

Improvement of Non-Local Means Algorithm Using Similarity in Image

Jeongwoo Lee*, Heeyeon Jo*, Jiyun Byun**, Hongrae Lee***

*Student, Division of Digital Healthcare, Yonsei University, Wonju, Korea

**Student, Software Division, Yonsei University, Wonju, Korea

***Professor, Software Division, Yonsei University, Wonju, Korea

[Abstract]

With the widespread adoption of smartphones, acquiring images has become easier. However, challenges arise due to uneven lighting conditions at night and the degradation and noise introduced during image transmission and compression. To minimize this noise and improve image quality, Non-Local Means (NLM) techniques are used, which unlike traditional methods, seek out patches within the image that are similar to the current patch to eliminate noise. However, a drawback of NLM is the diminishing utility as the similar patches become larger. This paper proposes a noise reduction method that utilizes the Sum of Absolute Differences to calculate similarity and applies weights accordingly. The proposed algorithm demonstrates an average improvement of 6.911dB in Peak Signal-to-Noise Ratio (PSNR) on Salt and Pepper noise images, showing a 0.713dB improvement over traditional NLM. When the proposed algorithm is applied to existing NLM optimization papers, performance improvements can be expected.

▶ **Key words:** Non-Local Means, SAD, Image Denoising, Salt and Pepper, Similarity, Look Up Table

[요 약]

스마트폰의 보급화에 의해 영상은 쉽게 획득할 수 있지만 야간의 조명 조건의 불균형성 그리고 영상 데이터의 전송과 압축 과정에서 열화와 잡음(Noise)이 생성되기도 한다. 이러한 잡음을 최소화하고 영상 화질을 개선하기 위해 Non-Local Means(NLM)은 주변의 픽셀값이 아닌 이미지 내에서 현재 patch와 유사한 patch를 찾아 잡음을 제거한다. 하지만, 유사도를 측정하는 데 있어서 유사한 patch가 커질수록 유용성이 떨어지는 단점을 가지고 있다. 본 논문에서는 Sum of Absolute Differences 연산을 이용하여 유사도를 계산하고 유사도에 따라 가중치를 적용하는 잡음 제거 방법을 제안한다. 제안한 알고리즘을 사용하여 Salt and Pepper 잡음 이미지에 PSNR 개선이 평균 7.98dB 향상되며 NLM 대비 1.06dB 개선된다. 제안한 알고리즘을 기존 NLM 최적화 논문에서 적용하였을 때 성능 향상을 기대할 수 있다.

▶ **주제어:** 비 지역적 평균 기법, SAD, 영상 잡음 제거, 소금 후추 잡음, 유사도, 룩업테이블

- First Author: Jeongwoo Lee, Corresponding Author: Hongrae Lee
- *Jeongwoo Lee (jwlee87@yonsei.ac.kr), Division of Digital Healthcare, Yonsei University
- *Heeyeon Jo (7531hy@yonsei.ac.kr), Division of Digital Healthcare, Yonsei University
- **Jiyun Byun (pinkwin1004@yonsei.ac.kr), Software Division, Yonsei University
- ***Hongrae Lee (lukas0712@yonsei.ac.kr), Software Division, Yonsei University
- Received: 2024. 10. 29, Revised: 2024. 11. 18, Accepted: 2024. 11. 21.

I. Introduction

디지털 영상처리는 현대 기술 환경에서 다양한 응용 분야에 광범위하게 적용되고 있으며 의료, 멀티미디어, 통신, 국방 등에서 그 중요성이 강조되고 있다[1]. 고해상도 및 고품질의 영상을 효과적으로 생성하고 보존하는 것은 영상처리 분야의 주요 연구 주제이다. 그러나 센서로 인한 잡음, 양자화 과정에서의 오차, 조명 조건의 불균형성, 그리고 영상 데이터의 전송과 압축 과정에서의 오류 등이 주요 원인으로 인해 영상에는 잡음이 쉽게 형성된다[2]. 이러한 과정에서 발생하는 잡음으로 대표적인 것은 Salt and Pepper(SNP) 잡음, 가우시안(gaussian) 잡음, 균일(uniform) 잡음이다[3]. 영상 내의 잡음은 단순히 영상의 품질 저하만을 초래하는 것이 아니라, 후속 영상처리의 정확도와 효율성에도 부정적인 영향을 미친다[4]. CT 또는 MRI 영상에 잡음이 추가될 경우, 환자의 병명 진단에 영향을 주게 되어 심각한 문제를 초래할 수 있다[5]. 따라서 SNP 잡음, 가우시안 잡음을 제거하기 위해 가우시안 필터(Gaussian Filter), 중간값 필터(Median Filter), 위너 필터(Wiener filter), 그리고 Non-Local Means(NLM) 기법이 연구와 응용에서 주로 활용되었다[6][7]. NLM 기법은 잡음 영상에 대해 뛰어난 성능을 보여 주목받는 잡음 제거 방식으로 관련 연구가 활발히 진행되고 있다. [7]은 NLM 알고리즘의 최적화를 위해 Search Patch(SP)와 Search Window(SW) 크기를 각각 7×7 , 21×21 로 설정하는 NLM 알고리즘을 제안하였다. [8]은 NLM에서 가우시안 함수 폭을 데이터에 따라 조절하기 어려운 점을 지적하고, NLM 알고리즘의 성능 향상을 위해 SURE-Based NLM을 제안하였다. [9]은 NLM의 계산 비용이 높은 점을 지적하고, 이를 해결하기 위해 이웃 SW의 유사성을 계산하는 데 있어서 근사치 측정, Summed Square Image(SSI) 방식, 그리고 Fast Fourier Transform(FFT)를 사용하여 계산 속도를 향상시키는 새로운 알고리즘을 제안하였다. [10]은 NLM의 가중치 함수의 한계를 지적하고, 알고리즘의 성능을 높이기 위해 가우시안 잡음의 patch 별 차이를 활용한 확률적 NLM 기법을 제안하였다. [11]은 NLM의 계산 비용이 높은 문제를 해결하기 위해 Singular Value Decomposition(SVD)을 사용하여 픽셀 쌍의 계산을 효율적으로 줄이는 방법을 제안하였다.

하지만, NLM 알고리즘은 잡음 제거를 수행하기 위하여 이미지 내의 SW에서 SP를 탐색하여 Euclidean Distance로 유사도를 계산하고, 그 결과에 따른 가우시안 함수값을 곱한 값의 가중평균을 계산한다[12]. Euclidean Distance

는 차수가 증가함에 따라 유사성 측정의 효율성이 감소하게 되어 SW가 15 이상일 때 PSNR이 감소하는 문제점이 있다[13]. 이러한 문제점을 해결하기 위하여 본 논문에서는 두 이미지 간의 절대 차이를 선형적으로 계산하는 Sum of Absolute Differences(SAD) 연산[14]으로 두 patch 간의 절대값 거리를 가중치로 이용하여 유사도를 측정하게 함으로 계산 복잡도 감소 및 안정적인 비교 측정 방법을 제안한다. 더불어, SP와 SW의 크기 변환을 통해 실험 이미지의 특성에 따른 제안된 알고리즘의 성능을 비교 평가하였다.

본 논문의 구성은 다음과 같다. II장에서 NLM 알고리즘의 관련된 연구에 관해 서술하며, III장에서는 제안하는 NLM 알고리즘에 대해 자세히 설명한다. IV장에서는 실험을 통해 제안하는 방법의 성능을 검증하고, 마지막으로 V장에서는 본 논문의 결론을 서술한다.

II. Related Works

영상의 잡음을 제거하는 방법은 크게 Local 영역에서의 잡음 제거 기법과 Non-Local 영역에서의 잡음 제거 기법 등으로 구분될 수 있다. Local 기법은 대상 픽셀의 인접한 정보를 사용한 smoothing 방식으로 대다수 잡음을 제거한다. 하지만 smoothing 방식은 edge와 같은 영상의 디테일이 소실되거나 texture가 뭉개지는 단점이 발생한다. 반면, Non-Local 기법은 영상의 광범위한 영역에서 유사성을 기반으로 한 패턴이나 구조를 탐색하여 잡음을 제거하므로 세부적인 특징을 보존하는 데에 유리하다.

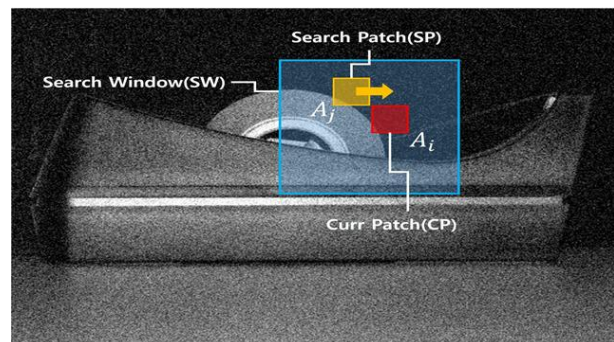


Fig. 1. Search Patch(SP), Curr Patch(CP) and Search Window(SW) in Gaussian Noise Image

Fig. 1은 NLM 알고리즘에서 사용되는 SP, CP 그리고 SW를 나타내는 그림이다. NLM은 잡음 제거할 영역인 CP와 유사한 영역을 찾기 위하여 SP를 sliding window 방

식으로 SW 내에서 이동하며 탐색하게 된다. 이 때 SP내의 픽셀과 CP의 픽셀 간의 값을 이용하여 유클리드 거리를 계산하여 유사도를 구하게 된다. 이렇게 계산된 유사도는 가중치 계산에 사용되며 이에 대한 식은 아래와 같이 나타낸다.

$$Z(i) = \sum_j \exp\left(-\frac{\|A_i - A_j\|_{2,a}^2}{h^2}\right) \quad (1)$$

식 (1)에서 가중치는 CP의 구성 픽셀(A_i)와 SP의 구성 픽셀(A_j) 간의 Euclidean Distance를 계산한 뒤에 표준편차가 a 인 Gaussian distribution를 적용하며 $Z(i)$ 는 exponential 항을 모두 합한 값으로 구해진다. 이렇게 계산된 가중치 값의 합 $Z(i)$ 는 잡음을 제거하기 위한 가중평균을 계산하는데 사용된다. patch 크기만큼의 가중치 값은 전부 합하여 SP의 중앙 픽셀에 곱해진다. SP가 SW를 전부 순회하며 구한 가중치가 곱해진 픽셀값의 합을 가중치의 합으로 나누어 가중평균으로 잡음이 있는 픽셀을 대체한다. 이에 대한 식은 아래와 같이 나타낸다.

$$\bar{x} = \frac{\sum_{i=1}^n z_i x_i}{\sum_{i=1}^n z_i} \quad (2)$$

식 (2)에서 n 은 SW 내의 SP의 수, z_i 는 식1에서 계산된 SP에 따른 가중치, x_i 는 SP의 중앙 픽셀값이다. 가중치 값이 곱해진 SW 내 모든 픽셀은 가중치의 총합으로 나누어져 가중평균 \bar{x} 를 계산하게 된다. 계산된 가중평균 \bar{x} 는 CP의 구성요소의 가운데 값으로 변경되어 잡음을 제거한다.

이러한 과정을 수행하기 위해서 각 patch 크기는 SW 내에서 sliding window 방식으로 순회해야 하기 때문에 SW 크기보다 항상 작아야 한다. CP 크기는 잡음 제거할 픽셀의 주변 픽셀들을 사용하므로 1을 제외한 홀수의 크기로 정해지며, SP 크기는 유사도가 두 patch 간의 차를 기반으로 계산되므로 CP 크기와 동일하게 설정된다. SW와 SP의 크기에 따라 알고리즘의 잡음 제거 품질에 영향을 주며[15], patch의 크기가 클수록 계산 복잡도가 커지게 된다. 또한 SW와 각 patch의 크기가 커질수록 두 patch 간의 직선거리를 이용하여 유사도를 측정하는 Euclidean Distance의 유용성이 떨어지는 문제점이 발생된다.

본 논문에서는 이러한 문제를 해결하기 위해 두 patch 간의 절댓값 거리를 이용하여 유사도를 측정하는 SAD 연

산으로 유사도를 측정하고, 유사 정도에 따른 가중치를 가우시안 함수로 차분 적용하는 방법을 제안한다. 이와 관련된 내용은 III 장에서 다루도록 한다.

III. The Proposed Scheme

본 논문에서 제안하는 방법은 기존의 NLM과 같이 SP가 SW 내에서 sliding window 방식으로 이동하며 CP와의 유사도를 계산한다. 하지만, 유사도를 계산하는데 기존 방식인 Euclidean Distance가 아닌 SAD 연산을 이용하여 가중치를 계산한다. Fig. 2는 제안하는 SAD 연산의 가중치 부여 방식의 전체적인 과정을 나타낸다.

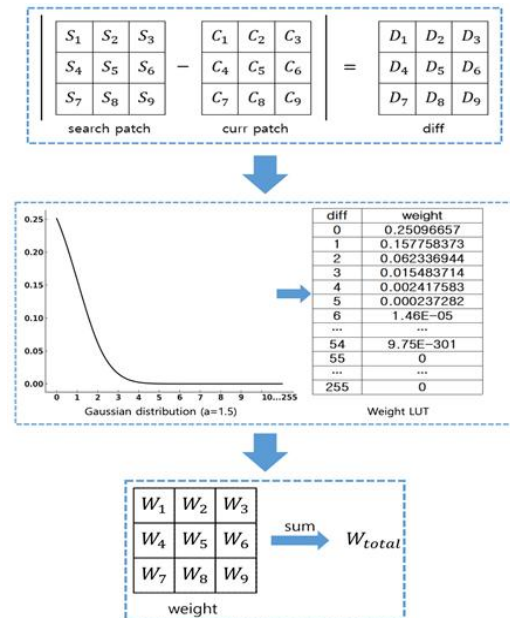


Fig. 2. The overall process of weighting method using SAD

Fig. 2에서 C_5 는 잡음 제거할 픽셀로, 처음 C_5 의 위치는 패딩된 이미지에서 원본 이미지의 좌측 최상단 픽셀이며 C_5 를 중심으로 patch 크기만큼 SP를 설정한다. 여기서 CP와 SP의 위치가 이미지의 가장자리 경우에는 Mirror 패딩을 이용하여 빈 픽셀에 대한 픽셀을 채워서 적용하였다. 그 후 S_1 은 C_1 , S_2 는 C_2 처럼 같은 좌표에 있는 픽셀 간의 차를 구한 뒤, 구한 값들 각각에 절댓값을 취한다. 예를들어 이미지의 픽셀은 0부터 255까지의 밝기 값을 가진다. SP 내 특정 픽셀값이 0이고 같은 위치의 CP 픽셀값이 255인 경우 차를 구하면 -255이 되고 이에 대한 절댓값으로 인해 연산 결과는 255가 된다. 이러한 계산 결

과를 patch 크기와 같은 diff에 저장하고 diff에 저장된 값들을 Gaussian distribution에 적용하여 각 픽셀에 대한 가중치를 계산하게 된다. 이에 대한 식은 아래와 같이 나타낸다.

$$weight(i, j) = \exp\left(-\frac{|d(i, j)|_a}{h^2}\right) \quad (3)$$

식 (3)에서 i, j 는 같은 크기의 비교 대상인 SP와 CP의 x축, y축 좌표를 의미하며, $d(i, j)$ 는 diff에 저장된 두 patch의 i, j 좌표에 있는 원소 간의 차이값을 나타내며 이 값을 Gaussian distribution에 적용하여 각 픽셀에 대한 가중치를 계산하게 된다. 하지만, 각 픽셀에 대한 가중치를 매번 계산하게 되면 계산 복잡도가 높아지게 된다. 이에 본 논문에서는 계산 복잡도를 줄이기 위해 diff가 가질 수 있는 가중치 값을 미리 계산하여 LUT(Lookup Table)로 작성하여 적용하였다. LUT에서 diff가 가질 수 있는 범위는 픽셀 간의 차이가 음수 또는 양수일 때 동일하므로 대칭으로 처리하여 0~255 범위에 대한 가중치 값만을 이용하였다. 유사도가 가장 높은 픽셀은 차이값이 0이므로 LUT의 인덱스가 0인 1.0 가중치 값이 부여된다. SP와 CP의 차이값이 가장 큰 경우에는 255이므로 LUT의 인덱스가 255의 가중치 값이 부여된다. 이와 같은 과정이 Patch의 크기만큼 진행되어 diff 내의 차이값인 D_1 부터 D_9 는 LUT로 매핑되는 가중치 값인 W_1 부터 W_9 를 가진다.

가중평균을 계산하여 잡음을 제거하는 과정은 NLM과 동일하게 진행된다. W_1 부터 W_9 까지의 가중치 값은 전부 더하여 W_{total} 이 되고, W_{total} 는 SP의 중앙값 S_5 와 곱해진다. SP가 SW를 sliding window 방식으로 순회하며 구한 SP마다의 W_{total} 을 누적하여 더하고, W_{total} 과 S_5 을 곱한 값도 누적되어 더한다. 누적되어 더해진 두 값을 나누어 가중평균을 구한 값으로 C_5 을 대체한다.

이러한 잡음 제거 과정은 SW가 이미지 전체를 sliding window 방식으로 순회할 때까지 진행된다.

IV. Experiments

본 장에서는 Proposed NLM(PNLM)의 성능을 실험을 통하여 검증하고, 잡음제거의 대표적인 필터인 Gaussian Filter와 Median Filter 및 기존의 NLM과의 성능을 비교한다. 또한, 제안 방법의 patch와 window 크기 변화에 따른 잡음 제거 성능을 비교한다. 기존 NLM 알고리즘은

[16]에 구현된 코드를 참조한다. 성능 비교에는 영상의 화질에 대한 손실 정보를 평가하는 PSNR 지표를 사용하며 이를 통해 잡음을 제거한 이미지가 원본 이미지에 비해 얼마나 손실되었는지를 비교한다. Table. 1은 본 논문의 실험 환경을 나타낸다.

Table 1. Experimental Setup

Component	Specification
CPU	11th Gen Intel(R) Core(TM) i7-1165G7 @ 2.80GHz
RAM	16.0GB
OS	Windows 11
Language	Python 3.10.11
IDE	PyCharm IDE 2023.2

실험에 사용된 이미지는 MATLAB의 Image Processing Toolbox에 포함된 예제 이미지 중 10장의 이미지와 표준 실험 이미지인 Lena를 사용하였다. Table. 2는 실험에 사용한 원본 이미지들의 이름과 크기를 나타내며 Fig. 3는 원본 이미지들을 보여준다.

Table 2. Information about Each Test Image

Tag & Name	Image Size	Tag & Name	Image Size
(a) peppers	512 × 384	(g) tire	232 × 205
(b) corn	312 × 415	(h) saturn	1200 × 1500
(c) cameraman	256 × 256	(i) tape	512 × 384
(d) coins	300 × 246	(j) fabric	640 × 480
(e) moon	358 × 537	(k) Lena	512 × 512
(f) rice	256 × 256		

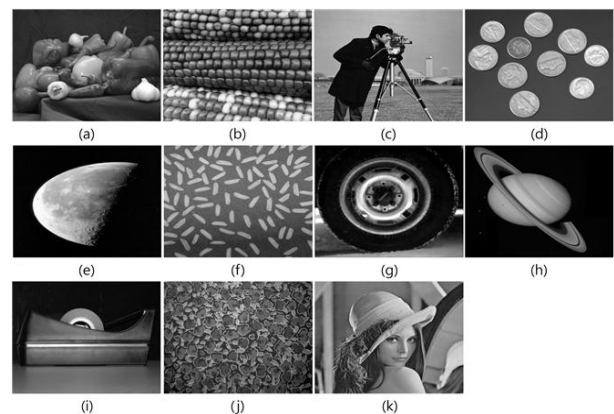


Fig. 3. Original Test Images

제안된 방법의 성능을 실험하기 위해 Fig. 3. 이미지들에 scikit-image 라이브러리를 사용하여 5%, 10%, 15%, 20%의 SNR 잡음을 적용하여 실험 이미지들을 제작한다. 잡음은 랜덤으로 적용되기 때문에 매 실험에 적용할 경우 차이가 발생된다. 본 논문의 실험은 동일한 SNR 잡음 이

미지로 실험을 진행하였다. Fig. 4는 제안된 방법의 성능을 실험하기 위해 Fig. 3 이미지 중 (c)에 5%, 10%, 15%, 20%의 SNP 잡음을 포함한 실험 이미지들을 나타낸다.

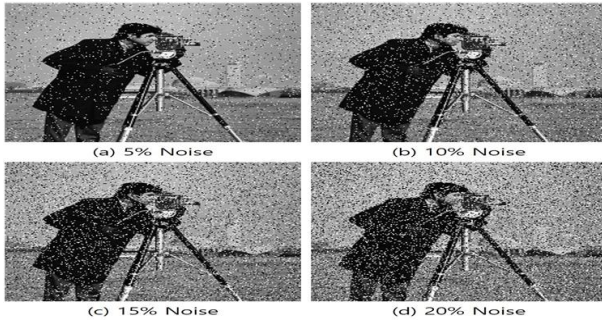


Fig. 4. Different Levels of Noise in Subfigure (c) of Fig. 3

첫 번째 실험은 Local 영역에서의 잡음 제거 기법과 Non-Local 영역에서의 잡음 제거 기법의 성능 비교를 위해 Gaussian Filter와 Median Filter 및 기존의 NLM과 PNLM을 사용하였다. 실험에서는 5% SNP 잡음을 Fig. 3의 이미지들에 적용하여 사용하였으며 Gaussian Filter는 OpenCV의 gaussianblur 함수[17]를 사용하였고, Median Filter는 [18]을 사용하였다. NLM과 PNLM의 SP와 SW 크기를 7×7, 21×21로 동일하게 설정하여 성능 비교를 하였다. Table. 3은 PNLM과 NLM, Gaussian Filter 그리고 Median Filter의 PSNR 잡음 제거 성능을 나타낸다.

Table 3. Comparison of PSNR Performance between PNLM, NLM, Gaussian Filter, and Median Filter in 5% SNP Noisy

Image Tag	Gaussian Filter	Median Filter	NLM	PNLM	Gaussian Filter - Noisy	Median Filter - Noisy	NLM - Noisy	PNLM - Noisy
(a)	24.08	26.81	31.38	31.66	1.99	4.72	9.29	9.58
(b)	24.02	26.13	27.03	28.11	1.95	4.06	4.95	6.03
(c)	23.91	25.37	26.13	28.07	1.87	3.33	4.09	6.03
(d)	24.31	26.71	28.46	30.29	1.96	4.36	6.10	7.94
(e)	22.62	25.49	30.45	31.16	1.93	4.80	9.76	10.50
(f)	24.29	25.96	28.56	29.96	1.91	3.57	6.18	7.58
(g)	22.93	25.52	27.19	28.56	1.92	4.51	6.18	7.56
(h)	22.63	25.45	32.32	32.59	1.94	4.76	11.6	11.90
(i)	23.48	26.22	30.74	31.36	1.95	4.69	9.21	9.84
(j)	24.25	26.61	26.07	27.86	1.94	4.29	3.76	5.54
(k)	23.30	24.59	26.66	27.02	1.63	2.92	4.98	5.35
Overall					1.91	4.18	6.92	7.98

Table. 3을 보면 대표적인 Local 영역에서의 잡음 제거 기법인 Gaussian Filter와 Median Filter는 잡음 픽셀의 인접한 정보를 이용한 smoothing 방식을 사용하여 Gaussian Filter는 평균 1.91dB, Median Filter는 평균 4.18dB 개선하였다. 반면 영상의 광범위한 영역에서 유사

성을 기반으로 잡음을 제거하는 Non-Local 기법을 사용한 NLM은 평균 6.92dB, PNLM은 평균 7.98dB 개선하였다. 이를 통해 잡음 주변만이 아닌 광범위한 영역에서 유사성을 기반으로 잡음을 제거하는 것이 더 효율적인 것을 알 수 있다. 이 결과에서 유사도 계산에서 차이가 있는 NLM과 PNLM의 잡음 제거 기법의 성능을 비교하면 (e), (h)와 같이 edge가 적고 평활한 이미지에서는 NLM과 PNLM 둘 다 높은 잡음 제거 효과를 나타냈다. 하지만, (c), (j)와 같이 edge가 많고 복잡한 이미지에서는 PNLM이 더 높은 개선 효과를 보였는데, 이는 NLM은 patch 단위로 유사도를 계산하지만 PNLM은 픽셀 단위로 유사도를 계산하여 가중치를 부여하기 때문에 더 정교한 잡음 제거가 가능하였다.

두 번째 실험은 잡음 정도에 따른 NLM과 PNLM의 잡음 제거 기법의 성능 비교를 하였다. 실험에서는 5%, 10%, 15%, 20% 잡음을 Fig. 3의 실험 이미지들에 적용하여 사용하였으며, NLM과 PNLM의 SP와 SW 크기를 7×7, 21×21로 동일하게 설정하여 성능 비교를 하였다. Table. 4는 NLM과 PNLM을 사용하여 잡음 제거한 결과를 PSNR로 보여주며, PNLM과 NLM의 차를 계산하여 잡음 제거 성능을 나타낸다.

Table. 4을 보면 5%, 10%, 15%, 20% 잡음에서 잡음 정도가 높을수록 전체적으로 NLM과 PNLM의 잡음 제거 성능이 저하되었지만, 전반적으로 NLM보다 PNLM이 높은 잡음 제거 성능을 보인다. 잡음 정도가 높아질수록 (c), (j)와 같이 이미지의 크기가 작으면서 edge가 많고 복잡한 이미지에는 잡음 제거 성능 차이가 줄어든 반면 (h)와 같이 이미지의 크기가 크면서 edge가 적고 평활한 이미지에서는 잡음 제거 성능 차이가 늘어난 것을 알 수 있다. 이는 잡음 정도가 높아질수록 픽셀 단위로 유사도를 계산하여 가중치를 부여한 방법이 평균 1dB 미만의 PSNR 차이를 보이므로 적은 연산량으로 평활한 이미지의 노이즈를 정교하게 제거하는데 도움이 된다는 것을 알 수 있다.

세 번째 실험은 제안된 유사도 계산법을 이용하였을 때의 계산 복잡도를 비교하였다. 실험에서는 5% SNP 잡음을 Fig. 3의 이미지들에 적용하여 사용하였으며 NLM과 PNLM의 SP와 SW 크기를 7×7, 21×21로 동일하게 설정하여 성능 비교를 하였다. Table. 5는 5% SNP 잡음 이미지에서 NLM과 PNLM의 유사도 계산에 따른 잡음 제거 수행 시간을 초(Seconds)단위로 나타낸다.

Table. 5를 보면 NLM과 PNLM은 이미지 크기가 가장 작은 (g)에서 가장 빠른 잡음 제거 시간을 보이고, 이미지 크기가 가장 큰 (h)에서 가장 느린 잡음 제거 시간을 보인다.

Table 4. Comparison of NLM and PNLM PSNR Performance at 7×7 and 21×21 Based on Noise Levels

Image Tag	Noisy	NLM	PNLM	NLM- Noisy	PNLM- Noisy	PNLM- NLM
(a)	22.086	31.379	31.665	9.293	9.578	0.286
(b)	22.072	27.026	28.107	4.954	6.035	1.081
(c)	22.040	26.133	28.069	4.093	6.029	1.936
(d)	22.354	28.458	30.295	6.105	7.941	1.837
(e)	20.692	30.447	31.157	9.755	10.465	0.710
(f)	22.382	28.560	29.958	6.179	7.576	1.398
(g)	21.009	27.191	28.565	6.182	7.556	1.374
(h)	20.689	32.323	32.593	11.634	11.903	0.269
(i)	21.522	30.735	31.363	9.213	9.841	0.628
(j)	22.313	26.070	27.857	3.757	5.544	1.787
(k)	21.678	26.662	27.024	4.984	5.346	0.362
Overall				6.923	7.983	1.061

Image Tag	Noisy	NLM	PNLM	NLM- Noisy	PNLM- Noisy	PNLM- NLM
(a)	17.702	22.135	22.345	4.433	4.642	0.210
(b)	19.012	24.629	25.152	5.617	6.141	0.523
(c)	19.114	24.410	25.936	5.296	6.823	1.526
(d)	19.050	25.685	27.304	6.636	8.254	1.618
(e)	17.574	26.077	26.757	8.503	9.183	0.680
(f)	19.509	26.034	27.179	6.525	7.670	1.145
(g)	17.960	24.396	25.292	6.436	7.332	0.896
(h)	17.506	26.449	27.086	8.943	9.579	0.636
(i)	18.489	26.765	27.360	8.276	8.871	0.595
(j)	18.594	22.334	23.013	3.740	4.418	0.679
(k)	17.240	20.051	20.198	2.811	2.959	0.148
Overall				6.111	6.898	0.787

(a) 5% Noise

Image Tag	Noisy	NLM	PNLM	NLM- Noisy	PNLM- Noisy	PNLM- NLM
(a)	16.341	21.454	21.631	5.113	5.289	0.176
(b)	17.225	22.894	23.099	5.669	5.874	0.206
(c)	17.152	22.659	23.710	5.507	6.558	1.050
(d)	17.248	23.677	24.942	6.429	7.694	1.265
(e)	15.595	22.819	23.467	7.223	7.872	0.649
(f)	17.695	24.129	24.916	6.434	7.222	0.787
(g)	16.040	22.118	22.698	6.078	6.659	0.580
(h)	15.636	23.274	24.009	7.638	8.373	0.734
(i)	16.610	24.102	24.611	7.491	8.000	0.509
(j)	17.101	21.732	22.202	4.631	5.101	0.470
(k)	16.172	19.803	19.923	3.631	3.751	0.120
Overall				5.986	6.581	0.595

(b) 10% Noise

Image Tag	Noisy	NLM	PNLM	NLM- Noisy	PNLM- Noisy	PNLM- NLM
(a)	15.286	20.797	20.878	5.510	5.591	0.081
(b)	15.859	21.371	21.340	5.512	5.481	-0.030
(c)	15.836	21.337	22.082	5.502	6.247	0.745
(d)	16.027	22.213	23.206	6.186	7.179	0.993
(e)	14.182	20.459	21.050	6.278	6.868	0.591
(f)	16.443	22.739	23.104	6.296	6.661	0.365
(g)	14.636	20.185	20.596	5.549	5.960	0.411
(h)	14.265	20.897	21.638	6.632	7.373	0.741
(i)	15.250	21.987	22.383	6.737	7.134	0.396
(j)	15.904	21.008	21.204	5.104	5.300	0.196
(k)	15.326	19.537	19.561	4.211	4.235	0.024
Overall				5.774	6.184	0.410

(c) 15% Noise

(d) 20% Noise

Table 5. Time Comparison (in seconds) between NLM and PNLM on 5% Noise Images

Image Tag	NLM processing time(s)	PNLM processing time(s)	Time Reduction (%)
(a)	63.667	31.082	-51.180
(b)	45.889	18.363	-59.983
(c)	26.655	9.847	-63.056
(d)	29.025	13.072	-54.963
(e)	62.555	28.650	-54.200
(f)	26.511	10.117	-61.838
(g)	20.199	7.803	-61.371
(h)	260.150	245.606	-5.591
(i)	53.037	27.342	-48.446
(j)	66.135	40.417	-38.887
(k)	60.296	34.138	-43.382
overall			-49.354

위의 결과에서 NLM 대비 PNLM 수행 시간의 감소율을 계산하기 위하여 다음의 식을 이용하였다.

$$TimeReduction = \frac{NLM_{PT} - PNLM_{PT}}{NLM_{PT}} \times 100 \quad (4)$$

식 (4)에서 NLMPT는 NLM의 수행 시간을 나타내며 PNLMPT는 PNLM의 수행 시간을 나타낸다. 위의 식을 통

해 전체적인 NLM 대비 PNLM 수행 시간이 평균적으로 49.354% 감소하였음을 알 수 있다. NLM과 PNLM의 계산 복잡도의 차이는 가중치 부여 단계에서 결정된다. NLM의 경우에는 patch 단위로 차이를 Gaussian distribution의 가중치에 적용하여 유사도를 계산하는 반면 PNLM의 경우 픽셀 단위로 차이를 미리 계산된 LUT을 가중치를 적용하여 유사도를 계산하기 때문에 동일한 이미지에 대해 NLM 보다 PNLM의 계산 복잡도가 약 50%정도 낮아지게 된다. 다만 이미지가 커질수록 PNLM은 patch 내 픽셀 단위로 가중치를 부여하므로 연산량이 늘어나게 된다. patch 내 픽셀 개수에 따라 연산량이 결정되는 PNLM은 patch 크기가 알고리즘의 시간을 결정하는 중요한 요인이 된다.

마지막으로 제안 방법의 SP와 SW 크기 변화에 따른 계산 복잡도 및 잡음 제거 성능 차이를 비교하였다. 실험에서는 5%, 10%, 15%, 20% 잡음을 Fig. 3의 실험 이미지들에 적용하여 사용하였으며, 각 잡음 정도에서 따라 SW 크기에 따른 SP 크기별 성능 비교를 위해 SW 크기는 21×21, 15×15, 9×9로, SP의 크기는 7×7, 5×5, 3×3으로 설정하였다. SP 3배 크기를 최대 SW 크기로 가정하여 SW가 21×21일 때, SP의 크기 7×7, 5×5, 3×3, SW가 15×15일 때, SP의 크기 5×5, 3×3, SW가 9×9일 때, SP의

크기 3×3으로 설정한 총 6개 case의 실험하였다.

Table. 6은 PNLM의 SW와 SP 크기 변화에 따른 계산 복잡도 차이를 비교하기 위해 각 잡음 정도에 대한 알고리즘 수행 시간을 SP 크기별로 평균을 내어 나타낸다.

Table. 6을 보면 각 잡음 정도와 상관없이 모든 이미지에 대한 평균 잡음 제거 연산 시간이 SW가 9×9이고 SP가 3×3 case에서 가장 빠르다는 것을 보여준다. 이는 SW의 크기가 작으면 SP의 탐색 횟수가 줄어들어 계산 요구가 줄어들고 더 빠른 잡음 제거가 가능함을 의미한다.

Table 6. Time comparison based on average (in seconds) for each noise level by SP and SW size

Image Tag	SP:3×3 SW:9×9	SP:3×3 SW:15×15	SP:5×5 SW:15×15	SP:3×3 SW:21×21	SP:5×5 SW:21×21	SP:7×7 SW:21×21
(a)	11.893	12.106	18.694	24.636	30.890	27.849
(b)	8.511	8.677	15.220	18.229	18.114	18.946
(c)	5.352	5.423	11.956	9.291	10.323	10.112
(d)	5.828	6.083	12.153	10.552	11.368	11.643
(e)	11.466	12.344	18.299	24.049	26.258	27.103
(f)	5.260	5.831	11.832	9.489	10.379	10.252
(g)	4.397	4.712	11.132	7.495	8.045	7.957
(h)	93.996	95.391	101.258	212.643	232.744	235.730
(i)	12.179	12.076	19.434	24.950	27.888	26.970
(j)	17.727	18.178	25.425	37.081	40.456	40.656
(k)	15.465	16.589	28.099	37.327	36.231	34.895

그리고 이미지 크기가 작은 (c), (d), (f), (g)는 SW가 15×15이고 SP가 5×5일 때 가장 느린 잡음 제거 수행 시간을 보이고, 크기가 큰 이미지는 각각 SW가 21×21일 때 (a)와 (i)는 SP가 5×5, (b), (e), (h), (j)는 7×7, (k)는 3×3일 때 가장 느린 잡음 제거 수행 시간을 보인다. 이는 이미지의 크기에 따른 SW와 SP크기가 잡음 제거 시간에 영향을 미치므로 최적의 SW와 SP 크기 설정이 필요함을 알 수 있다.

Table. 7은 PNLM의 SW와 SP 크기 변화에 따른 잡음 제거 성능을 확인하기 위해 각 잡음 정도를 제거하여 측정된 PSNR을 SP 크기별로 평균을 내어 나타낸다.

Table 7. PSNR performance based on average for each noise level by SP and SW size

Image Tag	noisy	SP:3×3 SW:9×9	SP:3×3 SW:15×15	SP:5×5 SW:15×15	SP:3×3 SW:21×21	SP:5×5 SW:21×21	SP:7×7 SW:21×21
(a)	17.854	19.627	20.876	22.826	21.537	24.231	24.129
(b)	18.542	19.837	21.506	22.875	22.478	24.447	24.425
(c)	18.535	20.425	21.897	23.970	22.629	25.328	24.949
(d)	18.670	20.838	22.441	24.922	23.192	26.730	26.437
(e)	17.011	19.486	21.079	24.035	21.845	26.384	25.608
(f)	19.007	20.545	22.153	24.279	23.043	26.301	26.289
(g)	17.411	19.292	20.778	22.918	21.558	24.585	24.288
(h)	17.024	20.307	22.047	25.216	22.636	27.540	26.331
(i)	17.968	20.268	21.886	24.592	22.701	26.834	26.429
(j)	18.478	19.776	21.183	22.582	21.936	23.629	23.569
(k)	17.604	18.746	19.734	20.839	20.259	21.629	21.677

Table. 7을 보면, (k)를 제외한 모든 이미지에서 SW가 21×21이고 SP가 5×5일 때 PSNR이 가장 높았고, (k)는 SW가 21×21이고 SP가 7×7일 때 PSNR이 가장 높았다. 그리고 모든 이미지에서 SW가 9×9이고 SP가 3×3일 때 PSNR이 가장 낮았다. 이는 SW가 클수록 많은 이미지 정보를 담고 있으므로 모든 이미지 유형에서 SW가 21×21일 때 가장 높은 잡음 제거 성능을 나타냈다. 또한 NLM의 경우 SW가 15×15 이상이 되면 PSNR이 감소한다고 하였으나[9], PNLM은 SP가 5×5, SW가 21×21에서 가장 높은 PSNR을 보이므로 NLM의 patch 단위로 유사도를 계산에 따른 Euclidean Distance의 유용성이 떨어지는 문제를 제안한 픽셀 단위로 유사도를 계산하는 방식으로 해결함을 알 수 있다.

V. Conclusions

본 논문에서는 SAD 연산을 사용하여 픽셀 단위로 유사도 계산하고 유사도에 따라 LUT을 이용하여 가중치를 부여하는 NLM 알고리즘을 제안하였다. PNLM을 통해 잡음 제거했을 경우, 5%의 잡음에서 PSNR이 NLM보다 평균 1.06dB 향상되었고 수행 시간은 평균 49.354% 감소하였다. 이는 잡음 정도가 높아져도 픽셀 단위로 유사도를 계산하여 가중치를 부여한 방법이 평균 1dB 미만의 PSNR 차이를 보이므로 적은 연산량으로 평활한 이미지의 노이즈를 정교하게 제거하는데 도움이 된다는 것을 알 수 있다. 또한 픽셀 단위로 가중치를 계산하는 PNLM의 최적의 수행 시간 및 잡음 제거 성능을 확인하기 위해 SW와 SP의 크기를 다양하게 변경하여 실험하였다. 전반적으로 SW와 SP의 크기가 작을 수록 잡음 제거 수행 시간이 빠르고 SW와 SP의 크기가 클수록 잡음 제거 성능이 높은 것을 알 수 있다. 이러한 결과를 통해 PNLM의 최적의 성능을 내기 위하여 이미지의 특성에 따라 적응적으로 SW와 SP의 크기를 설정하는 연구를 추후에 진행할 예정이다.

ACKNOWLEDGEMENT

This research was supported by the MISIP(Ministry of Science and ICT), Korea, under the National Program for Excellence in SW supervised by the IITP(Institute of Information & communications Technology Planning & Evaluation) (2019-0-01219).

REFERENCES

- [1] Jihun Jung, Subin Choi, and Seong-Eun Kim, "A Study on the detection of photo of urine test strip using white balancing algorithm," Proceedings of Symposium of the Korean Institute of communications and Information Sciences, pp. 1335-1336, 2022.
- [2] Ji-Hyeon Baek, and Nam-Ho Kim, "Noise Removal Method using Entropy in High-Density Noise Environments," Journal of the Korea Institute of Information and Communication Engineering, Vol. 24, No. 10, pp. 1255-1261, 2020.
- [3] Sung Jun Kim, and Yong Ju Jung, "Deep Network for Detail Enhancement in Image Denoising," Journal of Korea Multimedia Society, Vol. 22, No. 6, pp. 646-654, 2019.
- [4] MoonWon Choo, "Image Contrast Enhancement Based on Genetic Algorithm Using A Variant Histogram Equalization," The Journal of Image and Cultural Contents, Vol. 14, pp. 219-230, 2018. DOI: <https://doi.org/10.24174/jicc.2018.06.14.219>
- [5] Eon Seung Seong, Ji Hye Heo, Seong Hyun Han, Dong Hoon Lim, "Medical image denoising using convolutional dual-decoder autoencoder," Journal of the Korean Data And Information Science Society, Vol. 33, No. 6, pp. 1065-1083, 2022. DOI: <http://dx.doi.org/10.7465/jkdi.2022.33.6.1065>
- [6] Young-Ro Kim, "Spatio-temporal Non-local Means Filter for Noise Reduction," Journal of the Institute of Electronics and Information Engineers, Vol. 59, No. 12, pp. 97-103, 2022. DOI: 10.5573/ieie.2022.59.12.97
- [7] A. Buades, B. Coll and J. -M. Morel, "A non-local algorithm for image denoising," 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), pp. 60-65, Vol. 2, San Diego, CA, USA, 2005. DOI: 10.1109/CVPR.2005.38.
- [8] D. Van De Ville and M. Kocher, "SURE-Based Non-Local Means," IEEE Signal Processing Letters, Vol. 16, No. 11, pp. 973-976, Nov. 2009. DOI: 10.1109/LSP.2009.2027669.
- [9] J. Wang, Y. Guo, Y. Ying, Y. Liu and Q. Peng, "Fast Non-Local Algorithm for Image Denoising," 2006 International Conference on Image Processing, pp. 1429-1432, Atlanta, GA, USA, 2006. DOI: 10.1109/ICIP.2006.312698.
- [10] Y. Wu, B. Tracey, P. Natarajan and J. P. Noonan, "Probabilistic Non-Local Means," in IEEE Signal Processing Letters, Vol. 20, No. 8, pp. 763-766, Aug. 2013. DOI: 10.1109/LSP.2013.2263135.
- [11] J. Orchard, M. Ebrahimi and A. Wong, "Efficient nonlocal-means denoising using the SVD," 2008 15th IEEE International Conference on Image Processing, pp. 1732-1735, San Diego, CA, USA, 2008. DOI: 10.1109/ICIP.2008.4712109.
- [12] Antoni Buades, Bartomeu Coll, and Jean-Michel Morel, "Non-Local Means Denoising," Image Processing On Line, Vol. 1 pp. 208-212, 2011. DOI: https://doi.org/10.5201/ipol.2011.bcm_nlm
- [13] Jeonghwan Kim, Minjeong Lee, and Jechang Jeong, "A Study to Calculate an Efficient Covariance Matrix of Non-local Means with Principal Components Analysis," Proceedings of the Korean Society of Broadcast Engineers Conference, pp. 205-207, 2015.
- [14] Yang-Soo Kim, Jin-Woo Jeong, Yoon-Sik Choe, "Enhanced RGB Video Coding Based on Correlation in the Adjacent Block," The transactions of The Korean Institute of Electrical Engineers, Vol. 58, No. 12, pp. 2538-2541, 2009.
- [15] R. Verma and R. Pandey, "Non local means algorithm with adaptive isotropic search window size for image denoising," 2015 Annual IEEE India Conference (INDICON), pp. 1-5, New Delhi, India, 2015. DOI: 10.1109/INDICON.2015.7443193.
- [16] Praveen Venkatesh, Non-Local-Means, <https://github.com/praveenVnktsh/Non-Local-Means>
- [17] OpenCV, opencv, <https://github.com/opencv/opencv>
- [18] Metehan Çetinkaya, Python-Median-Filter, <https://github.com/MeteHanC/Python-Median-Filter/tree/master>

Authors



Jeongwoo Lee entered the Division of Digital Healthcare at Yonsei University, Wonju, Korea, in 2021. She is currently a student in the Division of Digital Healthcare, Yonsei University.

She is interested in wearable sensor, artificial intelligence, and image processing.



Heeyeon Jo entered the Division of Digital Healthcare at Yonsei University, Wonju, Korea, in 2021. She is currently a student in the Division of Digital Healthcare, Yonsei University.

She is interested in data analysis, data science and image processing.



Jiyun Byun entered the Division of Software at Yonsei University, Wonju, Korea, in 2023. She is currently a student in the Division of Software, Yonsei University. She is interested in image processing, forensic science, and

digital forensics.



Hongrae Lee received the B.S., M.S. and Ph.D. degrees in Computer Science from Yonsei University, Korea, in 2010, 2012 and 2019, respectively. Dr. Lee joined the faculty of the Division of Computer and

Telecommunications Engineering at Yonsei University, Wonju, Korea, in 2020. He is currently an Assistant Professor in the Software Division, Yonsei University. He is interested in image processing, image communication technology and computer vision system.