International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-11 Issue-12, November 2022

Object Detection using Different Point Feature Techniques: A Comparative Analysis

Suhas Reddy P, Bhargavi Rao B, Jayanth Anala, Megha Dangayach

***Abstract**: The image Recognition system is a vital problem in the field of computer vision because it must be precise, successful in the desired goal, strong, healthy, and self-loading. The following are the most critical essential phases in image alignment/registration: feature matching, feature detection, derivation of transformation function based on related features in pictures, and reconstruction of images based on generated transformation function. In many applications, the goal of computer vision is to create an ideal and accurate image, which is dependent on optimal feature matching and detection. This paper's inquiry summarizes the similarity among five alternative approaches for robust features/interest points (or landmarks) detector and picture identification. This research also focuses on the extraction of unique features from photos that may be utilized to conduct effective matching of diverse perspectives of the images/objects/scenes.*

***Keywords**: Affine, BRISK, Feature extraction and matching, In Lier, ORB, Rotation, Invariance, SURF, Scaleinvaraince, estimate Geometric Transform 2D.*

I. INTRODUCTION

Several studies in computer vision have been conducted on the basis of feature detection. Which are important aspects of computer vision. Bay and Tuytelaars (2006) employed integral images for image convolutions and the Fast-Hessian detector to accelerate robust features. Their tests revealed that it is speedier and performs well. SIFT was introduced by Lowe (2004) as a method for extracting separate invariant features from pictures that are invariant to image scale and rotation. It was then widely utilised in picture mosaicking, identification, retrieval, and other applications. Bay and Tuytelaars (2006) employed integral images for image convolutions and the Fast-Hessian detector to accelerate robust features.

Their tests revealed that it is speedier and performs well. Many computer vision applications include an image matching challenge to detect correspondences between two photographs of the same scene/object. Image registration, camera calibration, and object recognition are just a few examples.

This study is separated into two parts that explain the distinguishing characteristics of photographs. To begin, "key points" are retrieved from unique regions in the photos, such as edges, blobs, corners, and so on. Key point detectors should have a high degree of repeatability [1]. Following that, neighbouring areas are selected around each key point, and different feature descriptors are computed for each region. Image extraction characteristics in pictures can give consistent matching across diverse views of the same image for image matching.

Feature descriptors are retrieved and saved from sample photos during the procedure. Perhaps the most relevant work tackling this problem is SURF [3] which has been demonstrated to achieve robustness and speed, only, as evident in our results, BRISK achieves comparable quality of matching at much less computation time.

This descriptor must be unique while also being resistant to noise and detection mistakes. Finally, the feature descriptors are matched across pictures.

II. IMAGE MATCHING TECHNIQUES

A. SURF

The SURF method (Speeded Up Robust Features) is a fast and robust algorithm for local, similarity invariant representation and comparison of images. The main interest of the SURF approach lies in its fast computation of operators using box filters, thus enabling real-time applications such as tracking and object recognition. Matching the features are compared only if they possessa similar modes of contrast which allows for matching at faster rates [2].

SURF (Speed Up Robust Features) algorithm, is based on multi-scale space theory and the feature detector is based on the Hessian matrix. Since the Hessian matrix has good performance and accuracy [3]. Instead of using Gaussian averaging to approximate the picture, squares are employed since convolution with squares is considerably quicker when the integral image is used. This may be done in parallel at various scales. SURF also use wavelet responses for feature description.

Manuscript received on 03 October 2022 | Revised Manuscript received on 11 October 2022 | Manuscript Accepted on 15 November 2022 | Manuscript published on 30 November 2022.

*Correspondence Author (s)
Suhas Reddy P*, Department of Computer Science & Engineering, Vellore Institute of Technology, Chennai (Tamil Nadu), India. E-mail: penubakasuhas.reddy2019@vitstudent.ac.in
Bhargavi Rao*, Department of Computer Science & Engineering, Vellore Institute of Technology, Chennai (Tamil Nadu), India. Email: bhargavirao.b2019@vitstudent.ac.in
Jayanth Anala*, Department of Computer Science & Engineering, Vellore Institute of Technology, Vellore (Tamil Nadu), India. jayanth.srivaastava2019@vitstudent.ac.in
Megha Dangayach*, Department of Computer Science & Engineering, Vellore Institute of Technology, Chennai (Tamil Nadu), India. E-mail: megha.dangayach2019@vitstudent.ac.in

B. ORB

ORB (Oriented FAST and Rotated BRIEF) isone of the key algorithms in Feature Detection. It detects the key points and computes the feature descriptors. First key points are identified then it computes binary feature vectors and groups all of them in the ORB descriptor. It had been developed as a good alternative to SIFT and SURF. It's a product of mixing between FAST keypoint detector, and BRIEF descriptor and lots of modifications that help to boost performance. FAST is employed at first to find the main points. FAST is a rotation variant and doesn't compute orientation. to extend rotation invariance, moments are calculated. If there's an in-plane rotation, the descriptor BRIEF performs badly.

In ORB, a rotation matrix is made using the patch's orientation, then the BRIEF descriptors are guided based on the orientation. A multiscale picture pyramid is employed by the orb algorithm. a picture pyramid is a multiscale depiction of a single image made up of sequences of images, each of which may be a different resolution version of the image. Each level of the pyramid contains a down sampled replica of the preceding level's picture. After constructing a pyramid, ORB employs the fast method to find key points in the picture. Orb is effectively finding details at a different scale by detecting key points at each level. ORB is therefore partly scale invariant [4].

C. BRISK

BRISK (Binary Robust Invariant Scalable Key points) is an Image matching technique that has a feature point detection and description algorithm with scale invariance and rotation invariance, developed as a free alternative to SIFT and readily implemented in famous CV libraries. The BRISK algorithm mainly has three modules such as key point detection, description and descriptor matching. It is to acknowledge the fact that modularity allows to use the BRISK detector with any other key point descriptor and vice versa, used for optimizing the performance [3]. The key point detection methodology of BRISK is inspired by AGAST (Adaptive and Generic Accelerated Segment Test) [2]. This method's flexibility allows the BRISK detector to be used in conjunction with any other key point descriptor and vice versa, optimizing for desired performance and the job at hand.

III. PROPOSED WORK

In this paper, we are going to see object detection with various point feature matching techniques, i.e. SURF, ORB, BRISK. and we are going to decide the best point feature technique among them with the help of Inlier points accuracy and various metrics.

ALGORITHM I

1. Read Images
2. Detect Feature Points
3. Extract Feature Descriptors
4. Find Putative Point Matches
5. Locate the object in Scene by Putative Matches
6. Display the detected object
7. Detect Another Object

ALGORITHM II

1. Read Images
2. Detect Feature Points
3. Extract Feature Descriptors
4. Find Putative Point Matches
5. Find the matched points or features from the images
6. Find the inlier points using the function *estimateGeometricTransform2D*
7. *Calculate the inlier points percentage from the above step.*

*100 * sizeofinlierpoints/sizeofmatchedpoints*

IV. SIMULATED RESULTS

In this section, we are going to see the output and tabulate a number of feature points from the scene image and target image and matched points between both the image and time taken of each point feature matching algorithm in the Object Detection and we investigate the sensitivity of SURF, ORB and BRISK against each intensity, rotation and noise.

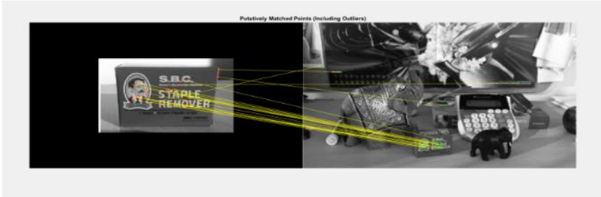

Fig. 1. SURF Matched Points

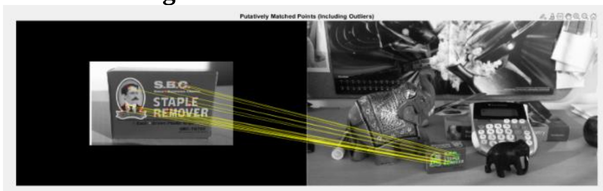

Fig. 2. ORB Matched Points

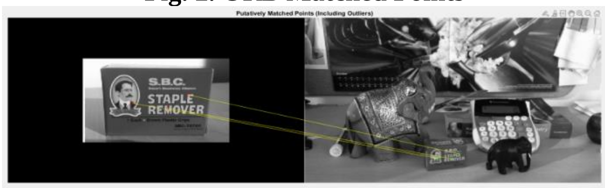

Fig. 3. BRISK Matched Points

From Table I, we can see that ORB has the most features extracted from the reference image and target image and Brisk has the second highest features. As for the time, ORB took the least time for extracting features, but for matching the features Surf took the least time. We will conclude which algorithm is better after seeing the analysis of inlier accuracy from the following data. (intensity, rotation, noise-altered images).

Table- I: Simulated Results

Measure	Time a	Ref	Target	Matches	Time b
SURF	0.162	1367	417	23	0.069
ORB	0.115	15707	3388	18	0.497
BRISK	0.418	2341	655	3	0.077

Time a: Extraction Time; Time b: Matching Time

Retrieval Number: 100.1/ijitee.L930811111222
DOI: 10.35940/ijitee.L9308.11111222
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

In each category, the images go through a modification for such as rotation and brightness for more required results. Figures for each output for individual algorithms can be seen in every category along with the comparison table of various metrics.

A. Intensity

Here we increased the brightness of the image by x2 times. As we can see the input image is whited out to some extent compared to the original images. The Matching points can be seen clearly in each algorithm.

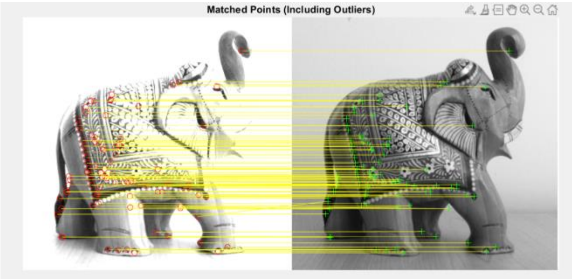

Fig. 4. SURF Matched Points

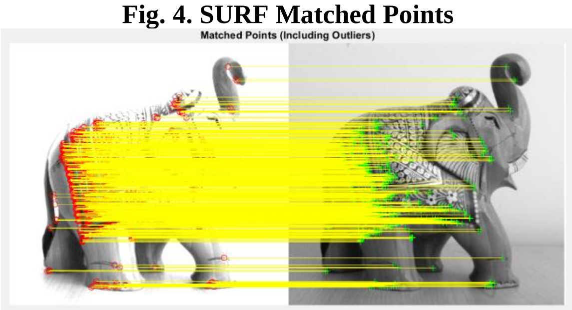

Fig. 5. ORB Matched Points

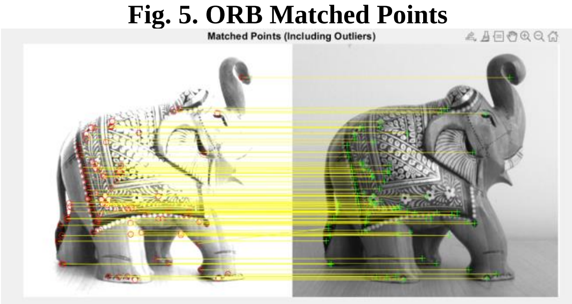

Fig. 6. BRISK Matched Points

Table- II: Intensity Measures

Measure	Time	INP	ALT	Matches	Accuracy
SURF	0.129	375	523	94	95.74
ORB	0.125	5820	6003	889	98.98
BRISK	0.449	766	1342	78	96.15

From Table II and Figs. 6, 7, 8 it can be said that ORB has the highest accuracy, more features extracted, and more matches than BRISK and SURF.

B. Rotations

We made the image rotate by 50 degrees. The Input and Output images can be compared in the figures. The Matching points made by each algorithm can also be visualised. As we can observe ORB made more matching points when compared to the rest of the algorithms.

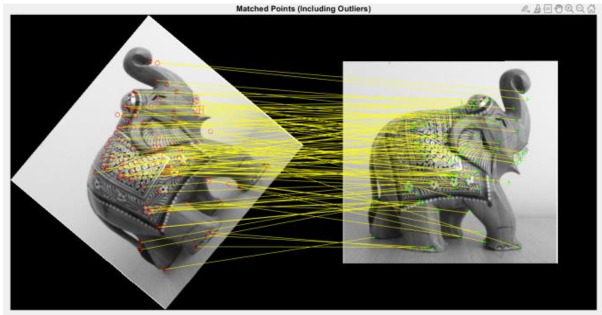

Fig. 7. SURF Matched Points

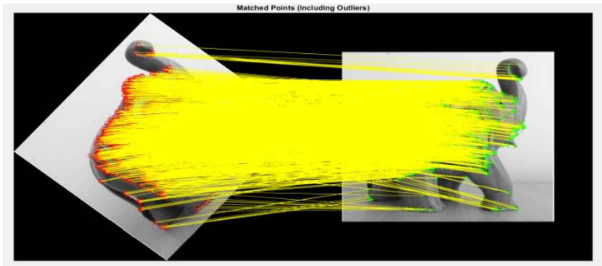

Fig. 8 ORB Matched Points

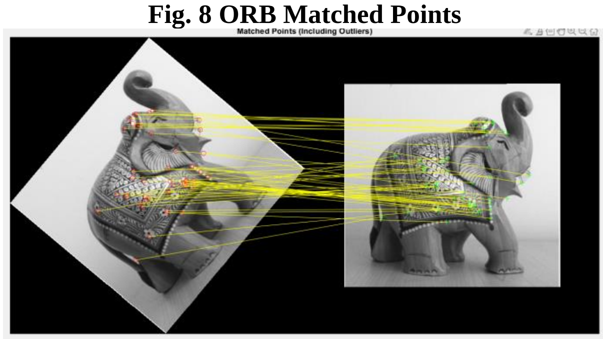

Fig. 9. BRISK Matched Points

Table- III: Rotation Measures

Measure	Time	INP	ALT	Matches	Accuracy
SURF	0.140	375	417	102	78.43
ORB	0.100	5820	7025	2279	87.31
BRISK	0.408	766	894	51	92.15

From Table III, and Figs. 7, 8, 9 it can be said that ORB has more advantages in time, features extracted and matches than SURF and BRISK. But BRISK has more accuracy.

C.Noise

We added 20\% (noise density) salt and pepper noise to the image. Fig. 11, 12, 13 can be seen with added noise and matched points by different algorithms.

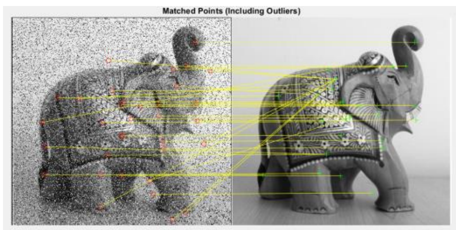

Fig. 10. SURF Matched Points

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

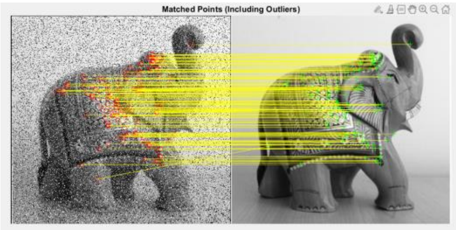

Fig. 11. ORK Matched Points

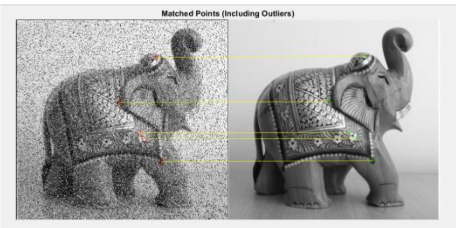

Fig. 12. BRISK Matched Points

Table- IV: Noise Measures

Measure	Time	INP	ALT	Matches	Accuracy
SURF	0.117	375	1811	34	41.22
ORB	0.119	5820	34307	147	65.30
BRISK	0.491	766	16565	6	83.78

From Table IV, it can be said that ORB has more advantages in, features extracted and matches than SURF and BRISK. But BRISK has more accuracy and SURF has an advantage in time.

V. CONCLUSION

In this paper, we used three different image matching techniques for object detection and compared accuracy using different kinds of transformations and deformations such as rotation and noise. For this purpose, we applied different types of transformations to original images/test images and compared the matching evaluation parameters such as the number of key points found in images, the matching rate of each image matching technique, and the execution time taken by each algorithm.

Feature Count Order: ORB > BRISK > SURF

Time Taken Order: BRISK > SURF > ORB

Matches Order: ORB > SURF > BRISK

Accuracy Order: BRISK > ORB > SURF

According to our comparisons, ORB is the most suitable among these three point feature matching algorithms for object detection.

REFERENCES

1. Bulbul, Farhad & Badsha, Md & Islam, Rafiqul. (2018). Object Detection by Point Feature Matching using Matlab. Advances in Image and Video Processing. 5. 10.14738/aivp.66.5619. [CrossRef]
2. Rublee, Ethan & Rabaud, Vincent & Konolige, Kurt & Bradski, Gary. (2011). ORB: an efficient alternative to SIFT or SURF. Proceedings of the IEEE International Conference on Computer Vision. 2564-2571. 10.1109/ICCV.2011.6126544. [CrossRef]
3. Liu Y, Zhang H, Guo H, Xiong NN. A FAST-BRISK Feature Detector with Depth Information. Sensors (Basel). 2018 Nov 13;18(11):3908. doi: 10.3390/s18113908. PMID: 30428580; PMCID: PMC6263410. [CrossRef]
4. Chuan Luo et al, Overview of Image Matching Based on ORB Algorithm, 2019 J. Phys.: Conf. Ser. 1237 032020 [CrossRef]

AUTHORS PROFILE

Suhas Reddy P is currently pursuing his bachelors in technology from the School of Computer Science & Engineering, Vellore Institute of Technology, Chennai. His main areas of research include Image Processing, Parallel Computing and Operating Systems. Currently focusing on Distributed Computing.

Bhargavi Rao, a final year undergraduate currently pursuing Computer Science Engineering at Vellore Institute of Technology Chennai. Focusing on my fundamentals like Data Structures, Algorithms, Database management systems, networks, formal theory. I have some experience in Java. As there is a lot to learn in technology, so currently I am exploring my interest in Graphic Designing, Image processing and UI/UX.

Jayanth Anala is currently pursuing his bachelors in technology from the School of Computer Science & Engineering, Vellore Institute of Technology, Vellore. His main areas of research include the Cybersecurity, Image Processing, Social Information Networks, Operating Systems. He is currently exploring cloud computing and NLP

Megha Dangayach, a final year undergraduate currently pursuing Computer Science Engineering Cyber Physical Systems at Vellore Institute of Technology Chennai. Her main areas of research include the Internet of Things, Networking, and Image Processing.

Retrieval Number: 100.1/ijitee.L930811111222
DOI: 10.35940/ijitee.L9308.11111222
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)