ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 4, April 2015

A Technical Analysis of Image Stitching Algorithm Using Different Corner Detection Methods

Pranoti Kale, K.R.Singh

Department of Computer Technology, YCCE, Nagpur, India

ABSTRACT: An image stitching is a method of combining multiple overlapping images of the same scene into a larger image without loss of information. Literature shows the use of various corner detection algorithms in image stitching. The most widely used are Harris corner detection method and SIFTS (Scale Invariant Feature Transform) method. In this paper, a comparative study is done forHarris corner detection algorithm and SIFT algorithm in image stitching using similarity matrix matching scheme. Total 30 pairs of different images have been used for simulation and comparison. The algorithms have been compared with respect to number of corner detected, number of matching pairs and matching time. From the simulation results it has been observed that SIFT corner detection method is more efficient in image stitching.

KEYWORDS: Image stitching, Corner detection, Harris corner, SIFT

I. INTRODUCTION

Image stitching is a sub branch of computer vision. It basically combines two or more different images to form one single image that is panorama. The word panorama is derived from the Greek words 'pan' and 'horama'. 'Pan' means everything and 'horama' means to view, and thus it means all round view. Panorama images can be created in a variety of ways, from the first round painting in the 18th and 19th centuries. The aim of stitching is to increase image resolution as well as the field of view; people used image stitching technology in topographic mapping. A topographic map is a type of map characterized by large-scale detail and quantitative representation of relief, using contour lines.

Typically, a camera is capable of taking pictures within the scope of its view only; it cannot take a large picture with all the details fitted in one single frame [2]. Panoramic imaging resolves this problem by combining images taken from different sources into a single image. Such images are useful for surveillance applications, video summarization, remote sensing etc. Image stitching algorithms create the high resolution photo mosaics used to produce today's digital maps and satellite photos. Creating high resolution images by combining smaller images are popular since the beginning of the photography.

To stitch images and form a panoramic image, the similarity of overlapping regions among adjacent images needs to be calculated in the first place.Intensity-based algorithms usually involve a large amount of computation and therefore are not appropriate for image alignment when there is image rotation and scaling. On the other hand, algorithms based on frequency-domain are in general faster and can handle well small translation, rotation, and scaling. Unfortunately, the performance of frequency domain-based algorithms will be degraded when dealing with scenarios where smaller overlapping regions exist. Feature-based algorithms utilize a small number of invariant points, lines, or edges to align images. One significant advantage of these algorithms is that the computational complexity will be reduced due to less information that needs to be processed. Additionally, feature-based algorithms are robust to changes in image intensity. However, there is one serious issue identified for many existing algorithms. Most of these algorithms make use of an exhaustive search that is based on template matching. As a result, the computation, although already decreased to some extent, is still intensive, which does not meet the real-time requirement usually found in panorama stitching [1].

In this paper we present two corner detection algorithms namely Harris corner detection algorithm and SIFT corner detectionalgorithm. We first detect corner of the input images, then perform image stitching by matching the corner

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 4, April 2015

points. This is done by calculating similarity matrix for each corner point. The image stitching process consists of three steps;first, filter out large numbers of candidate corners according to their position information using Harris and SIFT algorithm. Then, generate an initial set of matching-corner pairs based on gray scales of each corner's adjacent regions. Finally, combine the two images to get one single stitched image.

II. LITERATURE SURVEY

Over the period of last several years, many approaches have been proposed for image stitching.

Intensity based matching involves computation of similarity criteria like Sum of Squared Distances (SSD), correlation etc. These methods are capable of identifying the overlapping region in images which vary only by translation. If the images are subjected to complex geometric variations, these methods fail to capture the overlapping part of the scene in the source and target images [3].

Segmentation based methods are tried to determine the matching region from a pair of images. It consists of various methods working with grayscale and color images [4]. Segmentation algorithms generally classify the images based on histogram thresholding, fuzzy based approach, and region based approach etc. Histogram based methods identify the peaks and valleys in the image [5]. Though this method works fine for simple images, it fails when the input images vary by rotation or other complex transformations.

Another approach for matching region estimation is to operate the images in the frequency domain. Phase correlation method was employed to determine the overlapping region in [6]. These methods consider the properties of the cross power spectrum between the images. But it imposes a huge computational burden as computation has to be repeatedly performed at each pixel of the image.

Feature based methods identifies typical features from each image. These features are not affected by camera's perspective [7]. The discriminatingfeatures in the image include edge, corner, ridges etc. Feature based methods have gone through rapid development. Such methods include SUSAN detector [8], HARRIS corner detector [10] for feature extraction from images.

Shift Invariant Feature Transform (SIFT) is an efficient feature extraction algorithm for color images. These features are invariant to geometric transformations [11]. Many works were carried out using SIFT features for stitching the images. Though it gives good accuracy, it generates thousands of features for one image. Hence it imposes a huge computational burden.

Speeded-Up Robust Features (SURF) is another efficient invariant feature extraction algorithm and it is in wide use in many applications [3]. It provides a good balance between feature complexity and robustness to common deformations.

A research on feature-based image mosaic algorithm was given in [15]. It claims that SIFT is stable against rotation and scale variations, but it is very slow in computation. On the other hand, SURF functions are faster and with performance as good as SIFT [17].

III. METHODOLOGY

For stitching two images we need to detect the corners of each image. For corner detection we are using Harris corner detector and SIFT which is explained below in section A and B respectively.

A. Harris Corner Detector

Harris corner detection algorithm [6] was proposed by Harris C and Stephens MJ in the year 1988. It is an algorithm based on still image used for combined corner and edge detector. Reasonable amount of corner features are extracted which gives a better quantitative measurement by using a stable operator. A local detecting window in image is designed. The average variation in intensity is determined by shifting the window by a small amount in different direction. The centre point of the window is extracted as corner point. The point can be recognized easily by looking at the intensity values within a small window. Shifting the window in any direction gives a large change in appearance.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 4, April 2015

Harris corner detector is used for detecting corners. On shifting the window if it's a flat region than it will show no change of intensity in all direction. If an edge region is found than it will show no change of intensity along the edge direction. But if a corner is found than there will be a significant change of intensity in all directions. Harris corner detector gives a mathematical approach for determining the region is flat, edge or corner. Harris corner technique detects more features and it is rotational invariant and scale variant. To extract corner can give prominence to the important information. Those can be described by equation below.

$$E(u,v) = \sum_{x,y} w(x,y)[I(x+u, y+ v) I(x,y)]^2 \qquad (1)$$

Where,

- E is the difference between the original and the moved window.
- u is the window's displacement in the x direction.
- v is the window's displacement in the y direction.
- w(x, y) is the window at position (x, y). This acts like a mask. Ensuring that only the desired window is used.
- I is the intensity of the image at a position (x, y).
- I(x+ u, y+ v) is the intensity of the moved window.
- I(x, y) is the intensity of the original.

We've looking for windows that produce a large E value. To do that, we need to high values of the terms inside the square brackets. We expand this term using the Taylor series.

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

We tucked up this equation into matrix form

$$E(u,v) = [u \quad v]\left(\sum \begin{bmatrix} I_x^2 & I_xI_y \\ I_xI_y & I_y^2 \end{bmatrix}\right)\begin{bmatrix} u \\ v \end{bmatrix} (3)$$

After that rename the summed-matrix, and put it to be M:

$$M = \sum w(x,y) \begin{bmatrix} I_x^2 & I_xI_y \\ I_xI_y & I_y^2 \end{bmatrix} (4)$$

Harris corner can be defined as the maximum in local area by the following formula:

$$R = \mathrm{Det(M)} - k\,\mathrm{Trace}\,(M)^2 \ (5)$$

Where,

$$\mathrm{Det}\ (M) = \lambda_1 \lambda_2 \ (6)$$

$$\mathrm{Trace}\ (M) = \lambda_1 + \lambda_2 \qquad (7)$$

B. SIFT descriptor

SIFT [4] was first presented by David G Lowe in 1999. SIFT algorithm is very invariant and robust for feature matching with scaling, rotation, or affine transformation. We utilize SIFT feature points to find correspondent points of two sequence images. The SIFT algorithm is described through these main steps: scale space extrema detection, accuratekeypoint localization, orientation assignment and keypoint descriptor.

1) Scale space extrema detection

First, we build the pyramid of image by continuous smooth with Gaussian mask. DoG (Difference of Gaussian) pyramid of the image will be obtained by subtraction adjacent smoothed images. By comparing each pixel of current scale with upper and lower scales in the region 3x3, i.e. 26 pixels, we can find the maximum or minimum value

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 4, April 2015

among them. These points are also considered as candidate of keypoint. The equations below will be used to describe Gaussian function, scale space and DoG.

$$G(x,y,\sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} (8)$$

$$L(x,y,\sigma) = G(x,y,\sigma) * I(x,y) (9)$$

Where * is the convolution operation in x and y.

$$D(x,y,\sigma) = (G(x,y,k\sigma) - G(x,y,\sigma) * I(x,y)$$
$$= L(x,y,k\sigma) - L(x,y,\sigma) \quad (10)$$

2) *Accurate keypoint localization*

The initial result of this algorithm, considers keypoint location is at the central of sample point. However this is not the correct maximum location of keypoint then we need a 3D quadratic function to fit the local sample points to determine the true location, i.e. sub-pixel accuracy level of maximum value. Taylor expansion of the scale space function is shifted so the original is at the sample point.

$$D(x) = D + \frac{\partial D^r}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 D}{\partial x^2} x (11)$$

Where D and its derivatives are evaluated at the sample point and $x = (x, y, \sigma)^T$ is the offset from this point. The location of the extremum, $\hat{x}$, is determined by taking the derivative of this function with respect to x and setting it to zero, giving

$$\hat{x} = -\frac{\partial^2 D^{-1}}{\partial x^2} \frac{\partial D}{\partial x} (12)$$

The next stage attempts to eliminate some unstable points from the candidate list of key points by finding those that have low contrast or are poorly localized on an edge. For low contrast point finding, we evaluate $D(\hat{x})$ value with threshold. By substituting two equations above, we have: $\hat{x}$

$$D(\hat{x}) = \mathrm{D} + \frac{1}{2} \frac{\partial D^{-1}}{\partial x} \hat{x} (13)$$

If the value of $D(\hat{x})$ is below a threshold, this point will be excluded.

To eliminate poorly localized extrema we use the fact that in these cases there is a large principle curvature across the edge but a small curvature in the perpendicular direction in the difference of Gaussian function. A 2x2 Hessian matrix, H, computed at the location and scale of the key point is used to find the curvature. With these formulas, the ratio of principle curvature can be checked efficiently.

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} (14)$$

$$\frac{(D_{xx} + D_{yy})^2}{D_{xx} D_{yy} - (D_{xy})^2} < \frac{(r+1)^2}{r} (15)$$

So if inequality (15) fails, the keypoint is removed from the candidate list.

3) *Key points orientation assignment*

Each key point is assign with consistent orientation based on local image properties so that the descriptor has a character of rotation invariance. This step can be described by two equations below:

$$m(x,y) = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2} (16)$$

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 4, April 2015

$$\Theta(x,y) = \tan^{-1}\left(\frac{L(x,y+1)-L(x,y-1)}{L(x+1,y)-L(x-1,y)}\right) (17)$$

Two above equation are the gradient magnitude and theorientation of pixel (x, y) at its scale L(x, y). In actual calculation, a gradient histogram is formed from the gradient orientations of sample points within a region around the key point. The orientation histogram has 36 bins covering the 360 degree range of orientations, so each 10^0represents a direction, so there are 36 directions in all. Each sample added to the histogram is weighted by its gradient magnitude and by a Gaussian-weighted circular window with σ that is 1.5 times that of the scale of the keypoint. The highest peak in the histogram is detected and then any other local peaks have 80% of the highest peak value is used to create a keypoint with that orientation as the dominant direction of the key point. One keypoint can have more than one direction.

4) Key point descriptor

The repeatable local 2D coordinate system in the previous assigned image with location, scale and orientation are used to describe image region which is invariant with these parameters. In this step, descriptor of keypoint will becomputed. Each keypoint will be described by a 16x16 region around. Then in each of the 4x4 subregion, calculate the histograms with 8 orientation bins. After accumulation, the gradient magnitudes of the 4x4 region to the orientation histograms, we can create a seed point; each seed point is an
8-demensional vector, so a descriptor contains 16x16x8 elements in total.

Now, after corner detection a similarity matrix is generated to find the similar corners of the two images, which isexplained below in section C.

C. Estimation of geometric transformation:

Suppose that the image Ihas a resolution of $W\times H$, and the k^{th} corner in $I$is denoted by (with coordinates$x$and I_k. $y$and intensity(x, y_k)). One corner from the left image (I) and another corner from the right image ($I^r{}_j$) can be matched with each other if the following conditions are satisfied:

(i) The difference between $y$coordinates of these two corners is no greater than $H/3$.

(ii) The $x$coordinate of the left corner is greater than or equal to that of the right corner.

(iii) There is a high correlation between two corners.

Accordingly, we utilize (18) to calculate pair wise corner similarity and create a similarity matrix between adjacent images $I^l$and I^r:

$$Sim(i,j) = \begin{cases} \left|NCC(I_i^l, I_j^r)\right| if \left|I_i^l.y - I_j^r.y\right| < \lambda_h, I_i^l.x \geq I_i^r.x, \\ 0 \qquad\qquad else \end{cases} (18)$$

In (18), $\lambda_h$is the threshold of the difference between $y$coordinates of two corners, and normalized cross-correlation(NCC) function is the one described in [10]. Suppose that thesimilarity window size is $(2w+1) * (2w+1)$;

NCC is thencalculated as

$$NCC(I_i^l, I_i^r) = \frac{\sum_{u=-w}^{w}\sum_{v=-w}^{w} D_l(i,u,v).D_r(j,u,v)}{\sqrt{\sum_{u=-w}^{w}\sum_{v=-w}^{w} D_l(i,u,v)^2.\sum_{u=-w}^{w}\sum_{v=-w}^{w} D_r(j,u,v)^2}} (19)$$

Where,

$$D_l(i,u,v) = I^l(x_i+u, y_i+v) - \overline{I_i^l}$$

$$D_r(j,u,v) = I^r(x_j+u, y_j+v) - \overline{I_j^r} (20)$$

And$\overline{I_i^l}$and $\overline{I_j^r}$are the mean intensity of windows aroundcorners I_i^landI_j^r, respectively. In addition, we further filterout corner pairs with low similarity using (21), where λ_n is thesimilarity threshold.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 4, April 2015

$$Sim(i,j) = \begin{cases} sim(i,j) if sim(i,j) > \lambda_n \\ 0 \quad\quad else \end{cases} \quad (21)$$

Hence, we use equation (18) and (21) to calculate pairwise corner similarity, sim(i, j), resulting in a similarity matrix between two adjacent images.

D. Corner Matching:

After generating similarity matrix maximum number of matching corner pairs is determined. A set of indexes of matching-corner pairs is generated by the following procedure: in each row of the similarity matrix obtained previously, we find the column index so that the corresponding cell in the matrix has the maximum value for that row, and the pair of (row index, column index) is added into the set. After we process all rows in the matrix, we will obtain a set of index pairs.

IV. EXPERIMENTAL RESULTS

In order to perform the comparison of Harris and SIFT corner detection method experimentation has been performed on PC: CPU 2.30Ghz + 2.30Ghz, 4GB memory, Matlab R2013a. The experimentation has been performed on 30 different pairs of images. One sample pair is shown in Fig. 2. From the figure it can be seen the variation in two different images and the complexity of the stitching problem.

Fig. 2: Pair of sample images to be stitched

First, we detect the corners from both pairs of images to be stitched using Harris corner detection method. The detected corners have been further used for matching using similarity matrix. Fig. 3(a)shows the matching corner points of Fig. 2 and Fig. 3(b) shows the stitched image.

(a) (b)

Fig. 3: Matching corner pairs and stitched image using Harris algorithm

Here, we have demonstrated the results of only 8 images.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

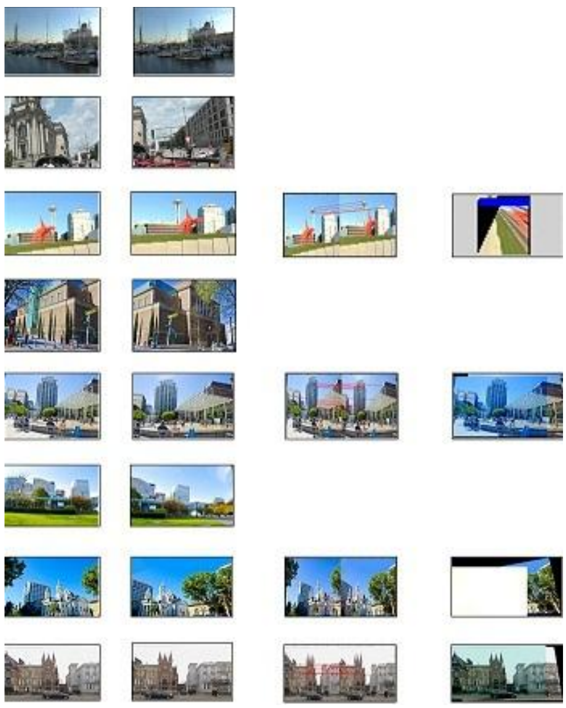
(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 4, April 2015

Table 1: Results of image stitching using Harris algorithm

Sample Pair	Key points (corner)		Matching points	Time in sec.
	Input image1 (I1)	Input image2 (I2)		
1	404	338	2	0
2	560	725	0	0
3	388	380	5	2.29
4	847	751	3	0
5	1063	1020	13	5.27
6	569	465	1	0
7	683	846	6	4.1
8	662	677	10	4.0

The number of corner points also called as key points detected using the Harris method for 8 images are tabulated in Table 1. These detected Harris points or Harris features have been used for matching. For instance, in Table 1 it is seen that for sample pair 1, 404 corner points (Input image1) and 338 corner points (Input image2) have been detected, however only 2 points have matched *i.e.,* only two points have similarity matrix in common. From this it is evident that although number of corner points detected are more in number but not so relevant to be matched for stitching. This affects the stitching process. From the results it has been observed that the efficiency of stitching depends on the number of matching points. Therefore, we have used a threshold (if number of matching points ≥ 5 then stitch). To demonstrate the aforementioned point results are demonstrated in Fig. 4. The first column depicts the first two images for sample pair 1 from Table 1. The matching point for I1 and I2 is 2 and therefore I1 and I2 have not been stitched.



(a) (b) (c) (d)

Fig. 4: Results of image stitching using Harris algorithm

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 4, April 2015

Fig. 4(a) and 4(b) are input images. Fig. 4(c) shows matching corners using Harris algorithm and Fig. 4(d) shows stitched image.

Secondly, the same stitching process was also implemented with SIFT corner method. Fig. 5(a) shows the matching corner points of Fig. 2 and Fig. 5(b) shows the stitched image using SIFT algorithm.

(a) (b)

Fig. 5: Matching corner pairs and stitched image using SIFT algorithm

Table 2: Results of image stitching using SIFT algorithm

Sample Pair	Key points (corner)		Matching points	Time in sec.
	Input image1 (I1)	Input image2 (I2)		
1	404	338	42	1.55
2	560	725	10	1.70
3	388	380	80	1.35
4	847	751	46	1.77
5	1063	1020	240	2.19
6	569	465	50	1.75
7	683	846	99	1.71
8	662	677	157	1.74

The number of key points detected using the SIFT method for 8 images are tabulated in Table 2. These detected SIFT features have been used for matching. For instance, in Table 2 it is seen that for sample pair 1, 404 corner points (I1) and 338 corner points (I2) have been detected, and 42 points have matched *i.e.*, 42 points have similarity matrix in common. Hence, these 42 points will be overlapped to get a stitched image. To demonstrate the aforementioned point results are demonstrated in Fig. 6. The first column depicts the first two images for sample pair 1 from Table 2. The matching point for I1 and I2 is 42 and therefore I1 and I2 have been stitched.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 4, April 2015

(a) (b) (c) (d)

Fig. 6: Results of image stitching using SIFT algorithm

Fig. 6(a) and 6(b) are input images. Fig. 6(c) shows matching corners using SIFT algorithm and Fig. 6(d) shows stitched image.

V. CONCLUSION

In this research work we have performed image stitching using two corner detection method namely Harris corner detector and SIFT descriptor. We have discussed the algorithms of these two methods.

From the results, we can see that the matching points of Harris algorithm is less than that of SIFT algorithm and also the image stitching done using SIFT algorithm is better than Harris algorithm. SIFT algorithm is more robust than Harris algorithm. Also the correspondence point from Harris features can be obtained with high time consuming and is very difficult to get high correct point. Whereas from SIFT features we can get high correctness and robustness correspondence points.

Based on extracting invariant scale features, we get potential feature matches for SIFT algorithm than that for Harris algorithm. SIFT can give better performance and when there are less rotations Harris corner detection algorithms can perform better.

REFERENCES

[1] Minchen Zhu, Weizhi Wang, Binghan Liu, and Jingshan Huang, "A Fast Image Stitching Algorithm via Multiple-Constraint Corner Matching", *Hindawi Publishing CorporationMathematical Problems in Engineering*, vol. 2013, pp. 1-6, sep 2013.
[2] RussolAbdelfatah, Dr.Haitham Omer, "Automatic Seamless of Image Stitching", *Computer Engineering and Intelligent Systems*, vol. 4, no.11, pp. 7-13, 2013.
[3] Ze-lang Wang, Fang-hua Yan, Ya-yuZheng, "An Adaptive Uniform Distribution SURF for Image Stitching", *IEEE 6th International conference on Image and Signal Processing (CISP)*, vol. 2, pp. 888-892, 2013.
[4] R.Karthik, A.AnnisFathima,V.Vaidehi, "Panoramic View Creation using Invariant Momentsand SURF Features",*IEEE International Conference on Recent Trends in Information Technology (ICRTIT)*, pp. 376-382, july2013.
[5] Chen-Hong Yuan, Jeng-Shyang Pan, Ming-HwaSheuand Tzu-Hsiung Chen, "Fast Image Blending and Deghosting for Panoramic Video", *IEEE 9th International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pp. 104-107, oct2013.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 4, April 2015

[6] Deepak Kumar Jain, GauravSexena,Vineet Kumar Singh, "Image mosaicing using corner technique", *IEEE International Conference on Communication Systems and Network Technologies*, pp. 79-84,may 2012.

[7] Zhen Hua, Yewei Li, Jinjiang Li, "Image Stitch Algorithm Based on SIFT and MVSC", *IEEE 7th International Conference on Fuzzy Systems and Knowledge Discovery*,vol. 6, pp. 2628-2632, aug 2010.

[8] Xianyong Fang, "Feature Based Stitching of a Clear/Blurred Image Pair",*IEEE International Conference on Multimedia and Signal Processing*, vol. 1, pp. 146-150, 2011.

[9] ChamindaNamalSenarathne, ShanakaRansiri, PushpikaArangala, AsankaBalasooriya, Dr. Chathura De Silva, "A Faster Image Registration and StitchingAlgorithm", *IEEE 6thInternational Conference on Industrial and Information Systems*, pp. 66-69, aug2011.

[10] Hyo-Kak Kim, Kwang-Wook Lee, June-Young Jung, Seung-Won Jung, and Sung-JeaKo, "A Content-Aware Image Stitching Algorithm for Mobile Multimedia Devices", *IEEE Transactions on Consumer Electronics*, vol. 57, pp. 1875-1882, no. 4,nov2011.

[11] Tao-Cheng Chang, Cheng-An Chien, Jia-Hou Chang, and Jiun-In Guo, "A Low-Complexity Image Stitching Algorithm Suitable for Embedded Systems", *IEEE International Conference on Consumer Electronics (ICCE)*,pp. 197-198, jan2011.

[12] Hyung Il Koo and Nam Ik Cho, "Feature-based Image Registration Algorithm for Image Stitching Applications on Mobile Devices", *IEEE Transactions on Consumer Electronics*, vol. 57, no. 3, pp. 1303-1310, aug2011.

[13] YingenXiong and Kari Pulli, "Fast Panorama Stitching for High-Quality Panoramic Images on Mobile Phones", *IEEE Transactions on Consumer Electronics*, vol. 56, no. 2, pp. 298-306, may 2010.

[14] Jubiao Li and Junping Du, "Study on Panoramic Image Stitching Algorithm", *IEEE 2ndPacific-Asia Conference on Circuits, Communications and System (PACCS)*, vol. 1, pp. 417-142, 2010

[15] TangfengXu, Delie Ming1, Liping Xiao, Chengkai Li, "Stitching Algorithm of Sequence Image Based on Modified KLT Tracker", *IEEE 5thInternational Symposium on Computational Intelligence and Design*, vol. 2, pp. 46-49, oct2012

[16] Xian-Guo Li, Chang-Yun Miao and Yan Zhang, "An Algorithm for Selecting and Stitching the Conveyer Belt Joint Images Based on X-ray", *IEEE International Conference on Intelligent Computation Technology and Automation*,vol. 1, pp. 474-477, may 2010.

[17] Oh-Seol Kwon and Yeong-Ho Ha, "Panoramic Video using Scale-Invariant Feature Transform with Embedded Color-Invariant Values", *IEEE Transactions on Consumer Electronics*, vol. 56, pp. 792-798, no. 2, may 2010

[18] Y. Zhang, G. Gao, and K. Jia, "A fast algorithm for cylindrical panoramic image based on feature points matching," *Journal ofImage and Graphics*, vol. 14, no. 6, pp. 1188–1193, 2009.

[19] W. Zhao, S. Gong, C. Liu, and X. Shen, "A self-adaptive Harris corner detection algorithm," *Computer Engineering*, vol. 34, no. 10, pp. 212–214, 2008.

[20] JiayaJia and Chi-Keung Tang, "Image Stitching Using Structure Deformation", *IEEE Transactions On Pattern Analysis And Machine Intelligence*, vol. 30, pp. 617-631, no. 4, april 2008.