

A Novel Blending Method for Automatic Seamless Image Mosaicing

Ahmet Hamdi VAROL, Emre SÜMER

Abstract— An image mosaic is a panoramic image generated by assembling sequential images belong to the same scene. This method is mainly used in mapping applications, the evaluation of the images acquired from unmanned aerial vehicles, and various computer vision applications. In this study, we proposed a novel blending method for automatic seamless image mosaicing from ground level photos taken by an digital camera. For the correction of consistent seams occurred during the mosaicing, we measured the color differences around the overlapping regions. For this purpose, first, we determined the intensity values of image pixels inside and outside the overlapping region boundary within a 1 pixel wide zone. Secondly, the intensity values were averaged and finally, the average values of color differences were applied to overlapping regions. Besides, a motion blur filter was performed for the same boundary regions in order to reduce the seams. In order to test the accuracy, a subjective evaluation metric was used. According to the results obtained, the seamless image mosaics were generated with a subjective accuracy of 74%. Our results indicate that the seams can be substantially reduced or completely eliminated, and the illumination differences are minimized.

Keywords— seamless texture, image mosaicing, motion blur, color intensity

I. Introduction

An image mosaic is a panoramic image generated by assembling sequential images belong to the same scene and it can be obtained by understanding geometric relationships between images. The geometric relations are coordinate transformations that relate the different image coordinate systems. By applying the appropriate transformations, - like affine, perspective and polynomial transformations -, via a warping operation and merging the overlapping regions of warped images, it is possible to construct a single image indistinguishable from a single large image of the same object, covering the entire visible area of the scene. In image mosaicing two input images are taken and these images are

merged to form a single large image. This merged single image is the output mosaiced scene [7].

Image mosaicing process consists of five steps. The first step is image extraction. In this step, image features are detected for each input images. In this task, Harris Algorithm is typically used.

The second step, image registration, is the task of matching two or more images. Registration methods can be divided into the following classes: (i) algorithms that use image pixel values directly, (ii) algorithms that use the frequency domain, (iii) algorithms that use low level features such as edges and corners and (iv) algorithms that use high-level features. The third step is the computation of homography using RANSAC algorithm. In this step, outlier points are removed. The last step is image warping and blending. Image warping is the process of digitally manipulating an image such that any shapes portrayed in the image have been significantly destroyed. Image blending blends the color pixels and aims to resolve contrast differences in the overlapped region in order to avoid seams [7].

In this study, we proposed a novel blending method for automatic seamless image mosaicing. For this purpose, firstly, we determined the intensity values of image pixels inside and outside the overlapping region boundary within a 1 pixel wide zone. Secondly, the intensity values were averaged and finally, the average values of color differences were applied to overlapping regions. Besides, a motion blur filter was performed for the same boundary regions in order to reduce the seams.

II. Feature Extraction

Feature extraction is the first step in image mosaicing. Once features have been detected, a local image patch around the feature can be extracted. This extraction may involve quite considerable amounts of image processing task. Transforming the input data into the set of features is called feature extraction. If the extracted features are carefully chosen then it is expected that the feature set will extract the relevant information from the input data in order to get mosaic image i.e. output [4].

There are feature elements like edges, corners, blobs, ridges in input images. Corners are good features used in several matching applications. The features of corners are that they are more stable features over changes of viewpoint. The other important property of corner is that if there is a corner in an image than its neighborhood will indicate an abrupt change in intensity. Corners are detected in images by applying corner detection algorithms. Some of the well known corner detection algorithms are Harris Corner detection Algorithm, SIFT (Scale

Ahmet Hamdi VAROL
Baskent University
TURKEY

Emre SÜMER
Baskent University
TURKEY

Invariant Feature Transform) corner detection algorithm, the machine learning based FAST (Features from Accelerated Segment Test) algorithm, SURF (Speeded-up robust feature) [5].

A. **Harris Algorithm**

Harris corner detection is a point feature extracting algorithm based on Moravec algorithm based by C. Harris and M.J Stephens in 1988. A local detecting window inside the image is designed. The average variation in intensity that results by shifting the window by a small amount in different direction is determined. At this point the centre point of the window is extracted as corner point. Shifting the window in any direction gives a large change in appearance. Harris corner detector is used for corner detection. 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 there will be 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 direction. Harris corner detector gives a mathematical approach for determining whether the region is flat, edge or corner. Harris corner technique detects more features and it is rotational invariant and scale variant. The result of feature matching using Harris corner detector is shown in Figure 1. For the change of intensity for the shift [u, v] [5].

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

where $w(x, y)$ is a window function, $I(x + u, y + v)$ is the shifted intensity and $I(x, y)$ is the intensity of the individual pixel. Harris corner algorithm is given below as:

1. For each pixel (x, y) in the image calculate the autocorrelation matrix M as;

$$M = \sum_{x, y} \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \quad (2)$$

2. For each pixel of image has Gaussian filtering, get new matrix M, and discrete two-dimensional zero-mean Gaussian function as:

$$Gauss = \exp(-u^2 + v^2) / 2\delta^2 \quad (3)$$

3. Calculating the corners measure for each pixel (x, y), we get:

$$R = \{I_x^2 x I_y^2 - (I_x I_y)^2\} - k \{I_x^2 + I_y^2\}^2 \quad (4)$$

4. Choose the local maximum point. Harris method considers that the feature points are the pixel value which corresponding with the local maximum interest point.

5. Set the threshold T, detect corner points.

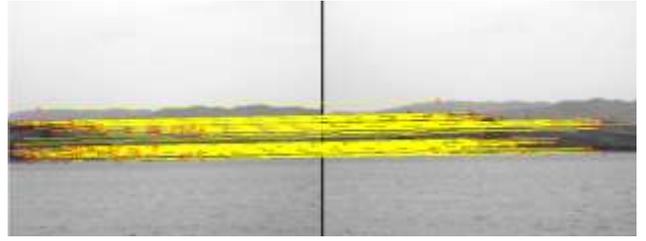


Figure 1. Features matching using Harris algorithm

B. **SIFT Algorithm**

SIFT Algorithm is Scale Invariant Feature Transform. SIFT is a corner detection algorithm which detects features in an image [5]. SIFT extracts from an image a set of descriptors. Each one of the extracted descriptors is invariant to an image translation, rotation and zoom-out. SIFT descriptors have also proved to be robust to a wide family of image transformations, such as slight changes of viewpoint, noise, blur, contrast changes, scene deformation, while remaining discriminative enough for matching purposes [6]. It can be also used to identify similar objects in other images. SIFT produces key-point-descriptors which are the image features [5]. Summary of the SIFT algorithm [6]:

1. Compute the Gaussian scale-space.
2. Compute the Difference of Gaussians (DoG).
3. Find candidate keypoints (3d discrete extrema of DoG)
4. Refine candidate keypoints location with sub-pixel precision.
5. Filter unstable keypoints due to noise.
6. Filter unstable keypoints laying on edges.
7. Assign a reference orientation to each keypoint.
8. Build the keypoints descriptor.

For a set of input frames SIFT extracts features. Image matching is done using Best Bin First (BBF) algorithm for estimating initial matching points between input frames. The false matches are removed in the image pair using RANSAC algorithm. Reprojection of frames are done by defining its size, length and width. Finally, stitching is performed to obtain a final output mosaic image. In stitching, each pixel in every frame of the scene is checked whether it belongs to the warped second frame. If so, then that pixel is assigned the value of the corresponding pixel from the first frame. It is a robust algorithm for image comparison but it is slow due to its running time [5].

C. **FAST Algorithm**

FAST is a corner detector algorithm founded by Trajkovic and Hedley in 1998. The detection of corner was prioritized

over edges in FAST as corners were found to be the good features to be matched because it shows a two dimensional intensity change, and thus well distinguished from the neighboring points. According to Trajkovic and Hedley the corner detector should satisfy the following criteria.

1. The detected positions should be consistent, insensitive to the variation of noise, and they should not move when multiple images are acquired of the same scene.
2. Accuracy; Corners should be detected as close as possible to the correct positions.
3. Speed; The corner detector should be fast enough.

FAST incremented the computational speed required in the detection of corners. This corner detector uses a corner response function (CRF) that gives a numerical value for the corner strength based on the image intensities in the local neighborhood. CRF was computed over the image and corners which were treated as local maxima of the CRF. A multi-grid technique is used to improve the computational speed of the algorithm and also for the suppression of false corners being detected. FAST is an accurate and fast algorithm that yields good localization (positional accuracy) and high point reliability [5].

D. SURF Algorithm

The Speed-up Robust Feature detector (SURF) uses three feature detection steps namely; Detection, Description and Matching. SURF speeded-up the SHIFT's detection process by keeping in view of the quality of the detected points. The Hessian matrix is used along with descriptors low dimensionality to significantly increase the matching speed. SURF is widely used in the computer vision community. It has proven its efficiency and robustness in the invariant feature localization [5].

III. Image Registration

Image registration refers to the geometric alignment of a set of images. The different sets of data may consist of two or more digital images taken of a single scene from different sensors at different time or from different viewpoints. In image registration the geometric correspondence between the images is established so that they may be transformed, compared and analyzed in a common reference frame [5]. It has been a central issue for a variety of problems in image processing such as object recognition, monitoring satellite images, matching stereo images for reconstructing depth, matching biomedical images for diagnosis. Image registration is the important step in image mosaicing [7]. Registration methods can be loosely divided into the following classes: (i) algorithms that use image pixel values directly, e.g., correlation methods; (ii) algorithms that use the frequency domain, e.g., Fast Fourier transform based (FFT-based) methods; (iii) algorithms that use low level features such as edges and corners, e.g., Feature based methods; and (iv) algorithms that use high-level features such as identified parts of image objects, relations between image features, for e.g., Graph-theoretic methods [5].

IV. Computing Homography

A. Homography

Homography is mapping between two spaces which often used to represent the correspondence between two images of the same scene. It is widely used for images where multiple images are taken from a rotating camera having a fixed camera centre ultimately warped together to produce a panoramic view [2].

A 2D point $(x; y)$ in an image can be represented as a 3D vector $x = (x_1; x_2; x_3)$ where $x = X_1/X_3$ and $y = X_2/X_3$. This is called the homogeneous representation of a point and it lies on the projective plane P^2 . A homography is an invertible mapping of points and lines on the projective plane P^2 . Other terms used for this transformation includes collineation, projectivity, and planar projective transformation. Hartley and Zisserman provide the specific definition that a homography is an invertible mapping from P^2 to itself such that three points lie on the same line if and only if their mapped points are also collinear. They also give an algebraic definition by proving the following theorem: A mapping from $P^1 \rightarrow P^2$ is a projectivity if and only if there exists a non-singular 3×3 - matrix H such that for any point in P^2 represented by vector x it is true that its mapped point equals to Hx . This tells us that in order to calculate the homography that maps each x_i to its corresponding x_i' . It is sufficient to calculate the 3×3 homography matrix, H . All of the homography estimation algorithms that are discussed require a set of correspondences as input. So far these algorithms are only robust with respect to noise if the source of this noise is in the measurement of the correspondence feature positions. There will be other situations where the input will be corrupted with completely false correspondences, meaning that the two features in the images don't correspond to the same real world feature at all. There is a need to discuss ways to distinguish inlier and outlier correspondences so that the homography can be estimated robustly using only inlier matches [8].

B. Ransac Algorithm

The RANdom SAMple Consensus (RANSAC) algorithm proposed by Fischler and Bolles is a general parameter estimation approach designed to cope with a large proportion of outliers in the input data. Unlike many of the common robust estimation techniques such as M-estimators and least-median squares that have been adopted by the computer vision community from the statistics literature, RANSAC was developed from within the computer vision community [3].

RANSAC is a resampling technique that generates candidate solutions by using the minimum number observations (data points) required to estimate the underlying model parameters. As pointed out by Fischler and Bolles, unlike conventional sampling techniques that use as much of the data as possible to obtain an initial solution and then proceed to prune outliers, RANSAC uses the smallest set possible and proceeds to enlarge this set with consistent data points [3].

The algorithm is summarized as follows:

1. Select randomly the minimum number of points required to determine the model parameters.
2. Solve for the parameters of the model.
3. Determine how many points from the set of all points fit with a predefined tolerance ϵ .
4. If the fraction of the number of inliers over the total number points in the set exceeds a predefined threshold T , re-estimate the model parameters using all the identified inliers and terminate.
5. Otherwise, repeat steps 1 through 4 (maximum of N times).

The number of iterations, N , is chosen high enough to ensure that the probability p (usually set to 0.99) that at least one of the sets of random samples does not include an outlier. Let u represent the probability that any selected data point is an inlier and $v = 1 - u$ the probability of observing an outlier. N iterations of the minimum number of points denoted m are required, where

$$1 - p = (1 - u^m)^N \quad (5)$$

and thus with some manipulation,

$$N = \frac{\log(1 - p)}{\log(1 - (1 - v)^m)} \quad (6)$$

v. Warping and Blending

A. Image Warping

Image Warping is the process of digitally manipulating an image such that any shapes portrayed in the image have been significantly distorted. Warping may be used for correcting image distortion as well as for creative purposes (e.g., morphing). While an image can be transformed in various ways, pure warping means that points are mapped to points without changing the colors. This can be based mathematically on any function from part of the plane to the plane. If the function is injective the original can be reconstructed. If the function is a bijection any image can be inversely transformed. The last step is to warp and blend all the input images to an output composite mosaic. First we need to make out the output mosaic size by computing the range of warped image coordinates for each input image. As described earlier we can easily do this by mapping four corners of each source image forward and computing the minimum x and y , maximum x and y coordinates to determine the size of the output image. Finally x -offset and y -offset values specifying the offset of the reference image origin relative to the output panorama needs to be calculated. The next step is to use the inverse warping as

described above for mapping the pixels from each input image to the plane defined by the reference image, is there to perform the forward and inverse warping of points, respectively [8].

B. Image Blending

The final step is to blend the colour pixels in the overlapped region to avoid the seams. Simplest available form is to use feathering, which uses weighted averaging colour values to blend the overlapping pixels. An alpha factor is generally used, which is often called alpha channel. It takes the value of 1 at the center pixel and becomes 0 after decreasing linearly to the border pixels. In case at least two images overlap in an output mosaic we will use the alpha values to compute the colour at a pixel in there [8]. An example image mosaicing result is illustrated in Figure 2.



Figure 2. Final mosaiced image

In this study, we proposed a novel blending method for automatic seamless image mosaicing. For this purpose, firstly, we found the border of overlapping area and then determined the corner points of that border. Then, we constructed a closed frame using that corner points. To apply motion blur filtering and to determine the intensity values of image pixels inside and outside the overlapping region, we need the points, which are on the line segment. Besides, the angle of that line segment is also computed. Secondly, the intensity values were averaged and finally, the average values of color differences were applied to overlapping regions. Besides, a motion blur filtering was performed for the same boundaries in order to reduce the seams. This filtering is capable of linear, radial, and zoom movements. The blurring size and direction can be altered by adjusting the length and angle parameters. In this study, the linear type of filter was used so that the blurring occurs in a single direction either horizontally or vertically. By applying this filter, for each point, the blur intensity was kept as 3, and for the motion blurring angle, we took angle which is found in the last step and add 90° . In Figure 3, the overlapping region boundary and image pixels inside and outside the overlapping region boundary within a 1 pixel wide zone are presented.



Figure 3. Overlapping region boundary and a 1 pixel wide zone



Figure 4. An example of successful blending (image set#1)

VI. Results

In order to evaluate the accuracy, a web page has been prepared for subjective assessment (www.baskent.edu.tr/~avarol/tez). In the web page, ten image pair were presented to be evaluated by the academic staff of Baskent University. The first image was created by merging two images without applying any blending process, while, the second was obtained after removing seams in overlapping areas. We used a rating scale for assessment, which is adapted from [1], shown in Table I. The experimental test results were also presented in Table II. According to the results, the average ratings were found to be between in the range 1.6 and 4.2. The most successful image pair was found to be image set#1 while the least successful pair was determined to be image set#9. As a result, the overall success rate was found to be 74%. The resulting images of set#1 and set#9 are illustrated in Figure 4 and 5, respectively.

TABLE I. RATING SCALE FOR THE EVALUATION OF SEAMLESS MOSAICING

Value	Rating	Description
1	Excellent	The success of seamless mosaicing is extremely in high quality
2	Fine	The success of seamless mosaicking is in high quality
3	Passable	The success of seamless mosaicking is in acceptable quality
4	Marginal	The success of seamless mosaicking is in poor quality
5	Unusable	The success of seamless mosaicking is so bad

TABLE II. THE RESULT OF SUBJECTIVE ASSESMENT FOR EACH IMAGE SET

Avg. Rating	IMAGE SET #									
	1	2	3	4	5	6	7	8	9	10
	1.6	1.7	2.9	1.9	3	1.7	1.8	1.9	4.2	3.2



Figure 5. An example of unsuccessful blending (image set#9)

VII. Conclusion

In this paper, we described how to construct image mosaics, its well known methods and algorithms. Besides, we suggested a method to remove the seams after a mosaicing process. The proposed intensity based approach was found to be quite promising since 74% success rate was reached for a test data set, which is composed of ten image pairs. Moreover, motion blurring was found to be successful in elimination of the seams. Further, we adapted a well-known subjective assessment technique in order to evaluate the blending accuracy. Consequently, we believe that the proposed blending method enhanced the mosaicing results aesthetically by removing the sharp transitions and reducing the illumination differences remarkably.

References

- [1] Rafael C. Gozalez, Richard E. Woods, Digital Image Processing , 3rd ed., pp. 557-558
- [2] Dushyant Vaghela, Prof. Kapildev Naina, "A Review of Image Mosaicing Techniques"
- [3] Konstantinos G. Derpanis, "Overview of the RANSAC Algorithm", May 2010

- [4] A.Annis Fathima, R.Karthik,V.Vaidehi, “Image Stitching With Combined Moment Invariants and Sift Features”, The 4th International Conference on Ambient Systems, Networks and Technologies (ANT 2013)
- [5] Hemlata Joshi, Mr. KhomLal Sinha, “A Survey on Image Mosaicing Techniques ”, International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 2, Issue 2, February 2013
- [6] Ives Rey-Otero, Mauricio Delbracio, “Anatomy of SIFT Method”, IPOL Image Processing On Line, <http://dx.doi.org/10.5201/ipol>, PREPRINT March 11, 2014
- [7] Prof. Mayur Dhait, Miss. Rashmi S. Ghavghave, “Image Mosaicing Using Feature Detection Algorithm”, International Journal of Informative & Futuristic Research, Vol-1 Issue -8, April 2014
- [8] Stafford michahial, Latha M, Akshatha S, Juslin F, Ms Manasa B, Shivani U, “Automatic Image Mosaicing Using Sift, Ransac and Homography”, International Journal of Engineering and Innovative Technology (IJETT), Vol 3, Issue 10, April 2014

About Author (s):

Ahmet Hamdi VAROL

Ahmet Hamdi Varol is a graduate student in the Department of Computer Engineering, Baskent University, Ankara, Turkey. His primary research interests are computer vision and image processing.



Dr. Emre SÜMER

Dr. Emre Sümer received his Ph.D. degree in Geodetic and Geographic Information Technologies from Middle East Technical University, Ankara, Turkey in 2011. He is an assistant professor in the Department of Computer Engineering, Baskent University, Ankara, Turkey. His primary research interests are computer vision, image processing and pattern recognition.

