2020 International Conference on Electrical Engineering (ICEE)
September 25-27, 2020, Istanbul, Turkey

Novel LabVIEW Implementation of FREAK Based Image Mosaicing and Object Tracking Algorithms

LATI Abdelhai [a] , MEKHERMECHE Haitham [a], BENHELLAL Belkheir [a] , BELHOCINE Mahmoud [b] and ACHOUR Nouara [c]

[a] Université de Kasdi Merbah Ouargla (UKMO), BP 511, 30000, Ouargla, Algérie.
[b] Centre du Développement des Technologies Avancées CDTA.
Cité 20 août 1956, Baba Hassen 16303 Alger, Algérie.
[c] Université de Sciences et Technologie de Houari Boumedian USTHB.
Laboratoire de Robotique, Parallélisme et Systèmes Embarqués LRPSE.
BP 32, El Alia, Bab Ezzouar, 16111 Alger, Algérie.

Mails : abdelhai.lati@gmail.com, mkhhaitham@gmail.com, bbenhellal@gmail.com , mbelhocine@cdta.dz , nachour@usthb.dz

***Abstract*—in this paper, we propose implementations of two important vision algorithms on LabVIEW tool; which are image mosaicing and object tracking. Features are used as a starting point for our algorithms; so the overall algorithms will only be as good as their used feature detectors and features matching techniques, for that, we propose using Harris corner detector to find key points in overlapping images, then describing the detected features using FREAK descriptors for finding correspondences, after estimating homography parameters; either the second image is transformed to the frame of the first image for image mosaicing purpose, or four corners of the tracked object are transformed for object tracking purpose. The two algorithms are tested on Matlab, and then they are implemented on LabVIEW platform; in which a sophisticated interface is created to facilitate code execution for users, the built interface on LabVIEW can be used to be executed on embedded systems. The performance of our implementation is verified on real images from different data base.**

Keywords- Image mosaicing; Object tracking; Features matching; FREAK descriptors; Matlab; LabVIEW.

I. Introduction

Visual information is the most powerful type of data that may be seen by eyes and analyzed by the human brain. Since few years, researchers have tried to understand this phenomenon; so they thought to invent a camera that takes place of the eye for capturing different views of a scene **[1]**. However, the problem of reproducing the real world from images is not easy and it requires performing hard calculations and implementing complex algorithms on images **[2]**. Scientists have been investigating various techniques for acquiring three dimensions (3D) information from objects and scenes for many years; this acquired data is useful in different tasks.

In most of image processing algorithms, features detection techniques are essentially used to compute abstractions of image information and making local decisions at every image part whether there is an image feature of a given type at that part or not **[3]**. Type of features depends on the treated problem or the type of desired application. Features are used as a key stage for many computer vision algorithms such as, image registration; which is the process of aligning two or more images, so that objects representing the same structures are eroded. Image mosaicing and object tracking have benefited from features based image registration to refine certain objectives. An image mosaic is a composition generated from a sequence of overlapped views of a scene, **[4]** and object tracking algorithm is designed to locate and keep a steady watch on a moving object or many moving objects over time in a video stream or image sequences.

Constructing image mosaics is an active area of research in the field of computer vision, and image processing including; scene stabilization, change detection, video compression, increasing the field of view and resolution of a camera **[5].** Object tracking is an important component of many computer vision systems, and it is widely used in video surveillance, robotics, 3D image reconstruction, medical imaging, and human computer interface.

In our work, the two above applications were performed by understanding geometric relationships between images; these relations are the coordinate systems that relate the different image coordinate systems, and the performance of the mentioned algorithms depends mainly on features detection and matching stages, because robust features detector and robust method for features matching ensure robust geometric image registrations. Registering images is based mainly on extracting the overlapping region between them, for large overlapping; correlation measures can be used to find coordinates of that common region, but for small overlapping; features description needs to be used. In our work; we are going to utilize binary descriptor based image registration for implementing our algorithms on LabVIEW platform.

Our work is organized in four sections as follows; in section 1, we start with a state of the art about different works done in the field of features based image mosaicing and object tracking. The second section presents the entire scheme of the mosaic of images and object tracking. In the third section, we illustrate the design of our algorithms, and describe the different steps and techniques used for their implementation on LabVIEW platform, in section 4, evaluation of the results will be discussed, and finally; we end up with conclusions and perspectives for futures works.

978-1-7281-8292-6/20/$31.00 ©2020 IEEE

II. Related Works

A. Image Mosaicing

In **[6]**, authors proposed an automated image mosaicing algorithm for building panoramic images from video sequences. Kanade Lucas Tomasi tracker (KLT) **[7]** was used to match the detected Harris corners or SIFT features **[8]**, the moving objects in the scene caused some wrong matches, therefore; RANdom Sample Consensus (RANSAC) method **[9]** was used for removing them. The set of correct matches were used to estimate the homography matrix, by which overlapped views were aligned. The proposed method in **[10]** applied the corner detector to extract key points from partially overlapping views. Then, Normalized Cross Correlation (NCC) was used to find correspondences between images. RANSAC technique was used to get the good matches, in order to register images to get finally seamless image mosaic. The method described in **[11]** was proposed for building mosaics from an underwater video sequence. Difference of Gaussian (DoG) technique was used for feature detection, then; binary descriptors were used to describe the key points. The created descriptors were matched using Nearest Neighbor Distance Ratio (NNDR) for homography estimation and image warping.

From the discussed previous works, it is clear that most of the proposed image mosaicing algorithms suffer from appearance of outliers after features matching, therefore; we have proposed to use efficient matching method that eliminates most of the outliers, this method is based on using FREAK descriptors to describe the detected features, then applying XOR operation to find the correct correspondences.

B. Object Tracking

The work in **[12]** was designed to follow a moving object by a mobile robot, authors used mobile camera to capture image of the colored moving object and processed those images to get coordinate of both the moving object and robot. Global features of the moving object such as color and shape of the target were used to track the moving object. In **[13],** mean shift algorithm was used for object tracking, in this algorithm current location was searched based on the histogram of the object in the previous image frame and result of mean shift was used to find the peak of probability density function near the object old position. Implementation of the algorithm on LabVIEW for different moving objects showed that it is suitable for real data analysis. Another work **[14]** used color features for implementing stereo vision tracking system using LabVIEW, the algorithm was based on detecting the tracked object in both web cameras images and maintaining the middle point between the cameras in the center of the image and at the same time; images were used to estimate the distance between the object and the stereo vision system, the advantage of the proposed algorithm is that other types of global features as shape or size can be used for object detection. The proposed system in **[15]** is generally applied for face detection and tracking process, the used webcam detected the color which was fed by the Region of Interest (ROI) coordinates to be tracked in the program, in that color tracking system, the camera was taking the color value and the setup of the camera was working as desired.

By looking to the above discussed literatures, we can easily recognize that most of developed object tracking algorithms are based on global features as size, color and shape, and these algorithms need processing all image for tracking purpose, therefore; we have proposed a robust tracking algorithm based on local features rather than global features.

III. Features Based Image Mosaicing and Object Tracking Algorithms

The flowcharts in *Figure 1* illustrate our proposed features based image mosaicing and object tracking algorithms, in which for each flowchart; the essential steps of the applied algorithm are given as follows:

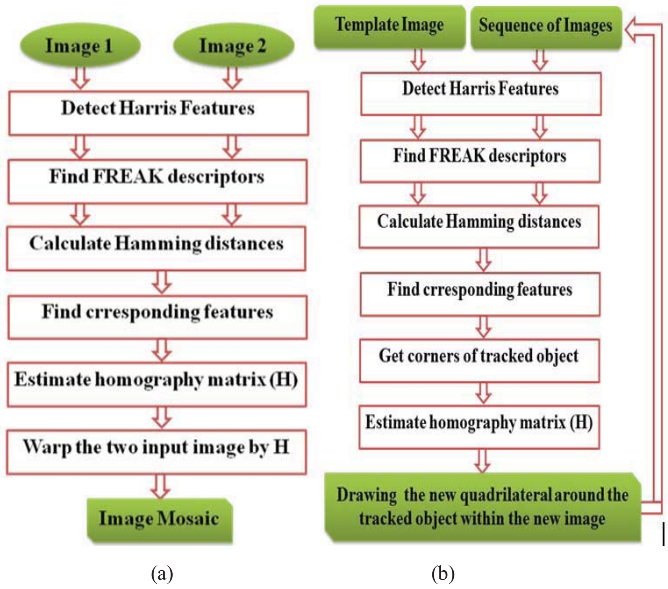

Figure 1: a) Flowchart for the proposed image mosaicing algorithm. b) Flowchart for the proposed object tracking algorithm.

A. Harris Based Features Detection

Harris corner detector is based on finding great changes in intensity when shifting a created window around candidate pixel in all directions. The algorithm of Harris corners detection is illustrated in the following steps **[16]**:

- *Computing derivatives Ix and Iy:*

Image derivatives can be computed by using small convolution filters of size 3x3, such as Sobel operators **[17]**

- *Constructing autocorrelation matrix M:*

By creating a window W around all pixels of the original image I, we compute M by equation 2:

$$M = \begin{pmatrix} A & C \\ C & B \end{pmatrix} \tag{1}$$

Where:

$$A = (\sum_{i \in W} \frac{\partial I_i}{\partial x})^2 \;,\; B = (\sum_{i \in W} \frac{\partial I_i}{\partial y})^2 \;\;,\; C = (\sum_{i \in W} \frac{\partial I_i}{\partial x} \frac{\partial I_i}{\partial y})$$

978-1-7281-8292-6/20/$31.00 ©2020 IEEE

- *Computing cornerness measure MSc*

To find out the interest points, Harris characterized the corner response as given by equation 3:

$$MSc(x,y) = \det(M) - k\,(trace(M))^2 \qquad (2)$$

Where:

k is a constant (usually 0.04)

$\det(M) = \lambda_1 \lambda_2 = AB - C^2$

$trace(M) = \lambda_1 + \lambda_2 = A + B$

- *Constructing threshold cornerness map*

For all image pixels, if MSc(x,y) is less than predefined threshold value, then MSc(x,y) will be zeroed.

- *No-maximal suppression*

We obtain the local maxima value, and remove noised pseudo corner points, if MSc(x,y) is less than MSc(i,j) for all (i,j) within the window centered at pixel of a key point (x,y), then MSc(x,y) is zeroed .

Finally, all pixels with a cornerness measure MSC(x,y) > 0 are considered as Harris corners.

B. FREAK Based Features Matching

Fast REtinA Key point or FREAK **[18]** descriptor is a modern binary descriptor computed based on the results of brightness comparison tests in a number of sampling locations around a key point. The algorithm behind FREAK descriptors is stated bellow:

- *Retinal sampling points*

These sample points are very important; because they form the basis for calculating the FREAK descriptor; the N sample points located around the given detected key point are smoothed with a Gaussian kernel. Here, the size of the kernel is varied with respect to the location of the sampling point to simulate the behavior of the human retina, the smoothed areas around the sampling points are referred to as receptive fields **[19]** ; and the centers of the receptive fields represent the sampling points of the FREAK descriptor.

- *Construction of descriptor*

FREAK descriptor is based on intensity comparisons between different pairs of smoothed sampling points **[20]**. By considering a pair of sampling points $P_a = (P_i, P_j)$, where i,j ∈{1, 2,..., N} and i≠ j ,the algorithm defines a binary encoded intensity comparison *s(P_a)* on this pair as follows:

$$s(P_a) = \begin{cases} 1 & if \quad P_i \rangle P_j \\ 0 & otherwise \end{cases} \qquad (3)$$

The above comparison creates the basis for building the FREAK descriptor F containing N bit string:

$$F = \sum_{0 \le a < N} 2^a s(P_a) \qquad (4)$$

- *Orientation of descriptor*

FREAK specifies a 45 sampling points for orientation, as shown in *Figure 2* **[21]**,

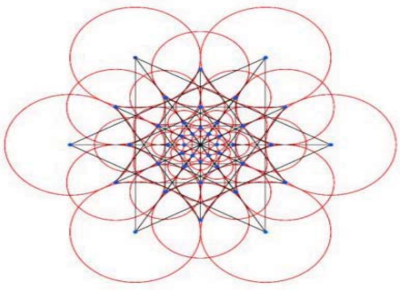

Figure 2. FREAK orientation pattern [21].

Where the blue circles are sampling points, red circles are smoothed receptive fields, and lines represent key point pairs. This pattern is symmetric and is used to generate the local gradient. The orientation *O* of a given key point can be computed using the following equation:

$$O = \frac{1}{M} \sum_{\substack{P_i, P_j \in G \\ i \ne j}} (P_i, P_j) \frac{T(P_i) - T(P_j)}{\left\| T(P_i) - T(P_j) \right\|} \qquad (5)$$

Where

G is the set of all the pairs used to compute the local gradients and ***M*** is the number of pairs in G.

T(P_i) denotes a function returning the 2D vector of the spatial coordinates of the center of receptive field, i.e. the vector of coordinates of the k^{-th} sampling point ***T(P_i)=(x_i,y_i).***

- *Matching of descriptors*

Matching of FREAK descriptors is performed using the hamming distance as a similarity metric, and can be computed efficiently by bitwise XOR followed by bit count **[22]**. Implementation of FREAK feature extraction and matching will produce a list of matched pairs of points. Having two FREAK descriptors F_1 and F_2 of length N, the measured hamming distance between them is written as:

$$D_H = \sum_{i=1}^{N} \left| F_1(i) - F_2(i) \right| \qquad (7)$$

$$with \begin{cases} D_H = 0 & if \quad F_1(i) = F_2(i) \\ D_H = 1 & if \quad F_1(i) \ne F_2(i) \end{cases}$$

Equation (7) is used to find the minimum number of substitutions needed to form one FREAK descriptor to another and the minimal value D_H is a sign of good matches.

C. Homography Based Image Transformation

Based on the found pairs of matched features, selecting an appropriate transformation model to compute the image alignments is an important step for image mosaicing **[23]**.

- *Homography Construction*

Homography can be represented as transformation of a system of four points into another one. The form of the projective model determines the type of represented geometric transformation **[24].** With a rotation angle θ , we can add translation t_x and t_y, and multiply the rotation matrix by *s* to get scaling and introducing skewing effect by multiplying a and *b* parameters. With adjusting perspective in the final row, by adding parameters P_1 and P_2, we get:

978-1-7281-8292-6/20/$31.00 ©2020 IEEE

$$H=\begin{bmatrix} s\times a\times\cos(\theta) & -s\times b\times\sin(\theta) & t_x \\ s\times a\times\sin(\theta) & s\times b\times\cos(\theta) & t_y \\ P_1 & P_2 & 1 \end{bmatrix}=\begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & 1 \end{bmatrix} \quad (10)$$

- *DLT based Homography Estimation*

The Direct Linear Transform (DLT) **[25-26]** is a simple algorithm used to solve for the homography matrix H given a sufficient set of point correspondences. Letting a point and its correspondent that can be written respectively as (x, y, 1); (u, v, 1), we can get:

$$\begin{aligned} -h1x-h2y-h3+\left(h7x+h8y+h9\right)u=0 \\ -h4x-h5y-h6+\left(h7x+h8y+h9\right)v=0 \end{aligned} \quad (11)$$

These equations can be written in matrix form as:

$$A\times H=0 \quad (12)$$

Where

$$A=\begin{pmatrix} -x & -y & -1 & 0 & 0 & 0 & ux & uy & u \\ 0 & 0 & 0 & -x & -y & -1 & vx & vy & v \end{pmatrix}$$

$$h=(h_1 \quad h_2 \quad h_3 \quad h_4 \quad h_5 \quad h_6 \quad h_7 \quad h_8 \quad h_9)^T$$

Since N pairs of correspondences provide 2N equations, 4 pairs are sufficient to solve for the 8 degrees of freedom of H.

D. Image Projection and Blending

In backward warping, points from the second image are transformed to the frame of the first image as shows equation:

$$x=H^{-1}*x' \quad (13)$$

Alignment of images may be imperfect due to registration errors resulting from incompatible model assumption. Therefore, different blending techniques can be used to compensate these generated errors **[27-28]**.

IV. Results and Evaluations

A. Material and Software

The algorithm for mosaicing of images and object tracking were executed on a computer that disposes 4 GB of RAM, CPU of Intel i7 generation and Intel graphic card, and our implementations were done on LabVIEW environment.

B. LabVIEW Results of Image Mosaicing

To test our system, we have designed an execution interface on LabVIEW, this allowed the user to upload two overlapping images, furthermore, we give the user the ability to select the type of corners detector (FAST or Harris) and its threshold value, also, the user can choose between FREAK or BRISK descriptor for the features matching. By the end; an execution summary will be shown including some essential information:
• Number of detected features in both images.
• Number of matched points between images.
• Type of transformation model.
• Image warping method.
Using the block diagram, we have programmed all steps of our algorithms graphically. *Figures (3-4-5-6-7-8)* show the obtained results on LabVIEW interface (Front Panel), of three different examples, the two first examples are from robotic data base [**29**] and the last one is of smartphone.

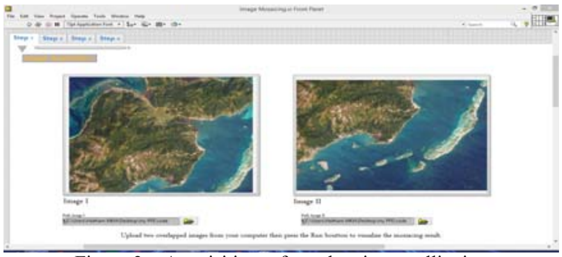

Figure 3. Acquisition of overlapping satellite images.

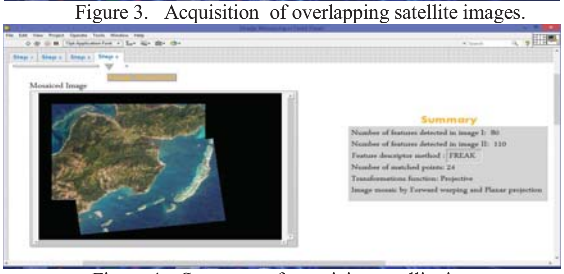

Figure 4. Summary of mosaicing satellite images.

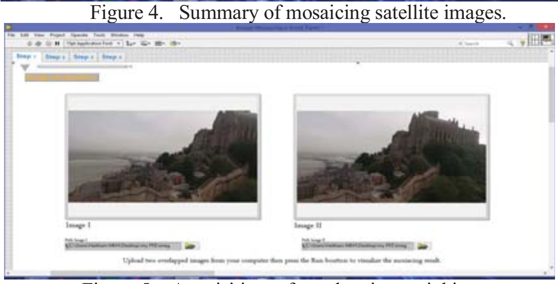

Figure 5. Acquisition of overlapping aerial images.

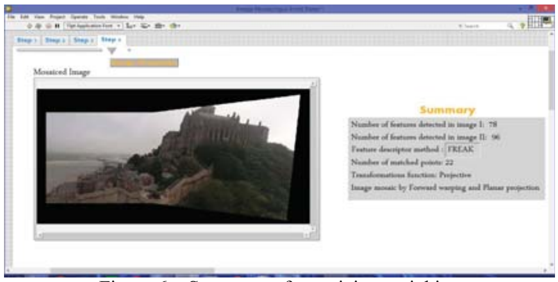

Figure 6. Summary of mosaicing aerial images.

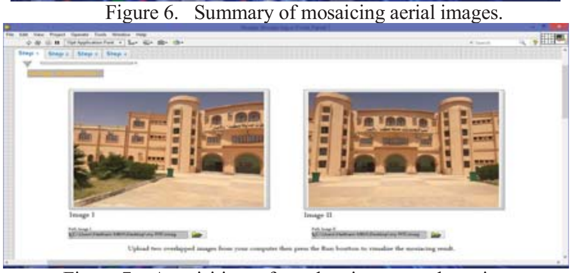

Figure 7. Acquisition of overlapping smartphone images.

Figure 8. Summary of mosaicing smartphone images.

978-1-7281-8292-6/20/$31.00 ©2020 IEEE

To compare the performance of the proposed method with other ones, visual comparisons can be subjective; moreover, numerical evaluations were used for comparison. We have used RMSE (Root Mean Squared Error) metric **[30]** for evaluation and *Table 1* summarizes our comparisons:

TABLE 1. COMPARISON OF RMSE VALUES.

Method	**RMSE**
Literature [31]	1.9
Literature [32]	1.6
Scene 1	0.2
Scene 2	0.14
Scene 3	0.05

From the above table, we can notice that our method gives very efficient results. RMSE value is reduced in our algorithm because Harris detector combined with FREAK descriptors gives good matches.

We have also evaluated the performance of our approach using recall measure **[33]**. *Table 2* shows the comparison of evaluation results with existing methods:

TABLE 2. COMPARISON OF RECALL VALUES.

Method	**Recall**
Literature [8]	0.62
Literature [11]	0.71
Literature [34]	0.27
Literature [35]	0.60
Scene 1	0.76
Scene 2	0.75
Scene 3	0.72

From the above table, our approach has high recall value compared to other methods, and this refers to the efficiency of the proposed technique. Since, texture information extracted using FREAK descriptors is robust to illumination variation and geometric deformation, it helps for enhancing the matching accuracy, thus recall value will be good enough.

The time cost of our method is reduced compared to other algorithms, because of the use of binary descriptors minimizes the computation time. *Table 3* compares the computation time of our algorithm to other works:

TABLE 3. COMPARISON OF TIME COSTS.

	Images size	**Mosaic time (s)**
Literature [36]	408 ×336	4.10
Literature [37]	381× 334	4.64
Scene ♯ 1	420 × 520	2.98
Scene ♯ 2	390 × 420	2.66
Scene ♯ 3	390 × 420	3.22

C. LabVIEW Results of Object tracking

Most previous object tracking works were done in higher level application that simply requires the shape of the object in each frame. We have identified effective method for robust tracking of the object which provides potential future research towards this field. We have used the key points based features for characterizing the tracked object, then; we have used the same image registration steps that were used for image mosaicing.

To test the performance of our application, we have used two different tracking cases, the first case is to track a moving object with a fixed camera and the second case is to track a moving object with a moving camera. *Figures (9-10)* show the obtained object tracking results on LabVIEW

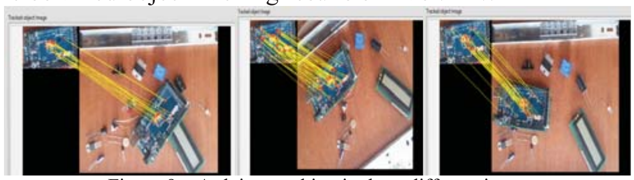

Figure 9. Arduino tracking in three different instances.

Figure 10. Car tracking in three different instances.

By applying our features based tracking algorithm on the two above examples (Arduino and car); we got excellent results, in which the object will be kept tracked while it is in the appearance of the camera. Advantages of our algorithms are:

1) The algorithm is based on local features (points) compared to other algorithms **[12, 13, 14, 15]**.
2) Tracking object can be done using single/different cameras.
3) The LabVIEW interface is simple and practical, and it can be used for real time applications.

V. CONCLUSIONS AND FUTURE WORKS

Our paper has introduced a novel image registration method for image mosaicing and object tracking algorithms; for overlapping images that were captured by single or different cameras. The novel algorithms consist of a feature correspondence through constructing FREAK descriptors and minimizing the Hamming distance between matched feature pairs to align the images geometrically and avoid the outliers in the matched features. Our applications were tested in Matlab, and then they were implemented on LabVIEW tool. The efficiency of the obtained results is verified by using different types of images. Finally, execution interfaces were constructed on LabVIEW, this allowed the user to upload overlapping images and get the result with one order.

The recommendations that can be made for the future investigation, is to use other kinds of features detectors, such, and also other techniques for features matching. We also recommend the generalization of our algorithm to create a video from a sequence of overlapped images and to track certain object in that video.

REFERENCES

[1] Keane, John F., and Stephen S. Carr. “A brief history of early unmanned aircraft”. Johns Hopkins APL Technical Digest 32.3 pp: 558-571. 2013.

[2] McCormack, Edward Donald, and Ted Trepanier. “The use of small unmanned aircraft by the Washington State Department of Transportation”. No. WA-RD 703.1. Washington State Department of Transportation, 2008.

978-1-7281-8292-6/20/$31.00 ©2020 IEEE

[3] Capel, D, and Andrew Z. “Automated mosaicing with super-resolution zoom”. Proceedings of Conference on Computer Vision and Pattern Recognition. Computer Society. 1998.

[4] Fischler, Martin A., and Robert C. Bolles. “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. 1981.

[5] Lin, Wen-Yan Daniel, et al. "Bilateral functions for global motion modeling." European Conference on Computer Vision. Springer, Cham, 2014.

[6] Bevilacqua, Alessandro, and Pietro Azzari. "A fast and reliable image mosaicing technique with application to wide area motion detection." International Conference Image Analysis and Recognition. Springer, Berlin, Heidelberg, 2007.

[7] Suhr, Jae Kyu. "Kanade-lucas-tomasi (KLT) feature tracker." Computer Vision (EEE6503). pp: 9-18. 2009.

[8] Lowe, David G. “Distinctive image features from scale-invariant keypoints”. International journal of computer vision 60.2 pp: 91-110. 2004.

[9] Raguram, Rahul, Jan-Michael Frahm, and Marc Pollefeys. "A comparative analysis of RANSAC techniques leading to adaptive real-time random sample consensus." European Conference on Computer Vision. Springer, Heidelberg, 2008.

[10] Ranjan, Nikhil, Braj Bihari Soni, and Brahmi Shraman. "An efficient technique for image Mosaicing using random sample consensus algorithm." International Journal of Computer Applications 118.16. 2015.

[11] Nagaraja, S., C. J. Prabhakar, and Praveen PU Kumar. "Parallax Effect Free Mosaicing of Underwater Video sequence based on Texture Features." Signal & Image Processing 5.5. 2014.

[12] Subramanian, T., Madbhavi, R., Potadar, S., D’souza, S. J., and Gangadharan, K. V. “Object Follower and Barrier Escaping Robot Using Image Processing”, International Journal of Innovative Research in Science, Engineering and Technology, ISSN(Online) : 2319-8753, 2015.

[13] Divya, M. “Single Object Tracking System By Using Labview”. International Journal on Recent and Innovation Trends in Computing and Communication 4.5 pp: 52-56. 2016.

[14] Holonec, Rodica, et al. “Object Tracking System using Stereo Vision and LabVIEW Algorithms”. Acta Electrotehnica 55.1–2. pp: 71-76. 2014.

[15] Chaitra, J. M., & AV, R. K. “Smart Autonomous Camera Tracking System Using my RIO With LabVIEW”. American Journal of Engineering Research (AJER). Volume-7, Issue-5, ISSN: 2320-0847. pp:408-413.2018.

[16] Dey, Nilanjan, et al. “A comparative study between Moravec and Harris corner detection of noisy images using adaptive wavelet thresholding technique”. arXiv preprint arXiv:1209.1558. 2012.

[17] Pratt, W.K.,. Digital image processing (4th edition.). John Wiley & Sons, Inc. pp. 465–522. 2007.

[18] Alahi, A., Ortiz, R., & Vandergheynst, P. Freak: “Fast retina keypoint”. In Computer Vision and Pattern Recognition (CVPR), Conference on IEEE. pp: 510-517. 2012.

[19] Leutenegger, Stefan, Margarita Chli, and Roland Y. Siegwart. “BRISK: Binary robust invariant scalable keypoints”. IEEE International Conference on Computer Vision (ICCV). 2011.

[20] R. Ortiz, “FREAK: Fast Retina Keypoint,” in Proceeding of CVPR. Washington, DC, USA: IEEE Computer Society, , pp: 510–517. 2012.

[21] Yao, Q., & Xu, X.. Freak Descriptor With Spatial Pyramid Kernel For Scene Categorization. Inte. Conference on Mechatronics, Electronic, Industrial and Control Engineering Published by Atlantis Press. (2015).

[22] Pandiselvam, P., Marimuthu, T., & Lawrance, R. A Comparative Study on String Matching Algorithms of Biological Sequences. In International Conference on Intelligent Computing(pp. 1-5). (2014).

[23] Patidar, D., and Jain, A., 2011. “Automatic Image Mosaicing: An Approach Based on FFT”. International Journal of Scientific Engineering and Technology . Page 1-4 .Vol 1. Issue 1. India .

[24] J-K Kamarainen and P. Paalanen, “ Experimental study on fast 2d homography estimation from a few point correspondences ”. ISBN 0978-952-214-773-8. ISSN 0783-8069 . Page 1-20 . Lappeenranta .2009.

[25] E. Dubrofsky , “ Homography Estimation ”. A master’s essay submitted in partial fulfillment of the requirements for the degree of in the faculty of graduate studies (computer science). University of British Columbia . 2009.

[26] R. Hartley and A. Zisserman, “ Multiple View Geomerty in Computer Vision ”. Cambridge University Press, 2nd edt. 2003.

[27] Richard, S., 2006. “Image alignment and stitching: a tutorial”, Found. Trends. Comput. Graph. Vis. 2 .pp1–104

[28] Shishir Maheshwari, Umesh C. Pati, “Mosaicing of images using Unsharp Masking algorithm for interest point detection”, Proceedings of 2014 IEEE International Conference on Advance Communication, Control and Computing Technologies, Ramanathapuram, Tamil Nadu, pp. 1431-1434, 2014.

[29] AerialRobotics Dataset. ftp://www.aerialrobotics.eu/ accessed: 2018-03-30

[30] P. Paalanen, J.-K. Kämäräinen, H. Kälviäinen , “Image Based Quantitative Mosaic Evaluation with Artificial Video”, 16th Scandinavian Conference, SCIA 2009, Oslo, Norway, LNCS 5575/2009, pp. 470-479, June 2009.

[31] S. Mehrdad, M. satari, M. Safdary and P. Moallem “Toward real time uavs’ image mosaicking”. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XLI-B1, Prague, Czech Republic. 2016.

[32] Manoj K, Nirupma T, Naveen H and Priya B, ‘Image Mosaicing Method by Using Sift and Grid Based’. IJLTEMAS. Volume III, Issue V, ISSN 2278 – 2540. 2014.

[33] Hassaballah, M., Abdelmgeid, AA., and Hammam, A.’Image Features Detection, Description and Matching’. Springer International Publishing Switzerland. 2016.

[34] Reeff, M., Gerhard, F., Cattin, P. C., & Székely, G. . ‘Mosaicing of Endoscopic Placenta Images’. GI Jahrestagun , pp 467-474. 2006.

[35] Laroze, M., Courtrai, L., & Lefèvre, S. . ‘Human Detection from Aerial Imagery for Automatic Counting of Shellfish Gatherers’. In VISIGRAPP (4: VISAPP) (pp. 664-671). 2016.

[36] Ming Li., Deren Li. And Dengke F. ‘A Study on automatic UAV image mosaic method for paroxysmal disaster ‘. International Archives of the Photo grammetry,. Remote Sensing and Spatial Information Sciences. Melbourne, Australia. 2012.

[37] Bhushan, R., & Sunkaria, R. K. . ‘A novel technique of image mosaicing based on discrete wavelet transform and its performance evaluation’. International Journal of Computer Applications, 98(15). 2014.

978-1-7281-8292-6/20/$31.00 ©2020 IEEE