Panoramic Image Stitching using SURF

Meng You*, Jong-Seok Lim*, Wook-Hyun Kim*

Abstract

This paper proposes a new method to process panoramic image stitching using SURF(Speeded Up Robust Features). Panoramic image stitching is considered a problem of the correspondence matching. In computer vision, it is difficult to find corresponding points in variable environment where a scale, rotation, view point and illumination are changed. However, SURF algorithm have been widely used to solve the problem of the correspondence matching because it is faster than SIFT(Scale Invariant Feature Transform). In this work, we also describe an efficient approach to decreasing computation time through the homography estimation using RANSAC(random sample consensus). RANSAC is a robust estimation procedure that uses a minimal set of randomly sampled correspondences to estimate image transformation parameters. Experimental results show that our method is robust to rotation, zoom, Gaussian noise and illumination change of the input images and computation time is greatly reduced.

Keywords : SURF, RANSAC, image stitching, image matching

I. Introduction

Panoramic image stitching is the process of combining multiple images with overlapping region or matching area in the scene to achieve a seamless transform result of images by using feature points and blending technique. Image features are a significant element in the image stitching. The features are invariant to image zooming, rotation, Gaussian noise and illumination. Image stitching algorithms have an extensive research work[1-2] and several commercial applications[3]. They also come bundled with most digital cameras currently being sold.

In the previous work the methods for image stitching fall largely into two categories-direct and feature based. Direct methods[4] have the advantage that they use all of the available image data and hence can provide very accurate registration, but they require a close initialization. Feature based methods[5] begin by establishing correspondences between points, lines or other geometrical entities. However, neither of these methods are robust to image zoom, rotation, noise and change in illumination.

Recently there has been great advance in the use of invariant features for image stitching and matching . These features can be found more repeatedly and matched more reliably than traditional methods such as correlation using Harris corner detector[6]. Harris corner detector is not invariant to scaling of the image, and correlation of image patches is not invariant to rotation. There are two ways to deal with invariant features-SIFT and . Th. Lowe's[2] Scale Invariant Feature Transform(SIFT) is geometrically invariant under similarity transforms and invariant under affine changes in intensity. The Speeded Up Robust Features(SURF)[7] approach approximates or even outperforms the SIFT scheme with respect to repeatability, distinctiveness, and robustness, yet can be computed and compared much faster.

Once features have been extracted from images, they must be matched for image stitching. However, it is an inefficient approach to use all the image features and a time consuming task.

In this paper we describe a rapid and an invariant feature based approach to panoramic image stitching. This has several advantages over previous approaches. Firstly, our method enables reliable matching of panoramic image sequences despite rotation, zoom, noise and illumination change in the input images. Secondly, we generate high-quality results using blending to render seamless output panoramas. Thirdly, we can decrease computation time through the homography estimation using RANSAC.

The remainder of the paper is organised as follows. Section Ⅱ describes SURF. Section Ⅲ describes image matching. In

* 영남대학교
투고 일자 : 2011. 1. 10 수정완료일자 : 2011. 1. 28
게재확정일자 : 2011. 2. 2

section Ⅳ we describe image stitching. Finally, section Ⅴ shows the experimental results and section Ⅵ concludes this paper.

Ⅱ. SURF

SURF(Speeded Up Robust Features) is a robust image detector and descriptor. It is partially inspired by SIFT descriptor. The detector is based on approximated Hessian matrix[8]. The descriptor, on the other hand, describes a distribution of Harr-wavelet responses within the interest point neighbourhood. Both the detector and the descriptor rely on integral images to reduce the computation time. Therefore, the SURF outperforms previously proposed schemes with respect to repeatability, distinctiveness, robustness and speed.

A. Fast Hessian Detector

The SURF detector is based on the Hessian matrix for good performance in computation time and accuracy. Given a point x=(x, y) in an image I, the Hessian matrix H(x, σ) in x at scale σ is defined as follows

$$\mathbb{H}(\mathrm{x},\sigma)=\begin{bmatrix} L_{xx}(\mathrm{x},\sigma) & L_{xy}(\mathrm{x},\sigma) \\ L_{xy}(\mathrm{x},\sigma) & L_{yy}(\mathrm{x},\sigma) \end{bmatrix}, \qquad (1)$$

where $L_{xx}(\mathrm{x},\sigma)$ is the convolution of the Gaussian second order derivative $\frac{\partial^2}{\partial x^2}g(\sigma)$ with the image I in point x, and similarly for $L_{xy}(\mathrm{x},\sigma)$ and $L_{yy}(\mathrm{x},\sigma)$.

The approximated Hessian detector use the approximated Hessian matrix using box filters as shown in Fig. 1 instead of Hessian matrix. The 9x9 box filters in Fig. 1 are approximations for Gaussian second order derivatives with σ =1.2(experimental value) and denote by D_{xx}, D_{yy} and D_{xy}. Hessian's determinant is as follows

$$\frac{|L_{xy}(1.2)|_F|D_{xx}(9)|_F}{|L_{xx}(1.2)|_F|D_{xy}(9)|_F}=0.912...\simeq 0.9, \qquad (2)$$

where $|x|_F$ is the Frobenius norm. This yields

$$\det(\mathbb{H}_{\approx})=D_{xx}D_{yy}-(0.9D_{xy})^2. \qquad (3)$$

Furthermore, the filter responses are normalised with respect to the mask size.

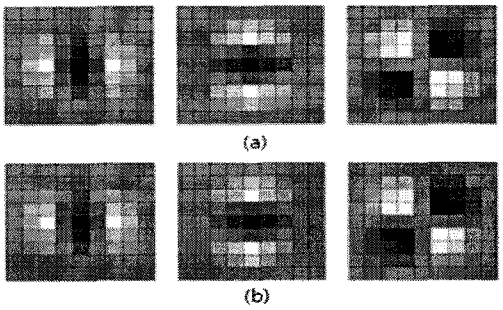

Fig. 1. (a) Second order Gaussian partial derivatives in x, y, xy direction
(b) Approximated box filters in x, y, xy direction

In computer vision, scale spaces are generally implemented as image pyramids. The images are iteratively smoothed with a Gaussian and subsequently sub-sampled in order to achieve a higher level pyramid images. The scale space is analysed by up-scaling the filter size rather than repeatedly reducing the image size. The initial scale layer is the output of the above 9x9 filter. The following layers are obtained by filtering the image with gradually bigger masks. As a result, the filter sizes result in 9x9, 15x15, 21x21, 27x27 etc.

B. SURF descriptor

The first step of the SURF descriptor extracting consists of fixing a reproducible orientation based on information from a circular region around the interest point. Then, it construct a square region aligned to the selected orientation. In order to be invariant to rotation, it is calculated the Haar-wavelet responses in x and y direction, shown in Fig. 2, and this is processed in a circular neighbourhood of radius 6s around the interest point, with s the scale at which the interest point was detected. The dominant orientation is estimated by calculating the sum of all responses within a sliding orientation window covering an 60 degree. The horizontal and vertical responses within the window are summed. The two summed responses then yield a new vector. The longest such vector lends its orientation to the interest point.

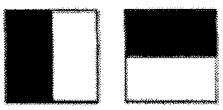

Fig. 2. Haar-wavelet responses in x and y direction

For the extraction of the descriptor, the first step consists of constructing a square region centered around the interest point, and oriented along the orientation selected above

mentioned. The size of this window is 20s.

The region is split up regularly into smaller 4x4 square sub-regions. For each sub-region, a few simple features are computed at 5x5 regularly spaced sample points. Usually d_x is called the Haar wavelet response in horizontal direction and d_y the Haar wavelet response in vertical direction.

Then, the wavelet responses d_x and d_y are summed up over each subregion and form a first set of entries to the feature vector. It also is extracted the sum of the absolute values of the responses, $|d_x|$ and $|d_y|$. Hence, each sub-region has a 4 dimensional descriptor vector v for its underlying intensity structure $V(\Sigma d_x, \Sigma d_y, \Sigma|d_x|, \Sigma|d_y|)$. This results in a descriptor vector for all 4x4 sub-regions.

III. Image Matching

Once features have been extracted from all images using SURF, they must be matched. We use RANSAC to select a set of inliers that are compatible with a homography between the images.

RANSAC(random sample consensus)[9] is a robust estimation procedure that uses a minimal set of randomly sampled correspondences to estimate image transformation parameters, and finds a solution that has the best consensus with the data. Essentially, it is a sampling approach to estimation **H**.

The image pixel (x, y) corresponding to a point (s, t) in the scene plane is obtained by

$$\begin{bmatrix} wx \\ wy \\ w \end{bmatrix} = \mathbf{P} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{4}$$

where **P** is projection matrix.

These two mappings together define a 3x3 matrix **H** mapping $(s, t, 1)$ to (wx, wy, w)

$$\begin{bmatrix} wx \\ wy \\ w \end{bmatrix} = \mathbf{H} \begin{bmatrix} s \\ t \\ 1 \end{bmatrix} \tag{5}$$

Such an matrix H is called a homography.

In the case of panoramas we select sets of 4 feature correspondences and compute the homography **H** between them.

The typical RANSAC approach is as follows:

1. Find many feature points in each of the two images: (x_i, y_i) and $(x'_i\ y'_i)$.

2. For each feature point (x_i, y_i) in one image, find a candidate corresponding feature point in the second image $(x'_i\ y'_i)$ whose intensity neighborhood is similar. This give a set of 4-tuples$(x_i, y_i, x'_i\ y'_i)$.

3. Randomly choose four 4-tuples and fit an exact homography **H** that maps the four (x_i, y_i) exactly to thier corresponding $(x'_i\ y'_i)$. Find the consensus set for that homography, namely find the number of other 4-tuples for whom the distance of the 4-tuple from the model is sufficiently small.

The distance could be defined in a variety of ways e.g. compute

$$\begin{bmatrix} w\tilde{x} \\ w\tilde{y} \\ w \end{bmatrix} = \mathbf{H} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{6}$$

and define

$$dist(\mathbf{H}, x_i, y_i, x_i', y_i') = \| (\tilde{x}, \tilde{y}) - (x', y') \| \tag{7}$$

4. After repeating step 3 a certain number of times(or until find a homography whose consensus set exceeds some pre-determined threshold), choose the homography with the largest consensus set and use that consensus set to re-estimate the homography **H** using least squares.

We repeat this with 400 trials and select the solution that has the maximum number of inliers. Fig. 3 show SURF features extracted from all of the images and RANSAC inliers. The number of SURF features in (c) and (d) is 1086 and 1115. But, the number of RANSAC inliers in (e) and (f) is 136 and 136.

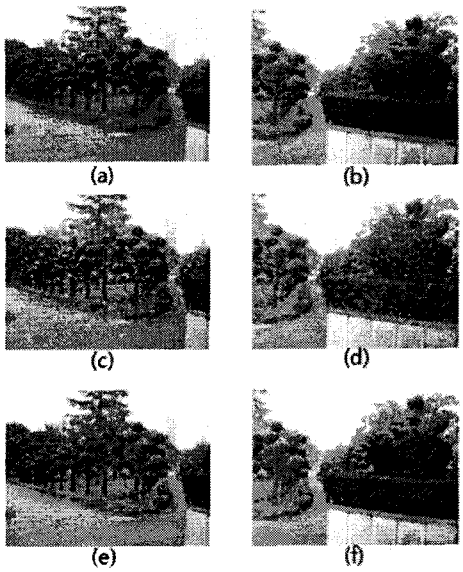

Fig. 3. (a) input image1 (b) input image2
(c) SURF feature1 (d) SURF feature2
(e) RANSAC inliers1 (f) RANSAC inliers2

IV. Image Stitching

A. Cylindrical Coordinates

Before we can align images to do panoramic image stitching, we need to establish the motion model that map pixel coordinates from one image to another. A variety of such parametric motion models are possible such as 2D transforms, planar perspective models and the mapping to non-planar surfaces. In this paper, we wish to project an image onto a cylindrical surface of unit radius[10]. Points on this surface are parameterized by an angle θ and a height h, with the 3D cylindrical coordinates corresponding to (θ,h) given by

$$(\sin\theta,\ h,\ \cos\theta) \propto (x,\ y,\ f) \tag{8}$$

From this correspondence, we can compute the formula for the warped or mapped coordinates[11]

$$x' = s\theta = s\tan^{-1}\frac{x}{f} \tag{9}$$

$$y' = sh = s\frac{y}{\sqrt{x^2+f^2}} \tag{10}$$

where s is an arbitrary scaling factor(sometimes called the radius of the cylinder) that can be set to s=f to minimize the distortion(scaling) near the center of the image. Fig. 4 shows original and cylindrically warped image respectively. And Fig. 5 results from image

Fig. 4. Original and cylindrically warped image

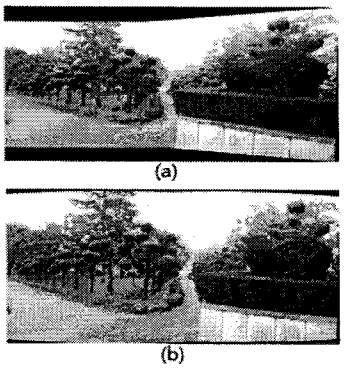

Fig. 5. (a) stitched image with original image (b) stitched image with cylindrically warped image

stitching using original image and cylindrically warped image.

B. Feature-based Alignment and image blending

Once we have chosen a suitable motion model to describe the alignment between a pair of images, we need to devise some method to estimate its parameters. One approach is to shift or warp the images relative to each other and to look at how much the pixels agree. The feature-based methods have advantage that computation time is faster than direct method that use pixel-to-pixel matching. In this paper, we have utilized the feature-based approach to align images. This method is to first extract distinctive features from each image, to match these features to establish a global correspondence, and to then estimate the geometric transformation between the images. This kind of approach has been used extensively until now[12].

After image alignment image blending is necessary to maintain the quality of the input images. In most cases neighbouring image edges show intensity discrepancies which are undesirable. In order to eliminate such problems, a blending method is used. The advantage of blending algorithm[13] is to improve visual quality of the composite image and making the edges invisible.

The composite image consists of a number of images that are initially placed next to each other. An empty composite image of adequate size is firstly created. Images are then placed in side by side.

Each image in turn is put in the composite image and its position is determined by feature-based method between the new image and the composite image. The blending algorithm is then applied and the process is repeated for all other images. In Fig. 6 the overlap between a new image and the composite image is shown(gray area). In the overlapped area the image blending algorithm calculates the contribution of the new image and the composite image at every pixel.

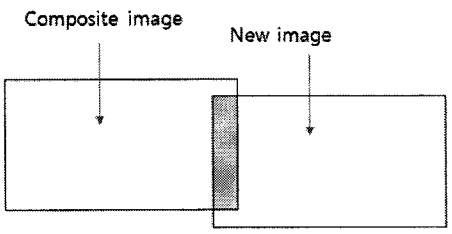

Fig. 6. Overlapped images

The blended image consists of pixels as follows:

$$N(x,y) = \alpha I(x,y) + (1-\alpha)\,C(x,y) \tag{11}$$

where $C(x,y)$ is the composite image pixel(before placing the new image), $I(x,y)$ is the new image pixel, $N(x,y)$ is the new composite image pixel(with new image added) and α is a weighting factor.

The blending algorithm minimizes effects of intensity variations, removes the edges.

V. Experimental Results

The proposed method has been tested on images acquired on outdoor environments. Several images have been acquired by changing zoom and rotation parameters of the camera. Some images were added Gaussian noise. The system was implemented in Microsoft Visual C++ 6.0 and was run on Pentium 4 3GHz PC with 2G RAM. The experiment have used 4 types of images such as zoom, rotation, Gaussian noise, and change of illumination. Two images used in the experiment have all 640x480 size. In this paper we have processed image stitching with two images. One is regular image, the other is modified image with a various of methods.

To evaluate performance of the proposed method, we first compared the number of feature points matched using each algorithm. Table 1 shows the results. In Table 1, N means normal image, Z means zoom image, R means rotated(90 degree) image in a clockwise, GN means Gaussian noise(10%) image, CI means changed illumination image, HC means Harris corner detector, and PA means proposed approach.

Table 1. The number of matching points

	N	Z	R	GN	CI
SIFT	250	291	235	145	254
HC	78	35	4	39	53
PA	90	69	80	61	55

This results show the number of matching points of the Harris corner detector is smallest among all the algorithms. However if the number of matching points is very small, result of the image stitching can not be good. Actually, in case of rotation, Harris corner algorithm failed to do image stitching. Fig. 7 shows example image of the matching points using proposed algorithm for each case of the Table 1.

Next, we compared detecting time of feature points, matching time of detected feature points, and stitching time of two images for each algorithm. Tables 2-6 show the results.

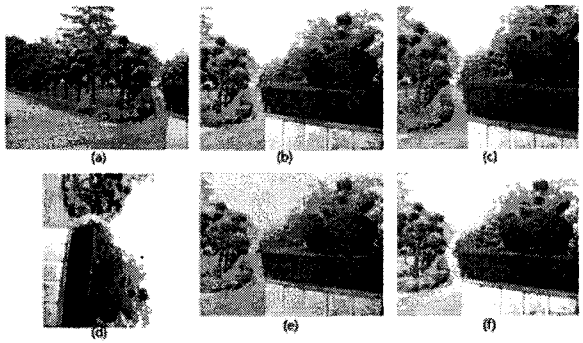

Fig. 7 matching points of (a) input image1 (b) input image 2 (c) Zoom image (d) Rotated image (e) Gaussian Noise image (f) Changed Illumination image for proposed algorithm

Table 2. DT, MT, ST, and TT for normal image

N	DT	MT	ST	TT
SIFT	2234	203	97	2,534
HC	63	1843	94	2,000
PA	1297	110	78	1,485

Table 3. DT, MT, ST, and TT for zoom image

Z	DT	MT	ST	TT
SIFT	2375	219	97	2,691
HC	63	2000	78	2,141
PA	1360	109	78	1,547

Table 4. DT, MT, ST, and TT for rotated image

R	DT	MT	ST	TT
SIFT	2203	219	97	2,519
HC	63	1359	141	1,563
PA	1359	109	94	1,562

Table 5. DT, MT, ST, and TT for Gaussian noise image

GN	DT	MT	ST	TT
SIFT	2000	203	94	2,297
HC	78	2656	78	2,812
PA	1328	109	78	1,515

Table 6. DT, MT, ST, and TT for changed illumination image

CI	DT	MT	ST	TT
SIFT	2200	219	97	2,516
HC	63	1797	78	1,938
PA	1281	109	78	1,468

In Tables 2-6, The time unit is millisecond, DT means detecting time, MT means matching time, ST means stitching time, and TT means total time. The result of Tables 2-6 show DT, MT, ST and TT of the each algorithms. The number of matching points is not an element to determine the performance of an algorithm. Because the larger the number of matching points, the longer the time of image stitching. Also if it is too small, image stitching is impossible. Actually in case of the rotated image, the number of matching points of the harris corner is very small but image stitching has failed.

In case of detection time(DT), harris corner detector(HC) took the shortest time. However it took the longest time in case of matching time(MT). As a result, the proposed algorithm took the shortest time in case of total time(TT). HC is only best in case of DT but image stitching can be failed in the specific image(in case of R). Therefore the proposed algorithm proved the best algorithm through experiment results.

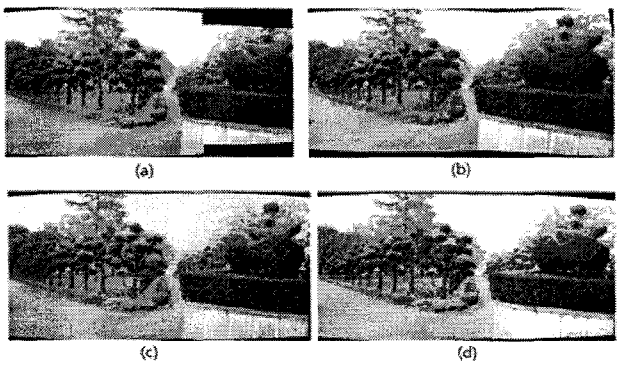

Fig. 8 stitching image of (a) Zoom image (b) Rotated image (c) Gaussian Noise image (d) Changed illumination image for the proposed algorithm

Fig. 8 shows stitched image using proposed algorithm for each case and can see good results in general.

VI. Conclusion

In this paper, we proposed a new method to process panoramic image stitching using SURF algorithm. And also we described an efficient approach to decreasing computation time through the homography estimation using RANSAC. The proposed approach has been compared the number of feature points matched and computation time with conventional methods. Experimental results showed that our method is robust to rotation, zoom, Gaussian noise and illumination change of the input images and computation time is fastest among other methods.

Future work will include an algorithm for both multi-images stitching and reduction of the blurriness.

References

[1] R.Szeliski, "Image alignment and stitching: A turorial," *Microsoft Research, Technicla Report*, MSR-TR-2004-92, 2004.

[2] D.Lowe, "Distinctive image features from scale- invariant key-points," *International Journal of Computer Vision*, vol. 60(2), pp. 91-110, 2004.

[3] S.Chen, "Quick Time VR-An image-based approach to virtual environment navigation," *In SIGGRAPH'95*. vol. 29, pp. 29-38, 1995.

[4] H.Shum and R.Szeliski, "Construction of panoramic mosaics with global and local alignment," *International Journal of Computer Vision*, vol. 36(2), pp. 101-130, 2000.

[5] P.McLauchlan and A.Jaenicke, "Image mosaicing using sequential bundle adjustment," *Image and Vision Computing*, vol. 20(9-10), pp. 751-759, 2002.

[6] C.Harris, "Geometry from visual motion," *Active Vision*, MIT Press, pp. 263-284, 1992.

[7] H.Bay, T.Tuytelaars, and I.V.Gool, "Surf:Speeded up robust features," *European Conference on Computer Vision*, vol. 3951, pp. 404-417, 2006.

[8] T.Lindeberg, "Feature detection with automatic scale selection," *International Journal of Computer Vision*, vol. 30(3), pp. 79-116, 1998.

[9] M.Fischler and R.Bolles, "Random sample consensus: A paradigm for model fitting with application to image analysis and autometed cartography," *Communications of the ACM*, vol. 24, pp. 381-395, 1981.

[10] R.Szeliski, "Image mosaicing for tele-reality applications, " *IEEE Workshop on Applications of Computer Vision*, pp. 44-53, 1994.

[11] R.Szeliski and H.Y.Shum, "Creating full view panoramic image mosaics and texture-mapped models" *Computer Graphics(SIGGRAPH'97)*, pp. 251-258, 1997.

[12] M.Brown, R.Szeliski and S.Winder, "Multi-image matching using multi-scale oriented patches," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition(CVPR'2005)*, pp. 510-517, 2005.

[13] V.Rankov, R.J.Locke, R.J.Edens, P.R.Barber and B.Vojnovic, " An algorithm for image stitching and blending," *Proceedings of SPIE*, vol. 5701, pp. 190-199, 2005.

Meng Yu received the B.S. and M.S. degree in computer engineering from Yeungnam University in 2007 and 2011. He is currently working towards Ph.D. degree on computer engineering.

His research interests include the areas of pattern recognition, computer vision and image processing.

Jong-Seok Lim received the B.S. gree in physics from Kyemyung University and the M.S. degree in Computer Science from Daegu Catholic University and the Ph.D. degree in Computer Engineering from Yeungnam University in 2004.

He is currently instructor in Yeungnam University.

His research interests include the areas of computer vision, image and video processing and pattern recognition.

Wook-Hyun Kim received the B.S. and M.S. degrees in electronic engineering from Kyungbuk National University in 1981 and 1983, respectively and the Ph.D. degree from University of Tsukuba in 1993.

He is currently a professor of Computer Engineering at Yeungnam University.

His research interests include the areas of computer vision, image processing and pattern recognition.