University of Science and Technology of Hanoi

Information and Communication Technology Department

Master Thesis

Academic year 2016 - 2017

Noise Evaluation and Reduction

---

presented by

DUC VIET HOANG

registered at University of Rennes 1

supervised by Dr. Tran Giang Son

Host organization : ICTLab, USTH

Hanoi - Vietnam

Attestation

I hereby, DUC VIET HOANG, certify that my report doesn't contain plagiarism (copy/paste) from other sources.

In case of plagiarism in my report, I know the consequences and I understand that my report won't be evaluated. In this case, my M2 internship will be noted as "fail".

Date 15/09/2017

Signature DUC VIET HOANG

Acknowledgements

Firstly, I am grateful to my supervisor Dr. Tran Giang Son of the ICTLab at University of Science and Technology of Hanoi (USTH). His regular supports, advices and encouragements have been a key driver for my work.

Next, I would like to thank Prof. Daniel Chillet who is my Tutor at Université de Rennes 1. I also want to thank all members of the ICTLab who were always open to welcome me. This was very helpful in creating good conditions for me to complete this work.

Finally, I would like to thank my family for their loves and supports to me. My parents and younger sister help me very much. From this, I can devote all of my time for the internship.

Abstract

Image processing applications are very popular nowadays. It can be seen in daily life, such as post-processing to improve quality of captured photos from digital cameras. Noise is an important aspect leading to degradation of image quality. This internship focuses on studying different methods to reduce noise in digital photos. In the context of SWARMS project at ICTLab - USTH, noise in captured photos potentially reduces computer vision tasks, such as object or scene recognition. The studied methods are evaluated using SWARMS image dataset to find out the best method to denoise images for this specific dataset.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This chapter presents introduction about the host institution (namely ICTLab), the internship context, the internship objective and the report organization.

1.1 Context

1.1.1 Company Introduction

ICTLab, a joint international research laboratory about Information and Communication Technology (ICT), was created on December 2014 at Hanoi, Vietnam. It is an international collaboration between University of Science and Technology of Hanoi (USTH), Institute of Information Technology (IOIT) of Vietnamese Academy of Science and Technology (VAST), IRD (Institut de Recherche pour le Développement) and the University of La Rochelle, France [1].

The research program at the ICTLab of USTH targets various scientific axes, including Modeling and Simulation, Health Bioinformatics, High Performance Computing, Machine Learning, Data Mining, Image Processing, Computer Vision and Document Analysis. Beside doing research, ICTLab also hosts internships for bachelor, master and doctoral programs.

Corresponding to these research axes, ICTLab researchers are working on different research projects, namely SWARMS, ARCHIVES, GPU4SPACE and HealthOmics.

1.1.2 Internship context

SWARMS (Say and Watch: Automated image/sound Recognition for Mobile monitoring Systems) is one of the 4 main projects at ICTLab. Its goal is to obtain a flexible and real-time monitoring network in order to feed decision support systems or support advanced visualization of the phenomenon to monitor [1]. During the first year of its implementation, ICTLab researchers are in progress of building a dataset for SWARMS project. This dataset contains a set of digital images which were collected by sending USTH bachelor students to the fields and taking photos.

This internship was happened in the context of SWARMS project where the goal is to improve the quality of image dataset collected by USTH bachelor students during their field trips. Since USTH bachelor students are not experts in photograpy, their taken photos may be added noise which, in turns, degrade the image by changing its brightness or its color information. Noise appearing in images represents unwanted components which degrades the image. It is the cause of errors from image when the pixel values do not reflect the true intensities of the real image scene. The noisy image may contain the information which is very different from the original normal image.

One common type of noise, called random noise, may easily exist in SWARMS images. This random noise is created by camera sensors with high ISO values in short exposure. ISO is a popular photography metric which describes the sensor's absolute sensitivity to light. Higher ISO value increases sensor sensitivity (i.e. how quickly the sensor absorbs light), but also adds noise to the captured photo. In other words, students tried to give more light to the camera to create lighter image, but also makes more noise appear to the image, especially in low light conditions, such as dark places or in the evening.

Since the appearance of noise degrades quality of images, it reduces accurary of SWARMS systems and consequently reduces outcomes of the project. Therefore, it is necessary to remove noise from taken images in order to enhance performance of SWARMS systems.

In this internship, we are going to study different image denoising techniques. The studied methods include Median filter, Average filter, Gaussian filter and Wiener filter. From these methods, we will compare their results by using two metrics: Mean Square Error (MSE) and Peak Signal to Noise Ratio (PSNR). Based on the quality characteristics of these two metrics, one efficient method is chosen to denoise the captured images for SWARMS project.

1.2 Problematic

1.2.1 Noise and Types of Noise

Noise definition Noise is an unwanted component of the image which can be additive or multiplicative. A captured image $i(.)$ can be decomposed into two components: a desired component $d(.)$, and a noise component $n(.)$. In the literature there exists two popular noise models: additive model and mulitplicative model.

The formula of the former is given by:

$$i(.) = d(.) + n(.) \tag{1.1}$$

Gaussian noise, quantization noise, poisson noise and uniform noise are examples of additive noise.

The formula of the latter is as follows:

Figure 1.1: Original image (left column) and salt and pepper noise (right column).

$$i(.) = d(.)n(.) \tag{1.2}$$

Speckle noise is an example of multiplicative noise.

Types of noise

There exists many types of noise in images. In this internship, only common and basic types of noise are studied, including Salt and Pepper Noise, Gaussian Noise, Quantization Noise and Speckle Noise.

Salt and Pepper Noise, also called Impulse Noise, is a type of noise having dark pixels in bright regions and bright pixels in dark regions. Salt and Pepper Noise can be caused by sharp and sudden disturbances in the image signal, bit errors in transmission or errors in analog-to-digital conversion.

The probability density function of salt and pepper noise is given as follows:

$$f_{ps}(x) = \begin{cases} f_p, & \text{for } x = p \\ f_s, & \text{for } x = s \\ 0, & \text{otherwise} \end{cases} \tag{1.3}$$

where f_p and f_s are occurrence probabilities of pixels having values equal to p (pepper) or s (salt). These salt and pepper pixels are also called dead pixels of the image. Figure 1.1 presents an example of image with salt and pepper noise compared to the original image.

Gaussian Noise, also called amplifier noise, is statistical noise where noise values obeys a Gaussian distribution. The probability density function p of a Gaussian random variable t is given by:

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{1.4}$$

where x represents grey level of the image, μ represents the mean value of density function and σ represents the standard deviation of the density function. Gaussian

Figure 1.2: Original image (left column) and gaussian noise (right column).

noise usually happens by poor illumination, high temperature or error transmission during acquisition process. Figure 1.2 presents an example of image with gaussian noise compared to the original image.

Quantization Noise, also called uniform noise, during acquisition process, when pixels of a sensed image are quantized into a number of discrete levels to perform image digitalization, quantization noise can appear. The values of quantization noise often follows approximately a uniform distribution. The probability density function p of a continuous uniform random variable t is given by:

$$p(t) = \begin{cases} \frac{1}{b-a}, & \text{for } a \leq t \leq b \\ 0, & \text{otherwise} \end{cases} \tag{1.5}$$

where a and b are the minimum and maximum values of uniform random variable t respectively, the mean μ and the variance σ of the probability density function is calculated as follows:

$$\mu = \frac{a+b}{2} \tag{1.6}$$

$$\sigma^2 = \frac{(b-a)^2}{12} \tag{1.7}$$

Figure 1.3 presents an example of image with quantization noise compared to the original image.

Speckle Noise, also called multiplicative noise: is a type of noise in which intensity values of image pixels are multiplied by undesired random values during the acquisition or transmission process. Speckle noise often happens in the cases of satellite images, medical ultrasound images or radar images, etc.

Figure 1.4 presents an example of image with speckle noise compared to the original image.

Figure 1.3: Original image (left column) and quantized image to 6 bits (right column).

Figure 1.4: Original image (left column) and image corrupted with speckle noise (right column).

1.2.2 Noise Removal

Noise removal (or denoising) is a very challenging problem in image processing. It is a process to remove noise from the image and rebuild the original image in the best quality. In modern digital image processing, image denoising is a hard problem which is used in many application areas such as Optical Character Recognition (OCR), image compression, image enhancement or image segmentation, etc.

Although image denoising techniques aim at removing noise from images, they should satisfy the following objectives [2]:

- Image edges should be well preserved
- Image details should not be lost
- Noise from flat regions should be removed totally
- Image global contrast should be well kept
- No artifacts should be added in the denoised image

With these objectives, image denoising algorithms usually follows two main steps, namely types of noise recognition and noise parameterization based on recognized noise. In the internship, we are going to apply fundamental image denoising methods to improve noisy images of SWARMS project.

1.3 Report organization

This report contains 5 chapters as follows:

- Chapter 1: Introduction
  This chapter presents context and problematic of the internship topic.
- Chapter 2: State of the art
  This chapter presents relevant research works about noise removal in digital images.
- Chapter 3: Contribution
  This chapter presents studied methods in this internship to solve noise removal problem, including Median filter, Average filter, Gaussian filter and Wiener filter.
- Chapter 4: Results
  This chapter presents the results of studied methods and an analysis of these results to choose an efficient method for SWARMS project.
- Chapter 5: Conclusion
  This chapter gives a conclusion about the methods and results obtained in this internship to solve the problematic. Possible future directions are also discussed.

Chapter 2

State of the art

This chapter presents a brief review of the state of the art works about image denoising. The studied methods belong to two categories: Spatial Filtering and Frequency Domain Filtering. The following sections detail various methods in each category.

2.1 Spatial filtering

Spatial filtering is a filtering operation where each pixel value of the output $o(u, v)$ is calculated from the input pixel at the same position $i(u, v)$ and its neighborhoods. There exists many methods to perform image denoising using spatial filtering. Two of the most popular spatial filters for denoising tasks are median filter and mean filter.

Chen et al., 1999 [3] proposed a tri-state median filter for noise reduction. The authors performed experiments on some properties of center weighted median filter. The experiments led to 3 statements: (1) if the center weight equals to one, the output of the center weighted median filter equals to that of standard median filter; (2) if the center weight equal to or is larger than the scanning window size, the output of the center weighted median filter has not effects on noise removal; (3) finally if the center weight is set to be three, the denoising performance of the center weighted median filter is maximal. From this study, the authors concluded that three important pixel values in median filter needed to be kept are (1) the last value before standard median, (2) the standard median, and (3) the first value after the standard median. From this, the authors also named their method as "Tri-state median filter".

Based on the study of Chen et al., 1999 [3], Chang et al., 2008 [4] introduced an adaptive median filter for image denoising. The main idea of this method is the manipulation of the values between the last value before the standard median and the first value after the standard media. The authors claimed that by changing these values appropriately, the output of noise reduction using median filter can be improved. The authors then used standard median as a base median and performed experiments on impulse noise with different images. The experimental results show that this adaptive median filter gives better noise removal results than standard median filter and tri-state median filter.

2.2 Frequency domain filtering

Another direction to denoise image is using frequency domain filtering. Frequency domain represents the original image in *frequency* space. In this domain, changes in image position correspond to changes in the spatial frequency. Common procedure of frequency domain filtering includes: (1) transforming the image into frequency domain; (2) performing filtering in this domain; and (3) transforming the image back to spatial domain for visualization or other post-processing steps.

One of the most popular methods is based on Wavelet transform. Wavelet transform is a type of image transformation which decomposes the image into multiple scale sub-bands with different time-frequency components. The process of using Wavelets for noise removal in image often contains three steps: (1) choose a wavelet type and a level N of decomposition. Using this wavelet, a wavelet transformation is performed on the image; (2) after having the decomposition image, the next step is to determine threshold values for each level from 1 to N. By performing these determined threshold values to the decomposed image, the frequency filtering is actually applied to the image; (3) reconstruction of the image after applying frequency filtering. This step is actually the use of inverse wavelet transform on denoised image.

Portilla et al., 2003 [5] proposed an image reduction method using Gaussian scale mixture model (GSM) in the wavelet domain. In detail, the authors modeled neighborhoods of oriented pyramid coefficients as a Gaussian vector and a hidden positive scalar multiplier. Using this model, Bayesian least square method is used to adjust the coefficients appropriately. Fan et al., 2001 [6] introduced a novel hidden Markov model (HMM) in wavelet domain in order to remove noise from images. The method exploits local statistic and intrascale dependencies of wavelet coefficients to denoise images. The experiments show that the method provides low computational complexity when denoising image noise.

Chapter 3

Contribution

In this internship, we follow the direction of using spatial filtering to remove noise from images of SWARMS project. The studied methods include Median Filter, Average Filter, Gaussian Filter and Wiener Filter. The following sections will present in detail each of these methods.

3.1 Median Filter

Median filter is a spatial filter which applies to a neighborhood (local method). It is a non-linear filter which replaces each pixel value in the image with the median of its neighbors and itself. In this method, the heavy task is the calculation of median for each neighbor area. Specifically, for calculating each median, we first order all pixel values in the neighborhood area, then replace the considered pixel with pixel value in the middle of the sorted list. If the neighborhood area contains an even number of pixels, median will be taken as the two middle pixel values. Table 3.1 shows an example to calculate median of a 5×5 image pixels having filter kernel of size 3×3. From this table, firstly the considered pixel is 156 in red color; secondly, the neighborhood pixels of size 3×3 are 123 121 223 223 156 178 125 211 178. By sorting these values, we have sorted pixel values are 121 123 125 156 178 178 211 223 223. Thirdly, the median value should be 178 and be the representative of its surrounding pixels.

Using median filter for noise removal proposes two advantages. First, median is a robust operator where single extreme values in the neighborhood do not affect median value. Secondly, median value is also one actual pixel value of the image so median filter does not create artifacts into the image. For this, median filter is useful in reducing noise of images but still keeping image details and edges. Figure 3.1 shows an example of using median filter to remove noise from a noisy cameraman image (first column). From this figure, we can see that median filter greatly reduces salt and pepper noise in the original image (second column).

Table 3.1: Example of median computation

234	147	225	214	134
231	123	121	223	189
124	223	156	178	196
219	125	211	178	124
210	185	221	189	134

		178		

Figure 3.1: Median filter on noise removal: original noisy image (left column), denoised image with median filter (right column).

3.2 Average Filter

Average filter is also a spatial filter applying to a neighborhood as median filter. However, the main goal of mean filter is to remove pixel values which are unrepresentative for their surroundings. Average filter is also a linear filter as a convolution. This means that each pixel value is replaced by the average of itself and its neighbors. As other convolution filters, average filter is also based on a kernel, which presents size and shape of the neighborhood to be sampled when computing the mean. The formula of the average filter is given by:

$$k(u,v) = \frac{1}{M} \sum_{(p,q)\in N} i(p,q) \tag{3.1}$$

Where $k(u,v)$ is new pixel value at position (u,v) of the image, N is neighborhood size, M is total number of pixels in the neighborhood N, $i(p,q)$ are original pixel

Table 3.2: Example of Average Filter

3×3 Average Filter

$$\frac{1}{9}\begin{array}{|c|c|c|}\hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline\end{array}$$

5×5 Average Filter

$$\frac{1}{25}\begin{array}{|c|c|c|c|c|}\hline 1 & 1 & 1 & 1 & 1 \\ \hline 1 & 1 & 1 & 1 & 1 \\ \hline 1 & 1 & 1 & 1 & 1 \\ \hline 1 & 1 & 1 & 1 & 1 \\ \hline 1 & 1 & 1 & 1 & 1 \\ \hline\end{array}$$

7×7 Average Filter

$$\frac{1}{49}\begin{array}{|c|c|c|c|c|c|c|}\hline 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ \hline 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ \hline 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ \hline 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ \hline 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ \hline 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ \hline 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ \hline\end{array}$$

values of the image in the neighborhood N.

The impact of average filter on noise removal depends on its size. It is often that the size of a filter is chosen to be much smaller than the image size. Table 3.2 presents three examples of average filter masks of sizes 3×3, 5×5 and 7×7 respectively. From this table, we see that the average filter has three main characteristics: (1) it is in odd ordered, (2) sum of its all elements is equal to 1, and (3) all elements are the same.

When applying average filter to denoise images, there is a trade-off between noise removal and image feature preservation. Specifically, the bigger size of the average filter results in less noise but produces more blurriness to edges and image details. Figure 3.2 shows impact of average filter size on removing salt and pepper noise of a noisy cameraman image (first column). From this table, 5×5 average filter (last column) removes more noise than 3×3 average filter, but also makes image edges and details more blurred. Due to this, it is important to choose an appropriate size when applying average filter to remove noise.

3.3 Gaussian Filter

Gaussian filter is a non-linear filter which performs 2D convolution operator over pixel values so as to blur images or to remove noise and image details. Gaussian filter

Figure 3.2: Impact of average filter size on noise removal: original noisy image (first column), denoised image with 3×3 average filter (second column), denoised image with 5×5 average filter (last column).

performs convolution similarly to average filter, but Gaussian filter uses a different convolution kernel which represents the Gaussian distribution. The formula of 2D Gaussian distribution is given below:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{3.2}$$

where x and y are distances from the origin in the horizontal axis and vertical axis respectively, and σ is the standard deviation of the 2D Gaussian distribution. The values from this distribution are sampled and used to build a 2D Gaussian convolution matrix using for image blur. Table 3.3 presents three examples of gaussian filter masks of sizes 3×3, 5×5 and 7×7 respectively.

In comparison with median filter, gaussian filter is faster in removing noise since multiplying and adding operations is often faster than sorting operations. With mean filter, Gaussian filter is slower but offers better performance of removing noise in frequency domain. This is since mean filter has little ability to separate one band of frequencies from another.

Figure 3.3 shows an example of using gaussian filter to remove noise from a noisy cameraman image (first column). From this figure, we can see that gaussian filter reduces noise from the image but also removes the image details and makes it more blurred.

3.4 Wiener Filter

Weiner filter is a frequency-domain method for image denosing. Wiener filter removes noise from an image signal by minimizing mean square error between the original noiseless image signal and the received image signal. Wiener filter assumes that both signal and noise are stationary linear stochastic processes with known spectral properties.

Figure 3.4 shows an example of using wiener filter to remove noise from a noisy

Table 3.3: Example of Gaussian Filters

3×3 Gaussian Filter

$\frac{1}{16}$

1	2	1
2	4	2
1	2	1

5×5 Gaussian Filter

$\frac{1}{273}$

1	4	7	4	1
4	16	26	16	4
7	26	41	26	7
4	16	26	16	4
1	4	7	4	1

7×7 Gaussian Filter

$\frac{1}{1003}$

0	0	1	2	1	0	0
0	3	13	22	13	3	0
1	13	59	97	59	13	1
2	22	97	159	97	22	2
1	13	59	97	59	13	1
0	3	13	22	13	3	0
0	0	1	2	1	0	0

cameraman image with additive gaussian noise (first column). From this figure, we can see that wiener filter reduces noise from the image but also makes it more blur.

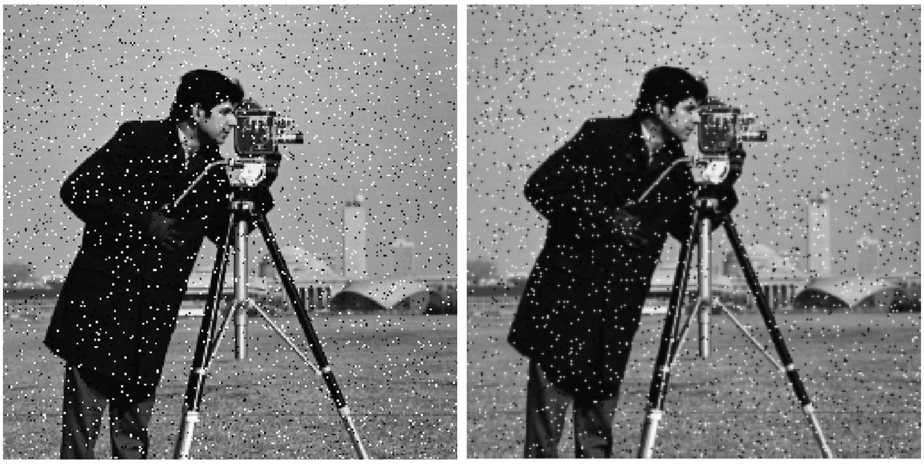

Figure 3.3: Gaussian filter on noise removal: original noisy image (first column), denoised image with 5×5 gaussian filter (second column).

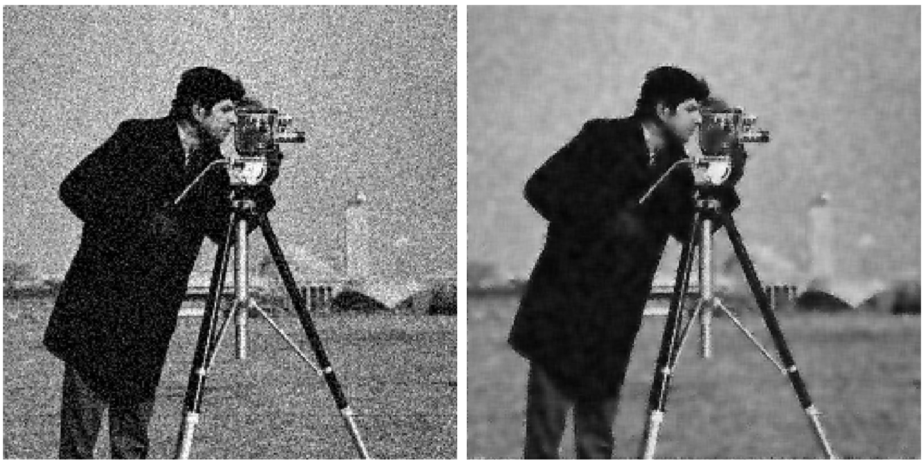

Figure 3.4: Wiener filter on noise removal: image with gaussian noise (first column), denoised image with 5×5 wiener filter (second column).

Chapter 4

Results

4.1 Experimental Setup

4.1.1 Dataset

4.1.2 Image Quality Assessment

Denoising noise helps in eliminating noise from images, but may also create image distortion. Due to this, it is important to evaluate quality of denoised image in order to assess performance of a noise reduction method. The measures used to evaluate image quality in this internship include Mean Square Error (MSE) and Peak Signal to Noise Ratio (PSNR).

Mean Square Error (MSE): Having the original image f of the size $m \times n$ and its denoised image f' of the same size, the mean square error is defined as follows:

$$MSE = \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} (f'_{ij} - f_{ij})^2 \tag{4.1}$$

MSE is determined as the mean of square of the difference between the original image and its denoised image. MSE presents the quality of denoised image in the sense that the smaller MSE is the better quality the denoised image has. The ideal case is when MSE is equal to zero. It means that the noise is completely removed from the image.

Peak Signal to Noise Ratio (PSNR): Having the MSE of the original image f of the size mxn and its denoised image f' of the same size, the peak signal to noise ratio measure is determined as follows:

$$PSNR = 10 \log_{10} \frac{R^2}{MSE} \tag{4.2}$$

where R is the maximum value of pixel intensity. For example, if the input image has an 8-bit unsigned integer data type, then R is equal to 255. The value of $PSNR$

Figure 4.1: Example of 4 photo classes. From left to right: Construction site, Rubbish, Pagoda, Pond

presents for the quality of image. In other words, the higher the value of $PSNR$ is, the better quality the image has.

Dataset: The experiments are performed using a total of 20 photos in the ICT-Lab's SWARMS project dataset. Each photo is artifically noise-added with Gaussian noise. As previously described, this dataset is composed of 800 photos, collected by USTH bachelor students. These images are categorized into 4 groups: Construction site, Rubbish, Pagoda and Pond, each containing 200 photos. Figure **??** shows examples of the four photo classes.

4.2 Results and Discussion

This section shows evaluation result of each described denoising method against 5 randomly chosen photos of each class. Expected results are answers to these questions:

- Are the denoising methods effective? (i.e. can they reduce noise?)
- What is the best method for denoising SWARMS photos?

The next part of this section will present evaluation result for each photo category.

Construction Sites

Figure 4.2 shows the images being used for the experiments in the first class. This class usually contains similar color tone and noise throughout the images.

Figure 4.2: Evaluated construction site photos. From left to right: construction1, construction2, construction3, construction4, construction5

Table 4.1: Evaluated metrics for construction site photos

	Noise		Median		Average		Gaussian		Wiener	
	MSE	PSNR	MSE	PSNR	MSE	PSNR	MSE	PSNR	MSE	PSNR
Construction 1	**0.0065**	**69.9985**	0.0097	68.2810	0.0122	67.2551	0.0095	68.3646	0.007	69.6586
Construction 2	0.0075	69.3964	0.0055	70.6970	0.0069	69.7596	0.0057	70.5950	**0.0045**	**71.6051**
Construction 3	0.0076	69.3395	0.0034	72.7804	0.0041	71.9945	0.0035	72.6697	**0.0028**	**73.6135**
Construction 4	0.0067	69.8582	0.0026	73.9885	0.0037	72.5057	0.0031	73.1572	**0.0022**	**74.6431**
Construction 5	0.0134	66.8630	0.0048	71.2836	0.0067	69.8649	0.0056	70.6357	**0.0046**	**71.4736**

Rubbish

Figure 4.3 shows the images being used for the experiments in the second class. This class shows a large diversity of colors due to existence of different small objects in the photo.

Figure 4.3: Evaluated rubbish photos. From left to right: rubbish1, rubbish2, rubbish3, rubbish4, rubbish5

Table 4.2: Evaluated metrics for rubbish photos

	Noise		Median		Average		Gaussian		Wiener	
	MSE	PSNR	MSE	PSNR	MSE	PSNR	MSE	PSNR	MSE	PSNR
Rubbish 1	0.0065	69.9809	0.0032	73.1106	0.0042	71.9250	0.0036	72.6244	**0.0024**	**72.2544**
Rubbish 2	0.0073	69.5154	0.0036	72.5118	0.0044	71.7414	0.0038	72.3736	**0.0029**	**73.4865**
Rubbish 3	0.0130	66.9853	0.0083	68.9621	0.0096	68.3076	0.0081	69.0201	**0.0063**	**70.1365**
Rubbish 4	0.0181	65.5581	0.0206	64.9940	0.0218	64.7541	0.0186	65.4329	**0.0160**	**66.0874**
Rubbish 5	0.0074	69.4444	0.0055	70.7083	0.0062	70.1895	0.0052	70.9374	**0.0040**	**72.0643**

Pagoda

Figure 4.4 shows the images being used for the experiments in the third class. This class show close color tone and noise for the images.

Figure 4.4: Evaluated pagoda photos. From left to right: pagoda1, pagoda2, pagoda3, pagoda4, pagoda5

Table 4.3: Evaluated metrics for pagoda photos

	Noise		Median		Average		Gaussian		Wiener	
	MSE	PSNR	MSE	PSNR	MSE	PSNR	MSE	PSNR	MSE	PSNR
Pagoda 1	0.0068	69.8109	0.0099	68.1878	0.0110	67.6997	0.0093	68.4539	**0.0055**	**70.7154**
Pagoda 2	0.0074	69.4260	0.0098	68.2235	0.0104	67.9661	0.0088	68.6992	**0.0053**	**70.8701**
Pagoda 3	0.0068	69.7773	0.0114	67.5480	0.0122	67.2754	0.0103	68.0138	**0.0065**	**69.9739**
Pagoda 4	0.0067	69.8514	0.0080	69.1193	0.0088	68.6955	0.0072	69.5489	**0.0047**	**71.4516**

Pond

Figure 4.5 shows the images being used for the experiments in the third class. A lot of similarity in pixel intensities and very little diversion of noise are showed in this class.

Figure 4.5: Evaluated pond photos. From left to right: pond1, pond2, pond3, pond4, pond5

Table 4.4: Evaluated metrics for pond photos

	Noise		Median		Average		Gaussian		Wiener	
	MSE	PSNR	MSE	PSNR	MSE	PSNR	MSE	PSNR	MSE	PSNR
Pond 1	0.0134	66.8492	0.0071	69.6058	0.0083	68.9171	0.0071	69.6413	**0.0057**	**70.5647**
Pond 2	0.0071	69.6281	0.005	71.1779	0.0056	70.668	0.005	71.1810	**0.0037**	**72.4836**
Pond 3	0.0071	69.6	0.0063	70.1105	0.0068	69.8103	0.0059	70.4217	**0.0045**	**71.6139**
Pond 4	0.0072	69.5324	0.0044	71.6815	0.0054	70.8081	0.0045	71.5637	**0.0034**	**72.8357**
Pond 5	0.007	69.6907	0.0076	69.3224	0.0079	69.1533	0.0065	70.0252	**0.0044**	**71.6963**

Discussion

Tables **??**, **??**, 4.3 and **??** show that in most cases of all categories, denoised images achieve smaller MSE and greater PSNR than the noisy image. This result indicates that all of the studied methods, namely Median filter, Average filter, Gaussian filter and Wiener filter, can reduce amount of noise in SWARMS dataset. These methods are therefore suitable for improving image quality for captured images.

To answer the second question from previous section, "What is the best method for denoising SWARMS photos?", it can be seen from all result tables that Weiner filter generally achieves best MSE and PSNR values, with one exception in rubbish2 photo. This accomplishment of Weiner filter can be explained as a result of its initial aim to minimize mean square error. However, this algorithm is performance-costly and (in average) takes 3.9 times longer than Average filter, 5.6 times longer than Gaussian filter and 1.3 times longer than Median filter.

Table 4.5 presents this runtime result for one photo as delegation for each class in the dataset.

Table 4.5: Method runtimes (in milliseconds)

Photo	Median	Average	Gaussian	Weiner
construction1	9.762	3.038	2.161	15.524
rubbish1	9.145	3.187	2.158	11.795
pagoda1	8.909	2.998	2.170	12.372
pond1	7.543	2.68	1.826	7.329

Chapter 5

Conclusion

This internship focuses on studying denoising methods for digital photos. These methods include Median filter, Average filter, Gaussian filter and Weiner filter. These methods are used in the context of SWARMS project with its captured photos in 4 categories: construction site, rubbish, pagoda and pond. The evaluation shows that all of these 4 methods are effective in denoising artifically-noised images. Additionally, Weiner filter achieves best noise reduction among these methods.

This work can be continued in several directions. First, other methods in frequency domain could be experimented, such as Stein's Unbiased Risk Estimator SureShrink [7], BlockShrink [8] or NeighShrink [9]. Hybrid denoising methods in both spatial and frequency domains [10] are also promising. Second, other image classes could be evaluated to see what kind of image could be best denoised. Last but not least, the experimented should also be quantitative and qualitative compared, not only based on two metrics (MSE and PSNR).

Bibliography

[1] ICTLab Website. http://ictlab.usth.edu.vn/. Accessed September, 2017.

[2] Ce Liu, Richard Szeliski, Sing Bing Kang, C Lawrence Zitnick, and William T Freeman. Automatic estimation and removal of noise from a single image. *IEEE transactions on pattern analysis and machine intelligence*, 30(2):299–314, 2008.

[3] Tao Chen, Kai-Kuang Ma, and Li-Hui Chen. Tri-state median filter for image denoising. *IEEE Transactions on Image processing*, 8(12):1834–1838, 1999.

[4] Chin-Chen Chang, Ju-Yuan Hsiao, and Chih-Ping Hsieh. An adaptive median filter for image denoising. In *Intelligent Information Technology Application, 2008. IITA'08. Second International Symposium on*, volume 2, pages 346–350. IEEE, 2008.

[5] Javier Portilla, Vasily Strela, Martin J Wainwright, and Eero P Simoncelli. Image denoising using scale mixtures of gaussians in the wavelet domain. *IEEE Transactions on Image processing*, 12(11):1338–1351, 2003.

[6] Guoliang Fan and Xiang-Gen Xia. Image denoising using a local contextual hidden markov model in the wavelet domain. *IEEE signal processing letters*, 8(5):125–128, 2001.

[7] David L Donoho. De-noising by soft-thresholding. *IEEE transactions on information theory*, 41(3):613–627, 1995.

[8] D. Zhou and X. Shen. Image denoising using block thresholding. In *2008 Congress on Image and Signal Processing*, volume 3, pages 335–338, May 2008.

[9] Zhou Dengwen and Cheng Wengang. Image denoising with an optimal threshold and neighbouring window. *Pattern Recognition Letters*, 29(11):1694 – 1697, 2008.

[10] Huanjing Yue, Xiaoyan Sun, Jingyu Yang, and Feng Wu. Cid: Combined image denoising in spatial and frequency domains using web images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2933–2940, 2014.