International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
IJISAE
ISSN:2147-6799 www.ijisae.org Original Research Paper

Optimal Process of Video Stabilization Using Hybrid RANSAC-MSAC Algorithm

S. Afsal[1], Arul Linsely[2]

Submitted: 11/11/2022 **Accepted**: 10/02/2023

Abstract: The diversity and amount of 360° cameras are growing as these sorts of cameras are used more frequently, with people filming 360° videos in a variety of settings. Trying to keep the camera stable and prevent shaking is not always simple, especially when using a handheld camera to record motion such as a walking tour or mountain bike ride. So far the majority of video stabilization technology has been created for recording video with a limited field of view, such as conventional videos shot with a smartphone, and it employs methods that don't translate well to 360-degree films. The architecture used by the majority of current video stabilization algorithms aid in attaining various benefits: they track gestures in the video, fit a motion model, smooth the motion, and then generate the stabilized output frames. Consequently, a feature extraction module is included in the video stabilization, and there are various ways to extract the feature. The fact that the SURF (Speeded-Up Robust-Features) is invariant to scale, rotation, translation, illumination, and blur makes them the most suitable techniques for feature detection and matching. To perform reliable estimation of inliers and outliers, hybridized RANSAC (Random sample consensus) and MSAC (M- estimator sample consensus) approaches are proposed in this work. Following this, a matched point pairs are fitted into an affine transformation model, thereby estimating the interframe motion.

Keywords: *360° videos, Speeded-Up Robust-Features, Feature detection, RANSAC, MSAC.*

1. Introduction

Videos are frequently used nowadays to capture significant moments in people's life. People frequently use handheld cameras to document important events, like trips and gatherings. Nowadays, a huge population is able to access 360-degree cameras because of the recent proliferation of its advanced features and their affordable price. They present Virtual Reality (VR) in its most approachable form through a spherical video [1]. Users of VR can enter and completely engage themselves in a digital world. The jitter occurs due to the shaky cameras has been present in these video clips because of the lack of professional stabilizing equipment. The video quality has been affected due to this instability, and it will also hinder the effectiveness of succeeding procedures like video coding or video surveillance. Decisions have been made to create a brand-new stabilizing technique specifically for 360° videos since more individuals are actively recording more realistic videos in real-life situations. Utilizing this technique, videos are stabilized in a fraction of the time it takes to play them normally. Another benefit of the stabilization technique is that it makes it easier to speed up a stabilized 360° video, transforming a drawn-out film into a fast-paced and enjoyable action experience. To equalize the camera velocity over time, the timing of the video frame timestamps has been modified. This is currently being tested, and intend to make it available to users in upcoming updates [2]-[5].

A cube map conversion has been performed in the framework to achieve omnidirectional viewing of 360° video [6]. The spherical signals in omnidirectional videos are obtained from cameras having a full 360° field of view or viewport. The area of the sphere in user's field of view is smoothly changed during the recording on the basis of user's head motion, improving user's experience of presence. When compared to conventional videos, the new interactive dimension and immersive element have a significant impact on how the end user perceives the quality of the experience [7, 8]. Thus the video stabilization includes a feature extraction module and there are several methods available to extract the feature. Among all, a suitable feature extraction approach has been selected to extricate the fine desired features from the image [9]. The video is upgraded in the preprocessing stage by taking out the haze, noise, and low illumination.

[1]*Assistant Professor, Department of Electronics and Communication Engineering, ACE College of Engineering, Thiruvallam, Trivandrum, Kerala, India.*
[1] *Head of the Department & Associate Professor, Department of Electrical and Electronics Engineering, Noorul Islam Centre for Higher Education, Thuckalay, Tamil Nadu, India.*
**Corresponding Author: yuvaafsals@gmail.com*

The initial step is to transform the video sequence into frames. The key frames are then acquired using a feature extraction approach [10].

To limit the amount of dense, large-scale content in a frame while maintaining the overview of the entire image, the feature vector has to be properly retrieved from the video frame. In contrast to global features, which utilize a single feature vector to represent all of the information in a video frame, local features use a variety of feature vectors to identify and describe essential parts or interest regions in the video frame. For global features, simple calculations and less memory storage is required, but they are not invariant to big movements and are vulnerable to clutter and occlusion. The resistance of local features to changes in rotation, scale, noise and light is higher than that of global features. The local characteristics take up a lot of memory but are more effective at matching pictures and identifying objects. Numerous techniques, such as SURF and SIFT (Scale Invariant Feature Transform) are utilized to find and match features, which are the popular methods for detecting and matching features. In terms of rotation invariance, warp transform, and blur, SURF is superior to SIFT. Due to the employment of a box filter and integral picture, SURF is three times faster than SIFT. Three steps make up the suggested algorithm: feature extraction, feature description and matching. To shorten the execution time, the robust features are extracted from the video frame using the SURF approach [11]-[13].

The movement of the platform causes unwanted jitter in cameras mounted on moving platforms, blurring and shaking the recorded footage. In order to eliminate unwanted image motion and create a corrected video sequence with only smooth global motions, a digital video stabilization technology is employed [14, 15]. The features are extracted using the SURF approach to estimate the robust inter frame motion and then the extracted features are matched among the adjacent frames. Then, the inliers and outliers are detected by the RANSAC algorithm [16, 17]. RANSAC is typically used to filter out local motion vectors and improper correspondences. However, it only works effectively on feature points of quick-moving objects. RANSAC is not suitable for properly eliminating the feature points if the object is moving slowly relative to the background [18]. In order to overcome the above difficulties, the hybrid stabilization approach has been developed as it holds the advantages of two effective algorithms [19]. Thus the hybrid RANSAC-MSAC algorithm is developed in this work for eliminating the inliers and outliers.

2. Proposed Methodology

Cameras installed on moving platforms experience undesirable jitter due to platform movement, which blurs and shakes the footage that is being recorded. To eliminate undesired image motion and create a rectified video sequence with only smooth global motions, digital video stabilisation is used.

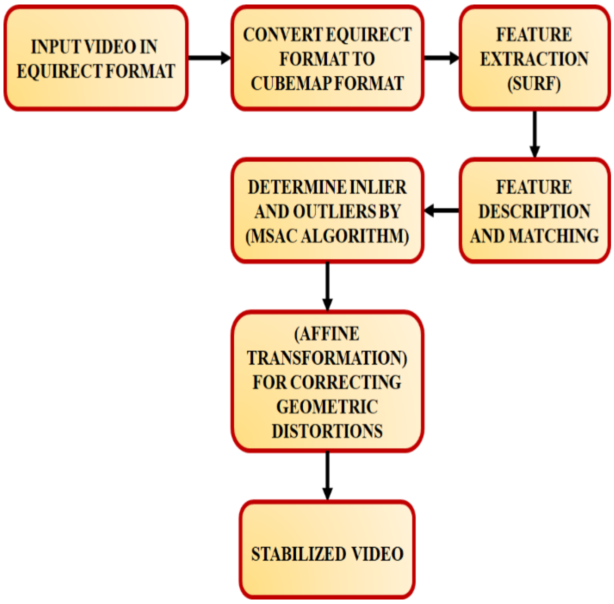

Fig.1. Proposed block diagram

The SURF feature descriptor is the foundation of our method for video stabilization. The SIFT is superseded by SURF, which also costs less to compute than SIFT. There are two primary steps in SURF: Identification of interest points and description of interest points.

2.1. Video Stabilization

The 360-degree cameras recording the entire viewing sphere is regarded as a benefit. Therefore, one need not worry about the frame clipping that occurs in recordings with small fields of vision when warping the frame randomly for video stabilization. Anyhow, this is a serious issue that needs an immediate solution. Imagine a situation in which a videographer is traveling ahead while maintaining a straight line of sight with a 360-degree camera. The cameraman takes a 90-degree turn to the right at a crossroads. If one stabilizes by distorting the frame to the reference, then the camera's perspective changes. A viewer turns the perspective and moves it back toward the front, but this is difficult and lowers the quality of the experience. Thus video stabilization technology plays a vital role in solving this issue.

2.2. Cubemap Representation

Typically, 360° videos are displayed in an equirect format. Although this format is good for seeing, computer vision systems have difficulty processing it. Therefore, it has been converted into cubemap format for further processing. The image is created by projecting the viewing sphere onto a unit cube's six faces. Each face corresponds to an image taken by a pinhole camera with a unit focal length placed in the cube's center. The majority of the computer vision technique for estimating camera pose has been applied to face images because they adhere to the typical epipolar geometry.

2.3. Surf Feature Extraction

A recently created framework called Speeded-Up Robust Features (SURF) is very likely to replace existing feature detectors. Due to its robust features, such as illumination invariance, translation invariance, size invariance, rotation invariance and contrast invariance, the SURF method is usually employed for performing object identification tasks. Also, it effectively recognizes the objects in photographs captured under various extrinsic and intrinsic settings. Fig.2 portrays the steps involved in the SURF feature extraction algorithm.

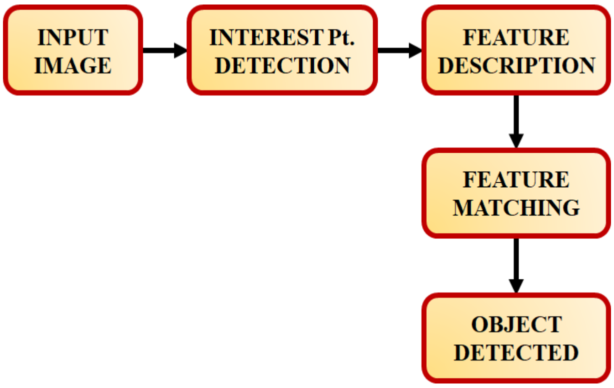

Fig.2. Process involved in SURF feature extraction algorithm

All subsequent components of the algorithm use integral images to dramatically increase their speed. In order to compute the surface integral of any size from the original image while utilizing an integral image, it is always important to read only four-pixel values.

$$I\sum(x,y) = \sum_{i=0}^{x}\sum_{j=0}^{y} I(i,j) \tag{1}$$

When calculating the responses of the Gaussian and Haar wavelet filters, this knowledge is frequently used.

$$\mathrm{H(x,y)} = \begin{vmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x\,\partial y} \\ \frac{\partial^2 f}{\partial x\,\partial y} & \frac{\partial^2 f}{\partial y^2} \end{vmatrix} \tag{2}$$

$$H(\bar{x}) = D_{xx}(\bar{x})D_{yy}(\bar{x}) - (0.9D_{xy}(\bar{x}))^2 \tag{3}$$

$$\therefore \bar{x} = (x, y, s)$$

To find the significant details in a picture, SURF employs determinants of Hessian matrices. Image convolutions with estimated Gaussian kernel second order derivatives are used to substitute second order partial derivatives. Box filters with coefficients $1, -1, 2, -2^2$ are used to approximate. This approximation is compensated by using the coefficient $0:9$ in equation (5). The size of the Gaussian kernels as well as their location in the image are defined.

$$\mathrm{H(x,y)} = \mathrm{H} + \frac{\partial \mathrm{H}^{\mathrm{T}}}{\partial \mathrm{x}}\mathrm{x} + \frac{1}{2}\mathrm{x}^{\mathrm{T}}\frac{\partial^2 \mathrm{H}}{\partial \mathrm{x}^2}\mathrm{x} \tag{4}$$

$$\hat{x} = \frac{\partial^2 H^{-1}}{\partial x^2}\frac{\partial H}{\partial x} \tag{5}$$

All of the sample points are given the same weight if the SURF algorithm is applied. By giving the dynamic weights to the representative points, this has been explained. It makes intuitive sense that genuine representative points will frequently appear in photos from the training set while fraudulent representative points will not.

Initially, the input video file is processed to extract the frames. A long time is required to compare every pair of successive frames. It selects the first frame and skips the adjacent frame and then the same has been applied to all the other frames in the video file, which thereby reduces the execution time. Therefore, half of the video files alone are processed using this method, hence it requires less time to remove and match the features. For better understanding, take a look at Table 1, in which $\mathrm{F1, F2, F3}$ & $\mathrm{F4}$ are similar but different from $\mathrm{F5, F6, F7}$ & $\mathrm{F8}$, in other words, the adjacent shot frames are different from the first shot frame. Moreover, frame F5 has been chosen as the abrupt transition frame. As a result, F1, F3, F5 and F7 are used for further processing whereas F2, F4, F6 and F8 are ignored. Thus, F1 is compared to F3, which is then compared to F5, and so forth.

Table 1. Frame selection

Shot 1				Shot 2			
$F1$	$F2$	$F3$	$F4$	$F5$	$F6$	$F7$	$F8$
$F1$		$F3$		$F5$		$F7$	
$Ft(1)$		$Ft(2)$		$Ft(3)$		$Ft(4)$	

The extracted frames are then converted into greyscale images after that it has been resized to $\mathrm{N \times N}$. The feature vector (Ft) is then estimated using the SURF feature descriptor. Key points are first found, and then they have to be described in a feature vector in the second step. Here, each key point's feature vector is measured at length 64, resulting in a descriptor features matrix (Ft) with a size of $(p \times 64)$, where (p) is the total key points found in each frame. It is important to note that because SURF features are scaled invariable and hence varying the image size has no influence over these features.

2.4. Feature detection and matching

Following feature extraction, feature matching is performed where the feature matrices $Ft(i)$ and $Ft(i+1)$

are compared. The distance function is used to find similarities across feature matrices based on the features matching. The respective feature vectors of feature matrices $\mathrm{Ft}(\mathrm{i})$ and $\mathrm{Ft}(\mathrm{i}+1)$ are compared using the Hamming distance formula. When the gap between adjacent feature vectors is below a given threshold value, they are deemed similar, and when the distance exceeds the threshold value, the feature vectors are ignored. The output is a vector $\mathrm{p} \times 1$ representing the number of matching features (MF) between the two feature matrices. Be mindful features are only taken from half of the video frames, therefore multiplying the sudden transition index by two will give the precise index for an abrupt transition. Following algorithm describes the SURF approach for feature extraction.

Step 1: Video ← Extract the input video frames

FrameNum ← Represents the frame number of input video

N ← Select the frame size

Th ← Set the threshold value

Initialize $C_1, C_2 \leftarrow 1$ //counters

Step 2: $for\ (i = 1, i + 2, i < FrameNum)$

Ft ← Convert the input video into grayscale

Ft ← resize Ft to N × N

$\mathrm{Ft}(\mathrm{C}_1) \leftarrow$ SURF feature (Ft) extraction

$\mathrm{C}_1 \leftarrow \mathrm{C}_1 + 1$

end for

Step 3: $for\ (i = 1, i + 1, i < C_1)$

MF(i) ← Feature matching (Ft (i), Ft (i+1))

Step 4: $for\ (i = 2, i + 1, i < C_1)$

If $(\mathrm{MF(i)} < \mathrm{Th}) \& \mathrm{MF(i+1)} > \mathrm{Th}) \& (\mathrm{MF(i-1)} > \mathrm{Th})$

$\mathrm{Ft}(\mathrm{C}_2) \leftarrow \mathrm{i} \times 2$

$\mathrm{C}_2 \leftarrow \mathrm{C}_2 + 1$

end if

end for

2.5. Hybrid RANSAC-MSAC Algorithm for Determining the Inliers and Outlier

Generally, from a set of correspondences, the RANSAC algorithm enables one to identify only one transformation. As mentioned earlier, it is frequently required to be able to recognize multiple groups of correspondences. RANSAC has been used sequentially to accomplish multiple detections, however, several writers have noted the challenges in using this method. Therefore, the RANSAC and MSAC algorithms have been combined together to rectify the aforementioned issue.

A summary of the final and comprehensive RANSAC-MSAC algorithm is given below, which includes the concepts that have been discussed in the following sentences. i_{max} is the only parameter used that denotes the maximal iteration. Since this is a contrary approach, the NFA is set to have a threshold value of $\varepsilon = 1$. To set the correspondence, the RANSAC-MSAC approach applies RANSAC at each iteration, hence the inlier groups discovered after each iteration has been removed. In fact, the self-similarity-driven repetitive matching result in artificial detections that "echo" the two images' genuine, unique transformations. The identical thing is then discovered multiple times in various postures. Typically, the proper transformation gets the highest score, making it the first one to be found. The exact description of repeating correspondences is then used to eliminate further detections of echoing transformations.

Repeating correspondence: The assigned consensus set has been represented as S and ℓ denotes the rest of the correspondences in the set ($\therefore\ \ell \cap S = \emptyset$). If the below conditions are satisfied then $c_i = (m_i, m_i') \in \ell$ is repetitive with respect to S.

Condition 1: $\exists\ m \in S\ s.t. \|m - m_i\|_2\ < min\{\sigma, \sigma_i\}$

Condition 2: $\exists\ m' \in S\ s.t. \|m' - m_i'\|_2\ < min\{\sigma', \sigma_i'\}$

The RANSAC-MSAC algorithm is written as follows,

Initialize the input by setting the non-redundant correspondence ℓ and i_{max} ($\therefore i = 0\ and\ \ell = \{\emptyset\}$.

Step 1: Detect the samples when $i < i_{max}$, minimizing $NFA\ (S, S')$ by uniform sampling of $S' \subset C$ and search of $S \subset C \backslash S'$(if $NFA\ (S, S') < 1\ and\ (S_{opt}, S'_{opt}) = (S, S')$, go to step 2, else $i = 1 + 1$).

Step 2: Repeat $i_{max}/10$ times. A sampling of S' and search of $\subset C \backslash S'$ minimizes $NFA\ (S, S')$ $\{\ if\ NFA\ (S, S')\ < NFA(S_{opt}, S'_{opt}), (S_{opt}, S'_{opt}) = (S, S')$.

Step 3: Optimal subset pair searching which includes S_{opt}. If the fusion has been detected then two meaningful subsets have been found (ie. (S_1, S_1') and (S_2, S_2') else, $S_1 = S_{opt}$ and $S_2 = \emptyset$.

Step 4: Discard the correspondences that are repetitions of S_1 ($\therefore C = C \backslash S_1$).

Step 5: S_1 has been added to the list $\ell, i = 0$. If $S_2 = \emptyset$, go to step 1, else, $(S_{opt}, S'_{opt}) = (S_2, S_2')$, go to step 2.

The output of disjoint group lists ℓ has been generated.

The repetitive correspondences for this group are now removed from the rest of the correspondences set C once each new group of correspondences is verified.

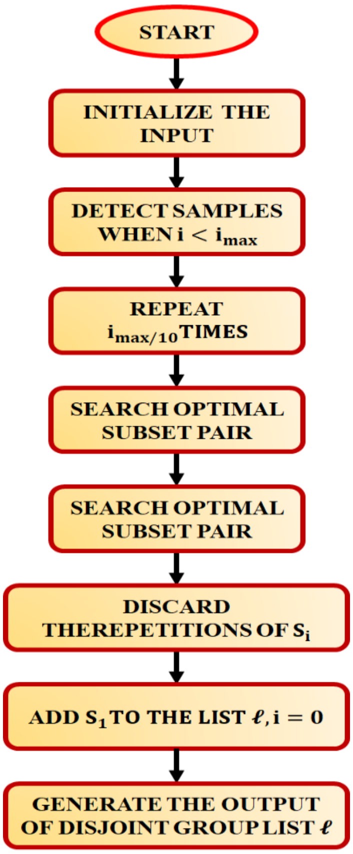

Fig.3. RANSAC-MSAC- Flowchart

The process flow of this hybrid algorithm in this video stabilization is significantly illustrated in Fig. 3.

3. Result and Discussion

The process of video stabilizing using SURF feature extraction is significantly carried out in this work for tracking the exact motion and gesture of the objects in the video with maximum clarity in an optimal manner. In addition, the implementation of a hybrid RANSAC-MSAC algorithm is preferred for estimating the inliers and outliers with high reliability. The entire work is evaluated using the MATLAB simulink and the attained outputs are evidently explained in this section for validating the introduced approach in an optimal manner.

Fig.4. Equirect format input video frame

The image representing the input video frame in equirect format is significantly portrayed in Fig. 4, which significantly displays the 360° horizontal view and 180° vertical view of the input frame in an efficient manner. Through this equirect format, it is highly possible to visualize the data originating from all directions as a points on a sphere since it consists of light data from all directions.

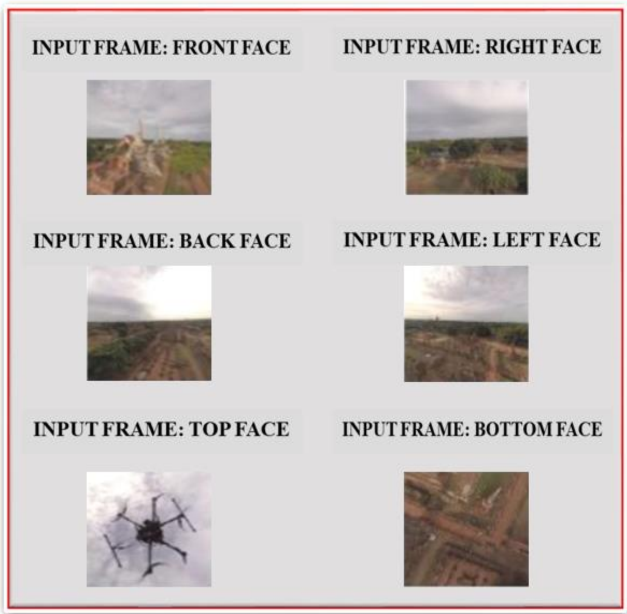

Fig.5. Cubemap representation

When editing the north and south poles of a spherical panorama, it is common to convert an equirect image into a cubemap, which has six cube faces like left, right, top, bottom, back and front faces in the cubic format to cover the entire sphere. In addition to eliminating the image distortions, viewpoint dependency and computational inefficiency, it efficiently reduces the time required for computation. The obtained cubemap illustration of the input video frame is remarkably portrayed in Fig.5.

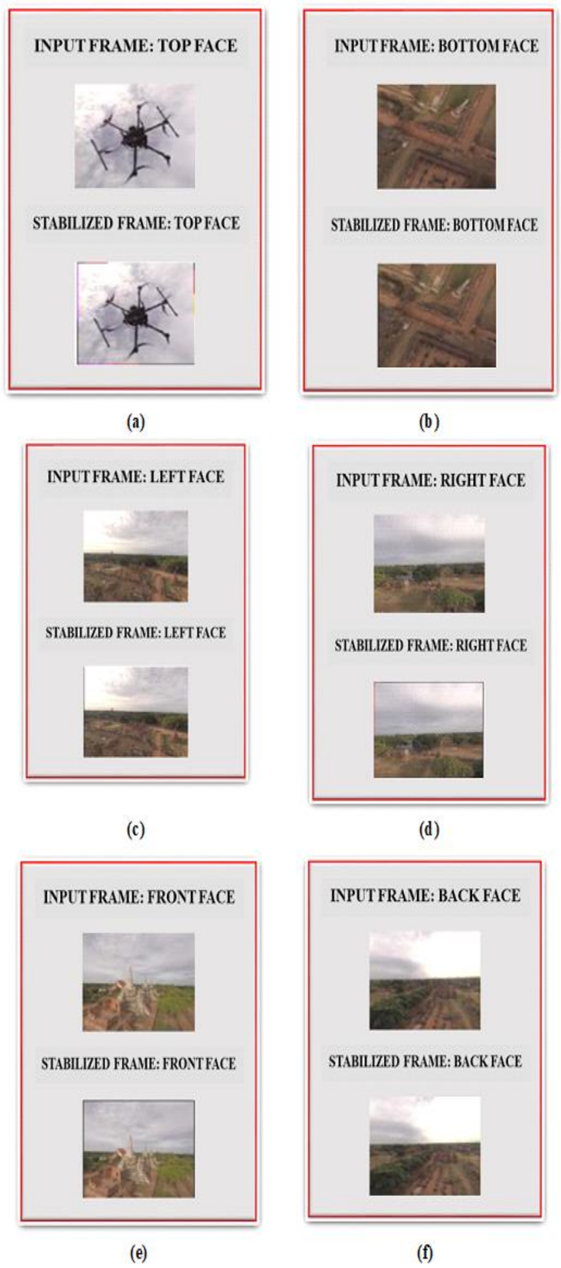

Fig.6. Input and Stabilized frames of (a) Top (b) Bottom (c) Left (d) Right (e) Front (f) Back Faces

To remove unwelcome camera vibration from a video sequence, video stabilization is effectively done in this study with the assistance of SURF based feature extraction. The input frame and the stabilized frame of the of all the six faces are individually illustrated in the Fig. 6 (a- (f)), which validates that the stabilization of the video frame is efficiently accomplished through the introduced approach using hybrid RANSAC-MSAC algorithm since the inliers and outliers are significantly assessed without any complication in an accurate manner.

The performance of the hybrid RANSAC-MSAC along with the SURF approach in the process of extracting the features in analogized with the SIFT based extraction approach for authenticating the contribution of introduced approach. The outcomes of the comparative analysis are significantly highlighted in Table. 2 to Table. 4 and Fig. 7 to Fig. 9 in an optimal way.

Table 2. Comparative analysis of SIFT and SURF based on Matching Pairs

Image	SIFT	SURF
1 *and* 2	37	40
2 *and* 3	40	43
3 *and* 4	39	43
4 *and* 5	40	40
5 *and* 6	38	39
6 *and* 7	40	43
7 *and* 8	42	38

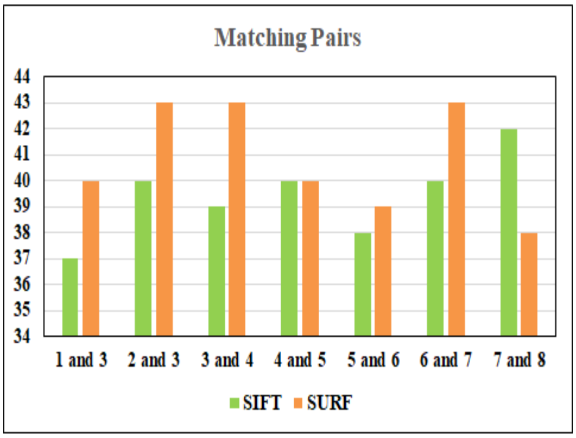

Fig.7. Comparison of Matching Pairs

Table 3. Comparative analysis of SIFT and SURF based on Computation Time

Image	SIFT			SURF		
	Detection time (s)	Matching Time (s)	Total (s)	Detection time(s)	Matching Time(s)	Total (s)
1 *and*	0.142	0.006	0.148	0.021	0.005	0.026
2 *and*	0.144	0.004	0.148	0.020	0.005	0.024
3 *and*	0.157	0.005	0.162	0.033	0.007	0.039
4 *and*	0.168	0.004	0.172	0.035	0.004	0.039
5 *and*	0.173	0.005	0.178	0.021	0.007	0.026
6 *and*	0.164	0.004	0.168	0.018	0.004	0.022
7 *and*	0.156	0.005	0.160	0.022	0.004	0.025

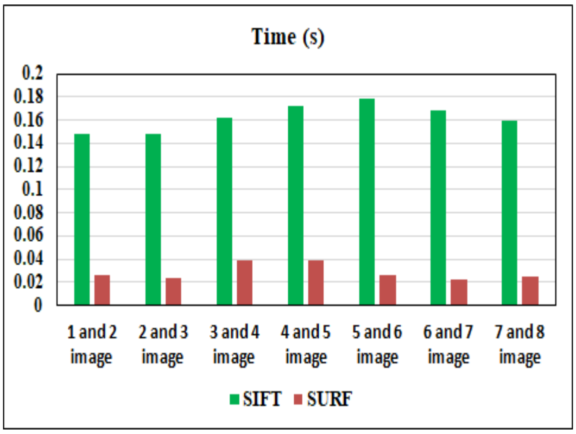

Fig.8. Comparison of Time

Table 4. Performance analysis based on accuracy

Approach	Accuracy (%)					
	Rotation	Scale	Blur	Illumination	Warp	Time Cost
SIFT	91.2	91.8	93.2	94.4	94.6	94.9
SURF	93.4	94.1	94.9	95.2	95.8	96.8

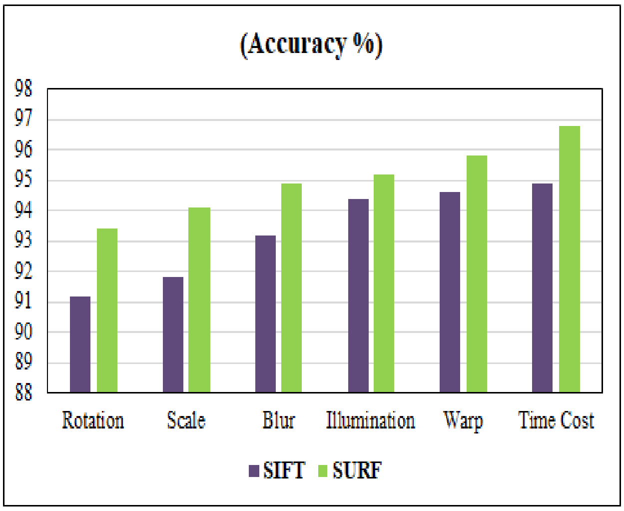

Fig.8. Comparison of Accuracy

Thus, it is proven from the obtained outcomes that the introduced approach significantly outperforms the existing approach in an efficient manner with optimal accuracy.

4. Conclusion

The elaborate analysis for validating the eminence of video processing through stabilization using SURF based feature extraction is remarkably provided in the present work since it owns plenty of meritorious impacts in enhancing the quality of the video in a wider range. The feature extraction using SURF algorithm efficiently involves in maximizing the reliability of the video frame in an effective manner, which in turn improves the performance capability of overall system. In addition, the estimation of Inliers and outliers is remarkably done using the introduced hybrid RANSAC-MSAC algorithm for obtaining the disruption free identification of the gestures and motions of the objects in the video without ant complexities. Comparative analysis done between the SURF and SIFT approaches proves that the SURF approach used in this work has delivered optimal outcomes than the SIFT with maximum accuracy.

References

[1] A. Luchetti, M. Zanetti, D. Kalkofen and De Cecco, M., “Stabilization of spherical videos based on feature uncertainty”, *The Visual Computer*, pp. 1–14, 2022.

[2] M. Zhao and Q. Ling, “A robust traffic video stabilization method assisted by foreground feature trajectories”, *IEEE Access*, vol. 7, pp. 42921–42933, 2019.

[3] A. Luchetti, M. Zanetti, D. Kalkofen and M. De Cecco, “Stabilization of spherical videos based on feature uncertainty”, *The Visual Computer*, pp. 1–4, 2022.

[4] T. Ma, Y. Nie, Q. Zhang, Z. Zhang, H. Sun and G. Li, “Effective video stabilization via joint trajectory smoothing and frame warping”, *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 11, pp. 3163–3176, 2019.

[5] T. Ma, Y. Nie, Q. Zhang, Z. Zhang, H. Sun *et al.*, “Effective video stabilization via joint trajectory smoothing and frame warping”, *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 11, pp. 3163-3176, 2019.

[6] H. Jiang, G. Jiang, M. Yu, Y. Zhang, Y. Yang *et al.*, “Cubemap-based perception-driven blind quality assessment for 360-degree images”, *IEEE Transactions on Image Processing*, vol. 30, pp. 2364–2377, 2021.

[7] G. de A. Roberto, N. Birkbeck, I. Janatra, B. Adsumilli and P. Frossard, Multi-feature 360 video quality estimation”, *IEEE Open Journal of Circuits and Systems*, vol. 2, pp. 338–349, 2021.

[8] X. Chen, A. T. Z. Kasgari and W. Saad, “Deep learning for content-based personalized viewport prediction of 360-degree VR videos”, *IEEE Networking Letters*, vol. 2, no. 2, pp. 81 – 84, 2020.

[9] P. V. Rouast and M. T. P. Adam , “Learning deep representations for video-based intake gesture detection”, *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 6, pp. 1727–1737, 2020.

[10] M. K. Asha Paul, J. Kavitha and P. A. J. Rani, “Key-frame extraction techniques: a review”, *Recent Patents on Computer Science*, vol. 11, no. 1, pp. 3–16, 2018.

[11] E. Hato and M. E. Abdulmunem, “Fast algorithm for video shot boundary detection using SURF features”, *In 2019 2nd Scientific Conference of Computer Sciences (SCCS)*, pp. 81–86, 2019.

[12] D. Mistry and A. Banerjee, “Comparison of feature detection and matching approaches: SIFT and SURF”, *GRD Journals-Global Research and Development Journal for Engineering*, vol. 2, no. 4, pp. 7–13, 2017.

[13] Y. Feng, “Mobile terminal video image fuzzy feature extraction simulation based on SURF virtual reality technology”, *IEEE Access*, vol. 8, pp. 156740–156751, 2020.

[14] J. Li, T. Xu and K. Zhang, “Real-time feature-based video stabilization on FPGA”, *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 4, pp. 907–919, 2017.

[15] L. Zhang, Q-K. Xu and H. Huang, “A global approach to fast video stabilization”, *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 2, pp. 225–235, 2017.

[16] T. N. Shene, K. Sridharan and N. Sudha, “Real-time SURF-based video stabilization system for an FPGA-driven mobile robot”, *IEEE Transactions on Industrial Electronics*, vol. 63, no. 8, pp. 5012–5021.

[17] Y. Nie, T. Su, Z. Zhang, H. Sun and G. Li, “Dynamic video stitching via shakiness removing”, *IEEE Transactions on Image Processing*, vol. 27, no. 1, pp. 164–178, 2018.

[18] C. Song, H. Zhao, W. Jing and H. Zhu, “Robust video stabilization based on particle filtering with weighted feature points”, *IEEE Transactions on Consumer Electronics*, vol. 58, no. 2, pp. 570 – 577, 2012.

[19] S. Liu, B. Xu, C. Deng, S. Zhu, B. Zeng *et al.*, “A hybrid approach for near-range video stabilization”, *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 9, pp. 1922–1933, 2017.