# CHAPTER 5

**ONE-STAGE NONLINEAR FILTERING SCHEME TO DETECT THE EDGES FROM A NOISY IMAGE**

<table>
<thead>
<tr>
<th>S.No.</th>
<th>Name of the Subtitle</th>
<th>Page No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1</td>
<td>Edge Localization and Detection</td>
<td>90</td>
</tr>
<tr>
<td>5.1.1</td>
<td>Edge Model and Localization</td>
<td>91</td>
</tr>
<tr>
<td>5.2</td>
<td>Principle of Existing Nonlinear Filtering Scheme for Noisy Image</td>
<td>93</td>
</tr>
<tr>
<td></td>
<td>Edge Detection</td>
<td></td>
</tr>
<tr>
<td>5.3</td>
<td>One-Stage Nonlinear Filter to Detect Edges from Noisy Images</td>
<td>94</td>
</tr>
<tr>
<td>5.3.1</td>
<td>Nonlinear Noise Suppression Edge Detection Scheme for Noisy Images</td>
<td>94</td>
</tr>
<tr>
<td>5.3.2</td>
<td>Subjective Assessment on grayscale images</td>
<td>98</td>
</tr>
<tr>
<td>5.3.3</td>
<td>Objective Assessment</td>
<td>105</td>
</tr>
<tr>
<td>5.4</td>
<td>Results and Discussions</td>
<td>109</td>
</tr>
<tr>
<td>5.5</td>
<td>Summary</td>
<td>111</td>
</tr>
</tbody>
</table>
CHAPTER 5
ONE-STAGE NONLINEAR FILTERING SCHEME TO DETECT
THE EDGES FROM A NOISY IMAGE

To reduce the computation cost needed in edge detection for the noisy images plays major role in edge detection. The structural properties of the image like edges obtained by the edge detector are used for recognition of objects. This chapter discusses a method to obtain the edge image from the images corrupted with Gaussian noise and impulse noise, without doing regularization or smoothing or denoising process.

This chapter is organized as follows. Section 5.1 presents discrete edge localization and detection problem. Section 5.2 presents the principle of existing nonlinear filtering scheme for edge detection in noisy images. Section 5.3 presents proposed Nonlinear noise suppression edge detection scheme for noisy images corrupted by either Gaussian noise or impulse noise. Section 5.4 presents the Results and discussions of proposed method in comparison with existing methods listed in literature. The chapter summary is presented in section 5.5.

5.1 Edge localization and detection

An edge pixel [81]-82 in an image corresponds to significant local changes of intensity in an image. Edges typically occur on the boundary between two different regions in an image. Based on the intensity profiles the edges can be modeled in the following way.

**Step edge:** The image intensity abruptly changes from one value to one side of the discontinuity to a different value on the opposite side.

**Ramp edge:** A step edge where the intensity change is not instantaneous but occur over a finite distance.

**Ridge edge:** The image intensity abruptly changes value but then returns to the starting value within some short distance.

**Roof edge:** A ridge edge where the intensity change is not instantaneous but occur over a finite distance.
5.1.1 Edge model and localization

Two points on both sides of the transition for the model of an ideal step edge, can be considered as edge-point candidates and are shown in fig.5.1. The exact or best edge point can be considered between the two pixels based on sub-pixel detection [68].

![Fig.5.1: Model of edge profile and edge candidates for normal images.](image)

Generally, from the two points one of the maximum will be selected. This kind of condition yields the local maxima extraction and is given in equation 5.1

\[ |g_{\text{neighbor in the gradient direction}}| < |g| \leq |g_{\text{neighbor in the inverse gradient direction}}| \]  

(5.1)

The role of the asymmetry in this condition is particularly obvious in the case of synthetic images. This condition is not same for noisy images, so asymmetry induces an ambivalent local maxima representation both in synthetic images and in real still images. The process of identify actual edge candidates is shown in fig.5.2.

In fig.5.2, \( S_k \) is the original signal and \( y_k \) is the signal shown along with edge candidature by convoluting with [1 -1] filter. Fig.5.2.a shows the ideal step edge transition in positive direction. Fig.5.2.c shows considered actual edge pixels based on local maxima among two pixels which comes under edge candidature. Fig.5.2.b shows the ideal step edge transition in negative direction. Fig.5.2.d shows considered
actual edge pixels based on local maxima among two pixels which comes under edge candidature.

![Diagram](image)

Fig.5.2: Edge localization varies $y_{k[1-1]}$ according to the sign of the transition.

Moreover, choosing one particular edge-point candidate in presence of noise implies that a change in the sign of the step edge induces a different localization result as shown in fig.5.3. In fig.5.3 $S_k$ is the original signal along with noise and $y_k$ is the signal shown along with edge candidature by convoluting with $[1\ -1]$ filter on the noisy signal. Fig.5.3.a shows the noisy ideal step edge transition in positive direction, here noise have no effect on edge candidates, so listed correctly in fig.5.3.c as a actual edge pixel. But the fig.5.3.b shows the noisy ideal step edge transition in negative direction, here noise has effect on edge candidates, so wrongly a pixel is localized as edge candidate shown in fig.5.3.d from the two edge candidatures.
The objective of any edge detector for noisy images is to overcome the above mentioned delocalization problem.

5.2 Principle of existing nonlinear filtering scheme for noisy image edge detection (NLFS)

The existing NLFS scheme [68] will localize the edge point according to the sign of the slope of the transition. If the slope of the transition is not negative, then the pixel under observation as edge pixel will be validated after the transition. Otherwise, if the slope of the transition is not positive then the pixel under observation will be validated before the transition itself. Nonlinear Filtering Scheme obtains the edge image directly from the noisy image in a single stage and already discussed in chapter 2.4.1 and again here briefly outlined in algorithm 5.1. It is a one-stage nonlinear filter for noisy images to detect the edge image without doing image denoising. This method adopts the following steps to obtain the edge image from the noisy image.

Algorithm 5.1: NLFS method to detect the edges from a noisy image without regularization
Input: Noisy image
Output: Edge image
Method:
1. Initially set zero for Threshold value.

2. By convoluting an noisy content with NLFS filter, calculate directional horizontal positive gradient \( \text{HOpg} \) and horizontal negative gradient \( \text{HOng} \) at each pixel of the image along the X-axis. If \( \text{HOpg} \leq \text{Threshold} \) then assign \( \text{dph} \) is with zero else \( \text{dph} \) is taken as \( \text{HOpg} \) and If \( \text{HOng} \leq \text{Threshold} \) then \( \text{dnh} \) is with zero else \( \text{dnh} \) is taken as \( \text{HOng} \).

The gradient along the X-axis direction is given by equation 5.2

\[
\text{Gh} = \text{dph} + \text{dnh} \tag{5.2}
\]

3. By convoluting an noisy content with NLFS filter, calculate directional vertical positive gradient \( \text{VEpg} \) and negative vertical gradient \( \text{VEng} \) at each pixel of the image along the Y-axis. If \( \text{VEpg} \leq \text{Threshold} \) then assign \( \text{dpv} \) is with zero else \( \text{dpv} \) is taken as \( \text{VEpg} \) and If \( \text{VEng} \leq \text{Threshold} \) then \( \text{dnv} \) is with zero else \( \text{dnv} \) is taken as \( \text{VEng} \).

The gradient along the Y-axis is given by equation 5.3

\[
\text{Gv} = \text{dpv} + \text{dnv} \tag{5.3}
\]

4. The final gradient magnitude \( \text{GrMag} \) is given by equation 5.4

\[
\text{GrMag} = \sqrt{ (\text{Gh})^2 + (\text{Gv})^2 } \tag{5.4}
\]

5. After processing all the pixels finally display the edge image.

5.3 One-stage nonlinear filter to detect edges from noisy images

Design of one-stage nonlinear filters for edge detection from noisy images avoids computations needed for image denoising.

5.3.1 Nonlinear noise suppression edge detection scheme for noisy images (NNSED)

Motivation

Though the existing two-stage color edge detectors for noisy images gives edge images with more accurate edge pixels, there is a scope for further improvement to design edge detectors which obtains the edge image directly without considering regularization process by eliminating time complexity for regularization. So, a new nonlinear filter for minimizing the delocalization of all true edges has been proposed in current study. Hence, in proposed work, the applicability of proposed nonlinear filters for detecting the edge image from images corrupted with Gaussian noise, impulse noise, and speckle noise are investigated. This Proposed NNSED for noisy
images is applied on a variety of standard real images corrupted with gaussian noise, impulse noise, and speckle noise. Upon obtaining the final edge image, identified that the obtained edge image is documented with thin edge pixels by covering uncovered edge pixels also, which is the drawback of existing NLFS method. The NNSED scheme has higher FOM value compared to the methods listed in the literature.

The proposed NNSED method for Noisy Images follows a principle that the gradients are computed in the directions called forward and backward for computing the slope in row-direction and also in column-direction of pixels of an image. By letting the difference between forward and backward direction gradient is considered as the actual edge candidate in the direction in which differences are identified. Block diagram of the proposed NNSED is shown in fig.5.4.
Fig. 5.4: Block diagram of proposed NNSED method for noisy images.
The process of detecting the edges from noisy image by applying NNSED algorithm is outlined in algorithm 5.2.

**Algorithm 5.2:** Nonlinear Noise Suppression Edge Detection (NNSED) scheme for Noisy Images.

**Input:** Given input noisy image, NI

**Output:** Final Edge image obtained without image denoising.

**Method:**

**STEP – I:** Read the noisy image (Gaussian, Impulse or speckle noise) \( I(x, y) \)

**STEP – II:** Compute the exact directional filters \( \text{ExactDfX} \) and \( \text{ExactDbX} \), for each pixel of a noisy image \( NI(p, q) \) in row-direction by considering the forward convolution and backward convolutions by using a nonlinear filter \[ 0; 1; -1 \].

a) First calculate \( \text{ExactDfX} \), the exact directional forward gradient in view of row-direction. This value \( \text{ExactDfX} \) is calculated by equation 5.5

\[
\text{ExactDfX} = NI(p, q) - NI(p, q+1) \tag{5.5}
\]

b) Next calculate \( \text{ExactDbX} \), the exact directional backward gradient in view of the row-direction. The \( \text{ExactDbX} \) is calculated by equation 5.6

\[
\text{ExactDbX} = NI(p, q-1) - NI(p, yq) \tag{5.6}
\]

c) If the calculated slopes \( \text{ExactDfX} \) and \( \text{ExactDbX} \) in row-direction, are the same then let the gradient along the row-direction \( \text{RowHg} \) is assigned with the differences among the \( \text{ExactDfX} \) and \( \text{ExactDbX} \).

**STEP – III:** Compute the exact directional filters \( \text{ExactDfY} \) and \( \text{ExactDbY} \), for each pixel of a noisy image \( NI(p, q) \) in column-direction by considering the forward convolution and backward convolutions by using a nonlinear filter \[ 0; 1; -1 \].

a) First calculate \( \text{ExactDfY} \), the exact directional forward gradient in view of column-direction. This value \( \text{ExactDfY} \) is calculated by equation 5.7

\[
\text{ExactDfY} = NI(p, q) - NI(p+1, q) \tag{5.7}
\]

b) Next calculate \( \text{ExactDbY} \), the exact directional backward gradient in view of the column-directions. The \( \text{ExactDbX} \) is calculated by equation 5.8

\[
\text{ExactDbY} = NI(p, q-1) - NI(p, yq) \tag{5.8}
\]

c) If the calculated slopes \( \text{ExactDfY} \) and \( \text{ExactDbY} \) in column-direction, are the same then let the gradient along the column-direction \( \text{ColumnVg} \) is assigned with the differences among the \( \text{ExactDfY} \) and \( \text{ExactDbY} \).
STEP – IV: Finally the image gradient at pixel NI(p,q) FinalGradientMagnitude as a edge pixel candidature is calculated by equation 5.9

\[
\text{FinalGradientMagnitude} = |\text{RowHg}| + |\text{ColumnVg}|
\]  \hspace{1cm} (5.9)

STEP – V: Iterate steps II to IV for each noisy pixel an finally display all the edge pixels in an output image.

5.3.2 Subjective assessment on gray scale images

To test the efficiency of the proposed method, algorithm is applied on variety of images. The total set of image database used for this method is around 600 gray scale images. To check the applicability of algorithm on variety of noise content, the test image dataset is corrupted with gaussian noise, impulse noise and speckle noise. Some of the tested results are shown from images of Lena, cameraman images corrupted with gaussian noise, impulse noise and speckle noise. The obtained edge images by applying NNSED method are shown in Fig.5.5 to 5.10 applying on gray scale images.
Fig. 5.5 shows obtained edge image by applying existing NLFS method and proposed NNSED scheme for noisy images on Lena image corrupted with gaussian noise.

(a) Original image.  
(b) Gaussian noisy image. 
(c) NLFS edge image.  
(d) NNSED edge image
Fig. 5.6 shows obtained edge image by applying NLFS method and NNSED scheme for noisy images on Cameraman image corrupted with gaussian noise.

(a) Original image.  
(b) Gaussian noisy image.  
(c) NLFS edge image.  
(d) NNSED edge image.
Fig. 5.7 shows obtained edge image by applying NLFS method and NNSED scheme for noisy images on Lena image corrupted with salt and pepper noise.

(a) Original image.          (b) Impulse noisy image.
(c) NLFS edge image.         (d) NNSED edge image

Fig. 5.7: Edge images after applying NLFS and NNSED methods on Lena image corrupted with salt and pepper noise.
Fig. 5.8 shows obtained edge image by applying NLFS method and NNSED scheme for noisy images on Cameraman image corrupted with salt and pepper noise.

Fig. 5.8: Edge images after applying NLFS and NNSED methods on Cameraman image corrupted with salt and pepper noise.

(a) Original image. (b) Impulse noisy image.
(c) NLFS edge image. (d) NNSED edge image.
Fig. 5.9 shows obtained edge image by applying NLFS method and NNSED scheme for noisy images on Lena image corrupted with speckle noise.

![Figure 5.9: Edge images after applying NLFS and NNSED methods on Lena image corrupted with speckle noise.](image)

(a) Original image.  (b) Speckle noisy image.
(c) NLFS edge image.  (d) NNSED edge image
Fig. 5.10 shows obtained edge image by applying existing NLFS method and proposed NNSED scheme for noisy images on Cameraman image corrupted with speckle noise.

Fig.5.10: Edge images after applying NLFS and NNSED methods on Cameraman image corrupted with speckle noise.

(a) Original image.  (b) Speckle noisy image.
(c) NLFS edge image.  (d) NNSED edge image
5.3.3 Objective assessment

The following performance measures are computed to ensure the, the count of all true edges called true positives (TP), the count of false edges which are labeled as edge pixels by mistake called false positives (FP) and the count of pixels which are edge pixels but the edge detector has not identified as edge one called false negatives(FN) for the NNSED method in comparison with existing NLFS method. For the NLFS method and proposed NNSED method based on above, here calculated the percentage of correctly detected edge pixels (PCD), the percentage of not detected edge pixels(PND), the percentages of erroneously detected edge pixels (PED), and based on PCD, PND,PED the figure-of-merit is also calculated and all these are computed using equations through 5.10 to 5.13.

\[ \text{PCD} = \frac{\text{TP}}{\max\{N_C,N_P\}} \]  
\[ \text{PND} = \frac{\text{FN}}{\max\{N_C,N_P\}} \]  
\[ \text{PED} = \frac{\text{FP}}{\max\{N_C,N_P\}} \]

\[ \text{FOM} = \frac{1}{\max\{N_C,N_P\}} \sum_{i=1}^{\text{NP}} \left\{ \frac{1}{1 + \alpha + \text{D}_i^2} \right\} \]

The FOM value lies between 0 to 1 and nearer to 1 is actual edge pixels are detected exactly.

Table 5.1 presents the calculated PCD, PND, PED and FOM values comparison with NLFS method and proposed NNSED method for Lena image corrupted by impulse noise(IN) with various noise densities between 1% to 50%.
Table 5.1: Computed PCD, PND, PED and FOM values for NLFS method and NNSED method for images corrupted with impulse noise.

<table>
<thead>
<tr>
<th>Impulse noise density</th>
<th>Method type</th>
<th>PCD</th>
<th>PND</th>
<th>PED</th>
<th>FOM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1%</td>
<td>NLFS Method</td>
<td>0.2516</td>
<td>0.0607</td>
<td>0.7484</td>
<td>0.3206</td>
</tr>
<tr>
<td></td>
<td>Proposed NNSED method</td>
<td>0.2589</td>
<td>0.0232</td>
<td>0.7244</td>
<td>0.3298</td>
</tr>
<tr>
<td>5%</td>
<td>NLFS Method</td>
<td>0.2429</td>
<td>0.0963</td>
<td>0.7571</td>
<td>0.4230</td>
</tr>
<tr>
<td></td>
<td>Proposed method</td>
<td>0.2459</td>
<td>0.0305</td>
<td>0.7563</td>
<td>0.5634</td>
</tr>
<tr>
<td>10%</td>
<td>NLFS Method</td>
<td>0.1918</td>
<td>0.1238</td>
<td>0.8082</td>
<td>0.6724</td>
</tr>
<tr>
<td></td>
<td>Proposed NNSED method</td>
<td>0.2232</td>
<td>0.0044</td>
<td>0.7749</td>
<td>0.7632</td>
</tr>
<tr>
<td>20%</td>
<td>NLFS Method</td>
<td>0.1322</td>
<td>0.2283</td>
<td>0.8678</td>
<td>0.4723</td>
</tr>
<tr>
<td></td>
<td>Proposed NNSED method</td>
<td>0.1429</td>
<td>0.1455</td>
<td>0.5507</td>
<td>0.6701</td>
</tr>
<tr>
<td>30%</td>
<td>NLFS Method</td>
<td>0.0756</td>
<td>0.1860</td>
<td>0.9244</td>
<td>0.5294</td>
</tr>
<tr>
<td></td>
<td>Proposed NNSED method</td>
<td>0.0871</td>
<td>0.1700</td>
<td>0.5651</td>
<td>0.6047</td>
</tr>
<tr>
<td>40%</td>
<td>NLFS Method</td>
<td>0.0645</td>
<td>0.1809</td>
<td>0.9355</td>
<td>0.6268</td>
</tr>
<tr>
<td></td>
<td>Proposed NNSED method</td>
<td>0.0808</td>
<td>0.1659</td>
<td>0.5197</td>
<td>0.7327</td>
</tr>
<tr>
<td>50%</td>
<td>NLFS Method</td>
<td>0.0932</td>
<td>0.2089</td>
<td>0.9068</td>
<td>0.6780</td>
</tr>
<tr>
<td></td>
<td>Proposed NNSED method</td>
<td>0.1002</td>
<td>0.2452</td>
<td>0.4886</td>
<td>0.6884</td>
</tr>
</tbody>
</table>
Fig. 5.11 shows the graph representation of computed FOM values by applying NLFS and NNSED method on lena image by varying impulse noise density between 1% to 50%.

Fig. 5.11: Graph representation of computed FOM values by applying existing NLFS and proposed NNSED methods on lena image by varying impulse noise.
Table 5.2 presents the calculated FOM values comparison with NLFS method and proposed NNSED method for Lena image corrupted by gaussian noise (GN) with varying variance and zero mean.

Table 5.2: Computed FOM Values for NLFS method and proposed NNSED method for images corrupted with gaussian noise.

<table>
<thead>
<tr>
<th>Gaussian noise</th>
<th>Method type</th>
<th>FOM</th>
</tr>
</thead>
<tbody>
<tr>
<td>m=0, v=0.1</td>
<td>NLFS Method</td>
<td>0.4557</td>
</tr>
<tr>
<td></td>
<td>Proposed NNSED method</td>
<td>0.7056</td>
</tr>
<tr>
<td>m=0, v=0.2</td>
<td>NLFS Method</td>
<td>0.7123</td>
</tr>
<tr>
<td></td>
<td>Proposed NNSED method</td>
<td>0.7235</td>
</tr>
<tr>
<td>m=0, v=0.3</td>
<td>NLFS Method</td>
<td>0.6281</td>
</tr>
<tr>
<td></td>
<td>Proposed NNSED method</td>
<td>0.7638</td>
</tr>
<tr>
<td>m=0, v=0.3</td>
<td>NLFS Method</td>
<td>0.3028</td>
</tr>
<tr>
<td></td>
<td>Proposed NNSED method</td>
<td>0.6539</td>
</tr>
<tr>
<td>m=0, v=0.5</td>
<td>NLFS Method</td>
<td>0.3028</td>
</tr>
<tr>
<td></td>
<td>Proposed NNSED method</td>
<td>0.6995</td>
</tr>
</tbody>
</table>

Fig.5.12 shows the graph representation of computed FOM values by applying NLFS and proposed NNSED method on lena image by varying gaussian noisy with zero mean and by varying variance.

Fig.5.12: Graph representation of computed FOM values by applying existing NLFS and proposed NNSED methods on lena image by varying variance with gaussian noise.
Table 5.3 presents the calculated FOM values comparison with NLFS method and proposed NNSED method for Lena image corrupted by speckle noise (SN) with various noise densities.

Table 5.3: Computed FOM Values for NLFS method and proposed NNSED method for images corrupted with speckle noise by varying the variance.

<table>
<thead>
<tr>
<th>Speckle noise</th>
<th>Method type</th>
<th>FOM</th>
</tr>
</thead>
<tbody>
<tr>
<td>v=0.15</td>
<td>NLFS Method</td>
<td>0.4754</td>
</tr>
<tr>
<td></td>
<td>Proposed NNSED method</td>
<td>0.5574</td>
</tr>
<tr>
<td>v=0.25</td>
<td>NLFS Method</td>
<td>0.3997</td>
</tr>
<tr>
<td></td>
<td>Proposed NNSED method</td>
<td>0.7446</td>
</tr>
<tr>
<td>v=0.5</td>
<td>NLFS Method</td>
<td>0.5840</td>
</tr>
<tr>
<td></td>
<td>Proposed NNSED method</td>
<td>0.7996</td>
</tr>
</tbody>
</table>

5.4 Results and discussions

The proposed NNSED scheme for Noisy Images has the advantage that it always produces thin edges by suppressing the noise without doing regularization. The NNSED method obtains edge image efficiently from the images corrupted by either a speckle noise, impulse noise, or gaussian noise. The amount of noise content forwarded along with the output edge image is minimized in comparison with the existing NLFS scheme.

To estimate the effect of producing edge image from noisy images by NNSED method experiments are done over 500 images corrupted by Gaussian noise, impulse noise, and speckle noise. Few sample output images obtained are depicted in fig.5.5 to 5.10. The output edge images obtained by applying NNSED and NLFS schemes on lena image and cameraman image corrupted by gaussian noise are compared and shown in fig.5.5 and 5.6. The output edge images obtained by applying NNSED and NLFS schemes on lena image and cameraman image corrupted by salt-and-pepper noise are compared and shown in fig.5.7 and 5.8. The output edge images obtained by applying NNSED and NLFS schemes on lena image and cameraman image corrupted by speckle noise are compared and shown in fig.5.9 and 5.10. From experimental results it has been identified that the amount of noise content carried along with the edge image is more in case of existing NLFS scheme and very less in
case of NNSED method. The NNSED method identifies the edge pixels efficiently even though noise content influences delocalizing the edge pixels in the image while detecting the edges. The main drawback of NLFS scheme is, it accounts noisy pixels also as an edge pixels in the resultant edge image and the ignores few edge pixels as a noisy pixels. All these are well documented with the NNSED method and the percentage of false edge pixels is reduced and the percentage of true edge pixels is improved even with variety of noises. From these it is clear that the proposed method exhibits improved performance by considering the possibility of extracting the all true edge pixels in the resultant edge image in comparison with NLFS scheme.

Table 5.1 indicated the calculated PCD, PND, PED and FOM values of the NNSED method comparision with NLFS method for Lena image corrupted by salt and pepper noise or impulse noise (IN) by varying noise densities. It is clear that the percentage of correctly detected edge pixels (PCD) is improved by identifying all true positive edge pixels, the percentage of not detected edge pixels (PND), and percentage of erroneously detected edge pixels (PED) are reduced by ignoring false positive and false negative edge pixels. It also shows that FOM value is more while compared to NLFS method even though noise with higher density. The experimental results documented in Table 5.2 shows the FOM values obtained by applying the NNSED method and NLFS scheme on Lena image corrupted with gaussian noise with zero mean and varying variance, it is clear that the FOM value is high means all exact edge pixels are extracted by NNSED method. The experimental results documented in Table 5.3 shows the FOM values obtained by applying the NNSED method and NLFS scheme on Lena image corrupted with speckle noise by varying variance , it is clear that the FOM value in this case also is high means all exact edge pixels are extracted by NNSED method.
5.5 Summary

The proposed NNSED method extracts all the true edge pixels efficiently by suppressing the percentage of noise content forwarded with the final edge image. It is observed that the proposed NNSED is an efficient approach for edge detection in noisy images without regularization. The edges obtained are compared with other nonlinear approaches and concluded that the edges obtained using NNSED scheme are more efficient.

The subjective analysis shows that the quality of edge image obtained by NNSED is more clear compared to NLFS. The carried noise content along with edge image may confuse the actual edge pixels in NLFS that percentage is minimized by proposed NNSED. The experimental results using FOM on few images shows that the proposed method produces a more clarity edge image in case of images corrupted with either gaussian noise, impulse noise or speckle noise than the existing NLFS method. Present study concludes that existing NLFS method fails in obtaining edge image if images are corrupted with high noise density. But, the proposed method may overcome this difficulty and produces the edge image without taking time complexity for regularization.