PRELIMINARY STUDY OF ANALYTICAL IMAGE MATCHING

Meng Qian Shen[1] and Yi Hsing Tseng[2]
Department of Geomatics, National Cheng Kung University, No. 1, Daxue Rd, Tainan City 70101, Taiwan
[1]Email: p66054167@mail.ncku.edu.tw; [2]Email: tseng@mail.ncku.edu.tw

KEY WORDS: IMAGE MATCHING, SIFT, IMAGE FEATURE.

ABSTRACT: Automatic image matching has been an essential task in the field of digital photogrammetry. Photogrammetric triangulation needs control points and tie points in the overlapped images to construct connections among images. Usually, we choose corners or some specific marks on images as tie points. On the other hand, SIFT (Scale Invariant feature transform) is known for image matching in computer vision, which extracts feature points in each image, and matches these images together according to their unique descriptors. This study discompose SIFT algorithm, in which contains octaves and levels, representing different spatial scales and image resolutions, to figure out the relationship hidden behind the algorithm. We first start from large scale images, step by step to match small scale ones, comparing accurate matches of different scale combinations to analyze the differences and accuracy. However, there could be some error matching, so we use RANSAC(RANdom SAmple Consensus) to remove outliers for higher accuracy and precision. By affine transformation and solving relative orientations of each image pair, we can get the residuals of images that go through several different levels of image matching. This study is tyring to analyze how different scales and image resolution affect image matching results.

1. INTRODUCTION

Nowadays, computer vision methods has become a widely used algorithm on image matching. In computer vision, it accelerates the procedure of image matching, making it faster than traditional photogrammetry, which do image matching based on space intersection. There are lots of commercial software now can deal with numerous photos in one time, matching these image and build a model automatically, such as Pix4D, Australis, PhotoScan……etc. The background algorithm of the software is based on the same points in the overlapped area of each images in photogrammetry.

SIFT (Scale-Invariant Feature Transform) (Lowe, 1999) is one of the most popular method for extracting image features which are robust to rotation, scale, illumination, and transformation. The purpose of this study is to discompose this image matching method and get to know more about how different factors and parameters such as image resolution, octaves, and levels affect the results.

Since SIFT algorithm has been published in 2004, more and more studies started to use this concept. Brown and Lowe (2007) used invariant features to do automatic panoramic image stitching, that make all images become one large image. Gong, Jiao, Tian, and Wang (2014) analyzed coarse-to-fine scheme image registration based on SIFT image features to find out the performance of SIFT on different scales. In addition, some researchers tried to improved SIFT and make it faster while processing images, such as SURF (Speeded-up robust features) (Bay, Ess, Tuytelaars, & Van Gool) (2008). They used box filter to accelerate image convolution processing, and according to Juan and Gwun (2009), SIFT is more stable relatively, and SURF is more efficiency.

2. METHODOLOGY

2.1 Detection of scale-space extrema

The first step of SIFT method is to detect keypoints from images, extracting candidate keypoints and then examined in further steps. First, it is necessary to identify locations of keypoints in different octaves and levels, which means to search for stable features across all scales to make sure these keypoints are invariant to scale change of the image. We used Gaussian function to simulate an image under different resolution. I(x, y) represents the input image, and G(x, y,σ) represents the Gaussian function, therefore, an image can be defined as a function, L(x, y,σ), that comes from the convolution of G and I. For each octave of scale space, the original image is convolved with Gaussians repeatedly; furthermore, the Gaussian image is down-sampled by a factor of two and repeat the process after each octave.

The difference-of–Gaussian, D(x, y,σ), images come from the subtraction of adjacent images, which is computed from the difference of adjacent levels separated by a constant factor k.

$$L(x, y,\sigma) = G(x, y,\sigma) * I(x, y)$$

$$D(x, y,\sigma) = (G(x, y,k\sigma) - G(x, y,\sigma)) * I(x, y)$$
$$= L(x, y,k\sigma) - L(x, y,\sigma).$$

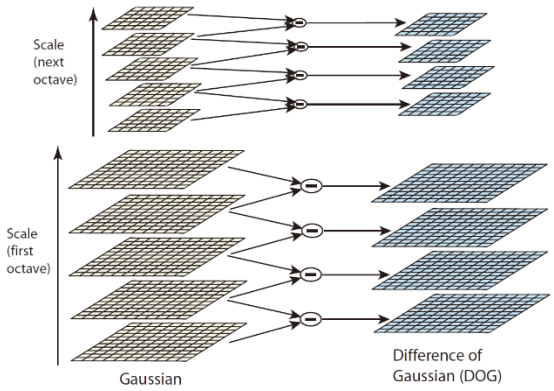

Figure 1: The Gaussian images come from repeatedly convolution of the original image and Gaussian function. The difference-of-Gaussian images come from the subtraction of adjacent levels. (David G. Lowe Copyright 2004).

The constant factor k is defined by the number of levels, and suppose each octave has s levels, then k will be $2^{1/s}$, because the sigma in next octave will double. To find the local maxima and minima, each pixel is compared to its eight neighbors in the same scale and nine neighbors in the above scale and below scale, therefore, one pixel would be compared to 26 pixels except for the edge ones. A pixel would be selected as a candidate keypoint if it is larger or smaller than all of them. This step would eliminate most of the sample points and get keypoints in all scales.

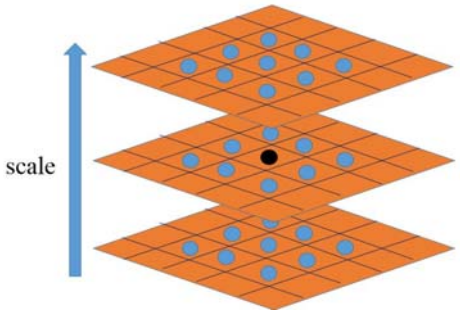

Figure 2. Non-maxima suppression

After the difference-of-Gaussian part is done, it will have a strong response along edges, and the response will become noise and make the result unstable. Follow by the research of corner and edge detector by Harris and Stephens (1988), it's easy to remove edge response by testing their principal curvatures which can be computed from a Hessian matrix, $\boldsymbol{H}$, that is constructed by taking second derivative for each extrema, which means calculating the difference of neighbour points. Suppose A is the eigenvalue with the larger magnitude and B is the smaller one for one extrema, so A and B will be proportional to the principal curvatures of the local correlation function. According to formulas, we can compute the summation of A and B from the trace of $\boldsymbol{H}$ and the product from the determinant:

$$D=\begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix}$$

$$\mathrm{Tr}(H) = D_{xx} + D_{yy} = A + B$$

$$\text{Det(H)}= D_{xx}D_{yy} - D_{xy}D_{xy}=AB$$

Let *r* be the ratio between *A* and *B*, so that *A*=*rB*. Then, according to the above equations, we have:

$$\frac{Tr(H)^2}{Det(H)} = \frac{(A+B)^2}{AB} = \frac{(rB+B)^2}{rB^2} = \frac{(r+1)^2}{r}$$

The equation only depends on the ratio of the eigenvalues, and the quantity (r+1)/r will have a minima while *A*=*B*, and increase with r. For now, we can set a threshold, r=10, in this study, and then we only need to check:

$$\frac{Tr(H)^2}{Det(H)} < \frac{(r+1)^2}{r}.$$

If one curvature is high and the other is low, it indicate it is an edge because it only shift along the edge,; on the other hand, if both curvatures are high, the local correlation is sharply peaked and shift in any direction that indicates it is a peak. This equation help us to eliminate keypoints that have a ratio between the principal curvatures greater than the threshold.

2.2 Description

To achieve image rotation invariance, we need to assign magnitude and orientation to each keypoint that is detect from the previous part. The gradient magnitude, m(x, y), and orientation, θ(x, y), for all sample points in the same scale can be computed by following the formulas bellow:

$$\text{m(x, y)}=\sqrt{(L(x+1,y)-L(x-1,y))^2+(L(x,y+1)-L(x,y-1))^2}$$

$$\theta\text{(x,y)}=tan^{-1}((L(x,y+1)-L(x,y-1))/(\ L(x+1,y)-L(x-1,y)))$$

An orientation histogram is built from the gradient orientation of sample points. We divide 360 degrees into 10 bins, and each sample point is added to the histogram before being weighted by its gradient magnitude and by a Gaussian-weighted window with σ, which is 1.5 times the scale of the keypoint. In addition, we only consider the range within 3σ. The peak in the histogram defines major orientation; however, there could be some peak that is within 80% of the highest one, so this keypoint will be recorded twice with the same locations and scale, but different orientation and magnitude.

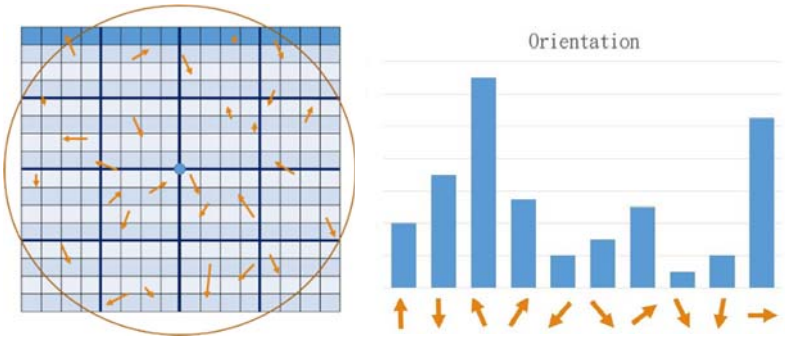

Figure 3. Image gradients under the range of Gaussian circular window as shown on the left. The peak value will be selected as the major orientation of a keypoint as shown on the right.

The last step of description is to build unique descriptors for every keypoint. We divided the range which lies in the Gaussian window around a keypoint into 4×4 subregions, and the pixels in each subregion is weighted by the Gaussian function, then assigned to several directions. This time, we separate 360 degrees into eight bins, which means each bin contains 45 degrees. In the end, each keypoint will have a 4×4×8 descriptor, and based on the unique descriptor that is robust to transformation, scale, and rotation, we will be able to match images.

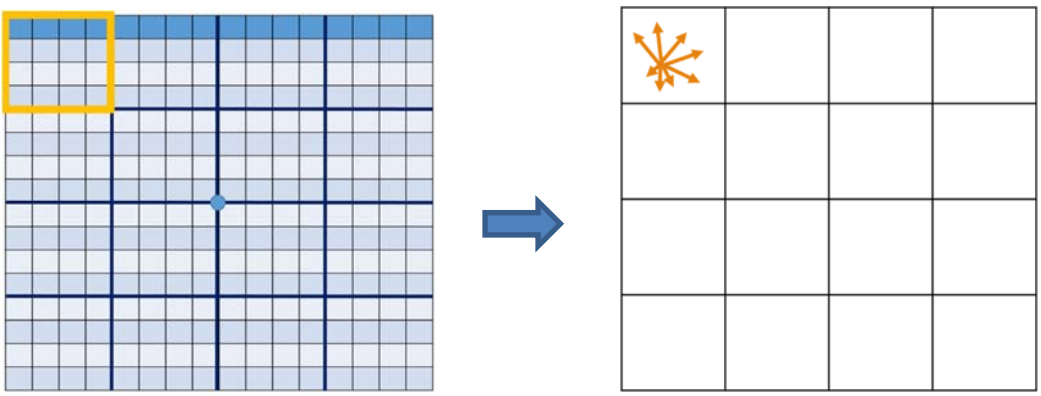

Figure 4. The pixels in each subregion will accumilate and be assigned to 8 bins, and gather all subregions will form a 4×4×8 descriptor for a keypoint.

3. EXPERIMENT AND ANALYSIS

The following figures show the results of image matching in different octaves and level. As long as we change the number of octave and level, the number of keypoints that were detected will change as well. First, we choose two images that were taken at different angles, downscaled by 0.5, starting from increasing the number of level in one octave.

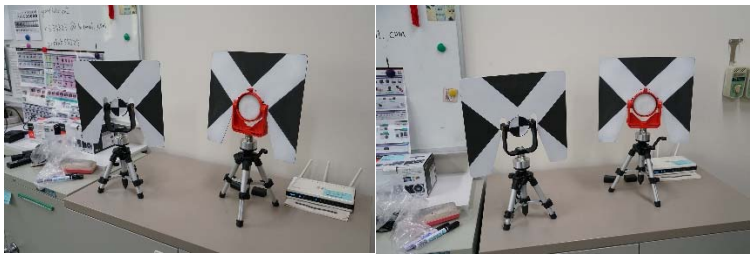

Figure 5. Original images (2652 pixels × 3976 pixels) at two positions.

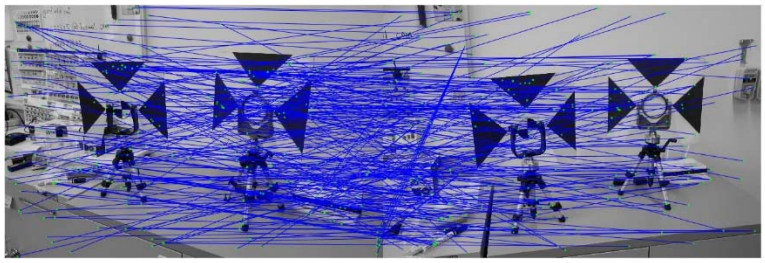

Figure 6. 1 octave and 3 levels (1326 pixels × 1988 pixels)

Though we can match the images by their feature points, there could exist some error matching as in Figure 6. Therefore, we use RANSAC (RANdom SAmple Consensus) (Fischler & Bolles, 1981) to remove outliers as figure 12 shows. It's an outlier detection method to estimate parameters of a mathematical model from a set of observed data. RANSAC can produce a model that is computed from inliers, providing that the probability of choosing only inliers in the selection of data is sufficiently high. We can rely on RANSAC to eliminate errors and achieve higher precision.

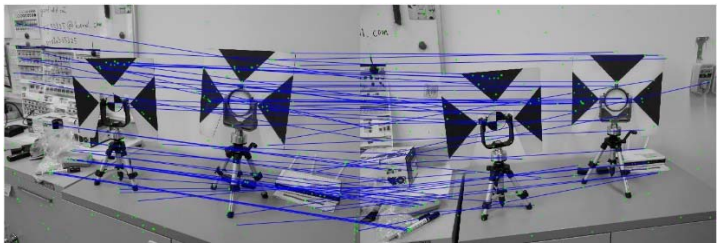

Figure 7. Remove error matching by RANSAC.

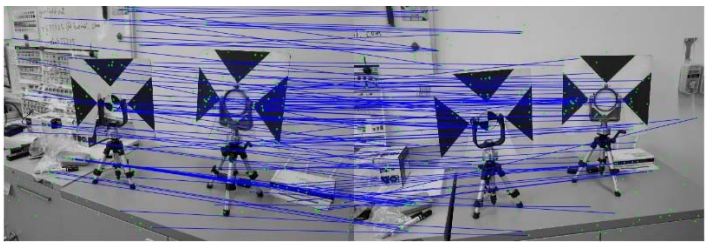

Figure 8. 1 octaves and 6 levels

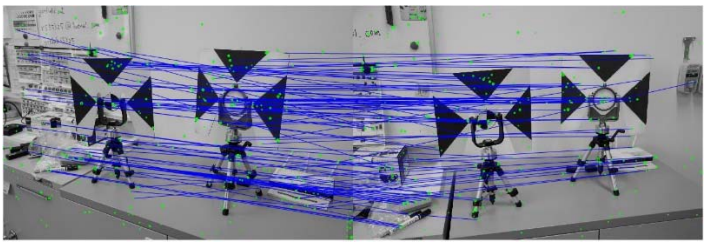

Figure 9. 2 octaves and 3 levels

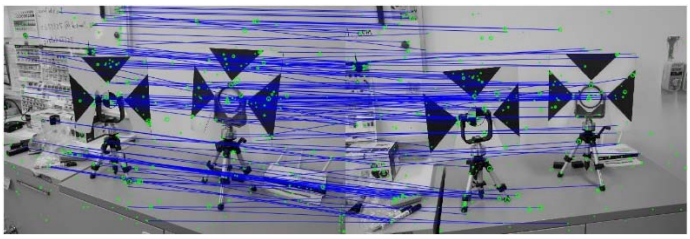

Figure 10. 3 octaves and 3 levels

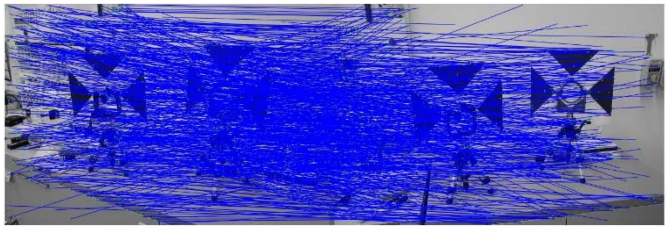

Figure 11. high resolution (2652×3976)

Table 1. Matching pairs before and after RANSAC.

RANSAC	1octave 3levels	1octave 6 levels	2octaves 3 levels	2octaves 6 levels	3octaves 3 levels	3octave 6 levels
Before	262	409	370	557	411	611
After	174	132	113	211	144	247

From table 1 we can tell that with the number of octave and level increase, the number of feature points increase as well. However, when we tried high-resolution one, it would detect too many keypoints that caused the matching result not good enough. From figure 11, we can see that although two images can matched successfully, there were some error matching that should be removed.

4. CONCLUSIONS

We can use SIFT and RANSAC to extract images features automatically, and more octaves and levels can detect more keypoints. By discomposing each step of SIFT algorithm, we know how two images are matched based on the unique descriptors. However, the more octaves and levels we use in the algorithm, the more time it will take while processing. It's very important to decide how many octaves and levels we need to use, and choose an appropriate way, or it may lead to time and source consuming. After image matching, there could exist some error matching that we have to rely on RANSAC to remove those error ones and improve the accuracy. Especially, in high-resolution images, we can find much more matching pairs, and in some cases, we can't use RANSAC to

eliminate those error ones. It is complicated to separate the successful ones and the error ones, so in future maybe we can try to detect the features starting from large scale to avoid detecting too many tiny features that should be considered as noise in the whole processing part.

REFERENCES

Bay, H., Ess, A., Tuytelaars, T., & Van Gool, L. (2008) Speeded-up robust features (SURF). Computer vision and image understanding, 110(3), pp.346~359.

Brown, M., & Lowe, D. G. (2007). Automatic panoramic image stitching using invariant features. International Journal of Computer Vision, 74(1), pp.59~73.

Fischler, M. A., & Bolles, B. C. (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM, 24(6), pp.381~395.

Harris, C., &Stephens, M. (1988). A combined corner and edge detector. Proceedings of the 4thAlvey Vision Conference, pp.147~151.

Juan, L., & Gwun, O. (2009). A comparison of sift, pca-sift, and surf. International Journal of Image Processing (IJIP), 3(4), pp.143~152.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision, 60(2), pp.91~110.

Acknowledgements

This research is supported and sponsored by Ministry of Science and Technology, Taiwan, R.O.C under Grant no. MOST 105-2627-M-006-012.