ_i,s_j} denotes the position of p in Gaussian pyramid σ_i and scale level s_j.

2.2.1 Scale Level Selection

In our work, there are two ways to localize the feature points. For the feature point is detected from DoG, it is also the extrema of DoG scale space, and the scale level is already determined. For the feature point localized manually, we select a proper scale level from a Gaussian pyramid, that is, the area around the feature point at the proper scale should have the highest information. Here, we use variance as the information measurement.

Suppose there is a feature point p located on image I. We apply various standard deviations of Gaussian to I, and build pyramids for them.(Figure 2.1). After that, Calculating the variances of the areas using the equation (2.1) and finding the scale level with maximum variance can obtain the proper scale level.

$$Var(p_{\sigma_i,s_j}) = \sum_{q=-n}^{n}\sum_{p=-m}^{m}\frac{I(p_{\sigma_i,s_j}(x+p,y+q)) - Mean(p_{\sigma_i,s_j})}{(2n+1)(2m+1)} \quad (2.1)$$

$$Mean(p_{\sigma_i,s_j}) = \sum_{q=-n}^{n}\sum_{p=-m}^{m}\frac{I(p_{\sigma_i,s_j}(x+p,y+q))}{(2n+1)(2m+1)} \quad (2.2)$$

where $I(.)$ denote the intensity value function, $p_{\sigma_i,s_j}(x,y)$ denote the coordinate of p in Gaussian pyramid σ_i and scale level s_j, and the size of area is $(2n+1)\times(2m+1)$.

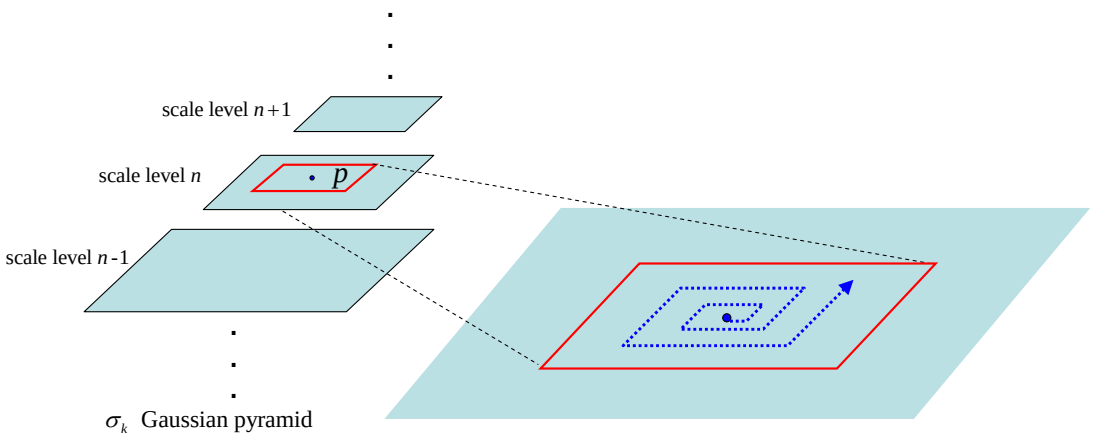


Figure 2.2: **Spiral profile in scale space.** The spiral descriptor of feature point p in σ_k Gaussian pyramid and scale level n, built by recording the intensity values in spiral order.

Table 2.1: **Size of spiral descriptor in Gaussian pyramid**

Block size	Descriptor size
8×8	64 (= 8×8)
16×16	256 (= 16×16)
32×32	1024 (= 32×32)
64×64	4096 (= 64×64)

2.2.2 Spiral Profile Extraction

Once the scale level is determined, we record the intensity values of block in spiral order starting from the position of feature point at determined scale level (Figure 2.2). These recording values are stored in a vector, called the spiral descriptor.

Consider the block is a square, Table 2.1 shows that the descriptor size is the same as block size. In order to reduce it, we merge some outer adjacent pixels (the position of feature is inner) into one point. In our application, we preserve the inner 16 pixels (including the feature point), and merge 4 pixels into one in first outer. The second outer consists of pixels merged from 16 pixels into one, and the nth outer consists of pixels

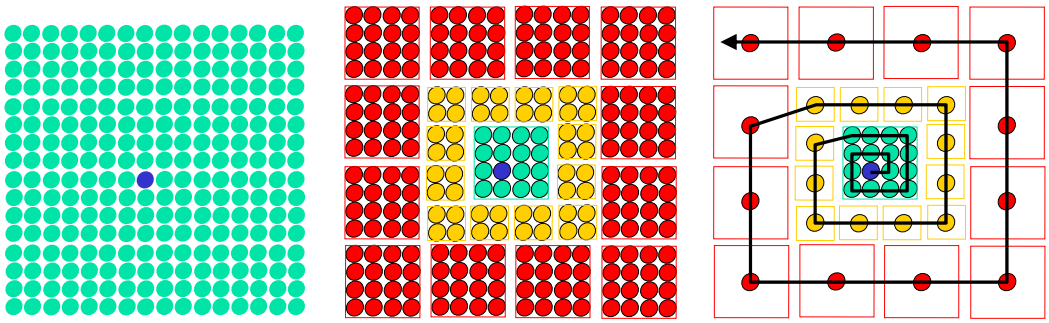


Figure 2.3: **Merge some outer adjacent pixels into one.** The blue point is the feature location shown in the left. We merge outer pixels into one and record them in spiral order (middle and right).

Table 2.2: **Size of spiral descriptor with merging pixels**

Block size	Descriptor size(with merging pixels)
8×8	28 (= 16 + 12×1)
16×16	40 (= 16 + 12×2)
32×32	52 (= 16 + 12×3)
64×64	64 (= 16 + 12×4)

merged from $(2 \times 2^n)^2$ into one (Figure 2.3).

Since merging outer pixels, the descriptor size is linear with respect to the side length of square block (Table 2.2). This strategy not only reduces the descriptor size but also gives a weighting filter to spiral profile. It emphasizes the inner pixels by using higher frequency to record, and de-emphasizes the outer pixels by using loose density to represent. However, there are redundant calculations for constructing all the descriptors. For instance, when building two neighboring descriptors, most of the content is overlapped, and some of the operations used to merge pixels are needless. The multi-level image pyramid proposed by Chen [4] can merge these pixels efficiently.

Assume the block size is $2^{(K+2)} \times 2^{(K+2)}$, we construct $K + 2$ levels of images, where each level is full resolution. Consider $K = 2$, an example shown in Figure 2.4. The value

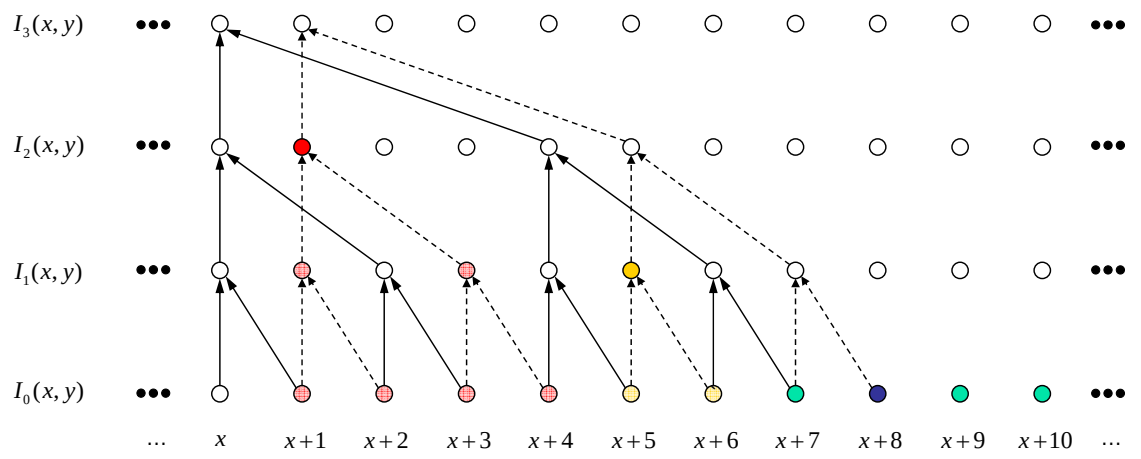


Figure 2.4: **Multi-level image pyramid.** Example of four-level images, where each level is full resolution, and I_0 is the original image. The solid line shows the pyramid at position x, and the dotted line shows another one. For simplicity, only the x-axis is shown, and the colored pixels is the partial profile for spiral descriptor centered at $x + 8$.

of pixel $I_l(x, y)$ at level l ($l > 0$) is calculated from its four corresponding pixels at level $l - 1$.

$$I_l(x,y) = \frac{1}{4}(I_{l-1}(x,y) + I_{l-1}(x+2^l,y) + I_{l-1}(x,y+2^l) + I_{l-1}(x+2^l,y+2^l)) \quad (2.3)$$

where l is the level number, $l = 0, \ldots, K + 1$, and I_0 is the original image. Notice that we can construct pyramids for all the positions in the whole image, expect for the last 2^l pixels at each row and each column due to boundary condition. After that, it is easy to access the merged pixels without redundant calculation from this pyramid.

2.3 Feature Matching using Dynamic Programming

Similarity of two points can be measured by aligning the spiral descriptors centered at these two points. Figure 2.5 shows an example of the spiral descriptors and their alignment results. The two images of toy truck are taken from different views; notice that the spiral descriptors of corresponding points form two images are intuitively similar. In this work, we use the alignment results to measure the similarity between spiral descriptors.

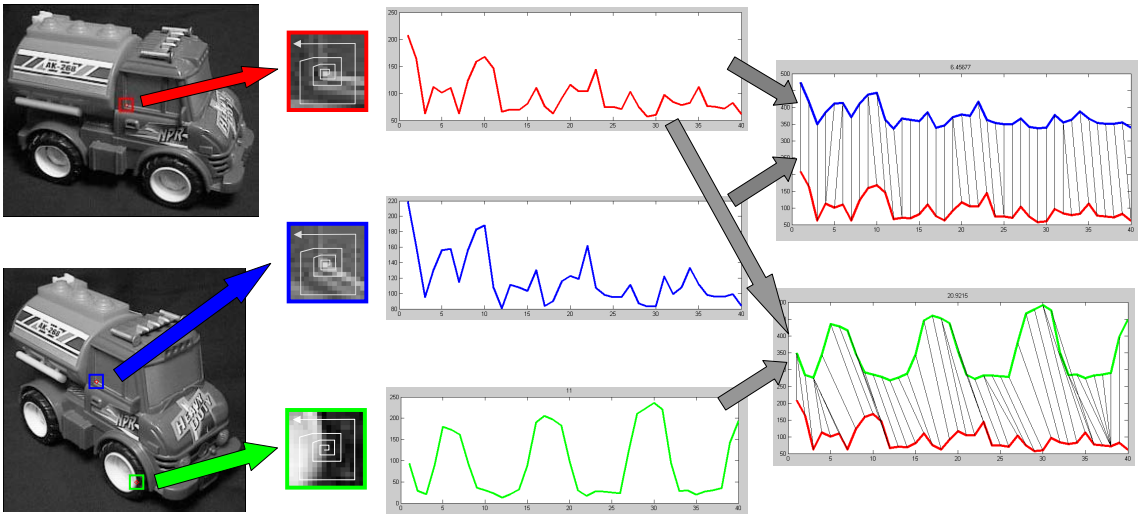


Figure 2.5: **Spiral descriptors and alignment results.** Images of toy truck taken from different views are shown on the left. The spiral descriptors are shown on the middle, and their alignment results are shown on the right. The correspondence points (red and blue) have better alignment result and warping cost (will talk later) than another pair (red and green) The warping cost for red and blue is 6.4, and 20.9 for red and green.

2.3.1 Dynamic Time Warping and Spiral Descriptor Alignment

Dynamic Time Warping (DTW) is a dynamic programming approach for efficiently aligning two sequences in time domain. Instead of solving the entire problem at once, it find the solutions to sub-problems (portions of the sequences), and used to find solutions to a slightly larger problem repeatedly until the solution is found for the entire sequences.

DTW has been widely used in gesture recognition [7], speech processing [25] and data mining [13, 28, 2]. Instead of aligning data sequences in the time domain here, we align image data of two descriptors in the spatial domain. Suppose there are two spiral descriptors D_a and D_b to be aligned, and their profiles are

$$D_a = a_1, \ldots, a_i, \ldots, a_n \tag{2.4}$$

$$D_b = b_1, \ldots, b_j, \ldots, b_n \tag{2.5}$$

where n is the length of spiral descriptor. In order to align two sequence using DTW, we construct a $n \times n$ table where the (i, j) element contains the difference value $d(a_i, b_j)$

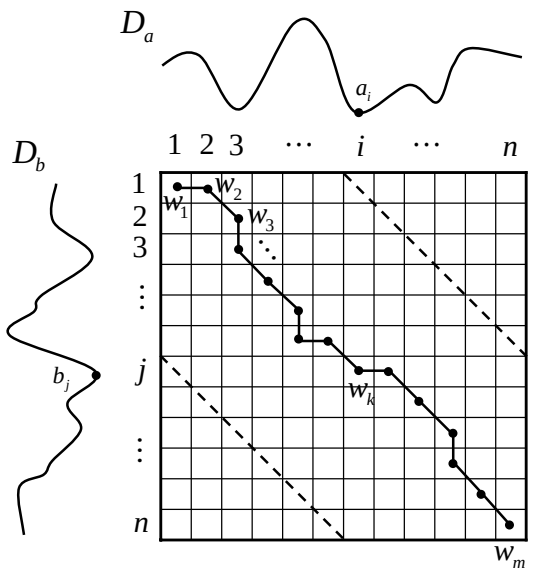


Figure 2.6: **Warping path.** The warping path starts at $w_1 = (1, 1)$ and ends at $w_m = (n, n)$. The element $w_k = (i, j)$ of the warping path describes that a_i should be aligned to b_j.

between a_i and b_j (here the Euclidean distance is used, that is $d(a_i, b_j) = |a_i - b_j|$). Once the table is constructed, the sequences alignment problem can be transform to optimal warping path finding problem.

A warping path W is a set of table elements that describes the correspondence mapping between D_a and D_b (Figure 2.6).

$$W = w_1, \ldots, w_k, \ldots, w_m \qquad , n \leq m \leq 2n - 1 \tag{2.6}$$

where $w_k = (i, j)$ is the kth element of W, and m is the length of W. The warping path has to satisfy several conditions.

Monotonicity For adjacency element of W, $w_k = (p, q)$ and $w_{k-1} = (p', q')$, where $p - p' \geq 0$ and $q - q' \geq 0$. This forces the element of W to be monotonically spaced in the table.

Continuity For adjacency element of W, $w_k = (p, q)$ and $w_{k-1} = (p', q')$, where $p - p' \leq 1$ and $q - q' \leq 1$. This restricts the warp path to step forward only into the adjacent element in the table (diagonally adjacent elements are included).

Boundary $w_1 = (1, 1)$ and $w_m = (n, n)$, this requires the warping path to start and finish in diagonally opposite corner cells of the table.

Windowing The element of W is restricted to fall into a warping window. This means that the corners of the table are pruned from consideration, as shown by dashed line in Figure 2.6.

There are exponentially many warping paths that satisfy the above conditions. However, the optimal warping path is to be provided with minimum warping cost.

$$C_W(D_a, D_b) = \min\left\{\sum_{k=0}^{m} d(w_k)\right\} \tag{2.7}$$

$$d(w_k) = d(a_i, b_j) = |a_i - b_j| \quad , w_k = (i, j) \tag{2.8}$$

where m is the length of warping path. We can find the optimal warping path efficiently using dynamic programming to evaluate the following recurrence.

$$\begin{cases} C(i,j) = d(a_i, b_j) + \min\begin{cases} C(i, j-1) \\ C(i-1, j-1) \\ C(i-1, j) \end{cases} \\ C(1,1) = 0 \\ C(p,q) = \infty \qquad , if \ |p-q| > R \end{cases} \tag{2.9}$$

where $C(i, j)$ is the minimum cumulative cost from state $(1, 1)$ to state (i, j), R is the window width for windowing condition, and $C(i, j-1)$, $C(i-1, j)$, $C(i-1, j-1)$ are the up, left, left up cell of $C(i, j)$ respectively. It make sure that the optimal warping path satisfy the conditions of continuity and monotonicity.

For solving this recurrence, all we need to do is make a decision for local optimal between three candidates until all the states are processed. Finally, the $C(n, n)$ is the cost for optimal warping path. In order to obtain the elements of optimal warping path, we also maintain the previous state of $C(i, j)$ by $pre(i, j)$.

$$pre(i,j) = (i', j') \quad , \ such\ that\ C(i', j') = \min\begin{cases} C(i, j-1) \\ C(i-1, j-1) \\ C(i-1, j) \end{cases} \tag{2.10}$$

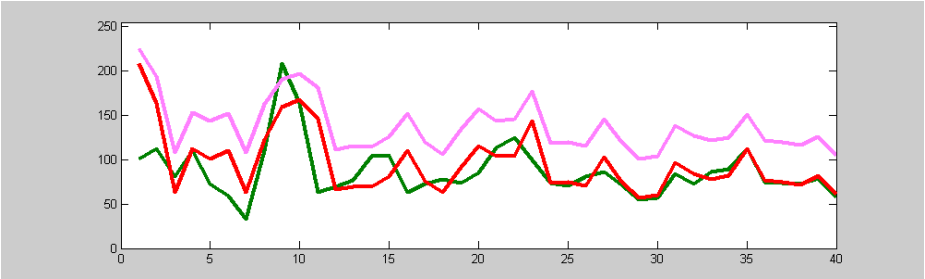


Figure 2.7: **Spiral descriptor with lighting effect.** The red one is a spiral descriptor. Because the lighting effect, it shifted slightly to become the pink one. The green one is another spiral descriptor.

Back-tracing the states from (n, n) using the information of $pre(,.,)$, and we can obtain the optimal warping path $W = w_1, \ldots, w_m$.

$$\begin{cases} w_k &= pre(i,j) \quad , where\, w_{k+1} = (i,j), \quad k > 0 \\ w_1 &= (1,1) \\ w_m &= (n,n) \end{cases} \tag{2.11}$$

There are many factors like lighting, white balance, which affect the intensity of image heavily. If we apply DTW to align two spiral descriptors in such images, the warping cost may not reflect the real similarity between them. Figure 2.7 shows that because of lighting effect, the spiral descriptor (red) shifted slightly (pink). Intuitively, the red one is similar to the pink one, and different to the green one. However, if we apply DTW to compute the warping cost between the red one and the pink one, it is higher than the cost between the red one and the green one. The reason is simple, DTW only take the absolute value of the data sequence into consideration. Keogh [14] propose Derivative Dynamic Time Warping (DDTW) that considers the shape-level representation. It can avoid aligning two data sequences that have similar values but different trend. The shape-level representations are achieved by computing the first derivatives for the data sequence

$$q'_i = \frac{(q_i - q_{i-1}) + (\frac{q_{i+1} - q_{i-1}}{2})}{2} \tag{2.12}$$

where q_{i-1} and q_{i+1} is the left and right neighbor points to the current data point q_i.

Given two spiral descriptors, we can apply equation (2.12) to obtain their shape-level

information, and compute the warping cost between them using DDTW. The cost is the similarity.

2.3.2 Spiral Descriptor Matching and Image Correspondence

We are interesting in the correspondence between images. Given two images, reference image and target image, we localize the feature points and extract the spiral descriptors for them in both images using the methods talked before.

For each feature point in reference image, a straightforward way to find the best matching in target image is to compute all the warping costs and pick the minimum. However, many features from reference image may have no correct matching in target image because they were not appeared or detected from target image. Therefore, it would be helpful to have a way to discard features that do not have any good match in target image. A global threshold to warping cost does not perform well. A more robust way is to consider the warping cost of second-minimum. If the second-minimum is too close to the minimum, we decide this feature in reference image have no matching point in target image. This measurement performs well because correct matches need to have significant difference to the closest incorrect match to achieve reliable matching.

$$w_{c1} < w_{c2} \times r \qquad , 0 < r \leq 1 \tag{2.13}$$

where w_{c1} is the minimum warping cost, w_{c2} is the second-minimum cost, and r is the threshold to the similarity between the best candidate and the second one. It can be set from 0 to 1 to represent the strictest and the loosest criteria for feature matching. Finally, we can obtain the image correspondence using the proposed spiral descriptor and its matching method based on dynamic programming.


E
S
A
1896

Chapter 3

Planar Image Registration

In this chapter, we introduce an image registration method for planar images. First, we estimate the homography matrix for two planar images, target image and source image captured from two views in one planar scene. In order to improve the registration accuracy, we decompose the homography matrix into its geometric parameters and adjust these parameters using iterative nonlinear optimization process. Finally, we can obtain the optimal homography by composing the optimal parameters.

3.1 Planar Homography

A homography is a non-singular linear relationship between points on two planes [10]. When the world points are on a plane, their images captured by two perspective cameras have homography relationship denoted by an 3×3 matrix H.

$$s\tilde{\mathbf{P}}_1 = {}^1\mathbf{H}_2\tilde{\mathbf{P}}_2 \tag{3.1}$$

where the homogeneous representation $\tilde{\mathbf{P}}_1$ and $\tilde{\mathbf{P}}_2$ are the corresponding points in the first and second view respectively. $\tilde{\mathbf{P}}_2$ is transfered to $\tilde{\mathbf{P}}_1$ by ${}^1\mathbf{H}_2$. Here, we denote the image of first view as target image, and the image of second view as source image.

3.1.1 Estimation of Homography Matrix

A homography matrix can be estimated from image information by matching several coplanar points. If there are more than four image correspondences of which no three in each image are collinear, we can estimate the homography matrix by applying least-square method.

In order to obtain the homography between source image and target image, we must to have several pairs of corresponding points. However, even SIFT can not find accurate correspondences in general case. Because SIFT descriptor is consisted of gradient histogram, to find correspondences of two images captured by difference cameras like satellite camera and compact digital camera is a difficult task for SIFT. Figure 3.1 shows that both SIFT and proposed spiral descriptor are failed in case of satellite image (left) and surveillance camera image (right).

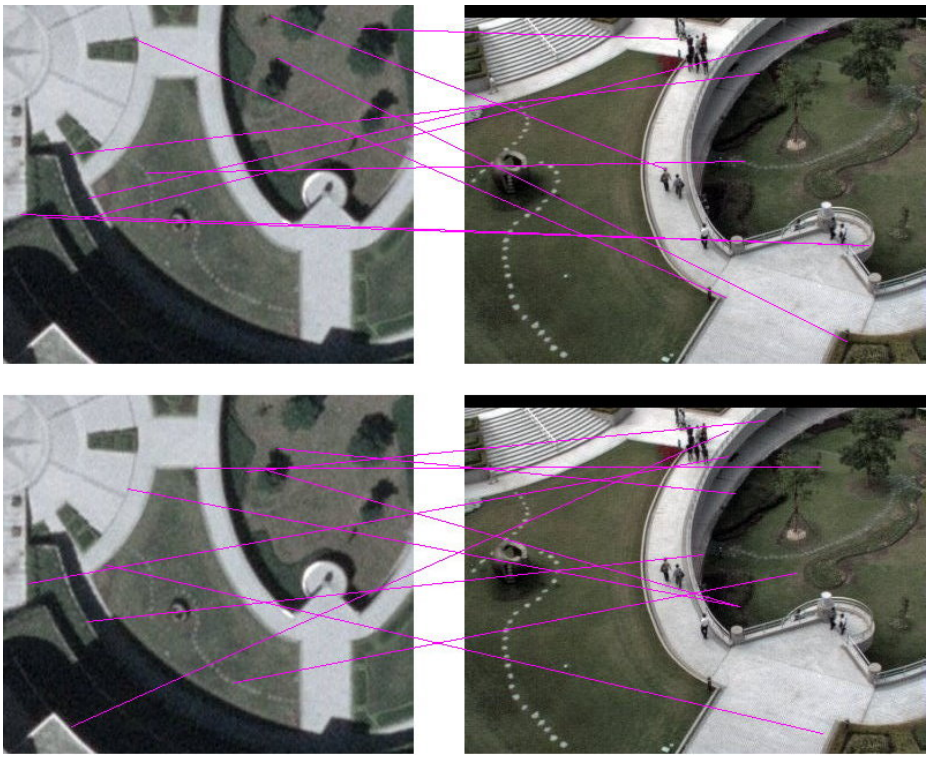


Figure 3.1: **SIFT and spiral descriptor matches the incorrect correspondences.** The Upper is the results of SIFT matching, and the lower is the results of proposed spiral descriptor. The left is satellite image, and the right is surveillance camera image. Top 10 corresponding points are all incorrect both in SIFT and proposed spiral descriptor.

Since finding the correspondences automatically is not reliable, we change the policy to select the corresponding points manually and refine by proposed spiral descriptor. Therefore, we select more than four pairs of corresponding points on the images. For each corresponding pair, we find the best matching point of target image by searching and matching a corresponding range of source image using proposed spiral descriptor.

$$Match(p(x_s, y_s), p(x_t, y_t)) = p(x'_t, y'_t) \tag{3.2}$$

such that

$$\begin{aligned} & C_W(\ D_s(p(x_s, y_s)), D_s(p(x'_t, y'_t))\ ) \\ = \min\{\ & C_W(\ D_s(p(x_s, y_s)), D_s(p(x_s + i, y_s + j))\ )\ \} \quad , -R \le i, j \le R \end{aligned} \tag{3.3}$$

where $Match(.,.)$ is a matching function returns the most similar point (measured by spiral descriptor) around the second argument in a searching area $(2R+1) \times (2R+1)$ with respect to the first argument. $p(x_t, y_t)$ is a point in target image, and $p(x_s, y_s)$ is another point in source image. $D_s(.)$ is spiral descriptor making function, $C_W(.,.)$ is the warping cost function mentioned in previous section, and both $p(x_t, y_t)$ and $p(x_s, y_s)$ are selected manually.

We can use the equation (3.2) to refine the pairs of corresponding points selected manually (Figure 3.2). Once the points are refined, the homography matrix can be estimated as following.

$$s\tilde{\mathbf{P}}_\mathbf{t} = {}^\mathbf{t}\mathbf{H}_\mathbf{s}\tilde{\mathbf{P}}_\mathbf{s} \tag{3.4}$$

$$s\begin{bmatrix} x_t \\ y_t \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & 1 \end{bmatrix} \begin{bmatrix} x_s \\ y_s \\ 1 \end{bmatrix} \tag{3.5}$$

where $\tilde{\mathbf{P}}_\mathbf{t}$ and $\tilde{\mathbf{P}}_\mathbf{s}$ are points in homogeneous representation on target and source image, respectively. $\mathbf{P_t} = (x_t, y_t)$ is transfered from $\mathbf{P_s} = (x_s, y_s)$ by ${}^\mathbf{t}\mathbf{H_s}$ up to a scalar factor. We simply spread the equation (3.5).

$$\begin{cases} s &= x_s h_{31} + y_s h_{32} + 1 \\ 0 &= (x_s h_{11} + x_s h_{12} + h_{13}) - x_t(x_s h_{31} + y_s h_{32} + 1) \\ 0 &= (x_s h_{21} + x_s h_{22} + h_{23}) - y_t(x_s h_{31} + y_s h_{32} + 1) \end{cases} \tag{3.6}$$

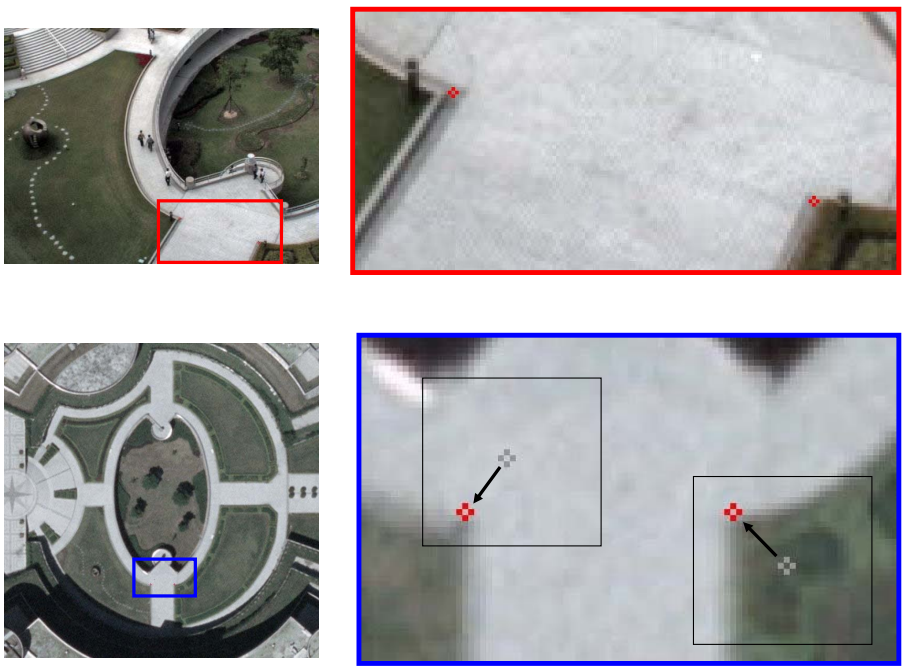


Figure 3.2: **Feature points refine using spiral descriptor.** The source image is shown on the left up, and the target image is shown on the left down. We refine the points in target image by searching and matching a corresponding range (the black block) of source image using proposed spiral descriptor.

and rearrange

$$\begin{bmatrix} x_s & y_s & 1 & 0 & 0 & 0 & -x_s x_t & -y_s x_t \\ 0 & 0 & 0 & x_s & y_s & 1 & -x_s y_t & -y_s y_t \end{bmatrix} \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \end{bmatrix} = \begin{bmatrix} x_t \\ y_t \end{bmatrix} \tag{3.7}$$

$$\mathbf{Ah} = \mathbf{b} \tag{3.8}$$

where $\mathbf{A}$ is a $2n \times 8$ matrix, $\mathbf{h}$ is a 8×1 vector, and $\mathbf{b}$ is a $2n \times 1$ vector. In equation (3.7), we have one correspondence pair, that is $n = 1$. Look at equation (3.8), $\mathbf{A}$ and $\mathbf{b}$ are known, and the degrees of freedom for $\mathbf{h}$ is 8. If there are more than four correspondences pairs of which no three are collinear (i.e., $n \geq 4$, equations ≥ 8), $\mathbf{h}$ can be solved in a least-square solution, then ${}^{\mathbf{t}}\mathbf{H}_{\mathbf{s}}$ is also estimated.

3.1.2 Geometry Interpretation

Consider there is a plane π in 3D space, and we take a photo for this plane. Hence, there exist a homography relationship between π and its photo image. This homography can be interpreted as planar perspective projection.

Look at Figure 3.3, the distance d between the plane π and camera optical center O_c can be calculated as following

$$d = \|\mathbf{P}_\pi\| \mathbf{cos}\theta \tag{3.9}$$

consider the inner product of $\mathbf{P}_\pi$ and unit normal $\mathbf{N}_\pi$

$$\langle \mathbf{P}_\pi, \mathbf{N}_\pi \rangle = \|\mathbf{P}_\pi\| \|\mathbf{N}_\pi\| cos\theta = d \tag{3.10}$$

$$\frac{1}{d} \langle \mathbf{P}_\pi, \mathbf{N}_\pi \rangle = 1 \quad (= \frac{1}{d} \mathbf{N}_\pi^{\mathbf{T}} \mathbf{P}_\pi) \tag{3.11}$$

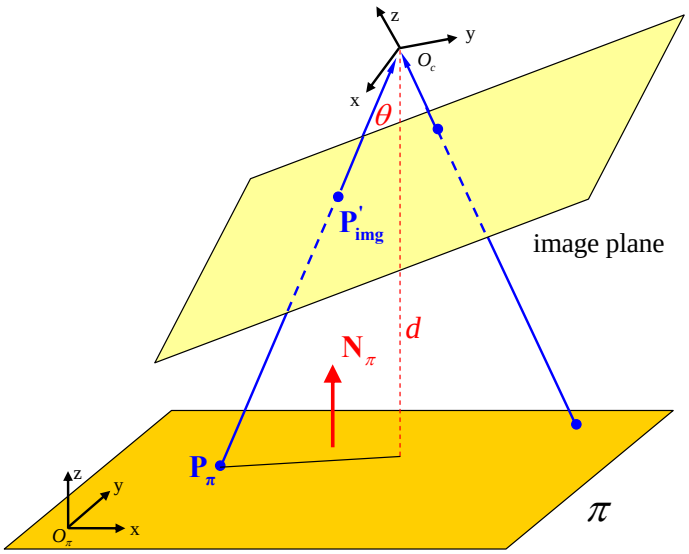


Figure 3.3: **Planar perspective projection.** $\mathbf{P}_\pi$ is a point on the plane π, and its perspective projection $\mathbf{P'}_{\mathbf{img}}$ in image coordinate can be related by rotation, translation, and scaling.

For all point on the plane π, we have constraint as equation (3.11). A point $\mathbf{P}_\pi$ on π and its perspective projection $\mathbf{P'}_{\mathbf{img}}$ have following relationship

$$\mathbf{P'}_{\mathbf{img}} \sim {}^{\mathbf{c}}\mathbf{R}_\pi \mathbf{P}_\pi + {}^{\mathbf{c}}\mathbf{t}_\pi \tag{3.12}$$

where $\sim$ indicates equality up to a scalar factor, ${}^{\mathbf{c}}\mathbf{R}_\pi$ is rotation matrix, and ${}^{\mathbf{c}}\mathbf{t}_\pi$ is translation vector from π coordinate to camera coordinate. Substituting equation (3.11) into equation (3.12) gives

$$\begin{aligned} \mathbf{P'}_{\mathbf{img}} &\sim {}^{\mathbf{c}}\mathbf{R}_\pi \mathbf{P}_\pi + {}^{\mathbf{c}}\mathbf{t}_\pi \\ &\sim {}^{\mathbf{c}}\mathbf{R}_\pi \mathbf{P}_\pi + {}^{\mathbf{c}}\mathbf{t}_\pi (\tfrac{1}{d}\mathbf{N}_\pi^{\mathbf{T}})\mathbf{P}_\pi \\ &\sim ({}^{\mathbf{c}}\mathbf{R}_\pi + \tfrac{1}{d}{}^{\mathbf{c}}\mathbf{t}_\pi \mathbf{N}_\pi^{\mathbf{T}})\mathbf{P}_\pi \end{aligned} \tag{3.13}$$

where $({}^{\mathbf{c}}\mathbf{R}_\pi + \frac{\mathbf{1}}{\mathbf{d}}{}^{\mathbf{c}}\mathbf{t}_\pi \mathbf{N}_\pi^{\mathbf{T}})$ is the relationship between $\mathbf{P'}_{\mathbf{img}}$ and $\mathbf{P}_\pi$ up to a scalar factor. In other word, it is ${}^{\mathbf{img}}\mathbf{H}_\pi$.

$${}^{\mathbf{img}}\mathbf{H}_\pi \sim {}^{\mathbf{c}}\mathbf{R}_\pi + \frac{1}{d}{}^{\mathbf{c}}\mathbf{t}_\pi \mathbf{N}_\pi^{\mathbf{T}} \tag{3.14}$$

Equation (3.14) gives a conclusion that any homography matrix $\mathbf{H}$ is consist of its motion parameters $\{\mathbf{R}, \mathbf{t}\}$ and structure parameters $\{\mathbf{N}, d\}$ of plane π. We call these parameters geometric parameters.

Table 3.1: **Four solutions for the planar homography decomposition, only two of which satisfy the positive depth constraint.**

Solution 1	Solution 2	Solution 3	Solution 4
$\mathbf{R}_1 = \mathbf{W}_1\mathbf{U}_1^T$	$\mathbf{R}_2 = \mathbf{W}_2\mathbf{U}_2^T$	$\mathbf{R}_3 = \mathbf{R}_1$	$\mathbf{R}_4 = \mathbf{R}_2$
$\mathbf{N}_1 = \mathbf{v}_2 \times \mathbf{u}_1$	$\mathbf{N}_2 = \mathbf{v}_2 \times \mathbf{u}_2$	$\mathbf{N}_3 = -\mathbf{N}_1$	$\mathbf{N}_4 = -\mathbf{N}_2$
$\frac{1}{d}\mathbf{t}_1 = (\mathbf{H} - \mathbf{R}_1)\mathbf{N}_1$	$\frac{1}{d}\mathbf{t}_2 = (\mathbf{H} - \mathbf{R}_2)\mathbf{N}_2$	$\frac{1}{d}\mathbf{t}_3 = -\frac{1}{d}\mathbf{t}_1$	$\frac{1}{d}\mathbf{t}_4 = -\frac{1}{d}\mathbf{t}_2$

3.2 Homography Matrix Optimization

In our application, the homography matrix can be estimated using the algorithm described in the previous section. But this homography between target image and source image is not sufficiently precise because the corresponding points are hardly perfect matching. Thus, we want to find the optimal homography matrix to improve the registration accuracy.

Given a homography matrix, we decompose it into geometric parameters, and then adjust these parameters according to an objective function in an iterative optimization process. Finally, we compose the optimal homography using the optimized parameters.

3.2.1 Decomposition of Homography Matrix

In the previous section, we introduce the geometric meaning of homography matrix. It consists of structure and motion parameters: $\left\{\mathbf{R}, \frac{1}{d}\mathbf{t}, \mathbf{N}\right\}$. Given a homography matrix $\mathbf{H} = (\mathbf{R} + \frac{1}{d}\mathbf{t}\mathbf{N}^\mathbf{T})$, there are at most two physically possible solutions for a decomposition into parameters $\left\{\mathbf{R}, \frac{1}{d}\mathbf{t}, \mathbf{N}\right\}$ given in Table 3.1 [19]. The elements of Table 3.1 can be estimated as follows:

1 For matrix $\mathbf{H_L} = (\mathbf{R} + \frac{1}{d}\mathbf{t}\mathbf{N}^\mathbf{T})$, calculate the its second largest singular value $\sigma_2(\mathbf{H_L})$.

2 Estimate the normalized matrix $\mathbf{H} = \frac{1}{\sigma_2(\mathbf{H_L})}\mathbf{H_L}$.

3 Estimate the SVD of the symmetric matrix: $\mathbf{H}^\mathbf{T}\mathbf{H}$.

$$\mathbf{H}^\mathbf{T}\mathbf{H} = \mathbf{V}\mathbf{\Sigma}\mathbf{V}^\mathbf{T} \tag{3.15}$$

where $\mathbf{V} = [\mathbf{v_1}, \mathbf{v_2}, \mathbf{v_3}]$ and $\mathbf{\Sigma} = diag\left\{\sigma_1^2, \sigma_2^2, \sigma_3^2\right\}$

4 Estimate $\mathbf{u_1}$ and $\mathbf{u_2}$.

$$\mathbf{u_1} = \frac{\sqrt{1-\sigma_2^3}}{\sqrt{\sigma_1^2-\sigma_3^2}}\mathbf{v_1} + \frac{\sqrt{\sigma_2^1-1}}{\sqrt{\sigma_1^2-\sigma_3^2}}\mathbf{v_3} \tag{3.16}$$

$$\mathbf{u_2} = \frac{\sqrt{1-\sigma_2^3}}{\sqrt{\sigma_1^2-\sigma_3^2}}\mathbf{v_1} - \frac{\sqrt{\sigma_2^1-1}}{\sqrt{\sigma_1^2-\sigma_3^2}}\mathbf{v_3} \tag{3.17}$$

5 Estimate $\mathbf{U_1}$, $\mathbf{U_2}$, $\mathbf{W_1}$, and $\mathbf{W_2}$.

$$\mathbf{U_1} = [\mathbf{v_2}, \mathbf{u_1}, \mathbf{v_2} \times \mathbf{u_1}] \tag{3.18}$$

$$\mathbf{U_2} = [\mathbf{v_2}, \mathbf{u_2}, \mathbf{v_2} \times \mathbf{u_2}] \tag{3.19}$$

$$\mathbf{W_1} = [\mathbf{Hv_2}, \mathbf{Hu_1}, \mathbf{Hv_2} \times \mathbf{Hu_1}] \tag{3.20}$$

$$\mathbf{W_2} = [\mathbf{Hv_2}, \mathbf{Hu_2}, \mathbf{Hv_2} \times \mathbf{Hu_2}] \tag{3.21}$$

where $\times$ denotes the binary operation of cross product.

Notice that there are only two solutions of Table 3.1 satisfies the positive depth constraint, that is, the plane is in front of the camera. We check the value of $\mathbf{N_i e_3}$ ($\mathbf{e_3} = [0\ 0\ 1]^T$). If $\mathbf{N_i e_3}$ is positive, then the solution i is accepted, else rejected. The degree of freedom for homography matrix is 8, so does the decomposition results (rotation matrix is 3, translation vector with distance is 3, normal direction is 2). The correctness of this algorithm has been proven in [19]. After applying the processes above that we can obtain the motion and structure parameters of homography matrix $\mathbf{H}$.

3.2.2 Proposed Objective Function

The main idea for the proposed objective function is the correlation ratio (CR) [26]. It is superior in accuracy, efficiency and robustness, and have been applied to multimodality registration of medical images [29, 23, 9]. The keynote of CR is that it measures the intensity dispersion for the voxels in target image which are corresponding to the points in source image that belong to the same histogram bin (Figure 3.4).

In our application, the original CR can not perform robustly. Because the pixels of the same histogram in target image are uncertain in the same histogram in source image due

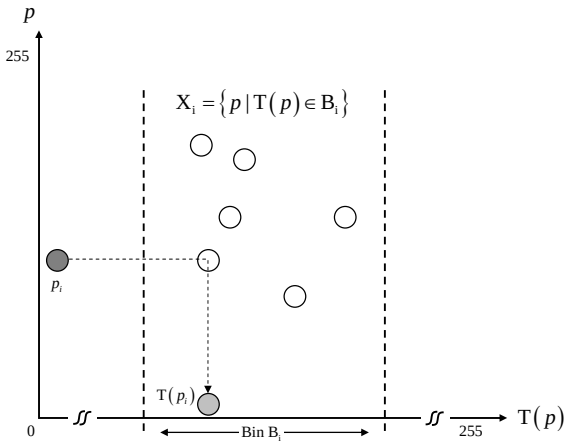


Figure 3.4: **Explaining of Correlation Ratio (CR).** CR measures the intensity dispersion for the voxels $\mathbf{X_i}$ in target image which are corresponding to the points in source image that belong to the same histogram bin $\mathbf{B_i}$.

to the lighting effect in the scene. Thus, instead of measuring the intensity dispersion of histogram bins, we consider the region similarity using intensity or color information.

First, we segment the source image into several regions using the method proposed by Felzenszwalb et al. [5]. For each region, we compute the corresponding region in target image using the initial homography and estimate the similarity within region (Figure 3.5). If the homography is precise for registration, the similarity within region should be high. However, the only consideration is insufficient, we further think about the similarity between regions. For each region, we also estimate the distances between all the other regions. If the distance between two regions in source image is short, it can be excepted that the distance between two corresponding regions in target image is also short. Thus, we propose the objective functions as follows.

$$\arg\min_{\Theta} \frac{\sum_{i=0}^{n} (\text{Area}((\mathbf{H}_\Theta R_i)) \text{Var}(\text{I}(\mathbf{H}_\Theta R_i)))}{\text{Correlation}(D_r, D_t)} \tag{3.22}$$

where

$$\begin{aligned} \Theta &= \{\mathbf{R}, \mathbf{t}, \mathbf{N}\} \\ D_s &= (d^s_{(1,1)}, d^s_{(1,2)}, d^s_{(1,3)}, \ldots, d^s_{(n,n)}) \\ D_t &= (d^t_{(1,1)}, d^t_{(1,2)}, d^t_{(1,3)}, \ldots, d^t_{(n,n)}) \end{aligned} \tag{3.23}$$

where

R_i is the region segmented in source image.

$\mathbf{H}_\Theta R_i$ is the corresponding region of R_i in target image.

$\mathbf{H_\Theta}$ is a homography consisted of Θ $(= \{\mathbf{R}, \mathbf{t}, \mathbf{N}\})$ that transfers the region from source image to target image.

$\mathrm{Area}(.)$ is a function that counts the size of region.

$\mathrm{Var}(.)$ is a variance calculation function.

$\mathrm{Correlation}(.,.)$ is a correlation estimation function.

$\mathrm{I}(.)$ represents the intensity of region.

D_s is an ordered sequence of which $d^s_{(p,q)}$ denotes the hue distance between R^s_p and R^s_q in source image.

D_t is an ordered sequence of which $d^t_{(p,q)}$ denotes the hue distance between $\mathbf{H}_\Theta R^s_p (= R^t_p)$ and $\mathbf{H}_\Theta R^s_q (= R^t_q)$ in target image.

It is a nonlinear optimization problem to achieve the equation (3.22), we use simplex method [24] to solve it. Once the iterative optimization process has done, we can obtain a set of parameters $\hat{\Theta}$ that can compose the optimal homography $\mathbf{H}_{\hat{\Theta}}$ which best describe the geometrical relationship between source image and target image.

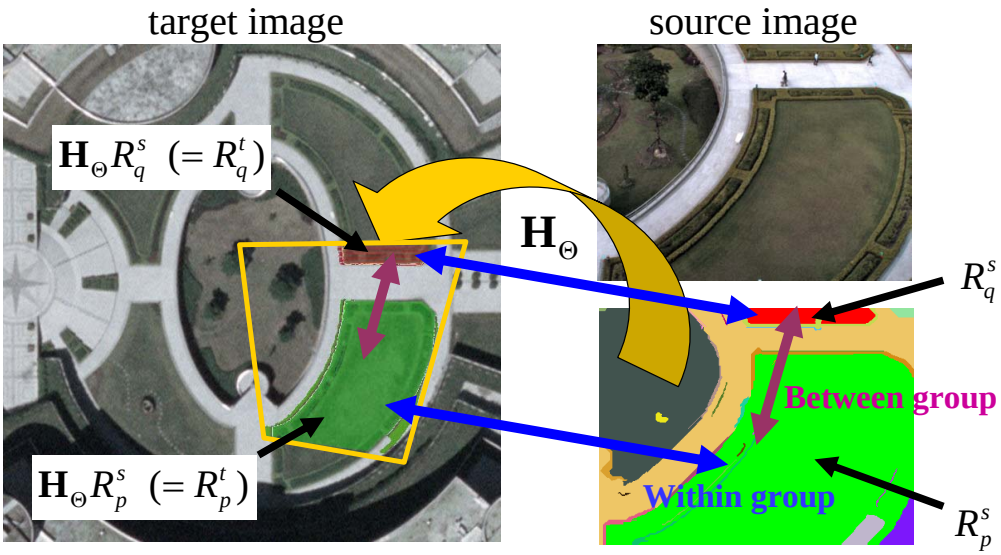


Figure 3.5: **Idea for proposed objective function.** We segment the source image into several regions using the method proposed by Felzenszwalb et al. [5]. For each region R_p^s, we compute the corresponding region $\mathbf{H}_\Theta R_q^s (= R_q^t)$ in target image.

Chapter 4

Experiment Results

To evaluate and demonstrate the performance of the proposed method. In the first section, we compare the results of image matching between SIFT and spiral descriptor. In the second section, we evaluate the performance of proposed point refinement method for points selected manually. In the third section, we evaluate the performance of the proposed objective for homography optimization. In the last section, we demonstrate the results of homography optimization.

4.1 Comparison between SIFT and Spiral Descriptor

Figure 4.2 shows an example of feature points found in different scale level of Gaussian space. Figure 4.5, Figure 4.4 and Figure 4.3 shows the matching results between SIFT and spiral descriptor. Spiral descriptor has almost the same performance as SIFT. However, SIFT have incorrect matchings in Figure 4.4 because the matchings labeled using dotted lines have similar gradient histograms, and can not eliminate by the ratio of nearest neighbor and second-nearest neighbor, that is the matching criteria of SIFT is weaker than spiral descriptor in this case.

We also do rotation testing for both proposed spiral descriptor and SIFT, and the results shown in Figure 4.7 and Figure 4.8. The matching results of SIFT is perfect, but some of ours have incorrect matchings. Because SIFT finds the major orientation for each feature, but we deal with the rotation problem using dynamic programming directly which just can bear small rotations. We can also use the orientation information of features to overcome large rotations in the future.

The weakness of proposed spiral descriptor is that images have small texture, and it is easy to produce the same spiral profiles. We can choose DTW instead of DDTW for feature matching to overcome this problem. SIFT is gradient based descriptor, and it is suitable for images have lots of gradient information. (Figure 4.9)

Figure 4.1: **SIFT feature points of truck image.** SIFT feature points are marked as crosses.

4.2 Results of Point Refinement

Figure 4.10, Figure 4.11, ,Figure 4.13 and Figure 4.12 shows the range of manual selection error which can refine by using the proposed point refinement method. Figure 4.10 and Figure 4.10 are the easy cases, but the drawbacks of this method are shown in Figure 4.13 and Figure 4.12. Points can not refine correctly when they have no “feature ” (Figure 4.13 right, center) or are similar to the others in its searching range.

4.3 Results of Objective Function

Figure 4.14 show that the proposed objective function is proper for the planar image registration, because the curves slides toward the values estimated using manual selection with high accuracy. Notice that the y-axis of the plots that have lots of local minimums (θ_z, and θ_{N_1}) are too small to affect the trend of proposed objective function.

4.4 Results of Planar Image Registration

Figure 4.15, Figure 4.17 and Figure 4.16 are the result images of proposed method for planar image registration. Figure 4.15 is the case that the right down of initial image have slight misalignment and proposed optimization process can fix it. Figure 4.17 shows that the optimized image can not register perfectly due to the objective have larger weighting

Figure 4.2: **SIFT feature points of truck image in Gaussian spaces.** SIFT feature points found in different scale level of Gaussian space are marked as crosses.

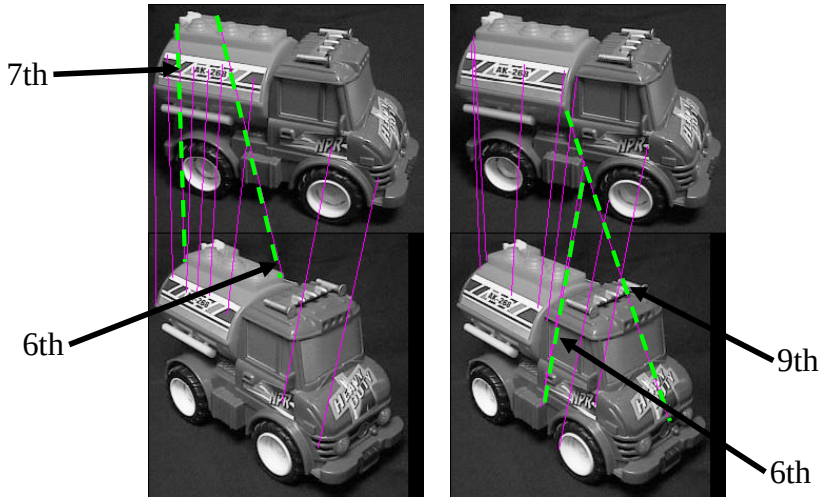


Figure 4.3: **Matching results in truck images taken from difference view point using SIFT and Spiral Descriptor.** SIFT 8/10 corrects shown on the left. Spiral descriptor 8/10 corrects shown on the right. The dotted lines with ranking are the incorrect matchings.

for larger area, that is, the small areas may not register well. Figure 4.16 shows that the initial registration have some blur effect because there are translation error between source image and target image. The optimized image is better than initial image.

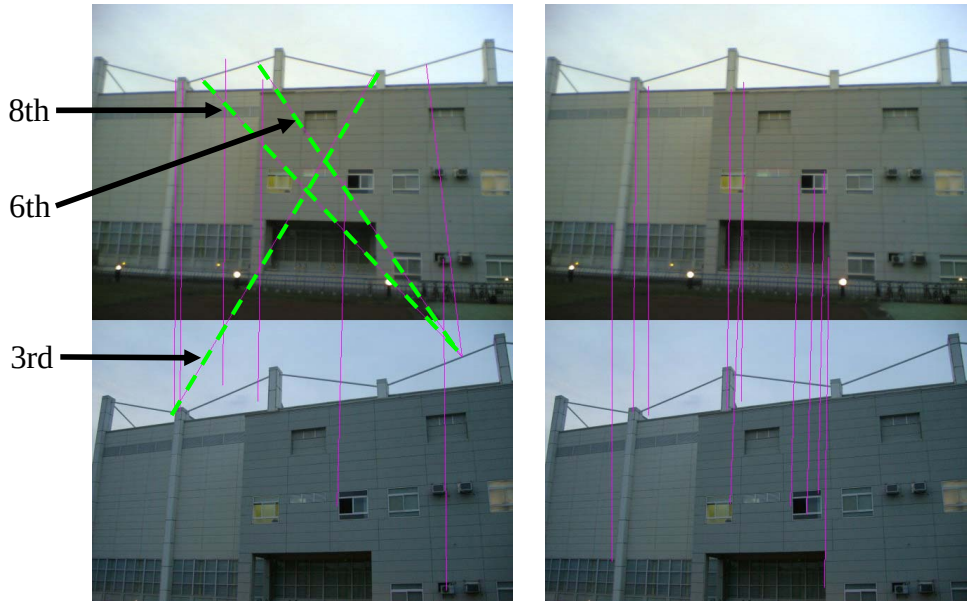


Figure 4.4: **Matching results in building images taken from difference view point using SIFT and Spiral Descriptor.** The two image are taken by difference cameras. Upper image is taken by mobile phone camera, and the lower one is taken by high resolution digital camera. SIFT 7/10 corrects shown on the left. Spiral descriptor 10/10 corrects shown on the right. The dotted lines with ranking are the incorrect matchings.

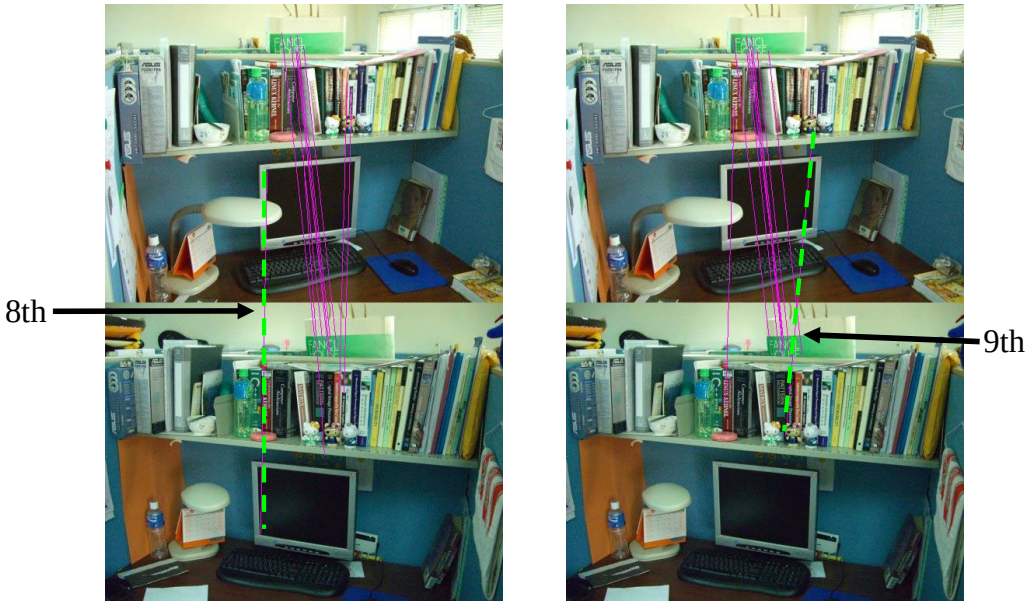


Figure 4.5: **Matching results in office images taken from difference view point using SIFT and Spiral Descriptor.** SIFT 9/10 corrects shown on the left. Spiral descriptor 9/10 corrects shown on the right. The dotted lines with ranking are the incorrect matchings.

SIFT

Spiral

Figure 4.6: **Matching results in wall-patting images taken from difference view point using SIFT and Spiral Descriptor.** Both SIFT and spiral descriptor have 10/10 corrects.

Figure 4.7: **Rotation testing for proposed spiral descriptor in wall images.** Two wall images are matched using proposed spiral descriptor from 10° to 90°. The incorrect matchings are marked using dotted lines.

Figure 4.8: **Rotation testing for SIFT in wall images.** Two wall images are matched using SIFT from 10° to 90°.

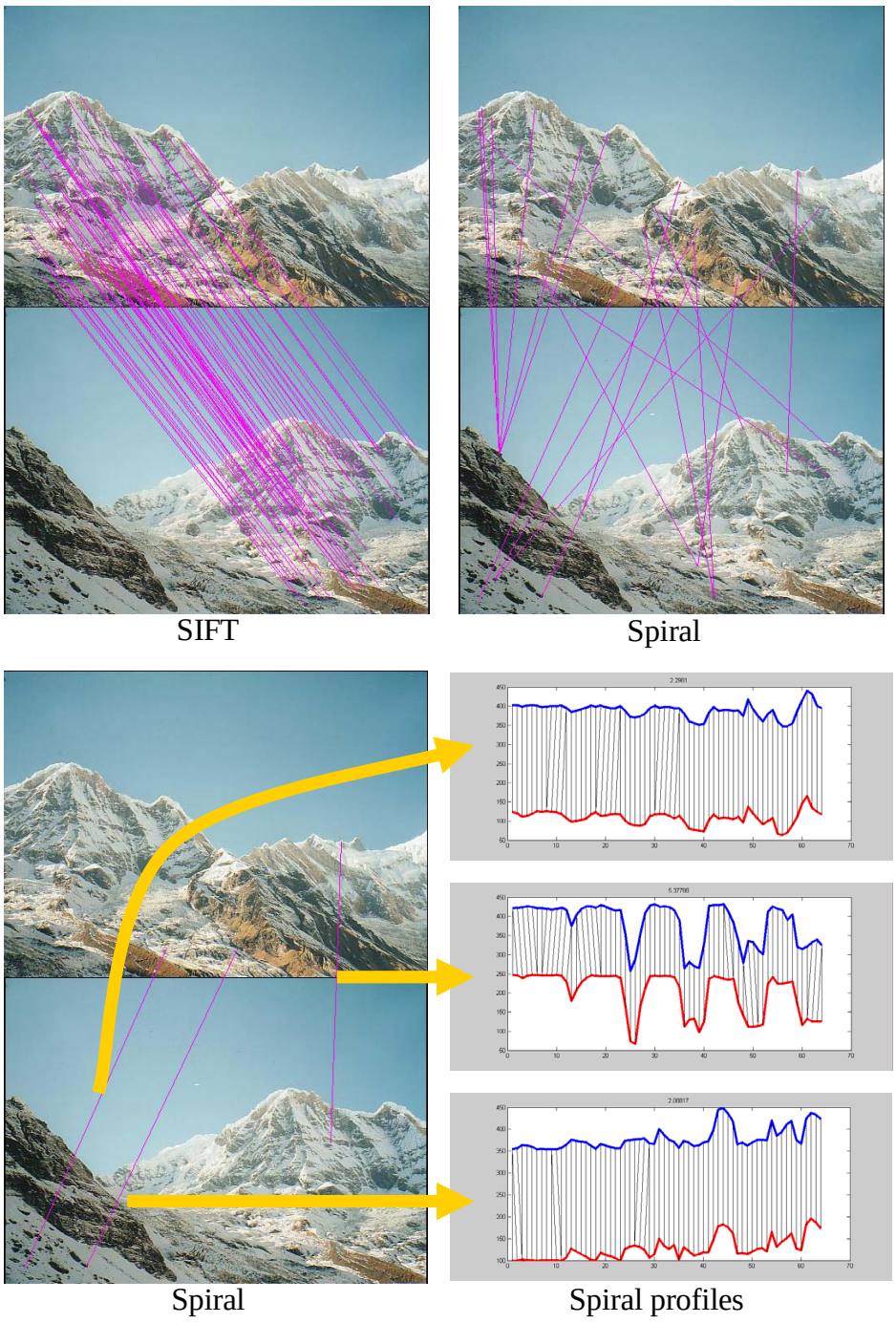


Figure 4.9: **The weakness of proposed spiral descriptor.** The weakness of proposed spiral descriptor is that images have small texture (right up), and it is easy to produce the same spiral profiles (right down). SIFT is gradient based descriptor, and it is suitable for images have lots of gradient information (left up).

Figure 4.10: **Range of point refinement in grassland images using spiral descriptor.** Red points on the left image and the white circles on the right image are the correspondences selected manually. The red regions are the points can be refined to the corresponding white circle in a 19×19 searching range within 1.5 pixels error.

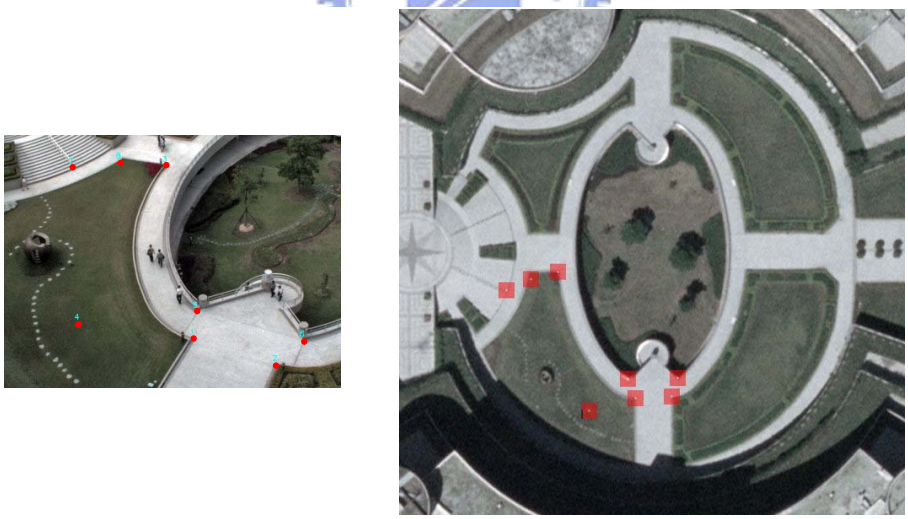


Figure 4.11: **Range of point refinement in grassland images using spiral descriptor.** Red points on the left image and the white circles on the right image are the correspondences selected manually. The red regions are the points can be refined to the corresponding white circle in a 19×19 searching range within 1.5 pixels error.

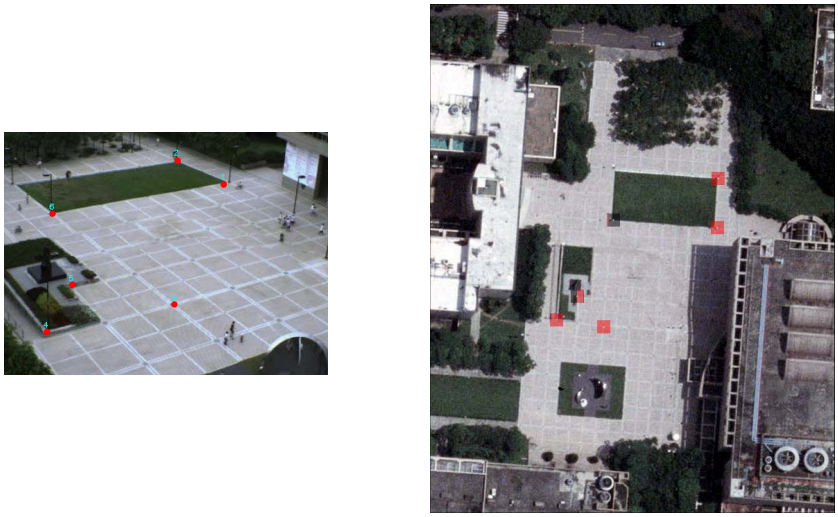


Figure 4.12: **Range of point refinement in concourse images using spiral descriptor.** Red points on the left image and the white circles on the right image are the correspondences selected manually. The red/black regions are the points can/can't be refined to the corresponding white circle in a 19×19 searching range within 1.5 pixels error.

Figure 4.13: **Range of point refinement in building images using spiral descriptor.** Red points on the left image and the white circles on the right image are the correspondences selected manually. The red/black regions are the points can/can't be refined to the corresponding white circle in a 19×19 searching range within 1.5 pixels error.

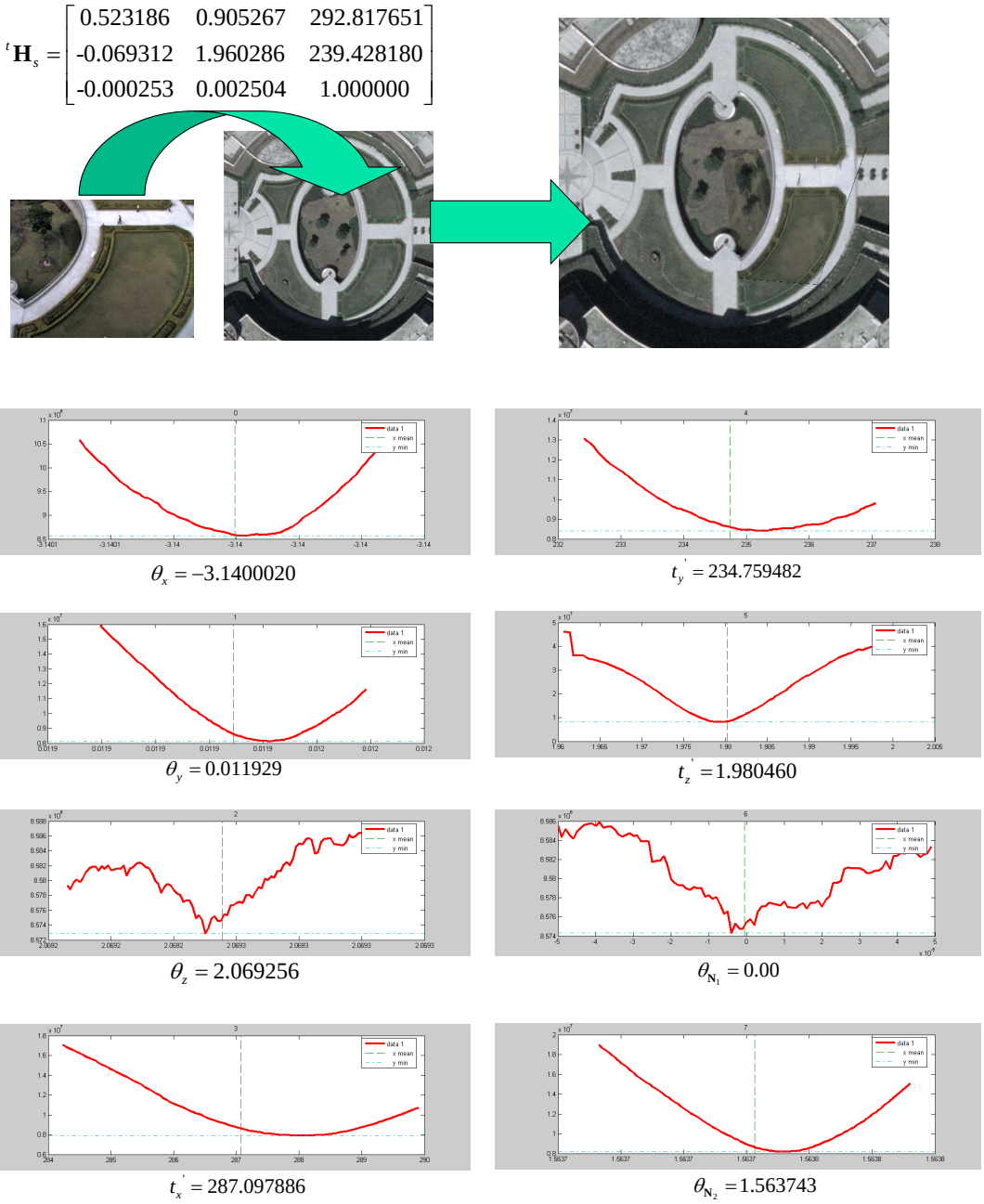
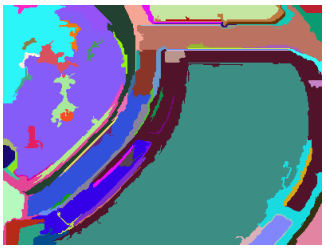


Figure 4.14: **Performance evaluation for proposed objective function in grassland image.** The registered image shown on the right up is composed of the source and target images shown on the middle and left up by the homography matrix ${}^{t}\mathbf{H}_{s}$. The x-axis and y-axis are the parameter values of ${}^{t}\mathbf{H}_{\mathbf{s}}$ and the cost for the proposed objective function, where $\frac{1}{d}\mathbf{t} = (t'_x, t'_y, t'_z)$, $\mathbf{R} = (\theta_{\mathbf{x}}, \theta_{\mathbf{y}}, \theta_{\mathbf{z}})$ and $\mathbf{N} = (\theta_{\mathbf{N_1}}, \theta_{\mathbf{N_2}})$. The values shown on the bottom of the curves represents the parameters of ${}^{t}\mathbf{H}_{\mathbf{s}}$ estimated using manual selection with high accuracy.

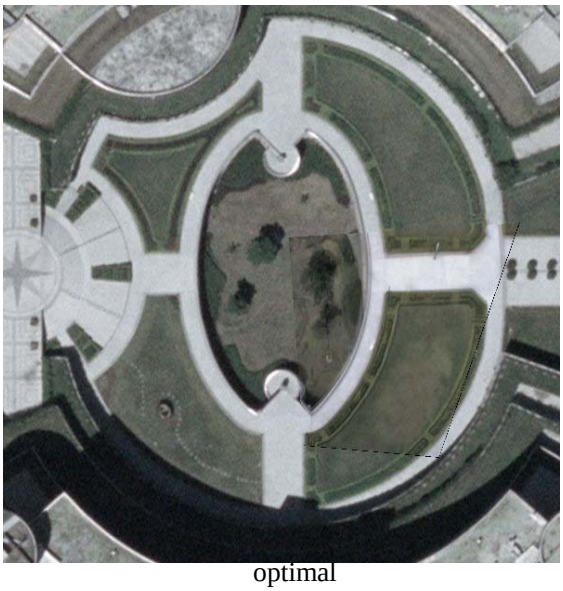
target image

initial

reference image



segmented reference image



optimal

Figure 4.15: **Optimization for image registration of grassland images.** The target image is shown on the left up, and the source image and it segmented image are shown on the left down. The initial homography estimated by correspondence points selected manually overlays the source image and target image shown on the right up. The optimized homography produces the final result shown on the right down.

Figure 4.16: **Optimization for image registration of wall-patting images.** The target image is shown on the left up, and the source image and it segmented image are shown on the middle up and middle down. The initial homography estimated by correspondence points selected manually overlays the source image and target image shown on the right up. The optimized homography produces the final result shown on the right down.

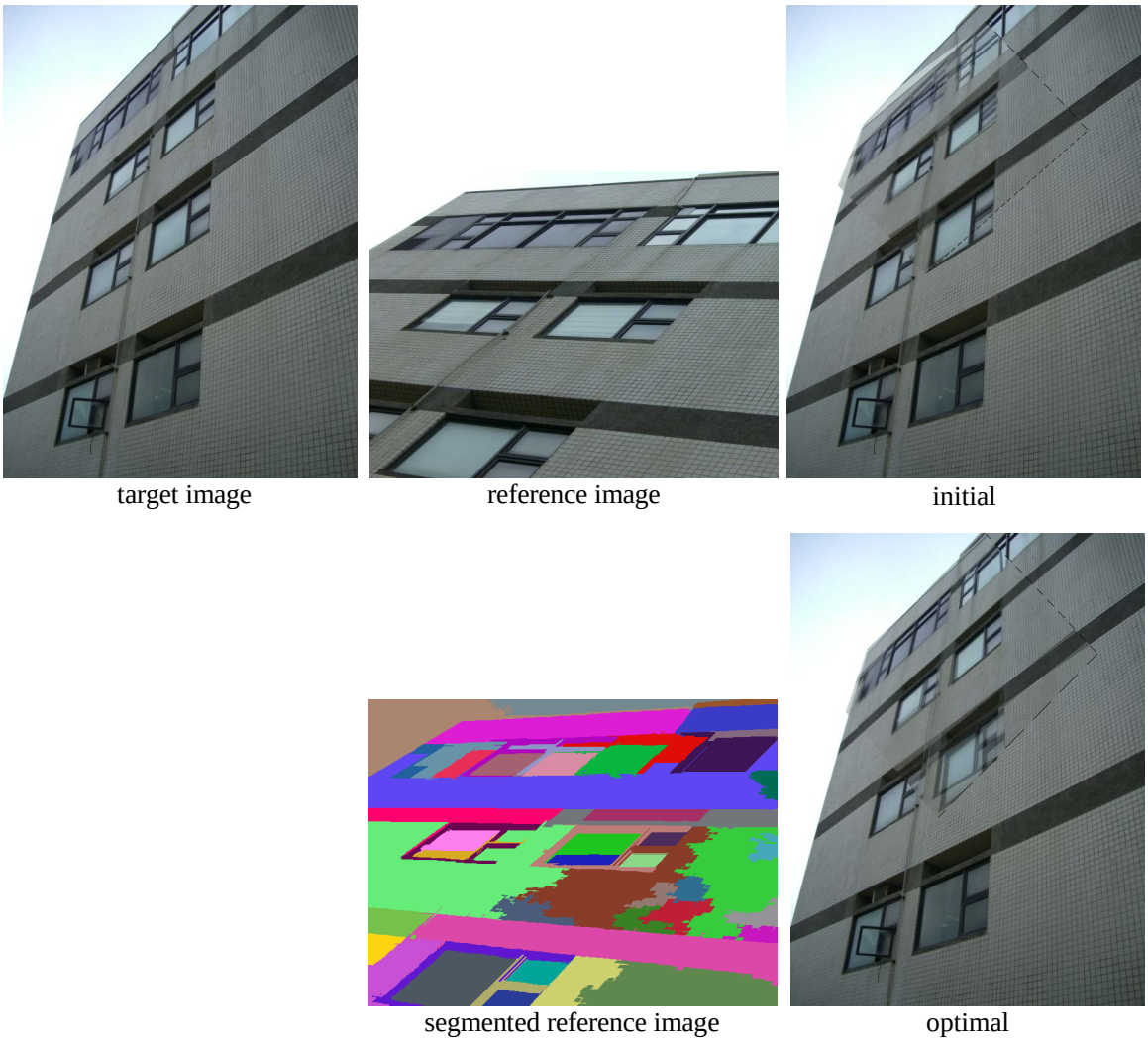


Figure 4.17: **Optimization for image registration of building images.** The target image is shown on the left up, and the source image and it segmented image are shown on the middle up and middle down. The initial homography estimated by correspondence points selected manually overlays the source image and target image shown on the right up. The optimized homography produces the final result shown on the right down.

Chapter 5

Conclusion

In this thesis, we propose a novel image feature descriptor: spiral descriptor, a planar image registration and its optimization method. The planar images, target image and source image are registered by estimating the transformation model: planar homography. This image registration process involves two major techniques, image correspondence detection/selection and homography matrix optimization.

For image correspondence, we propose a novel spiral descriptor based on SIFT feature point extraction. This descriptor is invariant to scaling, rotation and translation. The feature points are localized in scale space and the descriptors are built along spiral-shape profile, which can achieve scaling and translation. The novel feature matching method is also proposed. We use dynamic programming approach to align spiral descriptors and it is suitable for rotation invariant.

For planar image registration, we propose a novel method to promote the registration accuracy. First, we estimate the homography matrix by either detecting the image correspondences automatically or selecting image corresponding points manually and refining using proposed spiral descriptor. The initial homography matrix is decomposed into its parameters and the optimization process adjusts these parameters using iterative process. Finally, the optimal homography can produce high registration accuracy for planar images.

The experiment results show that the proposed spiral descriptor can match images automatically and robust as SIFT in some cases. For the cases failed, we select the correspondences manually and refine the positions in a searching range. The proposed planar image registration method not only promote the registration accuracy but also provides convenience process for user doing image registration.

Bibliography

[1] A. E. Abdel-Hakim and A. A. Farag. Csift: A sift descriptor with color invariant characteristics. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2:19781983, 2006.

[2] D. Berndt and J. Clifford. Using dynamic time warping to find patterns in time series. *AAAI Workshop on the Knowledge Discovery in Databases*, page 359370, 1994.

[3] P. Burt and E. Adelson. The laplacian pyramid as a compact image code. *Communications, IEEE Transactions on [legacy, pre-1988]*, 31(4):532–540, 1983.

[4] Y. S. Chen, Y. P. Hung, and C. S. Fuh. Fast block matching algorithm based on the winner-update strategy. *Image Processing, IEEE Transactions on*, 10(8):1212–1222, 2001.

[5] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2):167–181, 2004.

[6] K. Fukunaga and W. L. G. Koontz. Application of the karhunen-loeve expansion to feature selection and ordering. *Transactions on Computers*, 100(19):311–318, 1970.

[7] D. M. Gavrila and L. S. Davis. Towards 3-d model-based tracking and recognition of human movement. *Int. Workshop on Face and Gesture Recognition*, page 272277, 1995.

[8] J. M. Geusebroek, R. van den Boomgaard, A. W. M. Smeulders, and H. Geerts. Color invariance. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, pages 1338–1350, 2001.

[9] A. Gholipour, N. Kehtarnavaz, R. W. Briggs, and K. S. Gopinath. Kullback-leibler distance optimization for non-rigid registration of echo-planar to structural magnetic resonance brain images. *Image Processing, 2007. ICIP 2007. IEEE International Conference on*, 6, 2007.

[10] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2003.

[11] I. T. Jolliffe. *Principal component analysis*. Springer-Verlag New York, 1986.

[12] Y. Ke and R. Sukthankar. Pca-sift: A more distinctive representation for local image descriptors. *Proc. CVPR*, 2:506513, 2004.

[13] E. J. Keogh and M. J. Pazzani. Scaling up dynamic time warping for datamining applications. *Proceedings of the ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 285–289, 2000.

[14] E. J. Keogh and M. J. Pazzani. Derivative dynamic time warping. *SIAM International Conference on Data Mining*, 2001.

[15] T. Lindeberg. Scale-space theory: a basic tool for analyzing structures at different scales. *Journal of Applied Statistics*, 21(1):225–270, 1994.

[16] D. G. Lowe. Object recognition from local scale-invariant features. *International Conference on Computer Vision*, 2:1150–1157, 1999.

[17] D. G. Lowe. Local feature view clustering for 3d object recognition. *IEEE Conference on Computer Vision and Pattern Recognition, Kauai, Hawaii*, pages 682–688, 2001.

[18] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[19] Y. Ma, S. Soatto, J. Kosecka, and S. S. Sastry. An invitation to 3d vision: From images to geometric models., 2003.

[20] K. Mikolajczyk and C. Schmid. An affine invariant interest point detector. *Proc. ECCV*, 1:128142, 2002.

[21] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, 2005.

[22] H. Murase and S. K. Nayar. Detection of 3d objects in cluttered scenes using hierarchical eigenspace. *Pattern Recognition Letters*, 18(4):375–384, 1997.

[23] J. P. W. Pluim, J. B. A. Maintz, and M. A. Viergever. Mutual-information-based registration of medical images: a survey. *Medical Imaging, IEEE Transactions on*, 22(8):986–1004, 2003.

[24] W. H. Press. *Numerical recipes*. Cambridge University Press New York, 1986.

[25] L. Rabiner and B. H. Juang. *Fundamentals of speech recognition*. Prentice-Hall, Inc. Upper Saddle River, NJ, USA, 1993.

[26] A. Roche, G. Malandain, X. Pennec, and N. Ayache. The correlation ratio as a new similarity measure for multimodal image registration. *Proc. MICCAI*, 98:1115–1124, 1998.

[27] M. A. Turk and A. P. Pentland. Face recognition using eigenfaces. *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on*, pages 586–591, 1991.

[28] B. K. Yi, H. V. Jagadish, and C. Faloutsos. Efficient retrieval of similar time sequences under time warping. *Data Engineering, 1998. Proceedings., 14th International Conference on*, pages 201–208, 1998.

[29] B. Zitova and J. Flusser. Image registration methods: a survey. *Image and Vision Computing*, 21(11):977–1000, 2003.