Global Journal of Computer Science and Technology: F
Graphics & Vision
Volume 21 Issue 1 Version 1.0 Year 2021
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

Feature Matching with Improved SIRB using RANSAC

By Meet Palod, Manas Joshi, Amber Jain, Viroang Rawat & Prof. K.K. Sharma

Abstract- In this paper, we suggest to improve the SIRB (SIFT (Scale-Invariant Feature Transform) and ORB (Oriented FAST and Rotated BRIEF)) algorithm by incorporating RANSAC to enhance the matching performance. We use multi-scale space to extract the features which are impervious to scale, rotation, and affine variations. Then the SIFT algorithm generates feature points and passes the interest points to the ORB algorithm. The ORB algorithm generates an ORB descriptor where Hamming distance matches the feature points. We propose to use RANSAC (Random Sample Consensus) to cut down on both the inliers in the form of noise and outliers drastically, to cut down on the computational time taken by the algorithm. This post-processing step removes redundant key points and noises. This computationally effective and accurate algorithm can also be used in handheld devices where their limited GPU acceleration is not able to compensate for the computationally expensive algorithms like SIFT and SURF. Experimental results advocate that the proposed algorithm achieves good matching, improves efficiency, and makes the feature point matching more accurate with scale in-variance taken into consideration.

GJCST-F Classification: G.4

Feature Matching with Improved SIRB using RANSAC

Meet Palod [α], Manas Joshi [σ], Amber Jain [ρ], Viroang Rawat [Ѡ] & Prof. K.K. Sharma [¥]

Abstract- In this paper, we suggest to improve the SIRB (SIFT (Scale-Invariant Feature Transform) and ORB (Oriented FAST and Rotated BRIEF)) algorithm by incorporating RANSAC to enhance the matching performance. We use multi-scale space to extract the features which are impervious to scale, rotation, and affine variations. Then the SIFT algorithm generates feature points and passes the interest points to the ORB algorithm. The ORB algorithm generates an ORB descriptor where Hamming distance matches the feature points. We propose to use RANSAC (Random Sample Consensus) to cut down on both the inliers in the form of noise and outliers drastically, to cut down on the computational time taken by the algorithm. This post-processing step removes redundant key points and noises. This computationally effective and accurate algorithm can also be used in handheld devices where their limited GPU acceleration is not able to compensate for the computationally expensive algorithms like SIFT and SURF. Experimental results advocate that the proposed algorithm achieves good matching, improves efficiency, and makes the feature point matching more accurate with scale in-variance taken into consideration.

I. Introduction

Feature extraction and the matching continue to be the cornerstone of computer vision technologies. Essentially in feature detection, we abstract the image information into numbers and make a local decision at every point to see if there exists an image feature at that point. These local decisions taken at multiple spots are what we call interest points or key points. Ideally, techniques used for key point detection and matching should be impervious to different changes like - rotation, scale variance, illumination changes, noise and perspective changes.

Another desirable characteristic of these algorithms is that the key points generated should be unique to a greater degree to match a single feature with high rates of success.

The process of feature detection and matching can be broken down into some essential steps:

1. Detecting Interest points - It is done based on the brightness of the image or on the boundary extraction method.
2. Description of interest points - We generate a description vector for each feature point that describes the local appearance around every feature point that is invariant under changes in illumination, translation, scale, and in-plane rotation.
3. Matching the interest points across images - We match similar features across the images and map them, establishing a connection between two similar images.

Now, these general steps can be implemented using distinct Algorithms. To this extent, numerous algorithms have been proposed. The SIFT algorithm has been historically the most popular algorithm due to its application in several fields using visual features like object recognition, image stitching, visual mapping etc. The overhead of computational burden that accompanies SIFT led to an intensive search for its replacement.

Real-time systems and low-power devices like cell phones cannot bear that computational burden. SIFT generally performs better than SURF in computational cost and is the most accurate feature-detector-descriptor for scale, rotation and affine variations.

SURF detects more features than the SIFT algorithm and the features are detected in a scattered form generally all over the image. SURF has the highest computational cost for feature matching. Using SURF on low-end devices was not possible, and so a faster and more efficient algorithm was required.

ORB detects the highest number of features that are more concentrated on corners. In all practical fields of comparison ORB performs better than SURF and SIFT or any other algorithm for that matter. It is computationally the fastest and most efficient.

SIRB is the latest addition to this arsenal of keypoint detection and matching algorithms. The SIRB algorithm conflates the accuracy of the SIFT algorithm and combines it with the computationally efficient algorithm of ORB, thus making it fast as well as accurate while also being unaffected by scale, rotation and affine variations. We have proposed to take this one step further and improve on the existing SIRB(SIFT+ORB) algorithm using RANSAC(Random sample consensus). The idea behind using RANSAC is to reduce the number of interest points and thus eliminating the non matching points. This can potentially reduce more than 50% of the points and because the number of points to be matched have been reduced we can considerably reduce the matching time.

Author α σ ρ Ѡ ¥: e-mails: meetpalod18@gmail.com, manasj514@gmail.com, amberjain.98@gmail.com, viroangrawat@gmail.com, kkssgs@gmail.com

The RANSAC algorithm removes outliers or the data which does not fit the model. Its application in feature matching comes from the fact that it can handle more than 50% of the data as outliers, if applicable, and that has a drastic effect on time needed to generate a result. It is worth mentioning that RANSAC is a non-deterministic algorithm and that the results produced are reasonable only with a certain probability. This probability increases with the number of iterations. In RANSAC, we broadly perform these two steps iteratively:

1. Select a Minimal Sample Set(MSS) randomly from the input dataset and generate a model based on only these elements.
2. We now test which elements of the entire dataset are consistent with the model generated using the MSS. The set of elements which fit the model constitute a consensus set.

This process terminates when the probability of finding a better consensus set drops below a certain threshold.

The proposed method will reduce the number of points to be matched and offer a better alternative to the existing SIRB algorithm.

A feature or a keypoint is the distinctive piece of information, which is used for image matching, image stitching, and image registration. In an image, features are the specific structures like edges, corners, or objects. We detect corner points from an image which are also called interest points. An interest point in an image has a well defined position. These features extracted through the SIFT algorithm are invariant to rotation, scaling, and partially to illumination.

Keypoints are important because no matter how the image changes, we find the same key points in the modified image when comparing with the original image.

To detect keypoints, SIFT starts by generating the scale space for the image. Image at different scales is blurred by convolving a Gaussian kernel.

Scale-space is divided into octaves. In the original paper, Lowe [1] suggested that 4 octaves and 5 blur levels at each octave are ideal for the algorithm.

Each octave's image size is half of the previous octave and each image within an octave is increasingly blurred by a factor of k. The image is progressively blurred within an octave.

Mathematically, blurring is convolving the Gaussian operator and the image.

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \qquad (1)$$

Here, G is the Gaussian kernel, I is an image, x, y are the location coordinates and defines the amount of blur. A bigger value of implies more blur. Now with the blurred images, we create another set of images, known as Difference of Gaussians (DoG). This DoG images are used for finding keypoints in the image. It is obtained as the difference of two progressively blurred images in an octave. This process is done for all octaves of the image in the scale space, and the desired scale space is obtained. It is represented in below image:

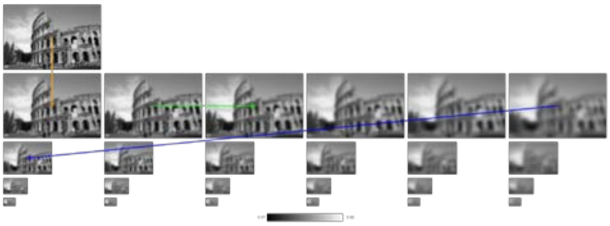

Fig. 1: Gaussian pyramid and Difference-of-Gaussian (DoG) pyramid

II. Local Maxima/Minima Detection

For computing local maxima or minima in the Difference of Gaussian, each point is compared to its eight neighbors in the image, 9 in the scale above and 9 in the scale below. Pixel is considered as a keypoint if its pixel intensity is more than all of the 26 neighboring points. Also, it is computationally efficient, because points are eliminated in the first few inspections.

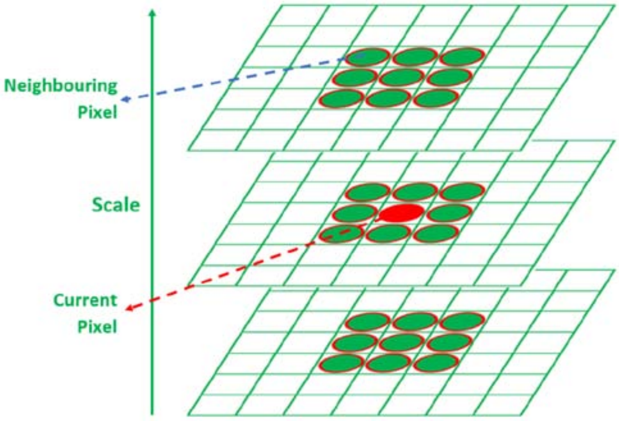

Fig. 2: Maxima and Minima are computed around the 3-dimensions x, y and σ.[15]

III. Key Point Localization Removing Keypoints with Low Contrast

After finding a possible keypoint by comparing it to neighboring pixels, the next task is to find datapoint location, scale, and ratio of principal curvatures. This process allows us to reject data points with low contrast which is sensitive to noise or localized along an edge. After it we fit a 3D quadratic function to the sample points to resolve the interpolated location of maximum and minimum. This approach uses Taylor expansion (2nd order) of the scale-space function, D (x, y, σ).

$$D(x) = D + \frac{\partial D^T}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 D}{\partial x^2} x \qquad (2)$$

Where D and the derivatives of D are calculated at the keypoint and x=(x, y, σ) T is the offset from this point. The correct location of keypoint extrema is x, and is evaluated by calculating the derivative of function D(x) concerning x and placing it to zero.

$$\hat{x} = -\frac{\partial^2 D^{-1}}{\partial x^2}\frac{\partial D}{\partial x} \tag{3}$$

The function value at extrema, (i.e. at $\widehat{x}$) D ($\widehat{x}$) is useful for removing keypoints with low contrast. The value D ($\widehat{x}$) can be calculated by substituting value of bx in D($\widehat{x}$).

$$D(\hat{x}) = D + \frac{1}{2}\frac{\partial D^T}{\partial x}x \tag{4}$$

Values less than 0.3 for the function value D($\widehat{x}$) are rejected. A final trial for removing key points/feature points on edges are performed because these are unnecessary points and will create redundant key points in the matching process. A poorly defined extrema located at the ridge in the DoG, which depicts the edge in the image has broader principle curvature across the ridge and a low one along with it which is not present in a blob or (corner) having broad principle curvature along with both directions. Principle of curvature is evaluated by Hessian Matrix H.

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \tag{5}$$

The eigenvalues corresponding to the Hessian matrix are tantamount to principle curvature. The ratio of eigenvalues λ1 and λ2 (which is proportional to principle curvature), are calculated with the help of Hessian matrix (H). This ratio is compared to certain threshold ratio r and high ratio points are repudiated. Where r = λ1/ λ2.

$$Tr(H) = D_{xx} + D_{yy} = \lambda_1 + \lambda_2 \tag{6}$$

$$Det(H) = D_{xx}D_{yy} - (D_{xx})^2 = \lambda_1\lambda_2 \tag{7}$$

$$\frac{Tr(H)^2}{Det(H)} = \frac{(\alpha+\beta)^2}{\alpha\beta} = \frac{(r\beta+\beta)^2}{r\beta^2} = \frac{(r+1)^2}{r} \tag{8}$$

Apply homogeneity to equations. The value of the term $(r+1)^2/r$ is minimum when both eigenvalues are equal, and it gradually increases with the value of r. Hence to check whether the ratio is below certain value, we used

$$\frac{Tr(H)^2}{Det(H)} = \frac{(\lambda_1+\lambda_2)^2}{\lambda_1\lambda_2} < \frac{(r+1)^2}{r} \tag{9}$$

We use the value of r =10 in this paper. Feature points above this threshold are repudiated.

IV. Orientation Assignment

Assigning orientation to keypoints ensures that it can be represented relative to its orientation and thus rotation invariance is achieved. To assign orientation, Histogram of Oriented Gradient (HOG) is used.

The scale of the keypoint is used to select the Gaussian smoothed image. Then, a 16x16 square window is considered around the detected keypoint.

For each pixel, L (x, y), at this scale, the gradient magnitude, m (x, y), and edge orientation, (x, y), is precomputed using pixel differences:

$$\theta(x,y) = tan^{-1}\left(\frac{L(x,y+1)-L(x,y-1)}{L(x+1,y)-L(x-1,y)}\right) \tag{10}$$

The edges below a threshold gradient magnitude are considered as weak edges and discarded. From the surviving edges, an orientation histogram is created. It has 36 bins covering the 360-degree range of orientations. Each pixel orientation added to the histogram is weighted by its gradient magnitude and by a Gaussian-weighted circular window with a that is 1.5 times that of the scale of the keypoint.

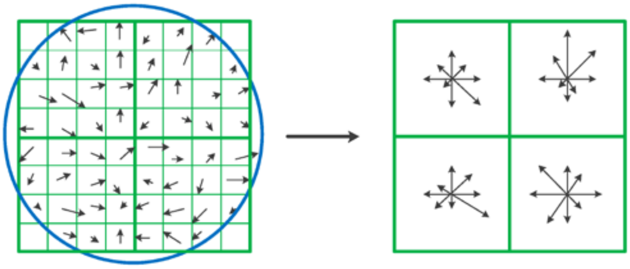

Fig. 3: Histogram of Gradient(HOG)[14]

Peaks in HOG correspond to the dominant orientation of the keypoint. If other peaks are detected within 80 percent of the dominant peak, then multiple keypoints are generated at the same location and scale with different orientation. These multiple orientation keypoint notably enhance the stability of matching.

V. rBRIEF: Rotation-Aware Briefs

After detecting all the key points, descriptors are generated for each keypoint. Feature descriptors encode unique information into a series of numbers and act like a numerical "fingerprint" that can be used to differentiate one feature from another. In this paper, we use the rBRIEF algorithm, a rotation-invariant version of the BRIEF algorithm. The BRIEF descriptor was proposed by M. Calonder [9]. BRIEF is an acronym for Binary robust independent elementary features, and it is robust to photometric and geometric image transformation.

The defined neighborhood, which is a square of some pixel width and height, around the keypoint is known as a patch. Brief depends on intensity difference tests to represent an image patch as a binary vector. Image patches could be effectively classified based on a relatively small number of pairwise intensity comparisons.

Each keypoint is described by a feature vector. Brief convert image patches into binary feature vectors, which are 128-512 bits string, so that together they can represent an object. The binary feature vector contains only 1's and 0's. Because BRIEF deals at the pixel level, it is very noise-sensitive. Hence it starts by smoothing the image with a Gaussian kernel. This reduces the sensitivity and also increases the stability and repeatability of the descriptors by pre-smoothing the patch.

To generate a 256-bit vector, it defines a randomly generated set of 256 pairs of pixels. The first pair in the random pair is taken from a Gaussian distribution drew around the keypoint with a standard deviation of 0.04 * S^2 (where S is the dimension of the patch). The second pair in the random pair is drawn from a Gaussian distribution drawn around the first pixel(x) with a standard deviation of 0.01 * S^2. rBRIEF achieves rotation invariance by moving these pixel locations by an angle equal to the orientation angle of the keypoint.

Then, it performs a binary test comparing the intensity of the first pixel in pair with the second pixel. If the intensity of the first pixel is less than the intensity of the second pixel, we append value 1 in the feature vector, else we append 0. When this binary test is performed on all the 256-pixel pairs in an image patch, we get a 256-bit binary vector, which is the desired descriptor.

VI. Ransac

RANSAC[8] algorithm is an extensively adapted mechanism in image processing for cleaning outliers from huge datasets. This algorithm helps in outlier removal, image noise filtration and optimal descriptors selection. RANSAC[6][7] does statistical estimations, which helps in evaluation for the likelihood of achieving accurate predictions.

RANSAC uses a minimal amount of data set to gain noise-repudiated data points, unlike other algorithms which initiate with huge datasets and then remove outliers.

Working of RANSAC:

- First, a minimum number of points are arbitrarily selected to define model parameters. From all the points, the points which fit according to the tolerance €(epsilon) are determined.
- Then we calculate the fraction of the number of inliers to total points present in the set.
- If the fraction is greater than a threshold (tau), which is predefined, we reevaluate parameters of the model using inliers.
- Else, we reiterate from step 1 to step 4, at most N times.

In this procedure, the value of N is taken large enough so that the minimum of one of the sets of random samples does not include any outlier. P represents probability which is set to 0.99, U represents chances of a selected data point as an inlier, V represents chances of selected data point as an outlier and N times the minimum number of points denoted M are required, so

$$1 - P = (1 - U^M)^N \tag{11}$$

After some formulations, N is chosen by the formula:

$$N = \frac{log(1-p)}{log(1-(1-V)^M)} \tag{12}$$

In our methodology, we aim to improve performance of SIRB by using RANSAC. RANSAC has been proven highly proficient in removing mismatched inconsistent sets of points. Since SIRB incorporates SIFT and ORB, therefore RANSAC has an intrinsic role in boosting the overall performance of SIRB. Due to the inaccuracy of SIFT, some points generated by it are mapped incorrectly, but RANSAC helps in filtering out these points, making SIFT more eficient. ORB [5] is an essential feature descriptor in low-end devices. So, it becomes crucial to eliminate shortcomings of ORB [12] and hence RANSAC is used as a post-quality improvement step to remove outliers and redundant key-points. RANSAC, thus makes SIRB more efficacious than some pre-existing methods, consequently making it an overall vigorous image recognition set-up.

VII. Match Feature Points Via Hamming Distance

The binary feature descriptor generated makes the matching process computationally more efficient because these binary vectors are stored in the form of bits. Performing operations like XOR on these binary vectors are much faster because of the ability to quickly compare descriptor pairs using few processor-level instructions. Algorithms like SIFT/SURF use Euclidean distance to compare binary feature vectors. We found that using hamming distance[11] for matching the key points is more efficient when comparing binary feature vectors. Therefore, we have used Hamming distance for our research.

The hamming distance is the value of the number of positions where both binary vectors differ. It is denoted by d(a, b), where a and b are two binary vectors. It can also be calculated by performing XOR operation on given two binary vectors.

$$d_{hamming}(a,b) = \sum_{i=0}^{n-1}(a_i \oplus b_i) \tag{13}$$

VIII. Conclusion

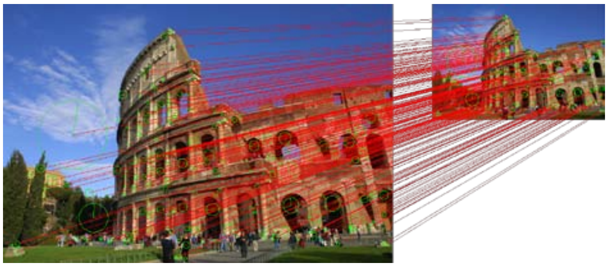

Fig. 4: Matching Result for Images with Scale Changes

We proposed a new algorithm, SIRB+RANSAC. It inherits the innate accuracy of the SIFT algorithm and the fast superiority of ORB. Its performance is further enhanced by using the RANSAC algorithm, which eliminates mismatches to a great extent. We start by describing the SIFT keypoint detection algorithm. We then introduced the rBrief method for binary descriptor generation. We used Hamming Distance to match the binary descriptors generated. Finally, we used the RANSAC algorithm to remove the mismatched keypoint pairs and eliminate the outliers.

The results of the experiment show that the new enhanced SIRB+RANSAC algorithm effectively resolves the problem that the ORB algorithm performs poorly in terms of scale invariance, and it is sensitive to image illumination. Based on the results of the experiment, we conclude that:

- The average matching accuracy of our algorithm is 89.85% as compared to the 86.95% accuracy of the traditional SIRB algorithm. The SIFT algorithm still presents an accuracy of 94.20%.
- The average computational time of the new improved SIRB+RANSAC algorithm is 87.69ms, which is about 42 times faster than the SIFT algorithm which takes an average of 3723ms. On the other hand, the SIRB algorithm takes 74.33ms.

The proposed algorithm will be exceptionally useful on low-end hand-held devices with low GPU power because it involves relatively less computational expenses than the prevalent algorithms. It also has applications in traditional systems where SIFT or SURF is currently being used.

IX. Future Work

It is evident from experimental results that the average matching time for SIRB is significantly less when compared with the SIFT algorithm. However, the matching accuracy has also dropped by 4.61%. We have released the source code of our algorithm so that it can be used by other researchers. There is the scope for improvement in our proposed algorithm, and these are the fields where future research can be directed:

- Instead of using Gaussian filters for convolution, filters like average filter and median filters can be used to reduce the run time of the algorithm while maintaining accuracy.
- Use improved RANSAC, modified RANSAC or optimal RANSAC to remove parts of the error feature point, thus increasing the proportion of correct matching features.
- Remove the repeatable and unstable key points generated after normalizing the scale space.

References Références Referencias

1. D. G. Lowe, Object recognition from local scale-invariant features, In Proceedings of the Seventh IEEE International Conference on Computer Vision, 1999, ICCV.1999.790410.
2. K. Yang and D. Yin and J. Zhang and H. Xiao and K. Luo, An Improved ORB Algorithm of Extracting Features Based on Local Region Adaptive Threshold, In 2019 6th International Conference on Systems and Informatics (ICSAI), 2019, ICSAI48974. 2019.9010209.
3. E. Rublee and V. Rabaud and K. Konolige and G. Bradski, ORB: An efficient alternative to SIFT or SURF, In 2011 International Conference on Computer Vision, 2011, ICCV.2011.6126544.
4. M. Zhao and H. Chen and T. Song and S. Deng, Research on image matching based on improved RANSAC-SIFT algorithm, In 2017 16th International Conference on Optical Communications and Networks (ICOCN), 2017, ICOCN.2017.8121270.
5. Y. Qin and H. Xu and H. Chen, Image feature points matching via improved ORB, In 2014 IEEE International Conference on Progress in Informatics and Computing, 2014, PIC.2014.6972325.
6. G. Shi and X. Xu and Y. Dai SIFT Feature Point Matching Based on Improved RANSAC Algorithm, In 2013 5th International Conference on Intelligent Human-Machine Systems and Cybernetics, 2013, IHMSC.2013.119.
7. O. Chum and J. Matas, Optimal Randomized RANSAC, In IEEE Transactions on Pattern Analysis and Machine Intelligence. 2008, TPAMI.2007.70787
8. S. Yang and B. Li, Outliers Elimination Based Ransac for Fundamental Matrix Estimation, In 2013 International Conference on Virtual Reality and Visualization, 2013, ICVRV.2013.63.
9. M. Calonder and V. Lepetit and M. Ozuysal and T. Trzcinski and C. Strecha and P. Fua, BRIEF: Computing a Local Binary Descriptor Very Fast, In IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012, TPAMI.2011.222.
10. Xue Leng and Jinhua Yang, Improvement of ORB algorithm, In Proceedings of the 2015 International Conference on Materials Engineering And Information Technology Applications, 2015 https://doi.org/10.2991/meita-15.2015.169.

11. M. Su and Y. Ma and X. Zhang and S. Li and Y. Zhang A Binary SIFT Matching Method Combined with the Color and Exposure Information, In 2017 International Conference on Network and Information Systems for Computers (ICNISC), 2017, IC-NISC.2017.00062.
12. F. Tian and P. Shi, Image Mosaic using ORB descriptor and improved blending algorithm, In 2014 7th International Congress on Image and Signal Processing, 2014, CISP.2014.7003867.
13. Wei He and Takayoshi Yamashita and Hongtao Lu and Shihong Lao, SURF Tracking, In 2009 IEEE 12th International Conference on Computer Vision, 2009, ICCV.2009.5459360.
14. Zhang Y, Wang C, Zhou X. RST Resilient Watermarking Scheme Based on DWT-SVD and Scale-Invariant Feature Transform. Algorithms. 2017; 10(2):41. https://doi.org/10.3390/a10020041.
15. Lowe, D.G. Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision 60, 91–110 (2004). https://doi.org/10.1023/B:VISI.0000029664.99615.94.