Digital Images Processing to reduce noise

Self-defined filters to process digital images with noise based on convolution and pooling, and get gradients of images

Yunzhe Li
College of Computer Engineering and Science
Shanghai University
Shanghai, China
1669077595@qq.com

ABSTRACT

In this document, we will investigate the best way to reduce different noises on images.

According to the mathematics principles, convolution and pooling methods are employed to define functions of median and mean filters to process images with varies kind of noise(i.e. **Gaussian noise, Salt&Pepper noise, Speckle noise, etc.**). Also, Fast Fourier Transform(which will be noted as ***FFT*** in the following) is used for getting image results on frequency domain.

On images with Salt&Pepper noise, median filter did well than mean filter when the density is lower than 30%. When the density is above 60%, both filters have poor performance for Salt&Pepper noise. Median filter is better than mean filter for Gaussian noise, while mean filter is better than median filter for Salt&Pepper noise. Both filters did not bad for Speckle noise.

On the reason of linearity, mean filter can't protect details on the images when reducing noise. While median filter has the nonlinear property, it's better to process impulse but there's trouble dealing with continuous polluted points, which accounts for the poor performance on reducing Gaussian noise.

To deal with Gaussian noise, mean filter is the best way. To deal with Salt&Pepper noise, median filter is the best way. Different filters should be applied according to different situations, for each filters has merits and demerits.

KEYWORDS

Median filter, Mean filter, **Sobel** operator

1 The practical significance of image filtering

Image, for computer vision is an important research object, through the precise processing of the image, can be applied to, such as: face recognition, automatic driving and other scenes. However, in the process of image acquisition, there may be interference due to environmental factors, such as light and rain, so that there is a deviation between the actual image and the image we want to get. This deviation is called "**noise**", which makes the subsequent processing of the image difficult.

Noise is a large or tiny extreme point that acts on the real gray value of image pixels, causing interference of bright and dark points and affecting the quality of the image.

The role of filtering is to remove the noise generated by external factors as much as possible and restore the original image for subsequent processing.

2 Mathematical principle of filtering

2.1 Mean pooling realized by convolution

Before processing, the original image needs to be transformed into a grayscale image, which is a two-dimensional matrix. The modification of the original data needs to be analyzed with the help of the surrounding data, and the convolution operation is carried out by analyzing the pixel values of the eight points to be processed, namely, the upper, lower, left and upper, right and lower, so the **weighting coefficient matrix** is used, as shown in **Figure 1**

1	1	1
1	1	1
1	1	1

Figure 1: Weighted coefficient matrix

For a pixel point in an 8x8 matrix, assuming that the mean filtering is used to operate on the pixel point (2,2), then the sum of the 9 surrounding points including itself is averaged and the mean obtained is assigned to the pixel point (2,2), as shown in **Figure 2**:

0	32	24					
23	1	90					
45	25	78					

Figure 2: Mean pooling is done for (2,2) pixels

By calculating

$$\frac{0+32+24+23+1+90+45+25+78}{9}$$

The value of (2,2) pixels is 35.3.

In the previous experiments, when the boundary pixels are convolved with the weighted coefficient matrix of 3x3, the problem of boundary-crossing will occur. In the case of boundary-crossing, the convolution is usually completed by filling "0" outside the edge to reach the size of the weighted coefficient matrix.

2.2 Filter template

Filtering template is also the **weighting coefficient matrix** mentioned above, which can be defined by itself according to requirements. For edge detection, **Sobel** operator is a well-known filtering template, as shown in **Figure 3**. For edges, there are obvious differences between surrounding pixels and themselves, which makes the setting of weighting coefficient different.

1	0	-1
2	0	-2
1	0	-1

Figure 3: Sobel operator

2.3 Median filter

Median filtering was proposed by Turky in 1971. It was initially used for time series analysis and later used for image processing, and achieved good results in reducing noise and restoration. Median filter is a typical nonlinear filter that completes signal recovery based on sequence statistics. Its basic principle is to replace the value of the center point of the image or sequence with the median value of the domain, which has the advantages of simple operation, fast speed and good denoising effect.

In other words, for a pixel, sort the gray values of the nine surrounding points from smallest to largest, and fill the pixel with the median as the result, as shown in **Figure 4**:

			0	76	12		
			87	56	34		
			21	50	99		

Figure 4: To median-filter one point

Sorting

$$\begin{bmatrix} 0 & 12 & 21 & 34 & 50 & 56 & 76 & 87 & 99 \end{bmatrix}$$

To get the median is 50, which is the filtered result of the median of this pixel.

3 Implementation by Matlab

According to the above mathematical principles, it can be understood as a gray graph matrix $\boldsymbol{A}$, customizing the size of the window, use median and mean filtering methods for processing, and slide on the matrix $\boldsymbol{A}$ from left to right and from top to bottom.

3.1 Image preprocess

aBefore filtering operation, it is necessary to preprocess the read image, that is, first convert RGB image to grayscale image, and apply the function:

$$rgb2gray(img)$$

Then add different noises to the image to observe the subsequent filtering effect.

3.2 Image resize

Since the convolution operation of filtering involves each pixel and its surrounding pixels, "0" should be added to the boundary points so that convolution can be operated correctly.

The size of the original image has been changed and elements have been changed as well.

3.3 Self-defined filter functions

3.3.1 Mean Filter The essence of mean filtering is to define a **weighted coefficient matrix** $\boldsymbol{M}$ with all elements is 1, convolved with each elements of $\boldsymbol{M}$ and $\boldsymbol{A}$. Sum the convolution values, take the average, assign the average to the center point, and proceed to the next point.

3.3.3 Median Filter Median filtering is to segment the processed matrix $\boldsymbol{M}$ according to the pre-set window size. For each of these partitioned matrices, apply the function

$$reshape(img, row, column)$$

to convert it to a 1 row, n column matrix. And apply the function

$$median(matrix)$$

to find the median of these numbers, assign the median to the middle, and proceed to the next point.

3.4 Unified processing after filtering

According to the filtered matrix and window size, the numerical range of the final result is determined, and the matrix is transformed into uint8 type.

3.5 Output images

Convert uint8 type to double type first, and apply the function

$$fft2()$$

to FFT and get images on frequency domain.Then apply

$$fftshift()$$

to shift the spectrum. Since we have a complex number, we can do the modulo operation. Finally, use the spectral logarithmic transformation

$$log(F+1)$$

to get the final images on frequency domain.

3.6 Result analysis

Analysis method: The attempts of "matrix subtraction" and "matrix division" are not suitable for this scene, and the error is large. Combined with the method of comparing matrix correlation in ***Lab 7***, a more accurate correlation comparison is obtained. So use the function

$$corrcoef(img1, img2)$$

to calculate the correlation.

3.6.1 Comparison of images with Salt&Pepper noise of different densities The origin image is shown in **Figure 5**

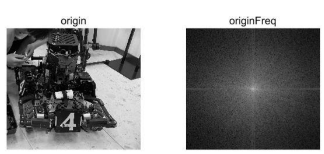

Figure 5: Origin image of gray & image on frequency domain

The filtering effect of images with Salt&Pepper noise of 5% is shown in **Figure 6:**

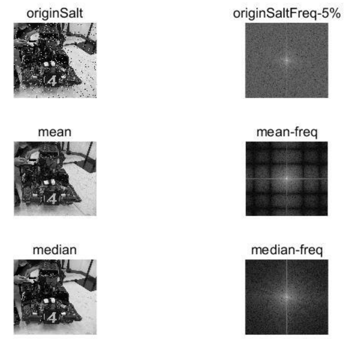

Figure 6: Filter effects of images with 5% salt&pepper noise

The filtering effect of images with Salt&Pepper noise of 30% is shown in **Figure 7:**

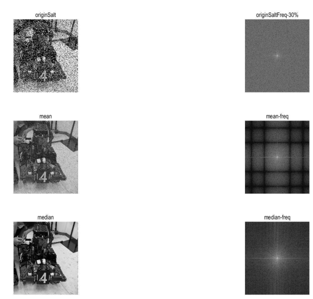

Figure 7: Filter effects of images with 30% salt&pepper noise

The filtering effect of images with Salt&Pepper noise of 55% is shown in **Figure 8:**

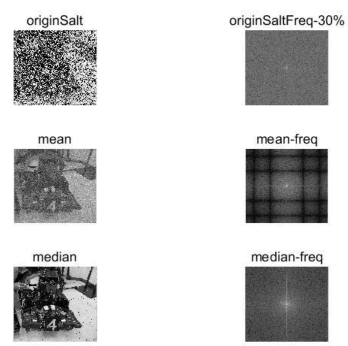

Figure 8: Filter effects of images with 55% salt&pepper noise

The filtering effect of images with Salt&Pepper noise of 75% is shown in **Figure 9:**

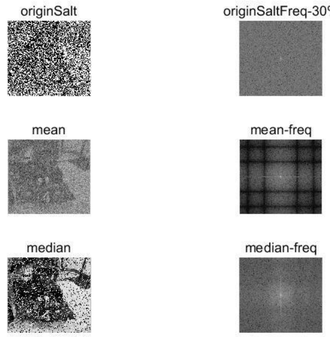

Figure 9: Filter effects of images with 75% salt&pepper noise

The test results of filtering effect correlation under different densities are shown in **Table 1**

Table1 Related coefficients of images with different densities of salt&peeper noise

Method/Density	0.05	0.3	0.55	0.75
Mean filter	98.8%	95.2%	85.2%	63.0%
Median filter	99.6%	99.2%	95.3%	67.4%
Original noise	91.1%	57.2%	32.5%	16.7%

As can be seen from Table 1, the median filter performs better than the mean filter for the calibration noise at full density. Below 30% density, the median filter can almost restore the original image, but above 60% density, the effect is not ideal, we need to find other more suitable filter.

From the analysis of principle, it can be seen that the mean filtering has some limitations: it can not protect the details of the image well, and it also destroys the details of the image in the process of image denoising, so that the image becomes blurred and the noise points, especially the salt and pepper noise, cannot be removed well.

3.6.1 Comparison of images with different noise Gaussian noise, Salt&Pepper noise and Speckle noise are used in this experiment.The gray scale and spectrum of the original filtered image are shown in **Figure 10:**

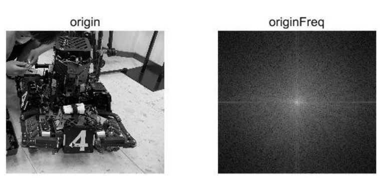

Figure 10 Origin image of gray & image on frequency domain

The filtering results of Gaussian noise are shown in **Figure 11**:

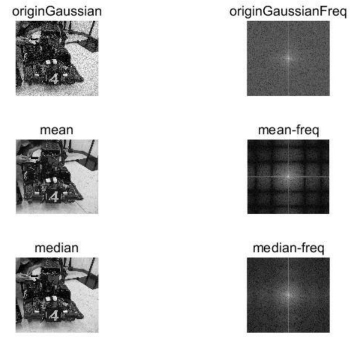

Figure 11 Process Gaussian noise & image on frequency domain

The filtering results of Salt&Pepper noise are shown in **Figure 12**:

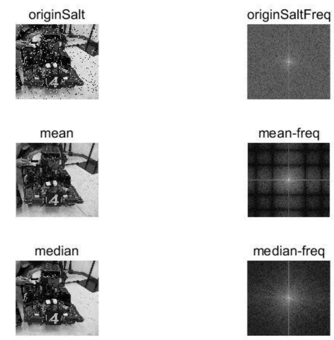

Figure 12 Process Salt&Pepper noise & image on frequency domain

The filtering results of Speckle noise are shown in **Figure 13**:

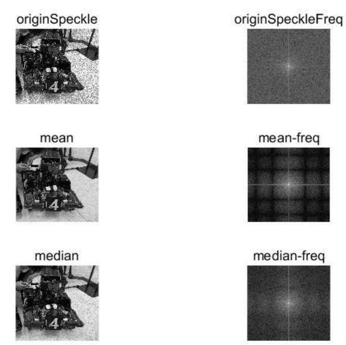

Figure 13 Process Speckle noise & image on frequency domain

The test results of filtering effect correlation under different densities are shown in **Table 2**

Table2 Results of images with different noise

Method/Noise	Gaussian	Salt&Pepper	Speckle
Mean filter	99.0%	98.8%	98.9%
Median filter	98.2%	99.6%	98.6%
Original noise	95.5%	91.2%	94.1%

It can be seen from **Table 2** that under Gaussian noise, the mean filtering is better. Under Salt&Pepper noise, the median filter is better. Under Speckle noise, the filtering effect of mean and median is ideal.

Mean filtering adopts linear method to average pixel values in the whole window range. Median filtering adopts nonlinear method, which is very effective in smoothing impulse noise. Meanwhile, it can protect sharp edges of images and select appropriate points to replace the values of polluted points. Therefore, median filter performs well to salt and pepper noise, but poorly to Gaussian noise.

3.7 Edge extraction preliminary: Sobel operator extraction gradient value

For the median filtered image, **Sobel** operator is used to extract the gradient of the image and check the filtering effect. In the future, relevant edge data will be consulted to determine the threshold value of the edge and then extract the edge. However, due to the current time constraints, no further development. Principle: Use the **Sobel** operator mentioned in 2.2, as shown in **Figure 13**:

-1	0	1
-2	0	2
-1	0	1

1	2	1
0	0	0
-1	-2	-1

Figure 13 Sobel operator on x and y

Convolved with the original image matrix, using the two dimensional convolution function

$$conv2(img1, Sobel)$$

Get the horizontal and vertical gradients and take the absolute values, and then follow the formula

$$grad = \sqrt{Gx^2 + Gy^2}$$

to obtain the overall gradient of the image.The result is shown in **Figure 14**:

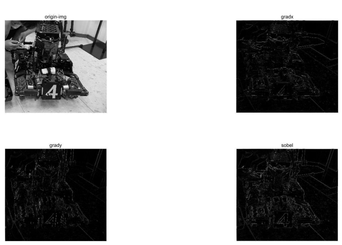

Figure 14 Gradients on x, y and whole

If the edges need to be extracted, the edges should be further processed, and the threshold value should be set according to the obtained grad, so as to distinguish what is the real boundary.

4 Conclusion

For Salt&Pepper noise of different densities, due to the linear nature of mean filtering, details will be lost and blurred during filtering, so the filtering effect for Salt&Pepper noise is poor.

For different noises of the same density, due to the nonlinearity of median filter, it is very effective in smoothing impulse noise, so it is very effective in dealing with salt and pepper noise, but for more cohesive Gaussian noise, the effect is poor. For Speckle noise, the two filtering effects are equivalent.

Each filtering method has its own advantages and disadvantages. For different situations, the most appropriate filter should be selected, and the filtering template and the **weighting coefficient matrix** should be adjusted to achieve the most appropriate filtering image.

ACKNOWLEDGMENTS

Thanks for Professor Duan's guidance in the past ten weeks. I benefited a lot from the teaching method not limited to books and exams, but also guided us to pay attention to practical problems and truly realize the combination of theory and practice.

The final data analysis also made me think more deeply, quantified the differences between different methods, and laid a foundation for future research.

The significance of signal processing is self-evident. I hope what I have learned in the past ten weeks can be applied in the future.

REFERENCES

[1] Xingyun Mao, Programming for OpenCV3, Publishing House of Electronics Industry.
[2] https://baike.baidu.com/item/图像滤波/1104221