InSAR Coherence Estimation using Multi-Stage Super Pixeling

1st Frincy Clement
dept. of Computing Science
University of Alberta
Edmonton, Canada
frincy@ualberta.ca

2nd Yashna Islam
dept. of Computing Science
University of Alberta
Edmonton, Canada
yashna@ualberta.ca

3rd Harshal Soni
dept. of Computing Science
University of Alberta
Edmonton, Canada
hsoni@ualberta.ca

***Abstract*—Interferometric Synthetic Aperture Radar (InSAR) is a satellite-based imaging technique which has been used to learn about earth's surface and sub-surface movements. It can measure earth's displacements by comparing phase information from the Synthetic Aperture Radar (SAR) images taken at different points in time. But, due to high level of noises from scatterers, the wrapped phases are distorted. Estimating the coherence of SAR image pair becomes rather important in this situation to denoise them and extract useful information from it.**

In our research, traditional methods of oversegmentation called Super Pixeling was performed in multiple levels, to capture the local as well as non-local coherence of images. The initial level of super pixeling was performed using SLIC [10] and Felzenszwalb's [11] algorithm and the second level using K-means clustering algorithm. We performed our tests on two datasets with 0.1 to 0 left to right and 0.3 fixed noise levels respectively. Our results show that, on the dataset with higher noise, our proposed method with best tuned parameters gave a mean-squared error(MSE) of 0.0157 which is 1.9 times better than the widely used traditional method called Boxcar (sliding window) which has an MSE of 0.0302. Also, it presented 30% more structural similarity of the coherence estimates compared to that of Boxcar.

Our proposed method has outperformed even the Convolutional Neural Network(CNN) learning-based approach [7] which gave a higher mean-squared error of 0.0299. Thus, our proposed approach estimates coherence between two SAR images, in a more robust and accurate way, compared to the existing traditional as well as learning-based state-of-the-art methods.

***Index Terms*—Image Coherence, Super Pixels, SAR Interferometry, InSAR, SLIC, K-means**

I. Introduction

Interferometric Synthetic Aperture Radar (InSAR), is an active remote sensing technique that acquires images of the Earth, and measure the surface deformation from glaciers, earthquakes and volcanoes. Even as early as 1990s, the technique of interferometric SAR (InSAR) [1] has been used to map topography and measure deformations. InSAR imaging technique can be used in quick surveying of remote areas that measures thousands of square kilometers and a spatial resolution of about 20m can be achieved. The pulses from the radar signals are combined to create an image of a ground scatterer, a signal processing technique called synthetic aperture radar (SAR), where the aperture includes motion of the antenna over many pulses.

A pixel in a SAR image consists of a complex number whose amplitude corresponds to the intensity of the returned radar energy and whose phase represents a fraction of a complete wavelength. The pixels in a SAR image is a complex function that varies from pixel to pixel due to the scattering features located on the surface. The SAR interferometry technique uses two SAR images of the same area acquired at different times and "interferes" (differences) them, resulting in maps called interferograms. It is created by subtracting the phases from two SAR images. The differences in fringes of a phase reveals the difference in distance between the ground and satellite. Each fringe corresponds to a displacement change of $\lambda/2$, where λ is the radar wavelength. InSAR images have deformation value that only follows the line-of-sight direction of the radar beam. Therefore, multiple SAR images with different observing geometries provides more than a single component of deformation.

The first step to generate an appropriate interferogram is the selection of SAR images, which has a strong impact on the quality of interferograms [2]. Certain parameters are required to be considered in order to get best results for InSAR images: (i) View angle (ii) Geometrical baseline, (iii) Temporal Baseline, (iv) Time of acquisition, (v) Coherence and (vi) Meteorological conditions. Coherence is the most important variable amongst the parameters as it provides a useful measure of the interferogram quality. However, determining the coherence between the SAR images suffers from noise caused by the delay in phase propagation by spatial and temporal variations. Other possible reasons include Doppler centroid difference, different look angle, volume scattering or surface reflectance properties. Another important noise is produced by the presence of multiple scatterers within one SAR resolution cell, which creates noise similar to salt and pepper called as speckle [3]. These are mainly created by smaller sized multiple scatterers which have a mixture of positive and negative signals adding to the randomness in resulting amplitude.

Hence, to understand the quality of the SAR images, coherence has to be estimated between SAR images pairs which will help in filtering the SAR images that goes into creating the InSAR which is used for measuring the terrain motion. The project focuses on repurposing conventional methods, by

estimating coherence after multiple levels Super pixeling. It is then decomposed back to the pixel level and compared with the ground truth.

Our results shows that Super pixeling using SLIC algorithm in the first level and K-means in the second level is able to get a mean squared error (MSE) of 0.0157 in a SAR image pair dataset with a fixed noise of 0.3 and 0.0143 in a dataset with left to right noise of 0.1 to 0. These values have outperformed the traditional method such as Boxcar [16] and even the recent learning-based approaches by reducing the MSE to 50%.

Once the coherence is calculated, SAR image pairs can be selected based on its quality, which can then be used to study the multivariate information available in the SAR pixels and separate them into individual components viz-a-viz motion of scatterer, the altitude of scatterer, atmospheric pressure, temperature and much more.

The rest of the paper is organized as follows. Section II will give an overview of the related works to our proposed approach. Section III amd IV introduces our proposed method and explains the implementation pipeline. Section V and VI concludes the report with experiments and results.

II. Related Work

The factors affecting the difference between SAR image pairs can be measured by the estimation of the coherence between the data. The methods used for estimating and classifying coherence in InSAR can be broadly categorized into three: (i) Filter-based, (ii) Block-matching and (iii) Learning-based approaches. One of the earliest methods for denoising and estimating coherence is Boxcar filtering approach [16]. The method involves calculation of average values using a rectangular window. Another filtering method was introduced by Goldstein and Werner in 1998 [12] that uses narrow band components from power spectra of interferograms to lower the phase noise. In 2004, Yanjie and Prinet introduced another method using phase correction and L-look sample along with Digital Elevation Model (DEM) [4]. The authors were able to find an asymptotically unbiased coherence estimation method for InSAR processing.

In 2011, Deledalle et. al introduced a non-local interferogram estimation (NL-InSAR) where they selected appropriate pixels by evaluating a patch-based similarity by considering noisy amplitudes, interferometric phases, and previous estimates [18]. Finally, the reflectivity, the actual interferometric phase, and the coherence are jointly estimated. Later in 2015, the same researchers proposed a framework based on nonlocal means for speckle reduction by resolution preserving called NLSAR [19]. The framework estimates the similarity between patches defined for arbitrary SAR modalities (SAR, InSAR, PolSAR, PolInSAR) and any number of looks. Then, the local selection of the most reliable estimate, defined as the estimate with least variance after a bias-variance trade-off.

In 2015, Cao et. al [5] adapted Patch-Based Locally Optimal Wiener Filter (PLOW), originally used for denoising digital images to SAR images. PLOW assumes that the images are geometrically similar, hence the improvised PLOW algorithm started with clustering the geometrically similar patches into groups and then performing denoising within each cluster. This filter-based method was able to provide better results at a higher computational cost.

Another approach for InSAR phase denoising was to use a block-matching method called BMINSAR by Barreto et. al [6]. In BMINSAR, initially the entire image is virtually divided into blocks using a search window and it is scanned in steps. Each search window is further sub-divided into patches which is the basis of evaluation. Using k-means clustering algorithm, every patch is clustered and labeled together based on inter-patch information and stacked into k cluster groups. Intra-patch information is used inside every stack and averaged phase, coherence, and amplitude is computed and replaced with previous information at every pixel position. These steps are repeated in parallel for every patch inside the window and thus iteratively for all the blocks inside the image. The authors have evaluated the performance of the algorithm using the PSNR metric by calculating an error between ground truth clean image and output image. It achieved an accuracy of 21cm in the resultant DEM which became the new state-of-the-art technique for phase denoising.

Recently, a lot of learning-based approaches has been introduced in the field of coherence estimation and filtering InSAR images. In 2018, Mukherjee et. al [7] introduced a CNN-based autoencoder to denoise filters and compute the coherence metric. The authors designed an autoencoder irrespective of image size. Thus, there is a flexibility to train the CNN using whole image or patches. The network was trained using 60X60 patches taken from real-world InSAR images. Then the denoised interferogram was fed into a 2-channeled CNN for coherence estimation. The execution time for their proposed method was 1.42 seconds which is lower than the other methods (Boxcar, NL-SAR, Goldstein and NL-InSAR). The phce (mean of cosine of absolute phase error) and cmse (coherence mean squared error) scores looked promising.

Again in 2018, the same researchers proposed a CNN-based InSAR image coherence classification as an extension to their previous work [8] using Markov-Random Field (MRF) method to threshold the InSAR image data and then a CNN classifier to measure the coherence. Their proposed method resulted in an improved demarcation of incoherent and coherent regions and reduced misclassifications in completely incoherent regions. The execution time of their CNN classifier took on 0.67 seconds outperforming Boxcar, NLSAR and NLInSAR methods.

In 2019, Sun et. al proposed DeepInSAR [13] as an extension to the previous work carried by Mukherjee. The authors proposed a deep CNN-based model DeepInSAR to solve the phase denoising and coherence estimation issue. Their proposed method includes a densely connected feature extractor whose input is a single tensor having the concatenated amplitudes and interferometric phases of two single look complex, (SLC) SAR images. The model was trained with a patch size of 128x128. Their proposed DeepInSAR outperforms all other methods (Boxcar, NL-SAR and NL-InSAR) 18 different levels of distortions. The model provides

better results without any human supervision as well as real-time performance.

A recent research by Ma et. al [9] in 2019 gave a completely new direction in segmentation of SAR imagery data using Super pixels and Graph Convolutional Networks. It differs from other learning-based methods in that it uses segmentation based on super pixeling rather than pixels, which in turn reduces computational cost while preserving the edges. They used Simple Linear Iterative Clustering (SLIC) [10] algorithm for super pixeling and used a majority voting-based strategy for getting the labels of super pixels. A SAR image after super pixeling will become graph structure and is a great candidate for Graph Convolutional Networks (GCN). The authors used a special GCN called Attention Graph Convolutional Networks shortly called AGCN and is said to be computationally efficient.

In 2020, another approach was proposed by Mukherjee et. al towards coherence estimation using generative convolutional neural network. Their proposed model "GenInSAR" [14] uses a sequential neural network to perform joint phase filtering and coherence estimation. It predicts the distribution of the center pixel using only its neighborhood pixels. The model takes the input of raw image divided into 11x11 phase patch and predicts Gaussian parameters to calculate centrality of image. Thus, it makes their network an efficient unsupervised approach for coherence estimation with a RMSE of 0.144.

Our proposed approach makes use of conventional methods of over segmentation called super pixeling for estimating the coherence of SAR images. Since the SAR images have nonlocal coherence, we plan to employ a graph-based segmentation introduced by Felzenszwalb et. al [11] to generate super pixels from pixels. Using a graph-based representation of an image, they found a base to measure the boundary between two regions. They used this to create greedy decisions to produce segmentations that go into higher detail in regions with low variation and vice-versa. Depending on the local contrast levels, the number and size of segments can vary. As per the authors, this method runs in the fraction of a second, which can be approximated as O(mlogm) for a graph with m edges.

Another proven choice of super-pixeling algorithm as mentioned before is using Simple Linear Iterative Clustering (SLIC) [10]. The algorithm is simple as it needs only one parameter which is the number of super pixels. When it was introduced, it outperformed the then state-of-the-art methods at a lesser computational cost but by preserving the segmentation quality. The super pixels generated using this method are similar in size and preserve the edge boundaries.

SLIC takes into account the proximity of pixels in the image plane and similarity of the different features. SLIC was initially performed on a 5-dimensional input because of which Euclidean distance became obsolete. SLIC created a new distance measure which considers the size of the super pixels. It is based on the K-means clustering algorithm where we start with specifying the number of clusters k. SLIC being a simple and fast algorithm, it is used as a pre-processing step for super-pixeling the pixel level input before passing to computer vision tasks such as recognition and segmentation.

III. Proposed Method

Our proposed method was structured on the understanding that a better estimate of coherence could be provided by identifying the non-local similarities between two SAR images. We employed the technique of super pixeling which uses process of over segmentation, while keeping perceptual [15] and meaning of the image, compared to just the pixels alone. In our experiments, we used two super pixeling algorithms namely Efficient graph based segmentation [11] and Simple Iterative Linear Clustering (SLIC) [10] to compare the performance. We also used K-Means clustering algorithm for second level of super pixeling. The coherence estimated obtained after two levels of super pixeling are then decomposed to pixel level to understand pixel-level coherence, which could then be compared with the ground truth using mean-squared error and structural similarity.

IV. Stages of Implementation

1) Datasets: We received the dataset of simulated SAR image pairs along with interferogram and ground truth coherence, from the Multimedia Research Centre at University of Alberta in coordination with 3V Geomatics. We worked with two sets of datasets with different levels of noise. The first dataset had 150 sets of inputs with 300x300 dimensions and a base noise of 0.1 to 0 from left to right. The second dataset had 60 sets of inputs with 1000x1000 dimension and a base fixed noise of 0.3.

For performing the initial round of experiments to fine tune the parameters, one sample image from the dataset was considered. After understanding the best value of parameters, the pipeline was repeated for all the inputs in the two datasets.

A. Preprocessing input

The first step in the proposed algorithm is to create the input data in the required format to be able to be passed into the pipeline. It starts with selecting a SAR image pair and the corresponding interferogram. A four-channel input is created using the following: (i) Amplitude of first SAR image, (ii) Amplitude of second SAR image, (iii) Real component of phase from the InSAR, and (iv) Imaginary component of phase from the InSAR. These 4 inputs are then stacked in the third dimension and is fed into the first stage of super pixeling. For the first dataset, we passed the 300x300 inputs directly without dividing it into patches. For the second input with higher noise, we have taken overlapping patches of 300x300 from the 1000x1000 input with a stride of 150.

Let each SAR pair be of size (N, N) , then the input to the pipeline would be of size $(N, N, 4)$.

B. Stage I Super pixeling

Super pixeling is a way of segmenting the input grid into smaller clusters, which are bigger than pixel size, there by preserving the semantic meaning of the input. In this stage,

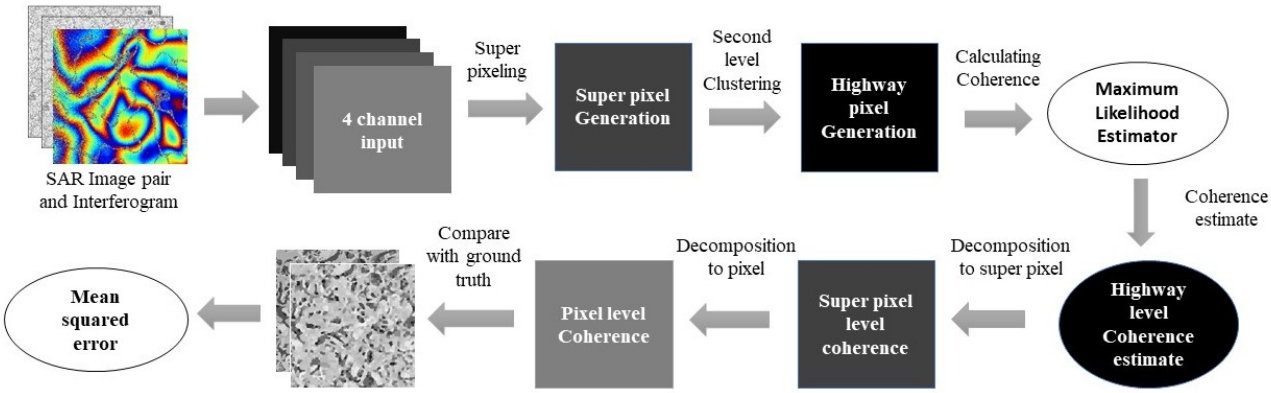

Fig. 1. Stages of implementation of Proposed Method

the constituent pixels of 4-channel input will be clustered into super pixels, which would group the pixels based on their proximity and similarity. This approach is used as opposed to blocking matching technique in BMINSAR [6] , as super pixeling generates the clusters which are not dependent on the window size of the block but rather based on the location and similarity of pixels. Super pixels would also preserve the relative dependence of the pixels.

For our experiments, we have compared two algorithms: (i) Felzenszwalb's efficient graph-based segmentation and (ii) SLIC (Simple Linear Iterative Clustering) which is a K-means based segmentation.

Felzenszwalb's algorithm works in a greedy manner and segment the inputs based on how it confirms to the global features. One of the peculiar features suggested by Felzenszwalb is that this algorithm is able to keep the details of regions where the variation is lower and at the same time ignore those in the region with high variation. To arrive at the best possible results, we have tuned the following parameters: the size of clusters, standard deviation of points in clusters and minimum component size.

SLIC performs clustering on the pixels in 5-dimensional space, 3 dimensions in the 3 colors channels and the remaining two dimensions in the pixel positions. As it is a simple algorithm, it is more robust. For this algorithm the parameter which decides the number of K-mean centers, compactness which balances between color and distance proximity and standard deviation of clusters has been tuned using various experiments.

After stage 1 super pixeling which generates C_1 clusters, the input of size $(N, N, 4)$ will generate an output of dimension (N, N) with C_1 unique cluster labels.

C. Stage II Super pixeling

After the initial superpixel generation, the intra-cluster pixels are closely correlated, but there will be some inter-cluster pixels which can also be closely correlated to each other. This would potentially limit our next steps in estimating the image coherence, as it would suffer from not capturing the non-local coherence. Hence, by adding a second level of clustering the super pixels generated from the first stage, would result in another set of segments, which we call *highway superpixels*. This would essentially capture non-local coherence that has not been found in the first level of super pixeling.

After stage 2 super pixeling with C_2 clusters, the input of size (N, N) will generate an output of dimension (1,C_1) with C_2 unique cluster labels.

D. Coherence Estimation

Once the highway pixels are generated, the coherence is estimated using Maximum likelihood estimator defined by the given formula.

$$\delta_L = \frac{\sum_{i=1}^{L} z_1^{(i)} z_2^{*(i)}}{\sqrt{\sum_{i=1}^{L} |z_1^{(i)}|^2 |z_2^{(i)}|^2}} \tag{1}$$

where z_1 and z_2 represents the pixel values in SAR image pair at location *i* and is summed up for all pixel values in each of the clusters separately. For this approach, an assumption is made that the observations of the smaller regions will be stationery. Thus for C_2 highway pixels, there will be C_2 coherence estimations.

E. Decomposition of Highway Coherence

The decomposition of highway coherence to pixel level is done in two stages. At first, it is decomposed to super pixel level and then from super pixel to pixel level. This means that the super pixels belonging to the same highway clusters receive the corresponding highway coherence of the centroid of the cluster, based on a weighted normalized distance function using L2-Norm. We have performed several experiments to arrive at different ways of decompositions, which will be explained in detail in the next section on Experiments.

The same process is repeated one more time from super pixel level to pixel level. For each stage of decomposition, the

centroids of the clusters are determined first, which assumes the mean value of coherence estimated for all the pixels in the cluster. This mean coherence value is then passed on to the corresponding super pixel and pixels.

C_2 highway pixels with dimension of (1,C_1) where C_1 is the number of super pixels, will then be decomposed to super pixels with a dimension of (N, N) where *N* is the width and height of input image, provided we pass a square grid of input.

F. Comparison with Ground Truth

Both the datasets that was provided to us had the simulated grouth truth of coherence included for each pair of SAR images. The performance of our proposed method was estimated by comparing the results with the ground truth using Mean Squared Error (MSE) and Structural Similarity Index (SSIM).

V. Experiments and Results

Coherence estimation of SAR image pair using our proposed method involves tuning the parameters of algorithms used for super pixeling. For each of the model parameters, for each of the datasets, we have performed few experiments with different methods of decomposition. These methods mostly differ in the weights or distance function used to pass the coherence values of cluster centroids to its cluster member pixels. Following are the details of experiments performed.

Notations used:
Let *I* be the 4 channel input of size $(N, N, 4)$ fed into the proposed pipeline. After first stage of clustering let *S* be the output with C_1 super pixels and a dimension of (N, N). After second stage of clustering let *H* be the output with C_2 high way pixels and a dimension of $(1, C_1)$. The cluster centroids of super pixels and highway pixels are denoted by S_c and H_c respectively and d_1 and d_2 are the L2-norm of normalized distance between member pixel and cluster centroid, for highway and super pixels respectively.

Experiment 1: Inverse Distance Decomposition

In this method, at each stage of decomposition, the coherence of cluster centroid is multiplied with a weight, before passing on to the corresponding member pixels. The weight increases as the member pixel is closer to the cluster centroid and becomes 1 when it is at the centroid. Similarly, as the member pixel is farther, the coherence received by the pixel will be much less and can be even zero. The coherence for *i*th member in cluster *j* can be estimated as:

$$\begin{aligned} For\ highway\ pixels: &\quad Coh(H_i^j) = Coh(H_c^j) * (1 - d_1) \\ For\ super\ pixels: &\quad Coh(S_c^j) = Coh(H_i^j) \\ &\quad Coh(S_i^j) = Coh(S_c^j) * (1 - d_2) \end{aligned}$$

Experiment 2: Differential Decomposition with level-specific weight

In this experiment, different weight function are applied based on whether the decomposition happens at highway to super pixel or from super pixel to pixel level. Since we wanted to cover the non-local coherence to the maximum, we gave higher weights to the first stage of decomposition, which is highway to super pixel. The coherence for i^{th} member in cluster *j* can be estimated as:

$$\begin{aligned} For\ highway\ pixels: &\quad Coh(H_i^j) = Coh(H_c^j) * \sqrt{(1 - d_1)} \\ For\ super\ pixels: &\quad Coh(S_c^j) = Coh(H_i^j) \\ &\quad Coh(S_i^j) = Coh(S_c^j) * (1 - d_2)^2 \end{aligned}$$

Experiment 3: Differential Decomposition with distance-specific weight

In this experiment, different weight functions are applied, based on how far the member pixel is from the cluster centroid. For closer pixels, a bonus weight is provided by applying square root of value between 0 and 1 and for farther pixels, penalty is applied by multiplying with a square of a value between 0 and 1. The coherence for i^{th} member in cluster *j* can be estimated as:

$For\ highway\ pixels:$

$$Coh(H_i^j) = \begin{cases} Coh(H_c^j) * \sqrt{(1 - d_1)},\ if\ d_1 < 0.3 \\ Coh(H_c^j) * (1 - d_1)^2,\ if\ d_1 > 0.3 \end{cases}$$

$For\ super\ pixels:$

$$Coh(S_c^j) = Coh(H_i^j)$$

$$Coh(S_i^j) = \begin{cases} Coh(S_c^j) * \sqrt{(1 - d_2)},\ if\ d_2 < 0.3 \\ Coh(S_c^j) * (1 - d_2)^2,\ if\ d_2 > 0.3 \end{cases}$$

Experiment 4: Direct decomposition of highway coherence

In this experiment, the coherence of cluster centroids are passed directly in the first and second stage of decomposition as-is, without adding any weights. The coherence for i^{th} member in cluster *j* can be estimated as:

$$\begin{aligned} For\ highway\ pixels: &\quad Coh(H_i^j) = Coh(H_c^j) \\ For\ super\ pixels: &\quad Coh(S_c^j) = Coh(H_i^j) \\ &\quad Coh(S_i^j) = Coh(S_c^j) \end{aligned}$$

Experiment 5: Direct decomposition of Super pixel coherence

In this experiment, the coherence is estimated at the super pixel level and then passed to the pixel level by doing one stage of decomposition. The coherence for i^{th} member in cluster *j* can be estimated as:

$$For\ super\ pixels: \quad Coh(S_i^j) = Coh(S_c^j)$$

Experiment 6: Combination of highway and super pixel direct decomposition

In this experiment, the coherence is estimated by combining the results of experiment 4 and 5. The coherence for i^{th} member in cluster *j* can be estimated as:

$$For\ highway\ pixels: \quad Coh(H_i^j) = Coh(H_c^j)$$
$$For\ super\ pixels: \quad Coh(S_i^j) = Coh(S_c^j)$$
$$Final\ Coherence = \sqrt{(Coh(H_i^j) * Coh(S_i^j)}$$

A. Evaluation Metrics

The resulting coherence of each experiment is compared with the ground truth using Mean squared error(MSE) and Structural Similarity Index (SSIM).

B. Results of Experiments

We conducted six experiments on two datasets with different noise levels. First dataset had 0.1 to 0 left to right noise level and second dataset had a fixed noise of 0.3. We compared all the combinations on different number of model parameters for Felzenszwalb's and SLIC super pixeling algorithm. Hence, different combinations of cluster numbers in first and second stage were repeated for all the six experiments, to arrive at the best tuned parameters.

The results of the experiments are added to the appendix section. "Fig. 3", "Fig. 4", "Fig. 5" and "Fig. 6" shows output for the first dataset using Felzenszwalb's, SLIC and Box car. "Fig. 7", "Fig. 8", "Fig. 9" and "Fig. 10" shows output for the second dataset using the same. Following are the inferences that can be made out of the results of the experiments:

1) As the number of clusters in super pixels increases, the MSE decreases and SSIM increases. But after a point, MSE increases and SSIM decreases.
2) For the same number of super pixel cluster, as the highway pixel cluster decreases, MSE decreases and SSIM increases.
3) The best number of clusters for super pixel and highway pixel changes with the level of noise in the dataset.
4) Top 3 experiments with the best results are 4,6 and 5. The highway pixels decomposed to pixel level without any weight function gives the best result for all the cluster numbers and for both datasets.
5) For the stage 1 super pixeling, SLIC algorithm performed better than Felzenszwalb's.

C. Final Results

Based on the learnings from the experiments, we performed the calculation of coherence on larger sized datasets. The first dataset with 150 300x300 input sets was used as-is without dividing into patches. The second dataset with 60 1000x1000 input sets, were divided into overlapped patches of 300x300 with a stride of 150. Since SLIC algorithm performed much better for the Stage I super pixeling, all the remaining tests were run using SLIC.

For the first dataset, 5625 and 600 cluster numbers were the best performing values for Stage I and II of super pixeling respectively. For the second dataset, the best number of clusters were 2498 and 75. The second dataset is the same as the one used in [7], hence the results were compared against their result table as well.

TABLE I
COMPARISON OF COHERENCE ESTIMATION ON FIRST DATASET

	Proposed	Boxcar
Average MSE	0.0143	0.0267
Average RMSE	0.1194	0.1633
Average SSIM	0.7615	0.6709

TABLE II
COMPARISON OF COHERENCE ESTIMATION ON SECOND DATASET

	Proposed	Boxcar	CNN-based	NLInSAR	NLSAR
Average MSE	0.0157	0.0302	0.0299	0.0265	0.0813
Average RMSE	0.1234	0.1704	0.1729	0.1628	0.2851
Average SSIM	0.6641	0.4890	N/A	N/A	N/A

The results shows that our proposed method performs better for both the datasets, compared to the most common conventional methods like Boxcar, NLSAR, NLInSAR as well as CNN learning-based methods.

VI. CONCLUSION

InSAR Coherence estimation is a vital step in the process of learning important information about earth's surface and sub-surface movements. Various contributions have been made in this field by many researchers to achieve accurate coherence estimation using traditional computer vision as well as learning-based methods.

Our proposed application, which is based on the traditional computer vision methods of oversegmentation called Super pixeling, has proven to be 1.9 times better compared to Boxcar, 5 times better than NL-SAR, 1.6 times better than NL-InSAR and 1.9 times better than CNN-based method, in terms of mean-squared error.

In the immediate future, we would like to work with our mentor and advisor for any additional experiments that may need for this research and also to replicate the efforts on standard datasets to compare our results for a publication.

VII. RESOURCES

1. Source Code: https://tinyurl.com/yaduklq5
2. Project Demo Video : https://tinyurl.com/ybevwhf5
3. Powerpoint for Project Demo: https://tinyurl.com/yax96rer
4. Algorithm and Documentation: https://tinyurl.com/y7fq83r7

VIII. ACKNOWLEDGEMENTS

We would like to thank Dr. Anup Basu and Dr. Irene Cheng for their advise in the process of working on this project.

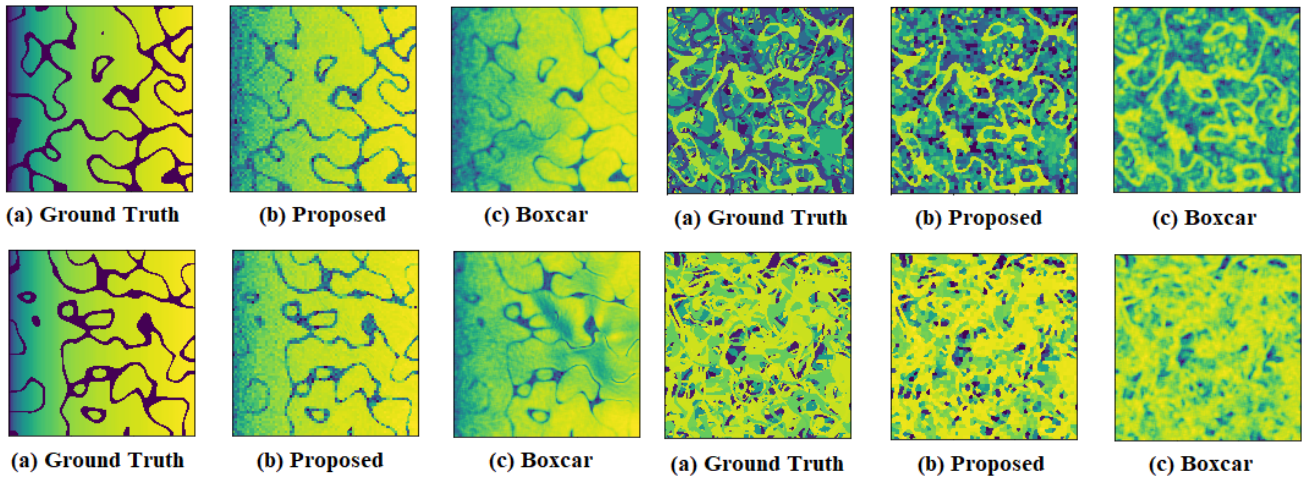

Fig. 2. Visual comparison of samples results from First (left half) and Second (right half) dataset

We would also like to thank Xinyao Sun, Phd. candidate at the Multimedia Research Centre, University of Alberta for his guidance and support throughout our research work. We would also like to acknowledge and thank 3V Geomatics and Multimedia Research Center for providing us with simulated InSAR datasets for our research.

REFERENCES

[1] Pritchard, Matthew E. "InSAR, a tool for measuring Earth's surface deformation." Physics Today 59, no. 7 (2006): 68.

[2] Alessandro Ferretti, Andrea Monti-Guarnieri, Claudio Prati, Fabio Rocca, and Didier Massonnet. InSAR Principles: Guidelines for SAR Interferometry Processing and Interpretation, Part B: InSAR processing: a practical approach. ESA, 2007

[3] Ferretti, A., A. Monti-Guarnieri, C. Prati, F. Rocca, and D. Massonnet. "InSAR processing: a mathematical approach (Part C)." InSAR Principles: Guidelines for SAR Interferometry Processing and Interpretation (2007): 120-234.

[4] Yanjie, Zhang, and Véronique Prinet. "InSAR coherence estimation." In IGARSS 2004. 2004 IEEE International Geoscience and Remote Sensing Symposium, vol. 5, pp. 3353-3355. IEEE, 2004.

[5] Cao, Mingyu, Shiqiang Li, Robert Wang, and Ning Li. "Interferometric phase denoising by median patch-based locally optimal wiener filter." IEEE Geoscience and Remote Sensing Letters 12, no. 8 (2015): 1730-1734.

[6] Barreto, Thiago LM, Rafael AS Rosa, Christian Wimmer, João R. Moreira, Leonardo S. Bins, Jurandy Almeida, and Fábio AM Cappabianco. "BMINSAR: A novel approach for InSAR phase denoising by clustering and block matching." In 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), pp. 2357-2360. IEEE, 2017.

[7] Mukherjee, Subhayan, Aaron Zimmer, Navaneeth Kamballur Kottayil, Xinyao Sun, Parwant Ghuman, and Irene Cheng. "CNN-Based InSAR Denoising and Coherence Metric." In 2018 IEEE SENSORS, pp. 1-4. IEEE, 2018.

[8] Mukherjee, Subhayan, Aaron Zimmer, Xinyao Sun, Parwant Ghuman, and Irene Cheng. "CNN-based InSAR Coherence Classification." In 2018 IEEE SENSORS, pp. 1-4. IEEE, 2018.

[9] Ma, Fei, Fei Gao, Jinping Sun, Huiyu Zhou, and Amir Hussain. "Attention Graph Convolution Network for Image Segmentation in Big SAR Imagery Data." Remote Sensing 11, no. 21 (2019): 2586.7

[10] Achanta, Radhakrishna, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. Slic superpixels. No. REP_WORK. 2010.

[11] Felzenszwalb, Pedro F., and Daniel P. Huttenlocher. "Efficient graph-based image segmentation." International journal of computer vision 59, no. 2 (2004): 167-181.

[12] Goldstein, Richard M., and Charles L. Werner. "Radar interferogram filtering for geophysical applications." Geophysical research letters 25, no. 21 (1998): 4035-4038.

[13] Sun, Xinyao, Aaron Zimmer, Subhayan Mukherjee, Navaneeth Kamballur Kottayil, Parwant Ghuman, and Irene Cheng. "DeepInSAR: A Deep Learning Framework for SAR Interferometric Phase Restoration and Coherence Estimation." arXiv preprint arXiv:1909.03120 (2019).

[14] Mukherjee, Subhayan, Aaron Zimmer, Xinyao Sun, Parwant Ghuman, and Irene Cheng. "A Novel Generative Neural Approach for InSAR Joint Phase Filtering and Coherence Estimation." arXiv preprint arXiv:2001.09631 (2020).

[15] Baig, Wajihullah, Luca, Adrian Rosebrock, Deven, Chris Viehoff, Ruben Garcia, Joe Marotta, et al. "Segmentation: A SLIC Superpixel Tutorial Using Python." PyImageSearch, April 18, 2020. https://www.pyimagesearch.com/2014/07/28/a-slic-superpixel-tutorial-using-python/.

[16] Weisstein, Eric W. "Boxcar function." From MathWorld–A Wolfram Web Resource. https://mathworld. wolfram. com/BoxcarFunction. html (2015).

[17] Seymour, M. S., and I. G. Cumming. "Maximum likelihood estimation for SAR interferometry." In Proceedings of IGARSS'94-1994 IEEE International Geoscience and Remote Sensing Symposium, vol. 4, pp. 2272-2275. IEEE, 1994.

[18] DDeledalle, Charles-Alban, Loïc Denis, and Florence Tupin. "NL-InSAR: Nonlocal interferogram estimation." IEEE Transactions on Geoscience and Remote Sensing 49, no. 4 (2010): 1441-1452.

[19] Deledalle, Charles-Alban, Loïc Denis, Florence Tupin, Andreas Reigber, and Marc Jäger. "NL-SAR: A unified nonlocal framework for resolution-preserving (Pol)(In) SAR denoising." IEEE Transactions on Geoscience and Remote Sensing 53, no. 4 (2014): 2021-2038.

APPENDIX

Stage 1 & 2 Super pixels		Exp 1	Exp 2	Exp 3	Exp 4	Exp 5	Exp 6
12853 & 600	MSE	0.02845	0.03040	0.03628	0.03418	0.04181	0.03682
	RMSE	0.16867	0.17436	0.19047	0.18488	0.20448	0.19189
	SSIM	0.49452	0.46184	0.41143	0.73693	0.64588	0.71165
12853 & 2000	MSE	0.02666	0.03216	0.03749	0.03601	0.04181	0.03799
	RMSE	0.16327	0.17933	0.19364	0.18977	0.20448	0.19491
	SSIM	0.48709	0.45286	0.40497	0.72245	0.64588	0.70032
6367 & 600	MSE	0.03502	0.02933	0.04274	**0.02787**	0.03077	0.02859
	RMSE	0.18714	0.17125	0.20673	**0.16694**	0.17541	0.16909
	SSIM	0.47065	0.44605	0.36851	**0.72681**	0.67906	0.71828
6367 & 1500	MSE	0.03037	0.02849	0.04126	0.02803	0.03077	0.02886
	RMSE	0.17427	0.16878	0.20312	0.16741	0.17541	0.16989
	SSIM	0.47235	0.44468	0.36828	0.72198	0.67906	0.71160
2676 & 600	MSE	0.04638	0.03413	0.05504	0.02705	0.02846	0.02735
	RMSE	0.21535	0.18476	0.23460	0.16447	0.16871	0.16539
	SSIM	0.43810	0.42245	0.28739	0.67637	0.64920	0.67253
1179 & 400	MSE	0.07032	0.04952	0.09219	0.03295	0.03439	0.03320
	RMSE	0.26518	0.22253	0.30363	0.18153	0.18543	0.18220
	SSIM	0.25539	0.38344	0.22624	0.46635	0.44177	0.59986

Fig. 3. Experiment Results for First dataset with left to right noise level of 0.1 to 0, using Felzenszwalb's algorithm in the Stage 1 Super pixeling

Stage 1 & 2 Super pixels		Exp 1	Exp 2	Exp 3	Exp 4	Exp 5	Exp 6
10000 & 600	MSE	0.0505	0.0329	0.0638	0.0147	0.0193	0.0157
	RMSE	0.2247	0.1814	0.2526	0.1212	0.1388	0.1252
	SSIM	0.4652	0.4488	0.3235	0.7624	0.6760	0.7409
10000 & 2000	MSE	0.0305	0.0322	0.0604	0.0152	0.0193	0.0162
	RMSE	0.1747	0.1793	0.2458	0.1231	0.1388	0.1273
	SSIM	0.4468	0.4529	0.3246	0.7426	0.6760	0.7252
5625 & 600	MSE	0.0565	0.0368	0.0744	**0.0146**	0.0179	0.0152
	RMSE	0.2376	0.1918	0.2728	**0.1210**	0.1337	0.1235
	SSIM	0.4545	0.4380	0.2992	**0.7773**	0.7181	0.7657
5625 & 1500	MSE	0.0547	0.0374	0.0777	0.0181	0.0211	0.0188
	RMSE	0.2339	0.1933	0.2788	0.1345	0.1452	0.1372
	SSIM	0.4406	0.4248	0.2870	0.7303	0.6824	0.7198
2500 & 600	MSE	0.0636	0.0438	0.0932	0.0175	0.0199	0.0179
	RMSE	0.2522	0.2094	0.3052	0.1323	0.1411	0.1340
	SSIM	0.4246	0.4035	0.2631	0.7534	0.7261	0.7519
1089 & 400	MSE	0.0754	0.0532	0.1150	0.0228	0.0241	0.0229
	RMSE	0.2746	0.2306	0.3392	0.1510	0.1553	0.1513
	SSIM	0.3870	0.3658	0.2265	0.7177	0.7105	0.7226

Fig. 4. Experiment Results for First dataset with left to right noise level of 0.1 to 0, using SLIC algorithm in the Stage 1 Super pixeling

	7x7	5x5
MSE	0.0286	0.0237
RMSE	0.1691	0.1539
SSIM	0.6675	0.6767

Fig. 5. Experiment Results for First dataset with left to right noise level of 0.1 to 0, using Boxcar (Sliding Window)

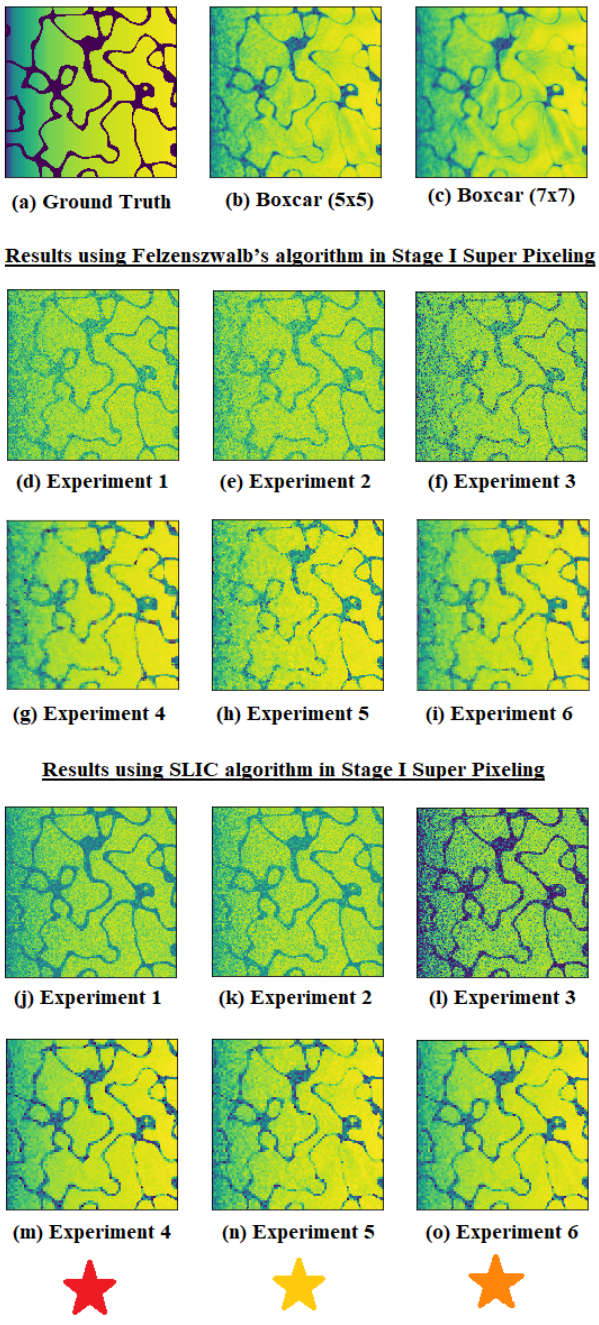

Fig. 6. Visual Comparison for the best tuned parameters for First dataset with left to right noise level of 0.1 to 0. Top 3 results were achieved by Experiment 4,6 and 5, which uses SLIC algorithm for Stage I Super pixeling

Stage 1 & 2 Super pixels		Exp 1	Exp 2	Exp 3	Exp 4	Exp 5	Exp 6
12092 & 600	MSE	0.0435	0.0455	0.0755	0.0546	0.0678	0.0587
	RMSE	0.2086	0.2134	0.2748	0.2336	0.2604	0.2423
	SSIM	0.4108	0.3472	0.2556	0.5222	0.4302	0.4947
5851 & 600	MSE	0.0531	0.0410	0.0924	0.0352	0.0417	0.0370
	RMSE	0.2304	0.2025	0.3040	0.1876	0.2043	0.1924
	SSIM	0.4533	0.4070	0.2860	0.5469	0.4970	0.5342
2444 &300	MSE	0.0803	0.0543	0.1430	**0.0299**	0.0331	0.0307
	RMSE	0.2834	0.2330	0.3781	**0.1730**	0.1818	0.1752
	SSIM	0.4457	0.4208	0.2652	**0.5221**	0.5008	0.5175
1079 & 200	MSE	0.0789	0.0519	0.1376	0.0353	0.0369	0.0357
	RMSE	0.2808	0.2279	0.3710	0.1878	0.1921	0.1890
	SSIM	0.4508	0.4409	0.2962	0.4573	0.4484	0.4556
1079 & 200	MSE	0.0813	0.0527	0.1393	0.0349	0.0369	0.0354
	RMSE	0.2851	0.2297	0.3732	0.1869	0.1921	0.1882
	SSIM	0.4479	0.4403	0.2945	0.4590	0.4484	0.4568

Fig. 7. Experiment Results for Second dataset with fixed noise level of 0.3, using Felzenszwalb's algorithm in the Stage 1 Super pixeling

Stage 1 & 2 Super pixels		Exp 1	Exp 2	Exp 3	Exp 4	Exp 5	Exp 6
10000 & 600	MSE	0.0635	0.0388	0.1044	0.0259	0.0337	0.0274
	RMSE	0.2519	0.1971	0.3232	0.1610	0.1836	0.1657
	SSIM	0.4870	0.4724	0.3325	0.5672	0.4947	0.5497
10000 & 2000	MSE	0.0570	0.0372	0.1005	0.0273	0.0337	0.0286
	RMSE	0.2388	0.1927	0.3170	0.1652	0.1836	0.1692
	SSIM	0.4868	0.4685	0.3349	0.5507	0.4947	0.5373
5624 & 600	MSE	0.0637	0.0381	0.1082	0.0205	0.0247	0.0213
	RMSE	0.2523	0.1952	0.3290	0.1431	0.1571	0.1461
	SSIM	0.5138	0.4983	0.3472	0.6258	0.5793	0.6157
5625 & 1500	MSE	0.0589	0.0367	0.1051	0.0213	0.0247	0.0220
	RMSE	0.2426	0.1916	0.3242	0.1458	0.1571	0.1483
	SSIM	0.5157	0.4957	0.3500	0.6168	0.5793	0.6085
2498 & 75	MSE	0.0789	0.0440	0.1230	**0.0169**	0.0198	0.0174
	RMSE	0.2809	0.2097	0.3508	**0.1299**	0.1408	0.1319
	SSIM	0.5042	0.5027	0.3379	**0.6683**	0.6392	0.6644
2481 & 150	MSE	0.0733	0.0420	0.1195	0.0171	0.0198	0.0176
	RMSE	0.2707	0.2049	0.3457	0.1308	0.1408	0.1328
	SSIM	0.5141	0.5048	0.3428	**0.6712**	0.6392	0.6650
1840 & 75	MSE	0.0805	0.0471	0.1330	0.0171	0.0196	0.0176
	RMSE	0.2838	0.2170	0.3646	0.1308	0.1400	0.1326
	SSIM	0.4959	0.4894	0.3210	0.6703	0.6440	0.6651
1840 & 300	MSE	0.0707	0.0437	0.1265	0.0177	0.0196	0.0181
	RMSE	0.2659	0.2089	0.3557	0.1331	0.1400	0.1346
	SSIM	0.5085	0.4899	0.3289	0.6645	0.6440	0.6602
1088 & 75	MSE	0.1126	0.0777	0.1977	0.0179	0.0196	0.0181
	RMSE	0.3356	0.2788	0.4447	0.1338	0.1399	0.1347
	SSIM	0.4138	0.3873	0.2191	0.6575	0.6445	0.6565
1088 &150	MSE	0.1081	0.0760	0.1946	0.0183	0.0196	0.0185
	RMSE	0.3288	0.2758	0.4411	0.1354	0.1399	0.1361
	SSIM	0.4185	0.3867	0.2217	0.6550	0.6445	0.6537

Fig. 8. Experiment Results for Second dataset with fixed noise level of 0.3, using SLIC algorithm in the Stage 1 Super pixeling

	7x7	5x5
MSE	0.0333	0.0295
RMSE	0.1824	0.1717
SSIM	0.4559	0.5089

Fig. 9. Experiment Results for Second dataset with fixed noise level of 0.3, using Boxcar (Sliding Window)

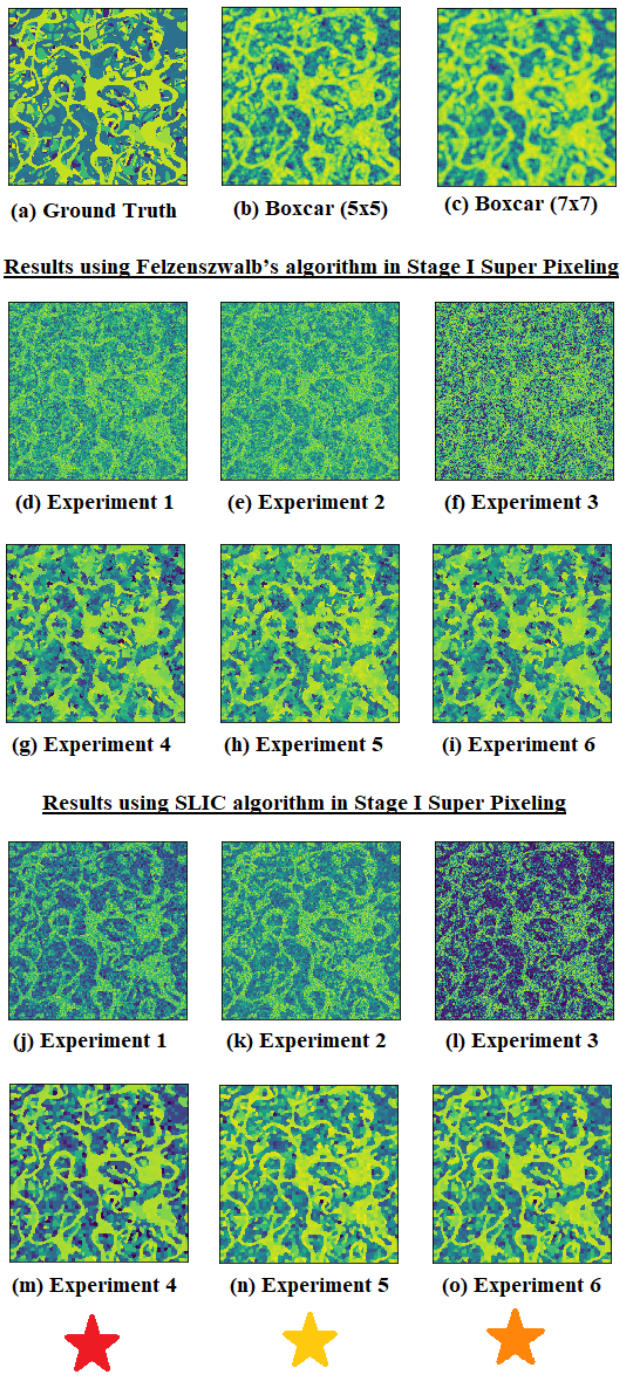

Fig. 10. Visual Comparison for the best tuned parameters for Second dataset with fixed noise level of 0.3. Top 3 results were achieved by Experiment 4,6 and 5, which uses SLIC algorithm for Stage I Super pixeling