**Object Detection Using Different Point Feature Techniques**

Suhas Reddy

SCOPE, VIT Chennai

Chennai ,India

penubakasuhas.reddy2019@vitstudent.ac.in

Megha Dangayach

SCOPE, VIT Chennai

Chennai, India

Megha.dangayach2019@vitstudent.ac.in

B Bhargavi Rao

SCOPE,VIT CHENNAI

Chennai ,India

[bhargavirao.b2019@vitstudent.ac.in](mailto:bhargavirao.b2019@vitstudent.ac.in)

***ABSTRACT***

**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 important 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 summarises 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 utilised to conduct effective matching on diverse perspectives of the images/objects/scenes.**

**keywords*:* SURF, ORB, BRISK, feature extraction, feature matching, Rotation Invariance,Scaleinvaraince,estimateGeometricTransform2D, Affine, Inlier**

**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 imagebscale and rotation. It was then widely utilised in picture mosaicing, 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 distinguishing characteristics from 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. Following that, neighbouring areas are selected around each key point, and different feature descriptors are computed for each region. Image extraction characteristics in pictures that can give consistent matching across diverse views of the same image for image matching.

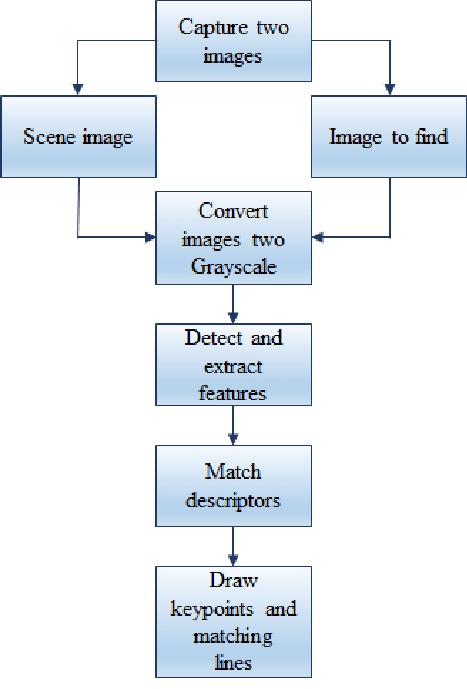
Feature descriptors are retrieved and saved from sample photos during the procedure. This descriptor must be unique while also being

resistant to noise and detection mistakes. Finally, the feature descriptors are matched across pictures.

**OVERVIEW OF IMAGE MATCHING TECHNIQUES:**

**SURF**

SURF uses box filters to approximate the DoG. 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 be done in parallel at various scales. To discover the sites of interest, the SURF employs a BLOB detector based on the Hessian matrix. It employs wavelet responses in both horizontal and vertical directions for orientation assignment, with appropriate Gaussian weights applied. SURF also use wavelet responses for feature description. SURF uses box filters to approximate the DoG. 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 be done in parallel at various scales. To discover the sites of interest, the SURF employs a BLOB detector based on the Hessian matrix. It employs wavelet responses in both horizontal and vertical directions for orientation assignment, with appropriate Gaussian weights applied. SURF also use wavelet responses for feature description.

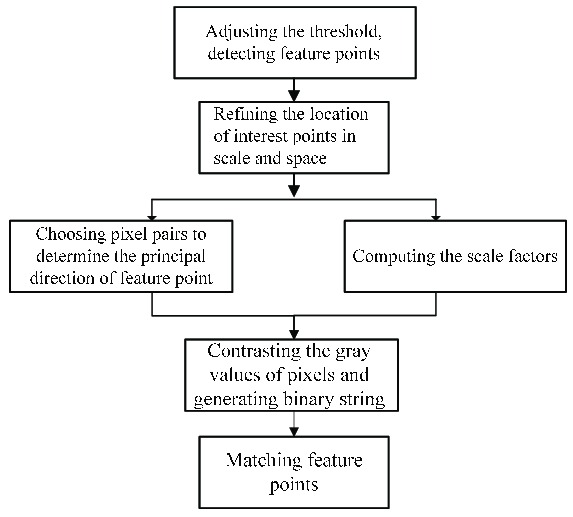


**ORB**

ORB is a modified fusion of the FAST key point detector and the BRIEF descriptor. FAST is used at first to find the main points. The top N spots are then determined using a Harris corner measure. FAST is rotation variant and does not compute orientation. It computes the patch's intensity weighted centroid with the centre corner identified. The orientation is determined by the direction of the vector from this corner point to the centroid. To increase rotation invariance, moments are calculated. If there is an in-plane rotation, the descriptor BRIEF performs badly.In ORB, a rotation matrix is constructed using the patch's orientation, and then the BRIEF descriptors are guided based on the orientation. A multiscale picture pyramid is used by the orb algorithm. An image pyramid is a multiscale depiction of a single image made up of sequences of images, each of which is a different resolution version of the image. Each level of the pyramid contains a downsampled replica of the preceding level's picture. After constructing a pyramid, orb employs the quick method to find keypoints in the picture. Orb is effectively finding important points at a different scale by detecting keypoints at each level. ORB is therefore partly scale invariant.

**BRISK**

BRISK's essential steps, namely feature identification, descriptor synthesis, and keypoint matching, to the degree of detail that a motivated reader can comprehend and duplicate It is crucial to note that the method's flexibility allows the BRISK detector to be used in conjunction with any other keypoint descriptor and vice versa, optimising for desired performance and the job at hand.



**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.

*Algorithm I*

Step 1: Read Images

Step 2: Detect Feature Points

Step 3: Extract Feature Descriptors

Step 4: Find Putative Point Matches

Step 5: Locate the Object in the Scene Using Putative Matches

Step 6: Display the detected object

Step 7: Detect Another Object

*Algorithm II*

Step 1: Read Images

Step 2: Detect Feature Points

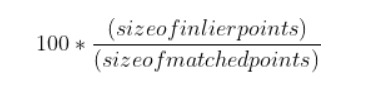
Step 3: Extract Feature Descriptors

Step 4: Find Putative Point Matches

Step 5: Find the matched points or features from the images

Step 6:Find the inlier points using the function estimateGeometricTransform2D

Step 7:Calculate the inlier points percentage from Step 6.



**SIMULATION RESULTS**

In this section we are going to see output and tabulate number of feature points from 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.

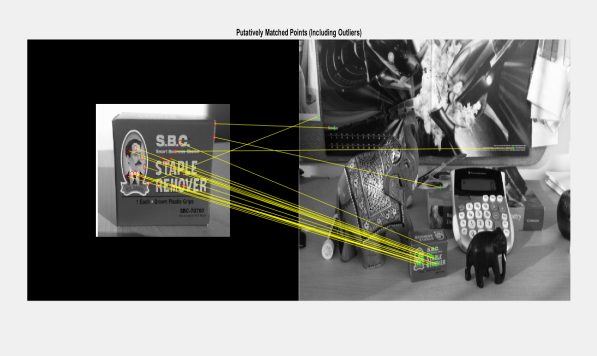


Figure a



Figure b

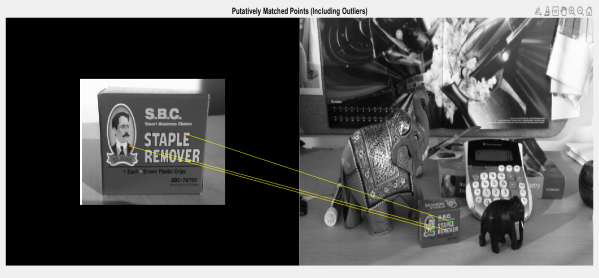


Figure c

Figure a:Surf Matched points, Figure b:ORB Matched points, Figure c:BRISK Matched points

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 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

From the above tabular column we can see that Orb has most features extracted from reference image and target image and Brisk has 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).

**Intensity:**

we made the brightness of image twice.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 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 |

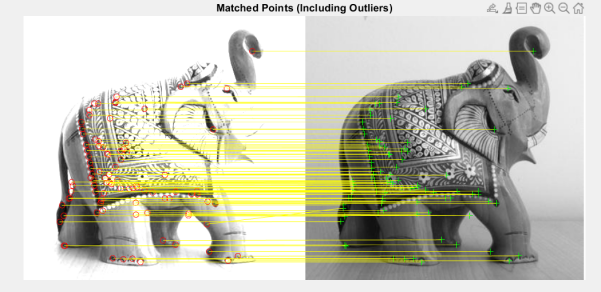


Figure d

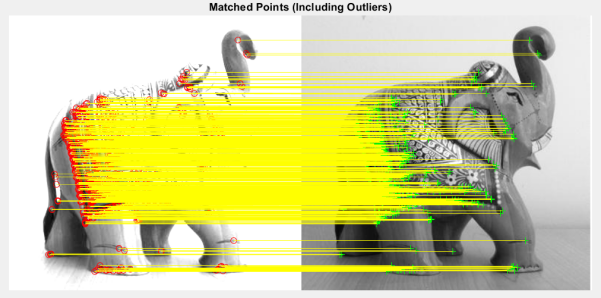


Figure e

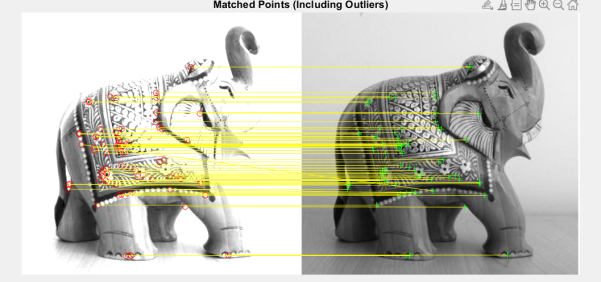


Figure f

Figure d:Surf Matched points, Figure e:ORB Matched points, Figure f:BRISK Matched points

From the table above, it can be said that Orb has the highest accuracy, more features extracted, and more matches than BRISK and Surf.

**Rotation:**

We made image rotated by 50 deg.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 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 |

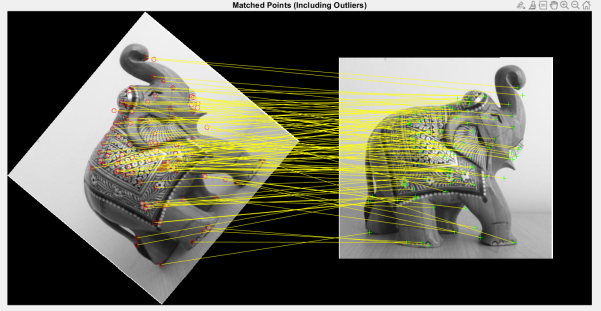


Figure g

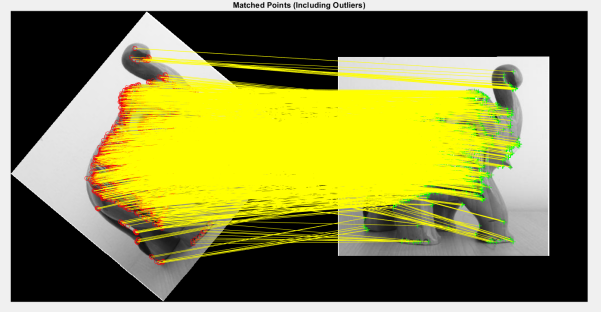


Figure h

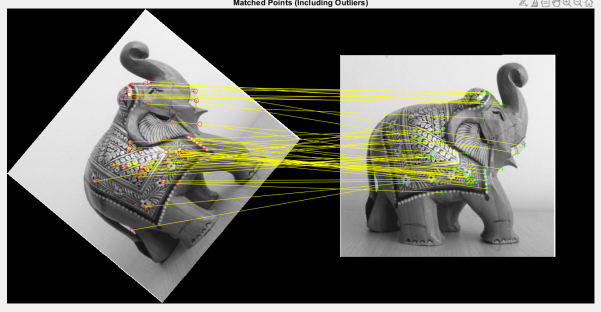


Figure I

Figure g:Surf Matched points, Figure h:ORB Matched points, Figure i:BRISK Matched points

From the tabulated results , it can be said that orb has more advantage in time , features extracted and matches than Surf and Brisk. But Brisk has more accuracy.

**Noise:**

We added 20% (noise density) salt and pepper noise to the image.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 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 |

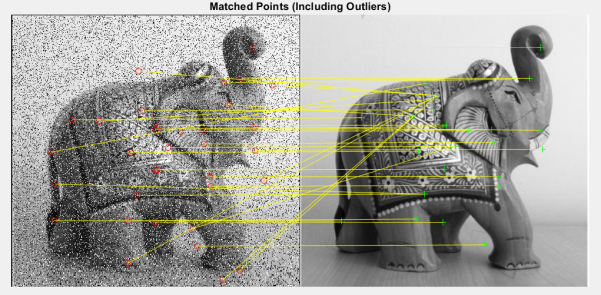


Figure j

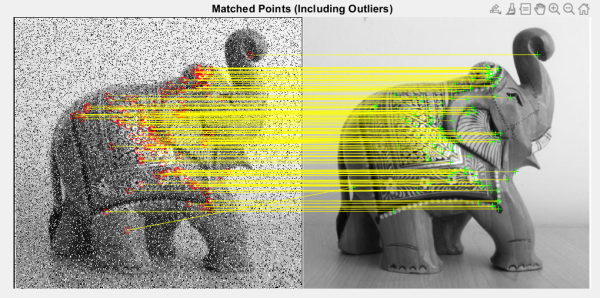


Figure k

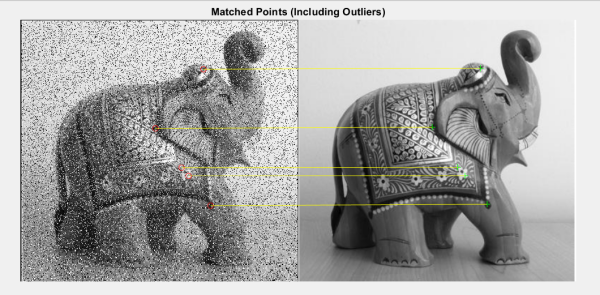


Figure l

Figure j:Surf Matched points, Figure k:ORB Matched points, Figure l:BRISK Matched points

From the tabulated results , it can be said that orb has more advantage in , features extracted and matches than Surf and Brisk. But Brisk has more accuracy and Surf has advantage in time.

**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 on original images and displayed the matching evaluation parameters such as the number of key points in images, the matching rate, and the execution time required for each algorithm.

In general,

Number of Features order:

WhatsApp Image 2021-11-26 at 6.50.23 PM

Time taken Order:

WhatsApp Image 2021-11-26 at 6.48.08 PM

Matches Order:

WhatsApp Image 2021-11-26 at 6.50.42 PM

Accuracy order:

WhatsApp Image 2021-11-26 at 6.52.35 PM

ORB is most suitable among these three point feature matching algorithm for object detection.

**REFERENCES**

<http://margaritachli.com/papers/ICCV2011paper.pdf>

<https://iopscience.iop.org/article/10.1088/1742-6596/1237/3/032020/pdf>

<https://www.researchgate.net/publication/330135867_Object_Detection_by_Point_Feature_Matching_using_Matlab>

<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6263410/>

<https://www.researchgate.net/publication/221111151_ORB_an_efficient_alternative_to_SIFT_or_SURF>

<https://arxiv.org/ftp/arxiv/papers/1710/1710.02726.pdf>

<https://towardsdatascience.com/improving-your-image-matching-results-by-14-with-one-line-of-code-b72ae9ca2b73>