SAR IMAGE CORRELATION

Contents

***Abstract*—There are many ways to correlate SAR images among them the popular method is object detection here we will see some other methods for correlating two SAR images. We will see how to remove multiplicative noise on SAR images, and after that we will see how the SIFT algorithm and SAR-SIFT algorithm works and the accuracy of both SIFT and SAR-SIFT algorithm and some methods like feature extraction to correlate SAR images.**

I. INTRODUCTION

Synthetic aperture radar (SAR) images are produced by an active system that sends a microwave signal from a sensor platform to the ground and detects backscattered waves that the ground reflects directly back to a receiver on the same platform, which can be borne aloft by either airplanes or satellites so any given image will have noise in it.

There are many applications for SAR image correlation, including mapping, monitoring, and surveillance. By analyzing the correlation between pairs of SAR images, it is possible to identify changes in the Earth's surface over time, such as the growth or decline of vegetation, the construction or destruction of buildings, or the movement of vehicles or other objects.

Noise is a common problem in synthetic aperture radar (SAR) images, and its presence can significantly impact the quality and usefulness of the images. Noise can be introduced into SAR images due to a variety of factors, such as interference from other sources, electrical or mechanical noise in the imaging system, or atmospheric conditions.

The removal of noise from SAR images is an important task that can improve the accuracy and reliability of the images for a variety of applications, such as mapping, monitoring, and surveillance. There are many different approaches that can be used to remove noise

One common approach to noise removal in SAR images is the use of image filters, which can smooth or blur the image to reduce the impact of noise. Filters can be designed to preserve important features in the image while reducing the impact of noise, and they can be applied using various techniques such as convolution or median filtering.

By using the SIFT algorithm to extract and match features in pairs of SAR images, it is possible to identify and measure the similarity between the images and to align them accurately. It is able to identify features in the images that are invariant to scale, orientation, and affine distortion. This means that the algorithm is able to identify and match features in the images even if they are scaled, rotated, or distorted in some way.

II. LITERATURE SURVEY

Some methods which are used to correlate SAR images

Object detection

It is a computer vision technique for locating instances of objects in images. It uses deep learning and some methods are:

CNN, RCNN, Fast R CNN, Faster R CNN, Histogram of Oriented gradients, Single shot detector, Spatial Pyramid Pooling (SPP NET), YOLO [1]

SIFT

SIFT (Scale-Invariant Feature Transform) is a feature detection and description method. It first detects distinctive points in the image, known as keypoints. It then describes the local appearance of the keypoints using a feature descriptor and matches the images.[2]

SAR-SIFT

SAR-SIFT is a variant of SIFT that was developed to improve the performance of SIFT in remote sensing images. SIFT descriptor allows it to better distinguish between different objects in the image. [2][3]

Hough Line Detector

The Hough line detection algorithm is a feature detection algorithm used to identify lines in an image. The algorithm then searches for clusters of points in the parameter space, which correspond to lines in the original image.[2]

Maximally Stable Extremal Regions

The Maximally Stable Extremal Regions (MSER) algorithm is a feature detection. The MSER algorithm works by constructing a scale space representation of the image, which is a series of images at different scales obtained by convolving the original image with a series of Gaussian kernels [2]

Cross Correlation

Cross-correlation is a statistical measure of the similarity between two signals or images. It involves comparing the two images pixel-by-pixel and calculating the similarity between the pixels at each location. [5][4]

Sparse representation-based algorithm

In the context of SAR images, sparse representation-based algorithms can be used to extract and analyze features in the images, such as edges, corners, or patterns. By representing the data in a sparse format, these algorithms can identify and analyze the most important or distinctive features in the images, and can effectively filter out noise and other artifacts.

Feature extraction-based algorithm

The performance of two feature extraction techniques is analyzed with respect to classification. The extracted features are grouped into different classes using classification techniques such as the k-means and fuzzy c means (FCM) clustering and are two unsupervised classification technique.[6]

III. METHODOLOGY

A. *Filters to remove noise*

1. *Mean Filter:*

Mean filtering is easy to implement. It is used as a method of smoothing images, reducing the amount of intensity variation between one pixel and the next resulting in reducing noise in images.

Fig. 1. Above picture is before and after mean filter is used

The idea of mean filtering is simply to replace each pixel value in an image with the mean (`average') value of its neighbours, including itself. This has the effect of eliminating pixel values which are unrepresentative of their surroundings. Mean filtering is usually thought of as a convolution filter. Like other convolutions it is based around a kernel, which represents the shape and size of the neighbourhood to be sampled when calculating the mean. Often a 3×3 square kernel is used I/9

Where I is identity matrix It has some effect on the salt and pepper noise. It just made them blurred.[12]

2. *Median filter*:

Unlike the previous filter which is just using mean value, this time we used median. Median filtering is a nonlinear operation often used in image processing to reduce "salt and pepper" noise.[12]

Fig. 2. Above picture is before and after median filter is used

B. SIFT Algorithm

SIFT, or Scale Invariant Feature Transform, is a feature detection algorithm in Computer Vision. SIFT helps locate the local features in an image, commonly known as the ‘keypoints’

of the image. These keypoints are scale & rotation invariant. We can also use the keypoints generated using SIFT as features for the image during model training.[10]
Broadly speaking, the entire process can be divided into 5 parts:
•Constructing a Scale Space: To make sure that features are scale-independent
•Keypoint Localisation: Identifying the suitable features or keypoints
•Orientation Assignment: Ensure the keypoints are rotation invariant
•Keypoint Descriptor: Assign a unique fingerprint to each keypoint
•Keypoint Matching: We can use these keypoints for feature matching.

1. *Constructing the Scale Space:*

We need to identify the most distinct features in a given image while ignoring any noise. Additionally, we need to ensure that the features are not scale-dependent.
Noise removal methods such as Mean Filter can successfully remove the noise from the images. Now, we need to ensure that these features must not be scale-dependent. This means we will be searching for these features on multiple scales, by creating a 'scale space'.
Scale space is a collection of images having different scales, generated from a single image. These blur images are created for multiple scales. To create a new set of images of different scales, we will take the original image and reduce the scale by half. For each new image, we will create blur versions

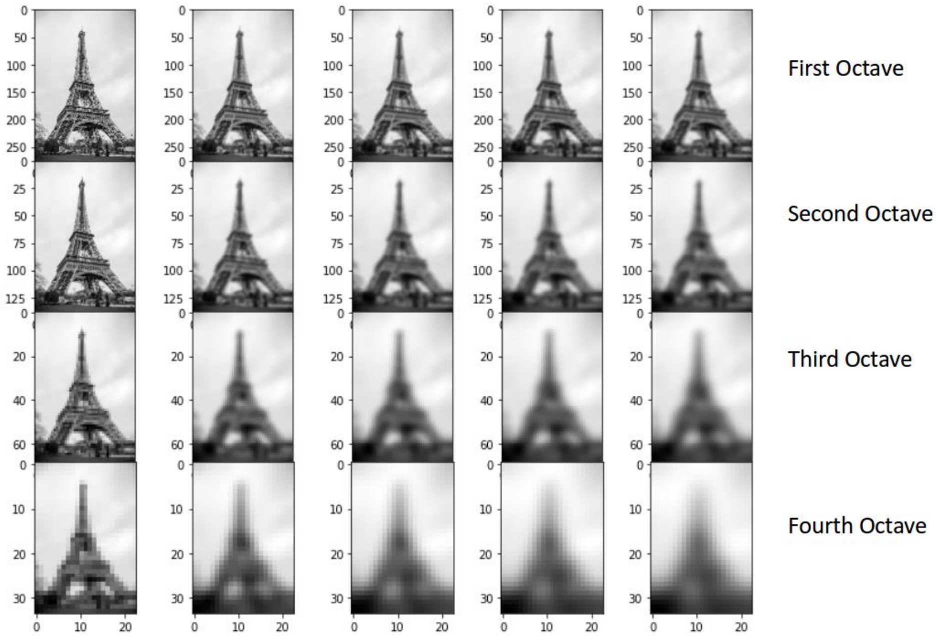


Fig. 3. We get this images by scaling and blurring

Difference of Gaussian (DoG)

We have created images of multiple scales and used Mean Filter for each of them to reduce the noise in the image. Next, we will try to enhance the features using a technique called Difference of Gaussians or DoG.
Originally, people used Laplacian of Gaussian to find the keypoints instead of DoG, but due to computational complexity people shifted to DoG. Laplacian of Gaussian approximates to scaling version of DoG.
Difference of Gaussian is a feature enhancement algorithm that involves the subtraction of one blurred version of an original image from another, less blurred version of the original.

DoG creates another set of images, for each octave, by subtracting every image from the previous image in the same scale.

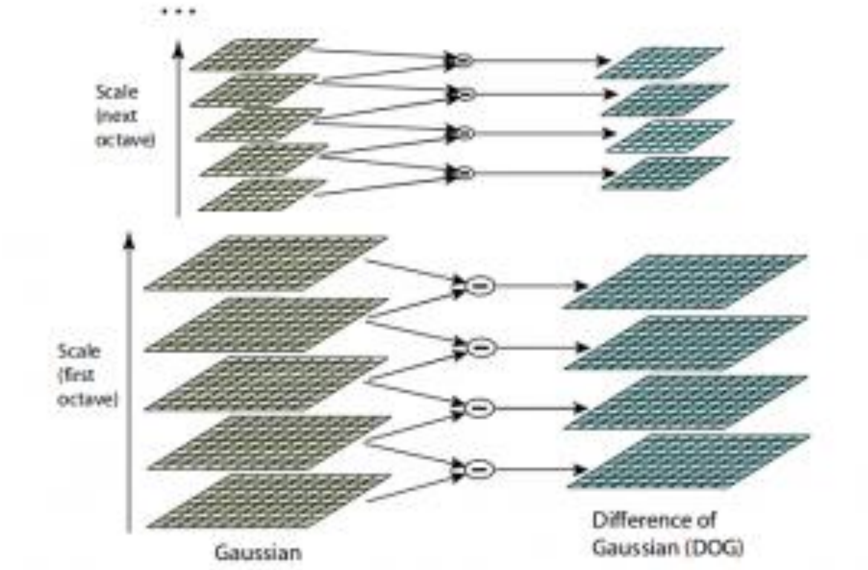


Fig. 4. Difference of Gaussian

Let us create the DoG for the images in scale space. Take a look at the below diagram.
On the left, we have 5 images, all from the first octave (thus having the same scale). Each subsequent image is created by applying the Gaussian blur or Mean Filter over the previous image.
On the right, we have four images generated by subtracting the consecutive Gaussians.

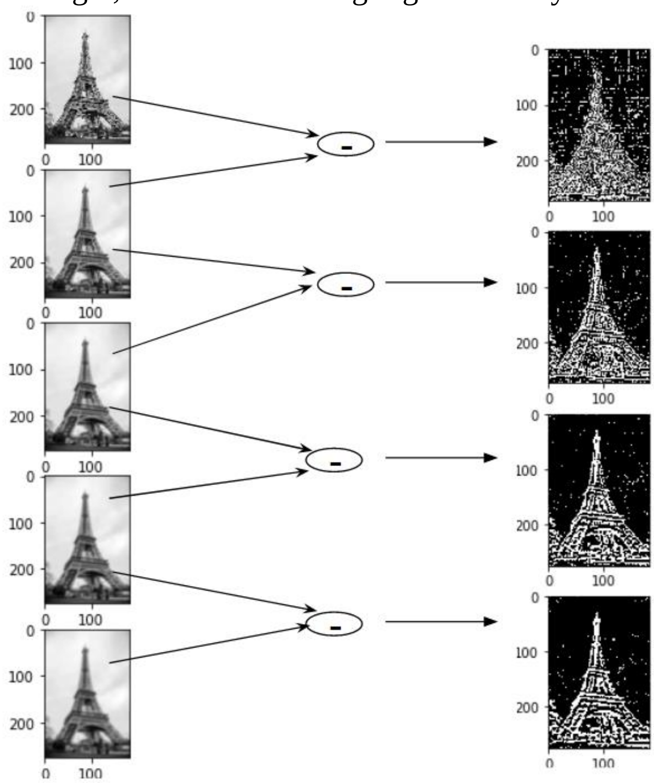


Fig. 5. Difference of Gaussians on the given image

2. *Keypoint Localization*

Once the images have been created, the next step is to find the important keypoints from the image that can be used for feature matching. The idea is to find the local maxima and minima for the images.

To locate the local maxima and minima, we go through every pixel in the image and compare it with its neighboring pixels.[10]

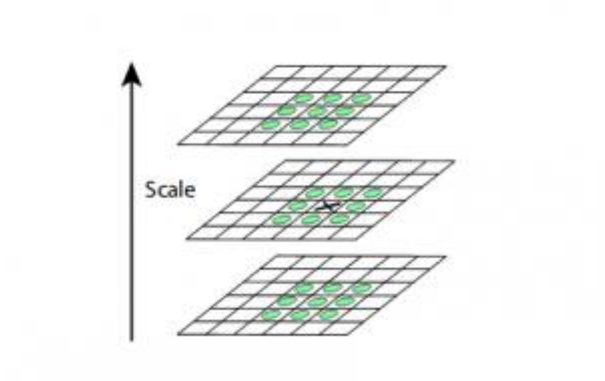


Fig. 6. Scale Space

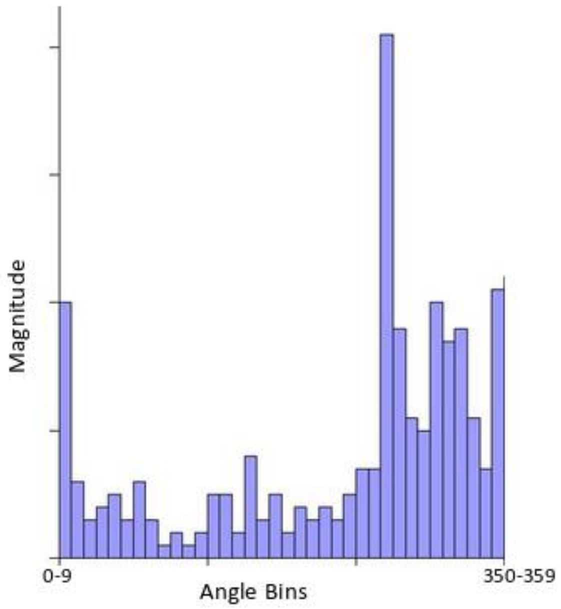
We now have keypoints that represent the images and are scale-invariant.

3. *Orientation Assignment:*

At this stage, we have a set of stable keypoints for the images. We will now assign an orientation to each of these keypoints so that they are invariant to rotation.

Creating a Histogram for Magnitude and Orientation

On the x-axis, we will have bins for angle values, like 0-9, 10 – 19, 20-29, up to 360.The histograms are represented like



This histogram would peak at some point. The bin at which we see the peak will be the orientation for the keypoint.

4. *Keypoint Descriptor:*

This is the final step for SIFT. So far, we have stable keypoints that are scale-invariant and rotation invariant. We will use the neighbouring pixels, their orientations, and magnitude, to generate a unique representation for this keypoint called a ‘descriptor’.

Since we use the surrounding pixels, the descriptors will be partially invariant to illumination or brightness of the images.
We will first take a 16×16 neighbourhood around the keypoint. This 16×16 block is further divided into 4×4 sub-blocks and for each of these sub-blocks, we generate the histogram using magnitude and orientation.

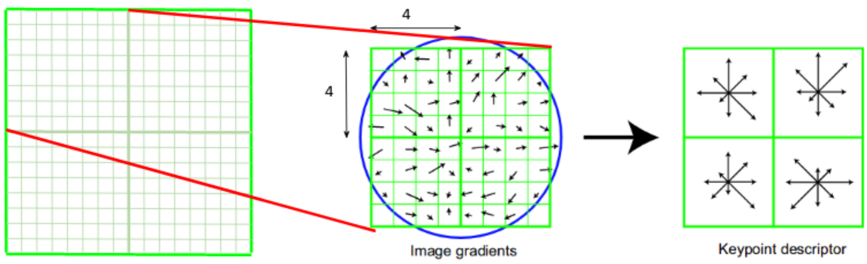


Fig. 7. Orientations of the pixels

5. *Keypoint Matching:*

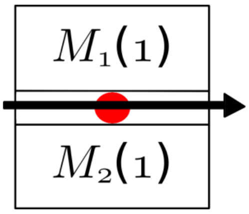
Keypoints between two images are matched by identifying their nearest neighbours.[9][11]

C. SAR-SIFT ALGORITHM

There are four main steps involved in SAR-SIFT algorithm
•Keypoint Detection
•Orientation Assignment
•Descriptor Assignment
•Keypoint Matching
Here in SAR-SIFT algorithm we define a new gradient function known as gradient ratio approach for computing above mentioned steps,

Gradient Ratio approach

Many works on edge detection have underlined the problem of using gradient by difference on SAR images. Traditional approaches in edge detection consist in thresholding the gradient magnitude.
For SAR images, this leads to higher false alarm rates in homogeneous areas of high reflectivity than in the ones of low reflectivity. The classical gradient by difference is thus not a constant false alarm rate operator.
Statistical studies have shown that the use of ratio is more suitable to multiplicative noise than the use of difference. Several edge detectors using ratio have been introduced in order to obtain a constant false alarm rate on SAR images
The Ratio of Average (ROA) consists in computing the ratio of local means on opposite sides of the studied pixel along one direction i



(a) Scheme of the ratio of local means for the first direction.

$$R_i = \frac{M_1(i)}{M_2(i)}$$

The ratio R_i is then normalized

The gradient magnitude and orientation are defined respectively as

$$D_n^1 = max_i(T_i)$$

$$D_t^1 = (argmax_i(T_i) - 1) \times \frac{\pi}{4}$$

Edges may then be obtained by thresholding the gradient magnitude.

Those operators have been designed for edge detection and provide a good estimate of the gradient magnitude.

The Ratio of Exponentially Weighted Averages (ROEWA) is an improvement of the ROA for a multi-edge context, obtained by computing exponential weighted local means

$$M_{1,\alpha}(1) = \int_{x=R}\int_{y=R^+} I(a+x, b+y) \times e^{-\frac{|x|+\alpha|y|}{\alpha}}$$

$$M_{2,\alpha}(1) = \int_{x=R}\int_{y=R^-} I(a+x, b+y) \times e^{-\frac{|x|+\alpha|y|}{\alpha}}$$

Where α is the weighted parameter

If i = 1 it is horizontal, if i = 3 it is vertical. We use this because it allows an adapting smoothing of the image

Proposed approach:

We propose here to define the horizontal and vertical gradient as:

$$G_{x,\alpha} = log(R_{1,\alpha})$$

$$G_{y,\alpha} = log(R_{3,\alpha}) \quad (8)$$

and to compute the gradient magnitude and orientation in the usual way as:

$$G_{n,\alpha} = \sqrt{(G_{x,\alpha})^2 + (G_{y,\alpha})^2}$$

$$G_{t,\alpha} = \arctan\left(\frac{G_{y,\alpha}}{G_{x,\alpha}}\right) \qquad (9)$$

We call this new gradient computation method Gradient by Ratio (GR).

1. *Keypoint Detection*

The multi-scale Harris matrix and function are defined respectively as

$$C(x,y,\sigma) = \sigma^2 \mathcal{G}_{\sqrt{2}\sigma} \times \begin{bmatrix} (\frac{\partial I_\sigma}{\partial x})^2 & \frac{\partial I_\sigma}{\partial x} \times \frac{\partial I_\sigma}{\partial y} \\ \frac{\partial I_\sigma}{\partial x} \times \frac{\partial I_\sigma}{\partial y} & (\frac{\partial I_\sigma}{\partial y})^2 \end{bmatrix}$$

$$R(x,y,\sigma) = \det(C(x,y,\sigma) - t \times tr(C(x,y,\sigma))$$

with $\mathcal{G}$ as a Gaussian kernel

By adapting the parameters on the multi-scale Harris criterion, the number of false detections can be decreased but so will the number of correct ones.
LoG and Hessian matrices do not seem convenient and easy to adapt to multiplicative noise since they rely on second derivatives
The convolution operator, I_σ the convolution of the original image by a gaussian kernel with standard deviation σ and t an arbitrary parameter
Considering this definition and the Gradient by Ratio, we propose the new SAR-Harris matrix and the multi-scale SAR-Harris function respectively as

$$C_{SH}(x,y,\sigma) = \mathcal{G}_{\sqrt{2}\sigma} \times \begin{bmatrix} (G_{x,\alpha})^2 & (G_{x,\alpha}) \times (G_{y,\alpha}) \\ (G_{x,\alpha}) \times (G_{y,\alpha}) & (G_{y,\alpha})^2 \end{bmatrix}$$

$$R_{SH}(x,y,\sigma) = det\,((C_{SH}(x,y,\sigma)) - d \times tr(C_{SH}(x,y,\sigma))$$

with d as an arbitrary parameter, and where the derivatives $G_{x,\alpha}$ and $G_{y,\alpha}$ are computed using this approach, called the SAR-Harris method, merges the two steps of the LoG method in order to avoid the use of second order derivatives
Several detections can then occur at the same position but for different scales. However, some of them are suppressed by thresholding the multi-scale SAR-Harris function.

2. *Orientations Assignment and Descriptors Extraction*

In the original SIFT algorithm, both the steps of orientation assignment and descriptor extraction rely on histograms of gradient orientation.
These histograms are computed on a neighbourhood of each keypoint and weighted by the gradient magnitude. Here we propose to use the Gradient by Ratio (GR) method (as discussed above), to compute those histograms.
The resulting descriptor is called Ratio Descriptor.

3. *Keypoint matching*

It is same as SIFT algorithm that is it uses nearest neighbouring points to detect the similar keypoints in two images

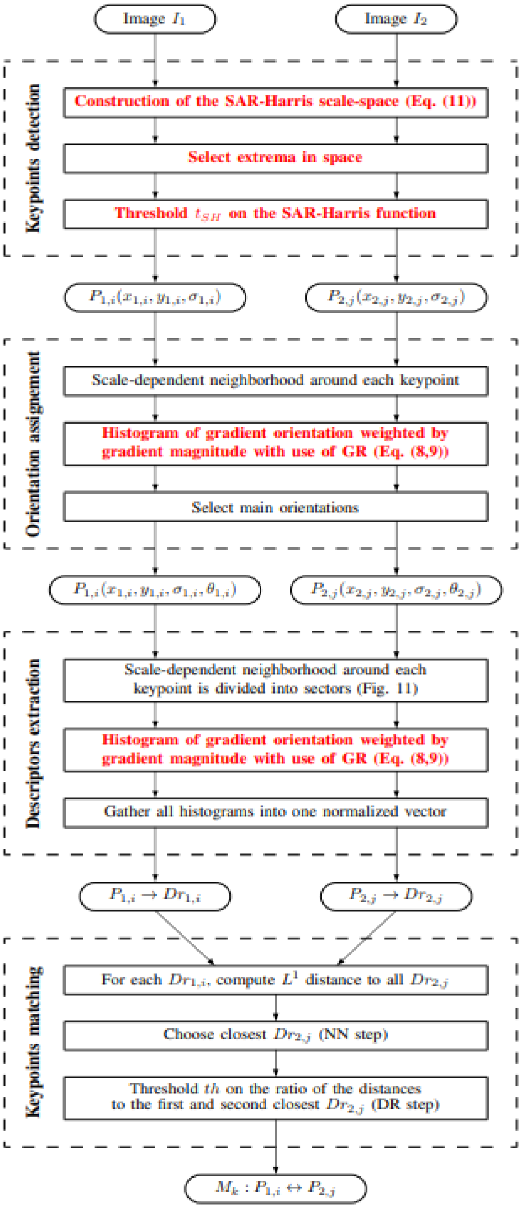


Fig. 8. Key differences of SIFT and SAR-SIFT methods

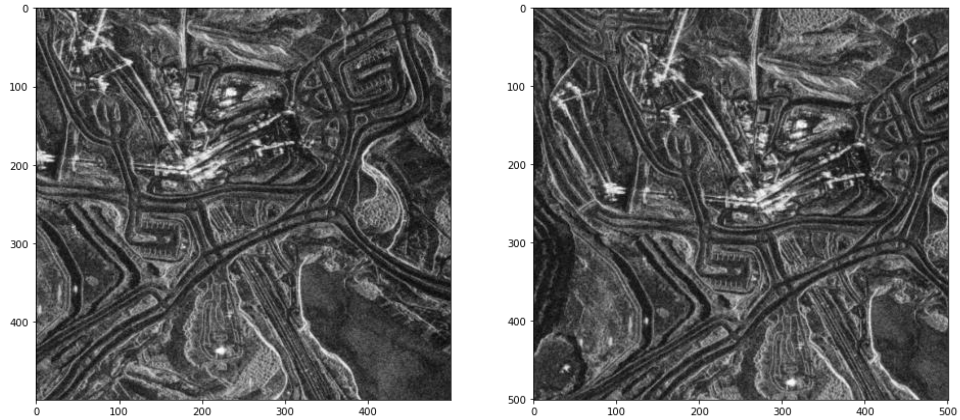
Here, the points in red are the points that are modified from SIFT algorithm to make the algorithm more suitable for SAR images and we call the modified algorithm as SAR-SIFT algorithm[7][8]

IV. EXPERIMENTAL RESULTS

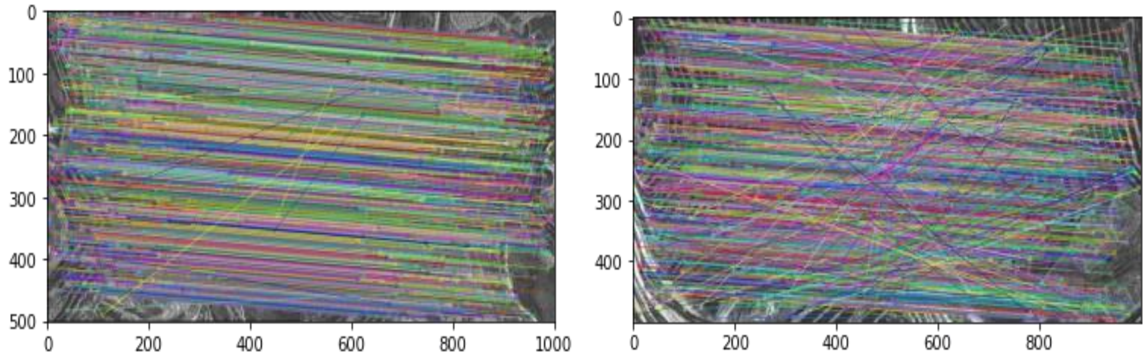
I have constructed dataset of 50 images to check out the mean accuracies of the above algorithms

MEAN OF SIFT ALGORITHM ACCURACY = 77.13%
MEAN OF SAR-SIFT ALGORITHM ACCURACY = 95.54%

Some of those image pairs are listed below
Example 1:



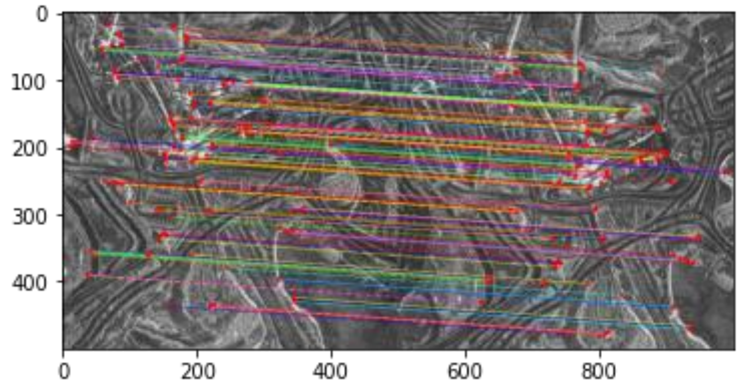
SIFT



m1=2200, m2=2567

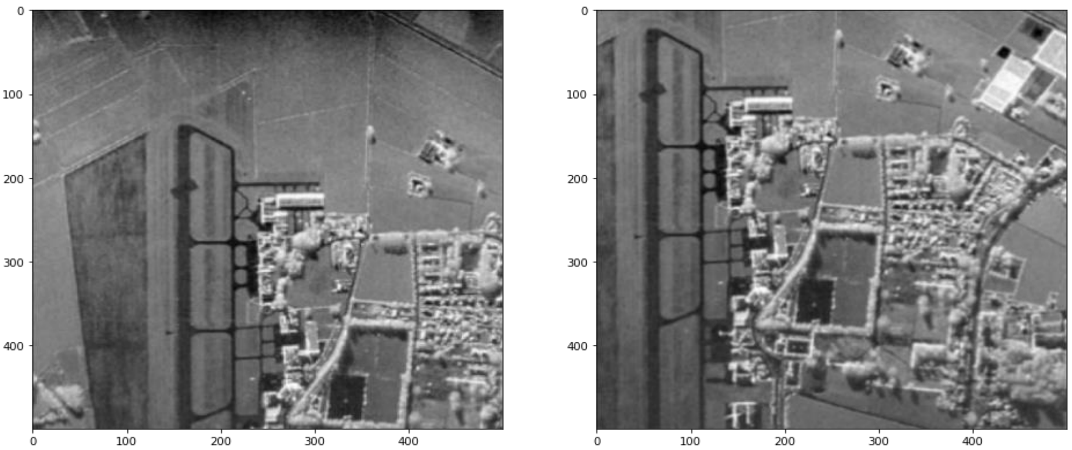
f1= 4384, f2=4594, tm=2567, a=85.03%

SAR-SIFT

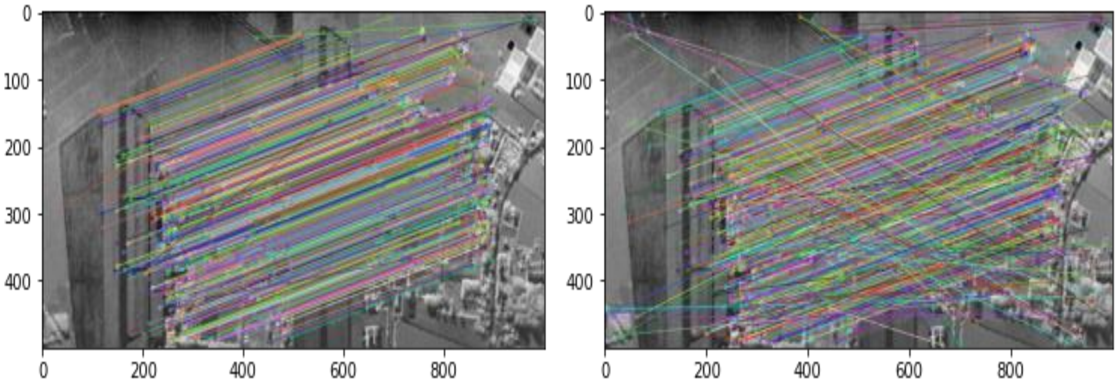


f1=525,f2=473,tm=155,a=99%

Example 2:



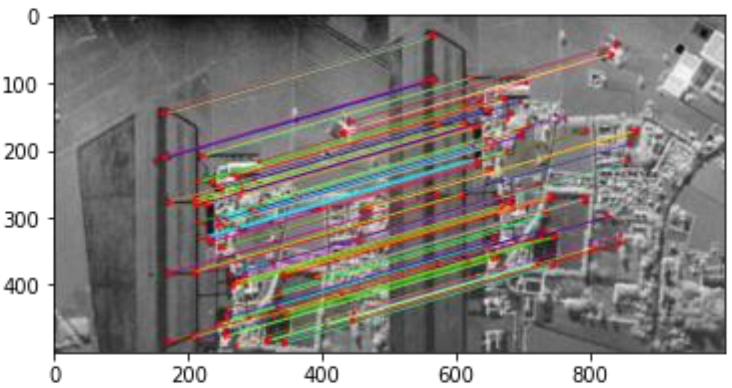
SIFT



m1=1050, m2=1170

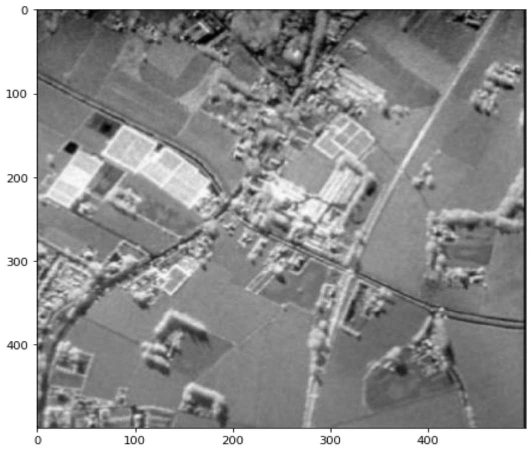
f1= 1740, f2=2429, tm=1170, a=89.74%

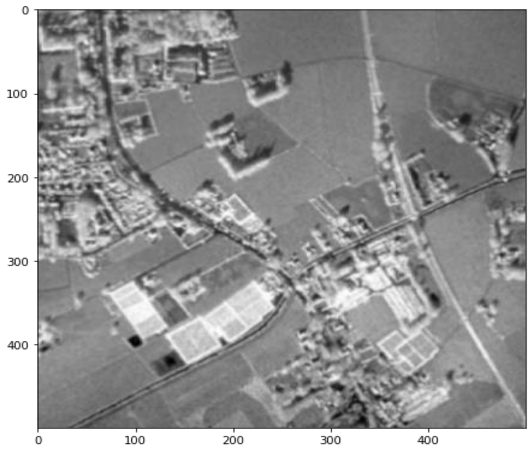
SAR-SIFT



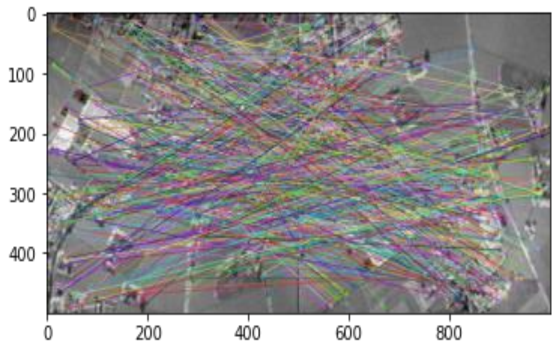
f1=430, f2=672, tm=189, a=98%

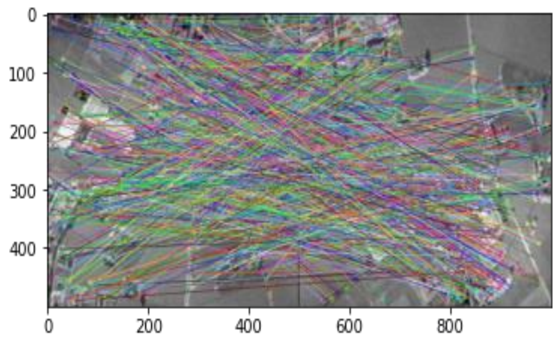
Example 3:





SIFT

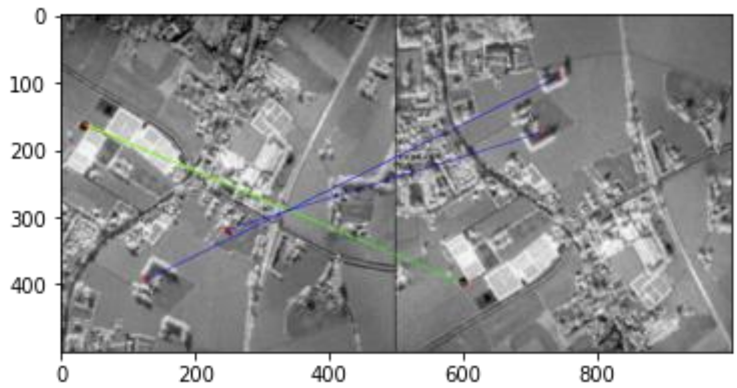




m1=600, m2=873

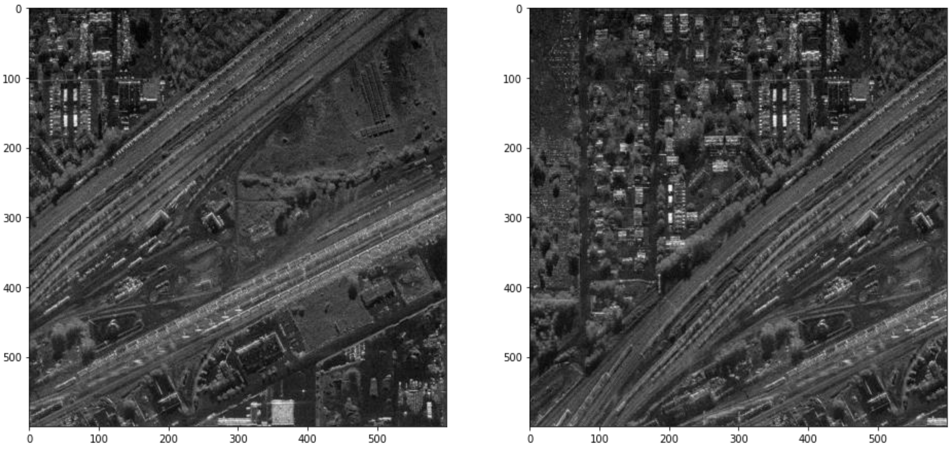
f1= 2723, f2=2630, tm=873, a=68.72%

SAR-SIFT

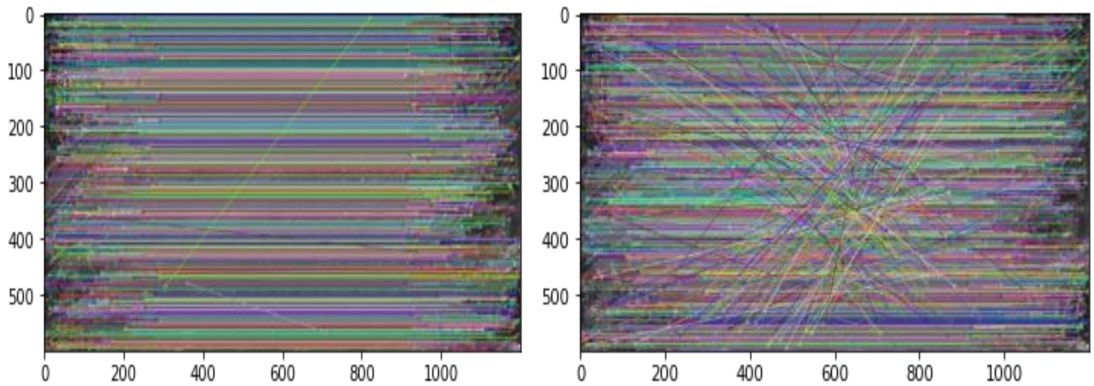


f1=477, f2=447, tm=4, a=75%

Example 4:



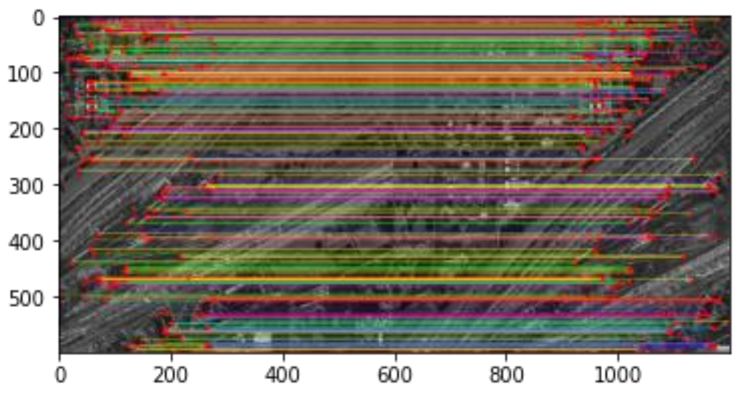
SIFT



m1=600, m2=757

f1= 978, f2=1222, tm=757, a=79.27%

SAR-SIFT



f1=222, f2=307, tm=112, a=100%

m1 = number of mapped features shown in combined image

m2 = number of mapped features shown in combined image

f1 = features points in 1st image

f2 =features points in 2nd image

tm = total number of features mapped in two images

a = accuracy of two images = m1/tm

V. CONCLUSION

Feature extraction is only suitable for correlation of two large SAR images because it only identifies large objects like water bodies, forest etc and is not suitable for correlation of small objects.

Hough line detector algorithm only detects any line that is present in SAR image and is not suitable for correlation

SIFT algorithm and SAR-SIFT algorithm is more suitable correlation of SAR images than any other correlation method

From this research,

Algorithm	Properties of object	Presence of Noise	Accuracy
SIFT	It does not give good results on objects with high reflectivity[7]	It works well on images with less noise[7]	77.13%
SAR-SIFT	It works on objects with high reflectivity also[7]	It works well even on images with speckle noise[7]	95.54%

VI. REFERENCES

[1]https://analyticsindiamag.com/top-8-algorithms-for-object-detection

[2]https://pdfs.semanticscholar.org/6856/c6025ddcd1ef63a6634704de6eaa14c37ab0.pdf

[3]https://towardsdatascience.com/sift-scale-invariant-feature-transform-c7233dc60f37

[4]https://www.isprs.org/proceedings/xxxvi/8-w27/tupin.pdf

[5]https://ieeexplore.ieee.org/document/6350961

[6]https://www.ijert.org/performance-analysis-of-feature-extraction-techniques-applied-on-sar-data

[7]https://perso.telecom-paristech.fr/tupin/PUB/Dellinger12.pdf

[8]https://hal.archives-ouvertes.fr/hal-00831763/document

[9]https://www.semanticscholar.org/paper/Improved-method-for-SAR-image-registration-based-on-Zhou-Zeng/b6cd548d6360246c5afacdb8314bf474e436dce0

[10]https://www.analyticsvidhya.com/blog/2019/10/detailed-guide-powerful-sift-technique-image-matching-python/

[11]https://towardsdatascience.com/sift-scale-invariant-feature-transform-c7233dc60f37

[12]https://www.bogotobogo.com/Matlab/Matlab_Tutorial_Digital_Image_Processing_6_Filter_Smoothing_Low_Pass_fspecial_filter2.php