ISSN 2278-3091
Volume 10, No.2, March - April 2021
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse1241022021.pdf
https://doi.org/10.30534/ijatcse/2021/1241022021

Multi-focus Image Fusion based on Genetic Algorithm using Local Features

Raheel Sarwar[1], Ammar Oad[2*], Mir Sajjad Hussain Talpur[1],Abida Luhrani[1], Akhtar Hussain[1], Taha Nuzhat[1], Hina Rehman,Shakir Hussain Talpur, Erum Saba Chang[1], Gaofeng Luo[2]
[1]Information Technology Centre, Sindh Agriculture University, TandoJam, Pakistan,
mirsajjadhussain@sau.edu.pk
[2]Faculty of Information Engineering, Shaoyang University, Shaoyang 422000, China,
ammar_2k309@yahoo.com

Recent developments in the domain of information technology have made it possible to extract a knowledge of ocean from input images. The knowledge extraction can be performed using a number of operations such as image segmentation. The major objective of image segmentation is to segment focused and non-focused regions from an input image. The field depth of optical lenses is limited. A camera focuses only on those objects which lie in its field depth, rest of the objects are appeared as non-focused or blurry. For image processing, it is a general requirement that an input image must be all in focus image. In almost each domain such as medical imaging, weapon and aircraft detection, digital photography, and agriculture imaging, it is required to have an all-in focused input image. Image fusion is a process which combines two or more input images to create an all in focused complimentary fused image. Image fusion is considered as a challenging task due to irregular boundaries of focused and non-focused regions. In literature, multiple studies have addressed this issue, however they have reported promising results in creating a fully focused fused image. Moreover, they have considered different features to identify focused and non-focused regions from an input image. For better estimation of focused and non-focused regions, an ensemble of multiple features such as shape and texture-based features can be employed. Furthermore, it is required to obtain optimal weights which are to be assigned to each feature for creating a fused image. The focus of this study is to perform a multi-focus image fusion using an ensemble of multiple local features by weight optimization using a genetic algorithm. To perform this experimentation, nine multi-focus image datasets are collected where each dataset indicates an image pair of multi-focused images. The reason of this selection is two-fold, as they are publicly available, and it contain different types of multi-focus images. For reconstruction of a fully focused fused image, an ensemble of different shape and texture-based features such as Sobel Operator, Laplacian Operator and Local Binary Pattern is employed along with optimal weights obtained using a Genetic Algorithm. The experimental results have indicated improvement over previous fusion methods.

I. INTRODUCTION

With the advancements in information technology, there is a deep knowledge of the ocean in images from which a user intends to retrieve relevant and specific information. By using a certain number of applications, a user can extract multiple types of knowledge from images, such as the creation of augmented reality-based three-dimensional simulation of the skull to perform an accurate on-line surgical treatment of oral and maxillofacial infections [1]. Similarly, radiologists can use multiple imaging modalities, including x-ray, positron emission tomography, computed tomography, and magnetic resonance imaging, to detect acute diseases [2]. A user must perform a set of operations at the time of knowledge extraction, which includes image segmentation and registration etc. The main goal of segmentation is to segment focused and non-focused regions in a particular image for image restoration [3]. An image is an artifact that draws a visual interpretation of a particular object in a two-dimensional photograph or a simple picture. Images can exist in multi-dimensions; however, the most common multidimensional images are two-dimensional [4] and three-dimensional images [5]. Image processing can be further classified into two categories such as computer graphics and computer vision. In computer graphics, a user manually creates a particular image using multiple image objects such as environment, lightings, etc., without capturing them from a natural environment. The most common examples of computer graphics are animated movies [6]. On the other hand, computer vision is a high-level image processing. A system intends to retrieve a signature of particular information from a sequence of images such as magnetic resonance imaging etc. [7]. Now a day, image processing has become the focus of multiple scientists because of its exponentially growing scope in scientific visualization. Currently, scientists are solving multiple problems of science using a powerful tool of image processing.

Field depth is also known as a focus range of optical lenses. Field depth is used to estimate the distance between nearest and farthest artifacts in an image. Due to the limited field depth of camera lenses, the camera focuses only on the areas which lie in its focus plane [8]. As a result, some regions in an image might be out of focus or blurred. A renowned methodology to address this issue is multi-focus image fusion. Data fusion is a process, which

uses information and data which is being generated from multiple sources to obtain refined information for decision making. Generally, image fusion is a process that combines spatial detail from two or more input images to create a new contemporary image by using a specific methodology [9]. During the process of image fusion, image $I_1(x, y)$ and $I_2(x, y)$ are fused into a new contemporary image IF (x, y), figure 1 below presents the visual interpretation of this phenomena.

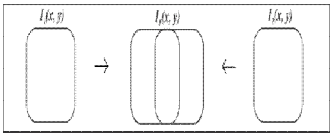

Figure 1: Principle of Image Fusion.

Similarly, Multi-focus image fusion generates a composite image that enhances the field depth of an image. Enhanced field depth creates an image with an efficient focus range using a set of multi-focus images. A composite image, also known as a fused image, will be more appropriate for multiple image processing applications and human perception. Image fusion can be of two types: Spatial Domain Image Fusion and Transform Domain Image Fusion [10]. Figure 2 presents a visual interpretation of this classification.

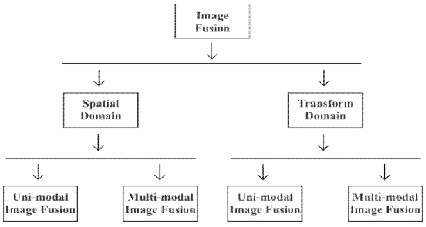

Figure 2: Domains of Image Fusion.

1.2 Applications of Image Fusion

There are multiple applications of image fusion which are enlisted below:

1. Weapon and Aircraft Detection
2. Medical Imaging
3. Agriculture Imaging
4. Digital Photography

1.3 Problem Statement: Image fusion is a challenging task due to irregular boundaries of focused and non-focused regions. In literature, different features have been employed to characterize focused parts in an image. For better results, an ensemble of different shape and texture-based features can be used to characterize better focused and non-focused regions. Furthermore, it is required to find optimal weights of different features to define their individual contribution in a fused image.

II. LITERATURE REVIEW

2.1 Spatial Domain Image Fusion

The study in [11] suggested a fusion methodology (illustrated in figure 3) to fuse source images into one single image with higher visual interpretation. In this paper, it is proposed to fuse input source images by weighted average, which can be computed by estimating weights obtained from the image detail extracted from the source images using a cross bilateral filter. The proposed approach was evaluated visually and quantitatively over number of multi-focus source images.

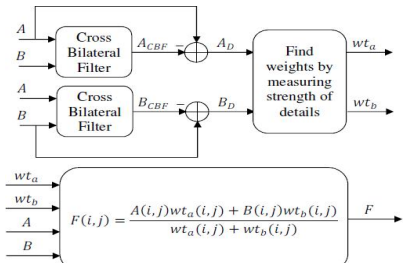

Figure 3: Fusion Methodology Proposed by Kumar etal

Chantara et al. [12] proposed a method for the extension of field depth in source images. Their study used a spatial frequency and sum of Laplacian for estimation of image coefficients from source images. Using extracted coefficients, they decided focused regions from images, which are later combined using image fusion.

Wang et al. [13] discussed multi-focus image fusion by suggesting their novel approach. The authors performed image fusion using the Laplacian operator and region optimization. They generated an initial decision map using the Laplacian operator, as Laplacian efficiently detects focused and non-focused areas from source images. Later, they refined their decision map by using region optimization.

Tang et al. [14] studied the problem of multi-focus image fusion by using a Convolutional Neural Network (CNN). Authors have used CNN for the classification of pixels of source images as focused, defocused, and uncertain. They generated an initial decision map by comparing intensities of input pixels. Later, they refined a decision map for image fusion by removing non-focused pixels.

Zhang et al. [15] suggested a novel approach for multi-focus image fusion using saliency analysis. The authors detected focused regions from multiple source images using the GBVS algorithm. Later, they used watershed and morphological methods Literature Review 18 to get a refined saliency map for fusion. Focused regions of source images are fused directly. However, the remaining regions are fused using shearlet transform methods.

2.2 Transform Domain Image Fusion

The study, Mitianoudis et al. [16] proposed the notion (presented in figure 4) of preserving the active areas of multi-focus input images in a composite image. In this study, the authors have implemented independent component analysis (ICA) approaches and topographical independent component analysis using novel fusion rules to create a particular composite image in the transform domain. According to study authors, their approach has performed slightly better than traditional wavelet approaches.

In another study of Liu et al. [17] proposed a multi-focus image fusion procedure in the transform domain with higher

complementary information of source images in convolutional representation for image fusion. However, convolutional sparse representation-based image fusion is the advanced form of sparse representation-based image fusion. To validate the effectiveness of the proposed approach, the authors have evaluated their approach visually and quantitatively.

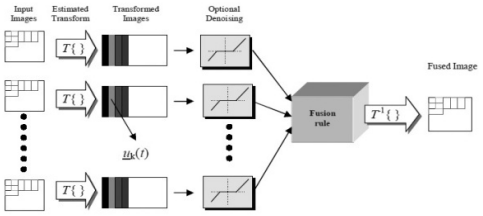

Figure 2.2: Fusion Methodology Proposed by Mitianoudis et al

2.3 Weight Optimization using Evolutionary Algorithm.

Kumar et al. [18] proposed a methodology for detecting ductal carcinoma based on image fusion using weight optimization through a genetic algorithm. Ductal carcinoma is the deadliest disease in women, which spreads exponentially and leads to death in its final stage. The proposed methodology creates an enhanced composite image from a set of X-ray mammography images based on discrete wavelet transform method using genetic algorithm-based weight optimization at pixel level weighted average.
Kulkarni et al. [19] addressed the problem of multi-focus and multi-spectral image fusion by proposing their approach. In their study, the authors have used a genetic algorithm for finding the optimal weights. However, the obtained optimal weights are multiplied with the input source images to create activity maps. Later pixel-level fusion is performed to create a visually enhanced optimal image.

III. METHODOLOGY

This comprehensive and diversified methodology comprises multiple steps such as a collection of benchmark datasets, feature extraction using local features, weight optimization using genetic algorithm, activity map generation and image fusion, etc. For this study, 09 benchmark datasets are collected. Each dataset represents a pair of two images, such as left and right focused images, foreground and background focused images, as these datasets are publicly available and contain multi-focus images. Preprocessing is not required in this study because each pixel of the input source image is essential for the fusion. Later, three blur segmentation maps are generated for each input image using shape and texture-based local features such as Local Binary Patterns (ωLBP), Sobel Operator (ωSO), and Laplacian Operator (ωLO). Blur segmentation map indicates the classification of the pixel of an image as focused or non-focused. The optimal weights are obtained using a genetic algorithm that decides each feature's contribution in an activity map. By comparing each pixel's intensity of both activity maps (i.e., left and right focused images), a fused image is generated. A fused image's effectiveness is evaluated using visual and quantitative 25 Image Fusion and Feature Extraction 26 analysis using specific statistical and mathematical-based quality measures such as Normalized Mutual Information, Average Gradient, and Yangs quality metric.

A. Local Binary Patterns

Local Binary Pattern (LBP) is a texture-based feature extraction technique [20]. It was initially proposed for the classification of texture in an image I(x, y). This feature is most commonly used because of its invariance to grey level changes. LBP takes a neighborhood value around a pixel, such as an image block of (3 × 3) pixels. It compares the intensity of the central pixel with the intensities of its surrounding pixels. If the pixel intensity is greater than or equal to the central pixel, then 1 is assigned to a corresponding pixel; otherwise, 0 will be assigned. This comparison returns an 8-bit binary sequence, which is to be converted into an equivalent decimal value. The intensity of a central pixel is replaced with a resultant intensity. The intensity of a corresponding pixel is replaced by the intensity of an image block's central pixel.
Similarly, local binary patterns are calculated for all pixels using the sliding window. The obtained binary patterns are collected in 255 bin histograms; however, the intensity values usually lie between 0 and 255. Mathematically, LBP can be expressed as equation 3.1.

$$\omega_{LBP} = \sum_{i=0}^{8} f(p_i - p_c) \times 2^n \qquad (3.1)$$

$$\text{where} f(x) = \begin{cases} 0 & p_c < p_i \\ 1 & p_c \geq p_i \end{cases} \qquad (3.2)$$

In equation **3.1**, *pc* is a central pixel. However, *pi* is a neighboring value. The working of LBP is expressed in figure 5

4	2	1
3	2	3
2	1	2

1	1	0
1		1
1	0	1

1	1	0
1	189	1
1	0	1

189		

Figure 5:Workingofω_{LBP}UsingaLocalPatchSizeof3×3

Acompletealgorithmispresentedin1

Algorithm 1 Local Binary Pattern

INPUT: I(x, y), local patch size of n × n where n mod 2 ≠ 0 AND n≠ 1.
OUTPUT: LBP image ωLBP (x, y).
Read the input image as I_g (x, y).
repeat
Apply an overlapping sliding window, with a block size of $n \times n$.
Take the first image block of nn along with its pixel intensities as $S_w(I_g(n,n))$.
Take the intensity of a central pixel as a threshold T in that particular window. Compare the intensities of neighboring pixels with T:
if ($p_n(x, y) \geq$ T), return 1.
if ($p_n(x, y)$<T), return 0. Return an 8-bit binary code.
Convert 8-bit binary code into decimal.
Replace the value of central pixel, with obtained decimal value. Collect binary codes in a 255-bin histogram.
until The image is not completely compared.
return Normalized LBP image as ωLBP (x, y).

B. Sobel Operator

Sobel operator is a shape-based feature extraction technique [21]. It was initially developed for edge detection in an image $I(x, y)$; however, an image's edge indicates the most prominent part of changed intensity. It is a gradient-based feature extraction method. The Sobel operator takes a neighborhood region $R(x,y)$ of 3 × 3 around a pixel. It computes the gradient in *the x-direction* and similarly in *the y-direction* using equations 3.3 and 3.4. The gradient in *the x-direction* can be calculated by convolving the selected region with M_1 using equation 3. The matrix M_1 can be expressed as:

$$M_1 = \begin{bmatrix} -1 & 0 & -1 \\ -1 & 0 & -1 \\ -1 & 0 & -1 \end{bmatrix}$$

However,thegradientin*y*canbecalculatedbyconvolvinga regionwithmatrix.

M_2whichcanbedefinedas:

$$M_2 = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 0 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

This convolution return two gradient matrices $F_x$and F_y. The total gradient G canbe calculated using equation 3.5. A complete algorithm is presented in algorithm2.

$$F_x(x,y)=R\times M_1 \quad \textbf{(3.3)}$$

$$Fy(x, y) = R \times M_2 \quad \textbf{(3.4)}$$

$$G(x,y) = \sqrt{(F_x)^2 + (F_x)^2} \quad \textbf{(3.5)}$$

$$\theta = \tan^{-1}\frac{F_x}{F_y} \quad \textbf{(3.6)}$$

$$\omega_{SO}(x,y) = G(x,y) \quad \textbf{(3.7)}$$

Algorithm 2 Sobel Operator

INPUT: $I(x,y)$, local patch size of $n \times n$ where n mod $2 \neq 0$ AND $n \neq 1$.
OUTPUT: SOB image $\omega_{SO}(x,y)$.

Read the input image as $I_g(x,y)$. repeat
Apply an overlapping sliding window, with a block size of n × n.
Take the first image block of nn along with its pixel intensities as $S_w(I_g(n,n))$.
Estimate the amount of gradient in horizontal direction as $F_x(x,y)$.
Estimate the amount of gradient in vertical direction as $F_y(x,y)$.
Estimate the magnitude of obtained gradients as $G(x,y)$.
Replace the intensity of a central pixel, with obtained magnitude.
Estimate the direction of gradient as $\theta^o \geq \theta \leq 180^o$.
Create a histogram, having nine bins of 20^o each.
Store the obtained magnitude, in the bins of the histogram.
until the image is not completely compared.
return Normalized SOB image as $\omega(x,y)$.

C. Laplacian Operator

Laplacian operator is also a shape-based feature extraction technique [22]. It is developed for edge detection in an image. As compared with the previously proposed edge detection techniques such as Sobel Operator, Laplacian operator is a second-order partial derivative. Mathematically, the Laplacian operator can be expressed in equation 3.8.

$$I^l(x,y)=\partial^2(x,y)\times I(x,y) \quad (3.8)$$

Where $\partial^2(x, y)$ is a Laplacian operation over an image $I(x, y)$, it takes a neigh-boring region such as 3 × 3 around a pixel, where it convolves the selected region $R(x,y)$withmatrixM_3usingequation3.9

$$M_3 = \begin{bmatrix} -1 & -1 & -1 \\ -1 & -8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

$$\omega_{LO}(x, y) = R \times M_3 \quad (3.9)$$

This convolution operation returns one large matrix containing a brief detail of edges of an image. A complete algorithm is given in algorithm 3.

Algorithm 3 Laplacian Operator

INPUT:I(x, y), local patch size of n × n where n mod 2 ≠ 0 AND n ≠ 1.
OUTPUT: Laplacian image ωLO(x, y).

Read the input image as $I_g(x, y)$.
repeat
Apply an overlapping sliding window, with a block size of n × n.
Take the first image block of nn along with its pixel intensities as Sw(Ig(n, n)). Estimate the amount of gradient in Ig(x, y) by using equation 11.
Replace the intensity of a central pixel with the obtained value.
untilThe image is not completely compared.
return Normalized Laplacian image as ωLO(x, y).

D. Weight optimization using genetic algorithm.

Genetic algorithm is one of the powerful optimization techniques [22] adopted from the phenomena of nature. A genetic algorithm makes a simulation of evolution to find the best chromosomes. Unlike other optimization techniques, the genetic algorithm never uses a partial derivative to find a local minimal. Randomness plays a quintessential role in genetic algorithms, making it hard to search the chromosome space to find the best chromosomes. Figure 6 below provides the flow chart of the genetic algorithm.

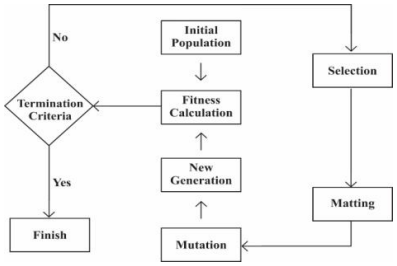

Figure 6: FlowDiagram of Genetic Algorithm.

Initially, a genetic algorithm initiates a population of randomly generated chromosomes. These randomly generated chromosomes are evaluated according to their fitness values. However, a fitness value is a numeric value. Suppose an initial population does not fulfill the termination criteria. In that case, the best chromosomes are selected based on higher fitness scores. The genetic algorithm's first operator is selection, where a pair of chromosomes is chosen based on higher fitness values.

Algorithm 4 Laplacian Operator

Initiate a random population of chromosomes.
repeat
Calculate fitness values of randomly generated chromosomes.
repeat
Select best chromosomes using higher fitness values. Perform matting using selected parents.
Perform mutation.
Compute fitness of new chromosomes.
until Creation of new population.
until Termination criteria is not fulfilled.
return best chromosome [θ0, θ1, θ2] from best chromosome population.

i) Matting

Amatting process is performed using a selected pair. It creates two more chromosomes, which replace the non-selected chromosomes in the previous generation. The number of chromosomes is to be the same as of the previous generation. This sequence of instruction is kept in execution continues until the termination criteria are not fulfilled.

ii) Initial Population In the first phase of the genetic algorithm, five chromosomes' random Population is generated. Each chromosome contains three genes. However, the sum of these genes should exactly be one. The main reason for applying this constraint is that we want to select the best combination of genes. These weights decide the contribution of each feature in an activity map for the image fusion.

iii) Selection In the second phase, the best chromosomes are selected for the next generation based on their fitness scores. However, this Selection can be made using specific selection methods such as Roulette Wheel, Fittest Half, and Random. In this study, the best chromosomes are selected using the roulette wheel methodology. In this type of selection method, each chromosome has a chance to be selected, including a lower fitness score. As it is generally observed that sometimes chromosomes having a lower fitness score generates outstanding results.

iv) FitnessFunction estimates the fitness value of a chromosome; calculating the fitness function depends on the optimization problem. If one wants to optimize a process, then a process should be implemented into the fitness function. In this dissertation, the fitness scores are calculated using the image entropy as in equation.3.10.

$$\text{Image Entropy} = -\sum_{i=0}^{255} (p_i) \log_2 (p_i) \qquad (3.10)$$

In each generation, there are five chromosomes and each chromosome consist of three genes, where θ_0, θ_1 and θ_2 are the individual genes of each chromosome which decide contribution of each feature in a fused image. Therefore, five images are generated using each chromosome such as $I_{F1}(x,y)$, $I_{F2}(x,y)$, $I_{F3}(x,y)$, $I_{F4}(x,y)$ and $I_{F5}(x,y)$ however, each image is evaluated using equation 3.10. The formulation of new images can be modeled using equations 3.11to 3.13where ω_{LBP} is a local binary pattern, ω_{LO} is a Laplacian operator and ω_{SO} is a Sobel operator.

$$I_1'(x,y) = \theta_0 \times \omega_{LBP} + \theta_1 \times \omega_{LO} + \theta_2 \times \omega_{SO} \qquad (3.11)$$

$$I_2'(x,y) = \theta_0 \times \omega_{LBP} + \theta_1 \times \omega_{LO} + \theta_2 \times \omega_{SO} \qquad (3.12)$$

$$I_{Fn}(x,y) = \begin{cases} I_1(x,y) & \text{if } I_1'(x,y) \geq I_2'(x,y) \\ I_2(x,y) & \text{Otherwise} \end{cases} \qquad (3.13)$$

In this experimentation, a fitness function maximizes function because we are interested in studying variations of pixel intensities in input source images. Therefore, a chromosome having a maximum entropy score is to be selected as the best chromosome.

v) Matting

The next step of the genetic algorithm is matting, which is also known as a crossover; the crossover can be multiple types, such as single-point crossover and two-point crossover. However, in this study, a procedure of single-point crossover is implemented. As a result, two new chromosomes are generated. At this stage, there is a clear chance that the sum of each newly created chromosome might be more significant or less than one. Therefore, the algorithm presented in 5 is implemented for scaling obtained weights.

Algorithm 5 Algorithm for Weight Scaling.

```
Initiate a random population of chromosomes
INPUT: Weights Array C[].

OUTPUT: Scaled Weights Array C[]
        Read weights array as C[].
        totalsum = sum(C[]) if totalsum >1 OR totalsum <1 then
                while totalsum ≠do
                for i in range(length(C)) do
                        C[i] = C[i] / totalsum
                                end for
                totalsum = sum(C[]) if totalsum == 1 then
                break
                end if
                end while
        end if
return Scaled Weights as C[].
```

vi) Mutation

The final operation of the genetic algorithm is mutation. The process of mutation is performed to improve the variety of chromosomes in the next generation. The chromosomes selected through elitism are not screened through mutation. The main reason for this constraint is that they are already the best solution. Newly generated chromosomes are screened through the process of mutation. Otherwise, they are identical alike their parents. The mutation can be performed using multiple methods such as Gauss and Reset etc. However, a mutation is implemented using the methodology of Gauss.

vii) New Generation

A genetic algorithm is a random optimization approach as it enhances a current solution by making random changes. However, It is not guaranteed that it can always generate outstanding results because of its randomness. Therefore parent solutions from the last generation are always kept in a new generation to obtain a better solution. At this point, one can create a new population of chromosomes which represent the next generation.

viii) Termination Criteria On successful creation of a new generation, specific termination criteria can be employed to validate whether a genetic algorithm should produce another generation of chromosomes or it gets halted. Therefore, multiple types of termination criteria can be employed at the same time. If any criteria among all fulfill termination requirements, then a genetic algorithm gets halted. In this study, two termination criteria are employed simultaneously, such as a fixed number of iterations and no significant change in chromosomes. A genetic algorithm is kept in execution for a limited number of generations such as 500, 1000, or higher in a fixed number of iterations-based criteria. Similarly, in another case, if there is no significant change in chromosome generations, then a genetic algorithm can be halted to create another generation.

ix)Generation of Activity Maps In this study phase, optimized feature weights are obtained using a genetic

algorithm discussed in previous sections. From generated populations of chromosomes, the best population is selected based on its fitness value. From the selected population, the best chromosome is to be selected using its fitness score. However, these chromosomes' genes are used as feature weights θ_0,$\theta_1$andθ_2, respectively. Later, shape and texture-based features are estimated from input images such as ω_{LBP},$\omega_{LO}$and ω_{SO}.. This process returns three images for each input image such as $\alpha_{\omega_{LBP}}(x, y)$, $\alpha_{\omega_{LO}}(x, y)$, $\alpha_{\omega_{SO}}(x, y)$ $\alpha_{\omega_{LBP}}(x, y)$,$\alpha_{\omega_{LO}}(x, y)$, $\alpha_{\omega_{SO}}(x, y)$ similarly for second input image,$\alpha_{\omega_{LBP}}(x, y)$, $\alpha_{\omega_{LO}}(x, y)$, $\alpha_{\omega_{SO}}(x, y)$ in total there are six activity images for eachpair of images. A combined activity map is created using each pixel of segmented maps. However, each obtained pixel is assigned a weight of the selected gene. The resultant value is a feature score for the activity maps. Mathematically, activity maps can be labeled as $\alpha_1^J(x,y)$and$\alpha_2^J(x,y)$. This image formulation is modeled using equation 3.14 and 3.15.

$$\alpha_1'(x,y) = \theta_0 \times \alpha_{\omega_{LBP_1}}(x,y) + \theta_1 \times \alpha_{\omega_{LO_1}}(x,y) + \theta_2 \times \alpha_{\omega_{SO_1}}(x,y) \quad (3.14)$$

$$\alpha_2'(x,y) = \theta_0 \times \alpha_{\omega_{LBP_2}}(x,y) + \theta_1 \times \alpha_{\omega_{LO_2}}(x,y) + \theta_2 \times \alpha_{\omega_{SO_2}}(x,y) \quad (3.15)$$

x) Improved Image Fusion Scheme For implementing an improved fusion scheme, the sequence of steps mentioned in figure 3.6 is employed. For convenience, this fusion procedure is presented only for two input images, such as $I_1(x,y)$and$I_2(x,y)$, However, a generalized algorithm for more than two input images is discussed at the end of this section. From figure 3.5, it can be seen that the proposed methodology consists of multiple steps. In the first step, the proposed method uses a renowned evolutionary algorithm such as a genetic algorithm to obtain optimal weights, i.e., θ_0, $\theta_1$and θ_2 whereθ_0 +θ_1+θ_2=1. First of all, a random population of five chromosomes is generated. However, each chromosome is composed of three genes, respectively. Later, each chromosome's fitness is evaluated using image entropy, as discussed in the earlier section. Each chromosome is ranked according to its fitness values. The top two chromosomes with higher fitness values are selected from a population of parent chromosomes to initiate matting. To produce new offspring, a single point crossover to be performed. As a result, two new offspring are produced and replace the chromosomes ranked in the 3rd and 4th grade. There is a need for genetic diversity in new offspring. Otherwise, they are identical to their parents. To introduce genetic diversity in newly born chromosomes, a mutation process is initiated, creating genetic diversity in the offspring. To maintain the elitism, parent chromosomes are also dispensed in the next generation, along with a chromosome having the lowest fitness value. Now the new generation comprises two-parent chromosomes, mutated chromosomes, and the last one with a lower fitness score. The primary reason for keeping the chromosome with a lower fitness value is that it might produce outstanding offspring in future generations. The process is kept in execution until a termination criterion is not fulfilled. In the end, we select the best population of chromosomes from all generated populations. From the best-selected population, a chromosome is extracted to obtain optimal weights. In the second step, LBP image using ω_{LBP}, Laplacian image using ω_{LO} and Sobel image by using ω_{SO} over a pair of input source images $I_1(x,y)$and$I_2(x,y)$ are calculated which return six segmented activity images such as $\alpha_{\omega_{LBP_1}}(x, y)$,$\alpha_{\omega_{LO_1}}(x, y)$, $\alpha_{\omega_{SO_1}}(x, y)$, similarly for the second input image, $\alpha_{\omega_{LBP_2}}(x, y)$, $\alpha_{\omega_{LO_2}}(x, y)and\alpha_{\omega_{SO_2}}(x,y)$ respectively. Using segmented images, two general activity images α $\alpha_1^J(x,y)$and$\alpha_2^J(x,y)$are created for each pair of input source images usingMethodology using Two Input Images.

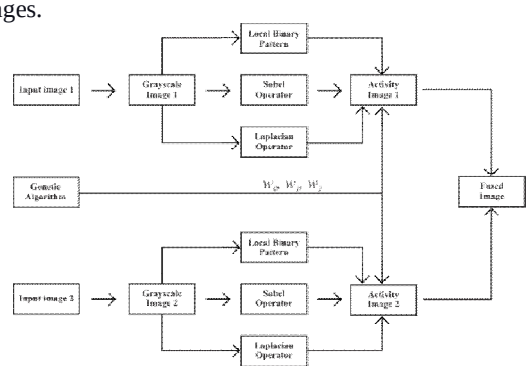

Figure 7: Block Diagram of Weighted Fusion

However, equation 3.15 and 3.16 obtained values of corresponding pixels to be stored in the newly created two black activity maps.

Here $\alpha_1^J(x,y)$ and $\alpha_2^J(x,y)$are the activity maps, which contain focused and non-focused pixels. In the final phase of the proposed methodology, the image is fused using activity maps $\alpha_1^J(x,y)$and$\alpha_2^J(x,y)$according to the scheme proposed in equation 3.17.

$$I_F(x,y) = \begin{cases} I_1(x,y) & \alpha_1'(x,y) \geq \alpha_2'(x,y) \\ I_2(x,y) & \text{otherwise} \end{cases} \quad (3.17)$$

In equation 3.19, $I_1(x, y)$ and $I_2(x, y)$ are the two input source images. In this step, both $\alpha_1^J(x,y)$ pixel intensities and $\alpha_2^J(x,y)$images are compared. If the pixel intensity of α $\alpha_1^J(x,y)$is more than or equal to $\alpha_2^J(x,y)$, then a corresponding pixel from $I_1(x, y)$ is registered in a newly created black image at the corresponding index. Otherwise, a corresponding pixel from $I_2(x,y)$ is to be registered. This scheme is kept in execution until the complete registration of all corresponding pixels at all corresponding pixels.

Algorithm 6 Genetic Algorithm based Multi-focus Image Fusion using Local Features.

Input (x,y),localpatchsizeof$n\times n$where$n$ mod2≠0$AND n\neq 1$
OUTPUT: Fusedimage$I_F(x,y)$.
Readtheinputimagesas$I_m(x,y)$.
$\theta_0, \theta_1, \theta_2$= 1= execute-Genetic-Algorithm(), where $\theta_0+\theta_1+\theta_2=1$
$\alpha_{\omega_{LBP1}}, \alpha_{\omega_{LO1}}, \alpha_{\omega \mathrm{SO}_1}$ = calculate-Sharpness($I_m(x,y)$).
$\alpha_{\omega_{LBP_2}}, \alpha_{\omega_{LO2}}, \alpha_{\omega \mathrm{SO}_2}$=calculate-Sharpness($I_{m+1}(x,y)$).
for$i$=0*toimageheight***do**
for j = 0 to imagewidth; **do**
$\alpha_1'(i,j) = \theta_0 \times \alpha_{\omega_{LBP_1}}(i,j) + \theta_1 \times \alpha_{\omega_{LO_1}}(i,j) + \theta_2 \times \alpha_{\omega_{SO_1}}(i,j)$
$\alpha_2'(i,j) = \theta_0 \times \alpha_{\omega_{LBP_2}}(i,j) + \theta_1 \times \alpha_{\omega_{LO_2}}(i,j) + \theta_2 \times \alpha_{\omega_{SO_2}}(i,j)$
end for
end for

for i = 0 to imageheight **do**
for j=0 *to imagewidth*; **do**
if($\alpha_1^J(i,j) \geq \alpha_2^J(i,j)$)
$I_F(i, j) = I_m(x, y)$
Else
$I_F(i,j) = I_{m+1}(x,y)$
endfor
end for

IV. RESULTS AND DISCUSSIONS

In this study, multiple experiments are being conducted to obtain best results of image fusion. The image fusion is performed using a certain number of shape and texture-based features such as ω_{LBP},ω_{LO}, ω_{SO}. These techniques are applied and verified over selected datasets and obtained multiple activity maps for image fusion. Later, these activity maps were combined using optimal weights which were obtained through an evolutionary algorithm such as genetic algorithm and the combined activity maps are used to restore a particular image. The list of performed experiments can be described as follow:

a) Effect of Blur on Image Texture

The first experiment is related to the effect of blur on image texture. Therefore, a vigorous investigation is conducted to evaluate impact of different blur magnitudes on image texture. This experimentation is performed over a selected set of input images such as a road image and snipper image using blur radiuses βr of 0.25, 0.5,0.75 and 1.0. The effectiveness of this experiment is evaluated using statistical measures of standard deviation and variance.

To study variations of quality degradations in image texture over multiple blurmagnitudesβ_r,anexperimentationisconductedovera selectedsetofinputimages.To identify texture in an input image, a texture-based operator such as ω_{LBP} is applied using a local patch size of 33.The obtained image is blurred in five iterations of β_r= 0.0, 0.25, 0.50, 0.75, 1.0 to obtain degraded image textures asD_{T_0} (x, y), D_{T_1} (x, y), D_{T_2} (x, y), D_{T_3} (x, y) and D_{T_4} (x, y), however, each obtained image is evaluated in each iteration.

Figure 7 below, draws an illustration of texture degradations in road image over multiple blur magnitudes. In figure 1(a), original input image of road is presented. Figure 1(b) illustrates image texture at βr = 0, however, figures 1(c) to 1(f) present texture degradations at βr = 0.25. 0.50, 0.75, 1.0. The evaluation of these degradations is illustrated in figures 1(g) and 1(h) using standard deviation and variance, respectively. From figures it can be observed that there is no degradation in image texture in between βr = 0.0 and βr = 0.25 as obtained results are closer which indicate that ωLBP is robust to small amount of blur. However, on other blur magnitudes, a significant degradation is observed in image texture. The results indicate that a texture degradation is directly proportional to a greater blur magnitude.

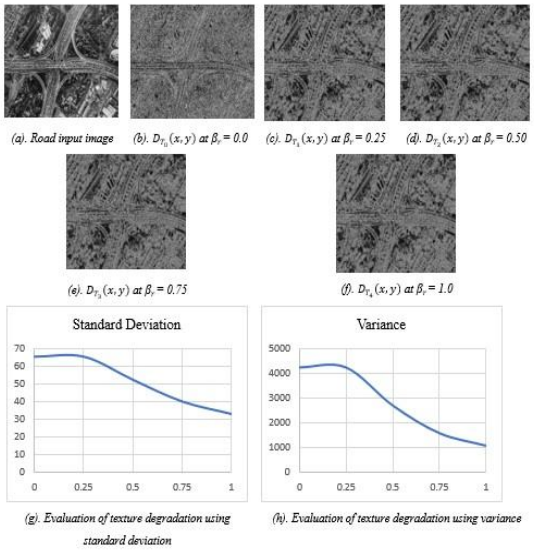

Figure 7: Evaluating Effect of Blur on Image Texture using Road Image.

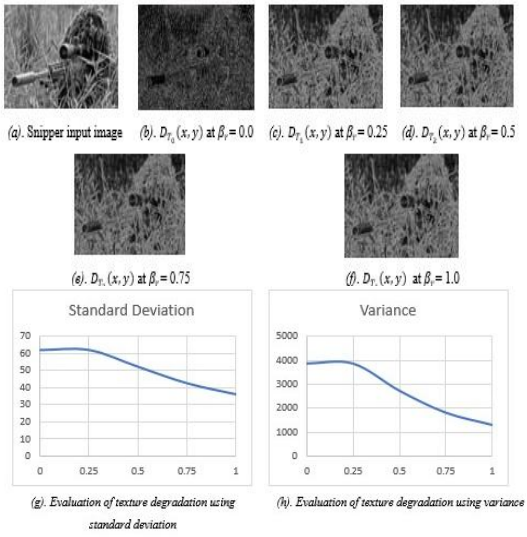

Figure 8: Evaluating Effect of Blur on Image Texture using Snipper Image.

Figure 8presents texture degradations in a snipper image over different blur radius. The original input image of a snipper is illustrated in figure 2(a). Figure 2(b) contain classified image texture at β_r = 0, however, the texture degradations on β_r = 0.25. 0.50, 0.75, 1.0 is illustrated in figures 2(c) to 2(f). The performance of these degradations is presented using figures of 1(g) and 1(h) using variance and standard deviation, respectively. It is evident from figures, there is no significant degradation in image texture in between β_r = 0.0 and β_r = 0.25 as obtained results are closer. However, other blur radiuses presented significant degradation in image texture.

The discussion in this section shows a relationship graphs of standard deviation and variance with overall consistency of a texture over a certain amount of blur. The major reason of evaluating this relationship is that a blur directly effects consistency of a texture. However, it is also evident from the obtained scatter plots, that texture is significantly degraded when the amount of blur increases

because regions of an image get smoother over a significant increase in blur magnitude.

b) Effect of Blur on Shape using Laplacian Operator

The second part of first experiment is related to the effect of blur on object edges in an input image. Therefore, another experimentation is conducted to evaluate significant impact of multiple blur magnitudes on image edges. This experimentation is performed over two selected images such as a road image and snipper image by employing blur radiuses βr of 0.25, 0.5, 0.75 and 1.0. The statistical measures of standard deviation and variance are used to evaluate its effectiveness.

To study variations of degradations in image edges using multiple blur magnitudes β_r, an experimentation is conducted over a selected set of input images. To classify edges in an input image, a second order derivative-based shape operator such as ω_{LO} is applied using a local patch size of 33. Later, obtained image is blurred in five iterations of β_r = 0.0, 0.25, 0.50, 0.75, 1.0 to obtain degraded image edges as $DE0(x,y)$, $DE1(x,y)$, $DE2(x,y)$, $DE3(x,y)$ and $DE4(x,y)$, however, each obtained image with degraded edges is evaluated in each iteration.

Figure 9, contains variations of edge degradations in a road image over multiple blur magnitudes. The original road image of road is illustrated in figure 3(a). The effect of blur on image edges at β_r = 0 is presented in figure 3(b), however, edges degradations at β_r = 0.25. 0.50, 0.75, 1.0 is dispensed in figures 3(c) to 3(f). The evaluation of these degradations is performed infigures of 3(g) and 3(h) using standard deviation and variance respectively. From obtained scatter plots a significant degradation is observed in image edges, as consistency of edges is decreasing significantly. The results indicate that edges degradation is directly proportional to blur magnitude.

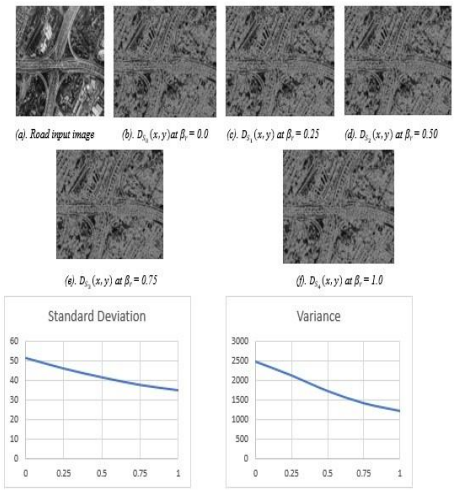

Figure 9: Evaluating Effect of Blur on Shape using Laplacian Operator on Road Image

Figure 10presents variations of edges degradations in a snipper image over multiple blur radiuses. Figure 3(a) contains an original road image. The impact of blur on image edges at β_r = 0 is illustrated in figure 3(b), however, variation of edges degradations on β_r = 0.25. 0.50, 0.75, 1.0 is presented in figures 3(c) to 3(f). The degradations are studied using standard deviation and variance which is dispensed using figures of 3(g) and 3(h) respectively. These two scatter plots show a strong degradation n image edges, as consistency of edges is decreasing significantly.

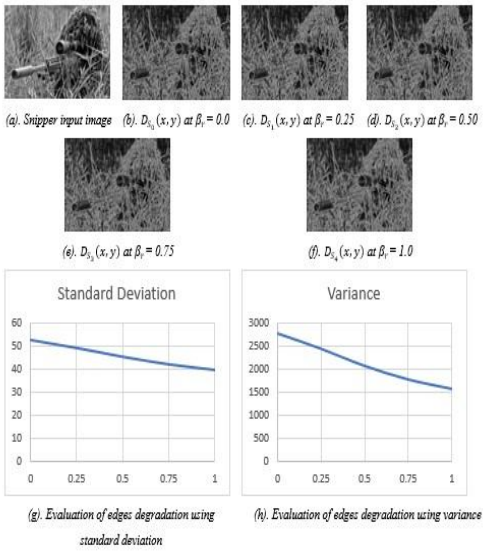

Figure 10: Evaluating Effect of Blur on Shape using Laplacian Operator on Snipper Image.

The discussions in this module delineate an effective relationship between edges consistency and amount of blur. It presents an association curves of standard deviation and image variance with overall consistency of edges in an image. The reason of finding this relationship is that a blur compromises the consistency of edges in an input image. However, it can be visualized from obtained curves, that edges information decreases when the amount of blur increases. Because smoothness increases over significant increase in blur.

c) Effect of Blur on Shape using Sobel Operator

The last part of experiment is related to the blur effect on object edges in a sobel image. Therefore, this experimentation is conducted to evaluate a significant role of different blur magnitudes on image edges. The experimentation is performed using road and snipper images by taking blur radiuses $_r$ of 0.25, 0.5, 0.75 and 1.0. The effectiveness of this activity is evaluated using statistical measures of standard deviation.

Detecting a particular edge in an input image employs a variety of multiple statistical and mathematical model-based procedures which aim at classifying relevant points in a relevant input image where a brightness is changed significantly or, more formally, has multiple loopholes. A set of points in an input image where brightness changes quintessentially are generally referred as a pool of curved line segment-based edges. In case of estimating loopholes in a single channel image is referred as a step detection and

resolving image signal loopholes in a unit time can be titled as detecting a change in pixel intensity. However, finding a relevant edge is a general requirement in processing an input image for multiple appliances such as machine vision, furthermore, it has been proven effective for extraction and detection of a relevant feature.

To study loss of edges information over multiple blur magnitudes $_r$, an experimentation is conducted over selected input images. To classify edges in an input image, a first order derivative-based shape operator such as ω_{SO} is employed using a local patch size of 33. Later, sobel image is blurred in five iterations of β_r = 0.0, 0.25, 0.50, 0.75, 1.0 to obtain degraded edges information as $D_{S0(x,y)}$, $D_{S1(x,y)}$,$D_{S2(x,y)}$, $D_{S3(x,y)}$ and $D_{S4(x,y)}$, however, each degraded image is evaluated in each iteration.

Figure 11 contains variations of edge degradations in a road image over multiple blur radiuses. The original image of road is illustrated in figure 5(a). The effect of blur on image edges at β_r = 0 is presented in figure 5(b), however, edges degradations at β_r = 0.25. 0.50, 0.75, 1.0 is dispensed in figures 5(c) to 5(f).

The evaluation of these degradations is performed in figures of 5(g) and 5(h) using standard deviation and variance respectively. These scatter plots present a significant degradation in image edges, as consistency of edges is decreasing significantly. The results indicate that edges degradation is directly proportional to blur magnitude.

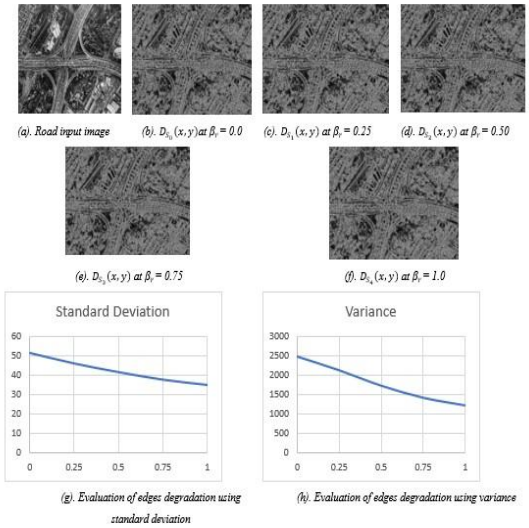

Figure 11: Evaluating Effect of Blur on Shape using Sobel Operator on Road Image.

Figure 12presents edge degradations in a snipper image over multiple blur magnitudes. The original snipper image of road is presented in figure 6(a). The effect of blur on image edges at β_r = 0 is illustrated in figure 6(b), however, edges degradations at β_r = 0.25. 0.50, 0.75, 1.0 are presented in figures 6(c) to 6(f). The evaluation of these degradations is performed in figures of 6(g) and 6(h) using standard deviation and variance respectively. These scatter plots indicate a significant degradation in image edges, as consistency of edges is decreasing significantly. The results indicate that edges degradation is directly proportional to blur magnitude because image smoothness increases when significantly blur magnitude increases.

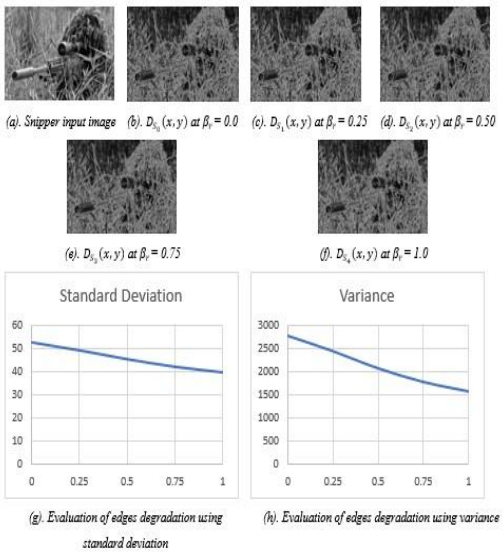

Figure 12: Evaluating Effect of Blur on Shape using Sobel Operator on Snipper Image

1) *Comparison with weighted approaches*

The goal of image fusion is to accurately preserve each detail of input images in a fused image. Therefore, fused images are evaluated using certain statistical and mathematical model-based quality measures such as NMI, AG and YC respectively. Furthermore, a comparison is performed with previously state-of-the-art fusion methods to find out accuracy of a proposed approach. Figure 13 presents a comparison of proposed fusion methodology with previous methodologies such as DCHWT, WSSM, CBF, ICA, CSR have produced lowest evaluation results. Other fusion methodologies such as SRCF and IFM have produced lower evaluation scores than proposed methodology. In this phase, a fused image is created from local features using a patch size of 3 × 3

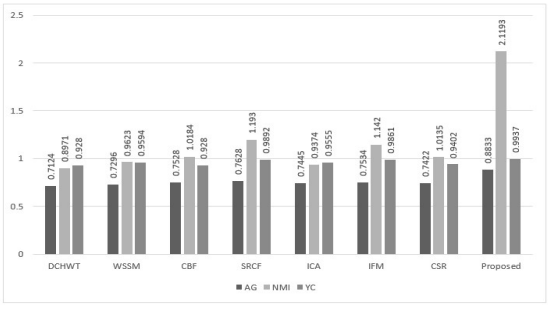

Figure 13: Comparison of Implemented Approach using Sliding Window of 3 × 3 with Weighted Approaches.

In Figure 14evaluation results of this experimentation are presented using a comparison with previous weighted fusion methodologies. This experimentation is performed using a local patch size of 5×5. There is a strong evidence that fusion methodologies such as SRCF and IFM have generated promising evaluation scores than this experimentation. However, other approaches such as DCHWT, WSSM, CBF, ICA, CSR have reported lowest evaluation results.

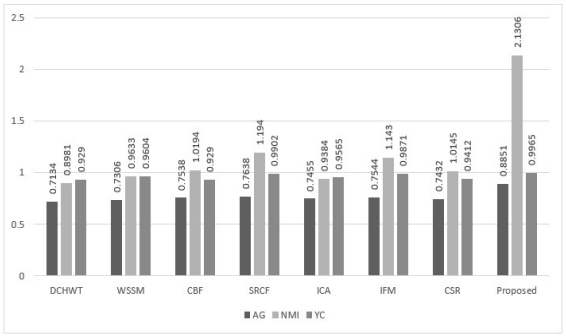

Figure 14: Comparison of Implemented Approach using Sliding Window of 5 × 5 with Weighted Approaches

The evaluation results of this experimentation over a local patch size of 7 × 7 are presented using a comparison with previous weighted fusion methodologies in figure 15. It has been observed from experimental evaluation that fusion methodologies such as DCHWT, WSSM, CBF, ICA, CSR have stated lowest evaluation results. Other weighted methodologies such as SRCF and IFM have generated promising evaluation scores than this experimentation.

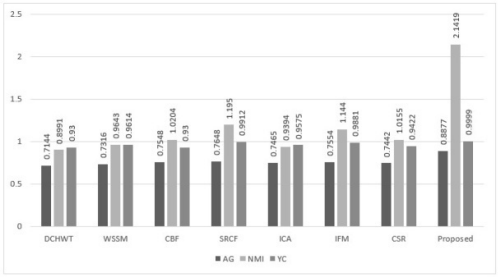

Figure 15: Comparison of Implemented Approach using Sliding Window of 7 × 7 with Weighted Approaches

2) *Comparison using non-weighted approaches.*

Figure 16, presents a comparison of proposed fusion methodology with previous non- weighted fusion methodologies. The fusion approaches such as DCTLP, MSTSR, PCA have produced lowest evaluation results. Other fusion methodologies such as PCNN, GIF, DCTV, IFGD, MWGF and CAB have produced closer but lower evaluation scores than proposed methodology. In this phase, a fused image is obtained from local features using a patch size of 3 × 3. Similarly, the evaluation is also performed over fused images which are created using patches of5 × 5 and 7 × 7 separately.

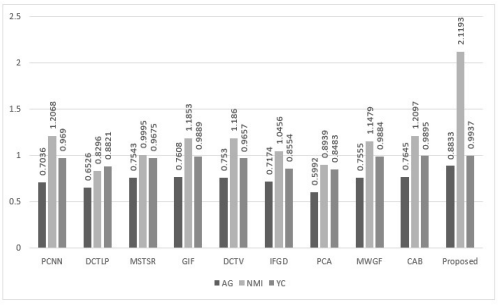

Figure 16: Comparison of Implemented Approach using Sliding Window of 3 × 3 with Non-Weighted Approaches.

In Figure 17evaluation results of this experimentation are illustrated using a comparison with previous non-weighted fusion methodologies. In this experimentation, a local patch size of 5×5 is used. It has been observed that fusion methodologies such as PCNN, GIF, DCTV, IFGD, MWGF and CAB have generated proximate but lower evaluation scores than this experimentation. However, other approaches such as DCTLP, MSTSR, PCA have formed lowest evaluation results.

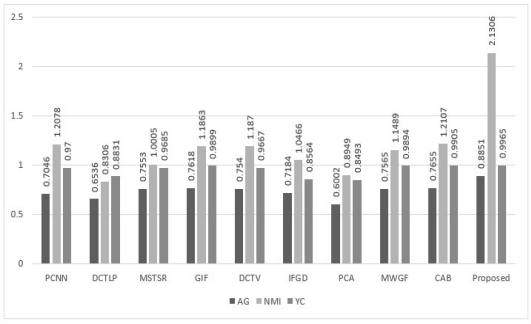

Figure 17: Comparison of Implemented Approach using Sliding Window of 5 × 5 with Non-Weighted Approaches.

Figure 18draws a chart of evaluation results of this experimentation over a local patch size of 7 × 7. In this chart, a comparison of previous non-weighted fusion methodologies with this experimentation is demonstrated. It is evident from this chart that methodologies of DCTLP, MSTSR, PCA have fashioned lowest evaluation results. Fusion methodologies such as PCNN, GIF, DCTV, IFGD, MWGF and CAB have provided proximate but lower evaluation scores than this experimentation.

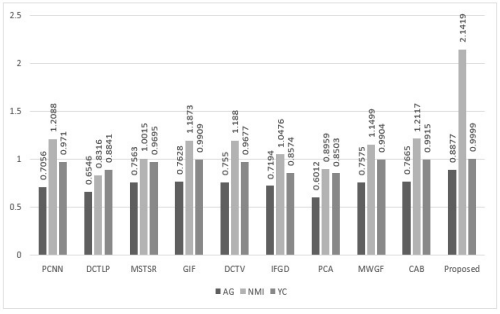

Figure 18: Comparison of Implemented Approach using Sliding Window of 7 × 7 with Non-Weighted Approaches.

V. CONCLUSION AND FUTURE WORK

The domain of image fusion is growing expeditiously and getting more and more researchers' attention day by day. Image fusion is a process that combines two or more input images to create an all in the focused fused image. Earlier studies have addressed this issue with promising results in creating an all in the focused complimentary fused image.

Furthermore, they have employed different features to characterize focused and non-focused regions in input images. For a reconstruction of all in focused fused image, an ensemble of shape and texture-based features such as ωSO, ωLO, and ωLBP is employed. To decide the contribution of each feature, the optimal weights are obtained using a Genetic Algorithm. The experimental results have indicated an improvement over previously proposed fusion method.Work In this dissertation, we have used nine datasets of multi-focus images, where each dataset is a pair of two images. These are currently the standard datasets available from the previous state of the art fusion methods. These datasets consist of eighteen multi-focus images. However, it is a small amount of data, which can only generate a generic conclusion as this domain lacks a large dataset of the multi-focus image dataset. One direction from this research is the creation of a novel dataset of multi-focus images. To perform this research, an ensemble of three features, such as two shape-based and one texture-based feature, is employed. For a better characterization of focused and non-focused regions, this ensemble can be increased to n features. Moreover, the optimal weights are obtained from a limited population of five chromosomes because of a higher computational cost. However, this population can also be increased up to n number of chromosomes.

REFERENCES

[1] C. Pohl and J. L. Van Genderen, "Review article multisensor image fusion in remote sensing: concepts, methods and applications," *International journal of remote sensing*, vol. 19, no. 5, pp. 823–854, 2001.

[2] K. Sims, "Artificial evolution for computer graphics," in Proceedings of the 18th annual conference on Computer graphics and interactive techniques, 2003, pp. 319–328.

[3] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[4] B. Goyal, A. Dogra, S. Agrawal, and B. Sohi, "A three stage integrated denoising approach for grey scale images," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–16, 2018.

[5] A. M. Chaudhry, M. M. Riaz, and A. Ghafoor, "A framework for outdoor rgb image enhancement and dehazing," *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 6, pp. 932–936, 2018.

[6] S. Li, J.-Y. Kwok, I.-H. Tsang, and Y. Wang, "Fusing images with different focuses using support vector machines," *IEEE Transactions on neural networks*, vol. 15, no. 6, pp. 1555–1561, 2004.

[7] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2818–2826.

[8] S. Li, J.-Y. Kwok, I.-H. Tsang, and Y. Wang, "Fusing images with different focuses using support vector machines," IEEE Transactions on neural networks, vol. 15, no. 6, pp. 1555–1561, 2004.

[9] C. Pohl and J. L. Van Genderen, "Review article multisensor image fusion in remote sensing: concepts, methods and applications," International journal of remote sensing, vol. 19, no. 5, pp. 823–854, 2008.

[10] P. Balasubramaniam and V. Ananthi, "Image fusion using intuitionistic fuzzy sets," Information Fusion, vol. 20, pp. 21–30, 2014

[11] C. Hsieh and W. Wu, "Pixel-based multi-focus image fusion by color appearance model," in *Software Engineering and Information Technology: Proceedings of the 2015 International Conference on Software Engineering and Information Technology (SEIT2015)*. World Scientific, 2016, pp. 255–260

[12] K. B. K. Shreyamsha, "Multifocus and multispectral image fusion based on pixel significance using discrete cosine harmonic wavelet transform," *Signal, Image and Video Processing*, vol. 7, no. 6, pp. 1125–1143, 2012

[13] N. VPS and E. Bindu, "A novel image fusion technique using dct based laplacian pyramid," *International Journal of Inventive Engineering and Sciences (IJIES)*, vol. 1, no. 2, pp. 1–9, 2013.

[14] M. Nikolaos and N. Tania, "Pixel-based and region-based image fusion schemes using ica bases," *Information Fusion*, vol. 8, no. 2, pp. 131–142, 2007.

[15] Y. Liu, X. Chen, R. K. Ward, and Z. J. Wang, "Image fusion with convolutional sparse representation," IEEE signal processing letters, vol. 23, no. 12, pp. 1882–1886, 2016.

[16] Kumar, Prema, R. P. M, and Kumar, "Enhancing bio-medical mammography image fusion using optimized genetic algorithm," *Journal of Medical Imaging and Health Informatics*, vol. 9, no. 3, pp. 1–6, 2019.

[17] Yu, S. Liu, Z. Liu, and Wang, "A general framework for image fusion based on multi-scale transform and sparse representation," *Information Fusion*, vol. 24, no. 1, pp. 147–164, 2014.

[18] Z. Jin-Yu, C. Yan, and H. Xian-Xiang, "Edge detection of images based on improved sobel operator and genetic algorithms," in *2009 International Conference on Image Analysis and Signal Processing*. IEEE, 2009, pp. 31–35.

[19] D. Marr and E. Hildreth, "Theory of edge detection," *Proceedings of the Royal Society of London. Series B. Biological Sciences*, vol. 207, no. 1167, pp. 187–217, 2011.

[20] X. Yi and M. Eramian, "Lbp-based segmentation of defocus blur," *IEEE transactions on image processing*, vol. 25, no. 4, pp. 1626–1638, 2016.

[21] Z. Jin-Yu, C. Yan, and H. Xian-Xiang, "Edge detection of images based

[22] on improved sobel operator and genetic algorithms," in 2009 International Conference on Image Analysis and Signal Processing. IEEE, 2009, pp. 31–35

[23] O. Kramer, Genetic algorithm essentials. Springer, 2017.