

영상에서 패치기반 CNN 모델을 이용한 잡음제거

2021210088 허지혜

1. 서론(Introduction)

<잡음>

- 과학기술의 발전에도 불구하고 영상의 전송 및 저장 과정에서 잡음이 첨가되어 영상의 질을 저하시킨다.
 - 위 같이 첨가된 잡음은 고주파성분에 해당되어 제거가 쉽지 않다.
-
- 영상에 존재하는 대표적인 잡음 : **가우시안 잡음, 임펄스 잡음**
 - 이러한 잡음을 제거하기 위한 선행 연구로 필터들이 존재한다.
 - 가우시안 잡음 제거 -> 평균과 관련된 필터
 - 임펄스 잡음 제거 -> 중앙값과 관련된 필터
-
- 선행연구 필터의 단점은 잡음이 덜 섞인 픽셀에 대하여도 균일하게 적용되어 중요한 영상 정보를 동시에 열화시킨다는 것이다.

1. 서론(Introduction)

<딥러닝 개념>

- 딥러닝은 컴퓨터 비전, 자연어 처리 등 많은 분야로 나뉘어 발전하고 있는데 그 중 CNN(Convolution Neural Network) 모델을 이용하면 영상인식, 물체인식, 얼굴인식 등 시각과 관련한 처리를 위한 역할을 하고 있다.

1. 서론(Introduction)

<본문에서 하고자 하는 내용>

- 논문에서는 영상의 잡음 특성과 분포에 상관없이 사용가능한 잡음제거 방법을 제안한다.
- 패치 상에서 **CNN** 모형 적용
- 복원 영상은 중첩된 패치에서 해당 픽셀들의 가중 평균에 의해 얻어진다.
- 잡음 데이터에 적응력이 좋은 **Adam** 알고리즘을 이용한다.
- 이를 평균값 필터, 중앙값 필터와 함께 비교한다.
- 판단척도로는 **MAE**(Mean A Error)와 **PSNR**(Peak Single to Noise Ratio)가 있다. (정량적, 정성적 비교가 동시에 수행된다.)

2. 관련 연구

<가우시안 잡음>

$$\mathbf{x} \in \mathbb{R}^{m \times n} \text{ 와 } \mathbf{y} \in \mathbb{R}^{m \times n}$$

- \mathbf{x} 를 원영상, \mathbf{y} 를 잡음영상이라고 할 때 가우시안 잡음 모형은 다음과 같이 나타낼 수 있다.

$$\mathbf{y} = \mathbf{x} + \sigma \times \mathbf{z},$$

2. 관련 연구

<임펄스 잡음>

- 잡음 비율이 p 인 임펄스 잡음 모형은 다음과 같이 나타낸다.
- 여기서 값은 0 또는 255인 잡음이다.

$$\mathbf{y} = \begin{cases} \mathbf{x} : \text{with probability } 1 - p \\ \boldsymbol{\eta} : \text{with probability } p, \end{cases}$$

2. 관련 연구 - CNN(Convolution Neural Network)

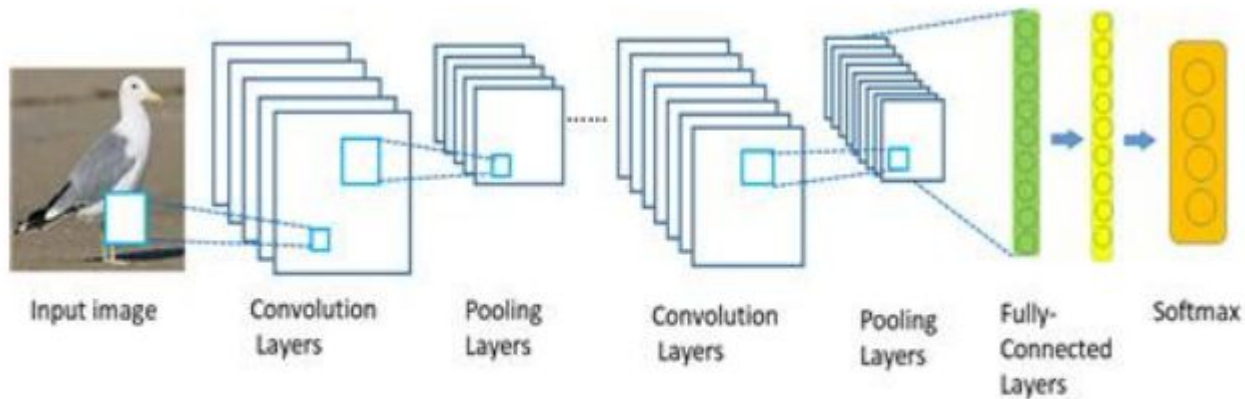


Figure 2.1 CNN architecture for bird recognition

input image -> convolution -> pooling -> fully-connected -> classification

2. 관련 연구 - Adam(adaptive moment estimation)

$$\Theta_{t+1} = \Theta_t - \alpha \frac{1}{\sqrt{h_t/(1-\nu^t) + \epsilon}} \left(\frac{m_t}{1-\mu^t} \right),$$

$$h_t = \nu h_{t-1} + (1-\nu) \left(\frac{\partial L(\Theta)}{\partial \Theta_t} \right)^2,$$

$$m_t = \mu m_{t-1} + (1-\mu) \frac{\partial L(\Theta)}{\partial \Theta_t},$$

[Adagrad]

개별 매개변수에 대해 드문 매개변수에는 학습률을 크게하고 빈번한 매개변수에는 학습률을 작게하는 등 학습률을 조정하면서 학습을 진행하는 방법.

[RMSProp]

감소평균을 이용하여 가중치를 주는 방법.

- **Adam** 알고리즘은 **Adagrad + RMSProp** 인 최적화 방법이다.
- Adagrad와 유사하게 학습이 진행되며, 계산된 기울기의 지수 평균을 저장하며 RMSProp와 유사하게 기울기의 제곱 값의 지수 평균을 저장한다.
- μ 는 이전의 매개변수 갱신값의 영향력을 감소시킨다.

2. 관련 연구 - 필터(평균값 필터, 중앙값 필터)

[평균값 필터]

픽셀값의 평균값을 선택하여 잡음을 제거한다. 가우시안 잡음 혹은 균일 잡음을 제거하는데 유리하다.

윈도우를 이용하여 전체 영상에 적용하여 잡음을 제거한다.

단, 에지도 잡음으로 취급하여 영상이 흐릿해질 수 있다.

[중앙값 필터]

에지는 그대로 두고 잡음만 제거하는 특성을 보인다.

하지만, 평균값 필터에 비해 동작시간이 증가한다.

2. 관련 연구 - 판단척도(MAE, PSNR)

평균 절대 오차

$$MAE = \frac{1}{n} \sum_{i=1}^n |x_i - x|$$

최대 신호 대 잡음비

$$\begin{aligned} PSNR &= 10 \log_{10} \left(\frac{MAX_I^2}{MSE} \right) \\ &= 20 \log_{10} \left(\frac{MAX_I}{\sqrt{MSE}} \right) \\ &= 20 \log_{10}(MAX_I) - 10 \log_{10}(MSE) \end{aligned}$$

3. CNN 모델을 이용한 잡음 제거

<기존 CNN 모델>

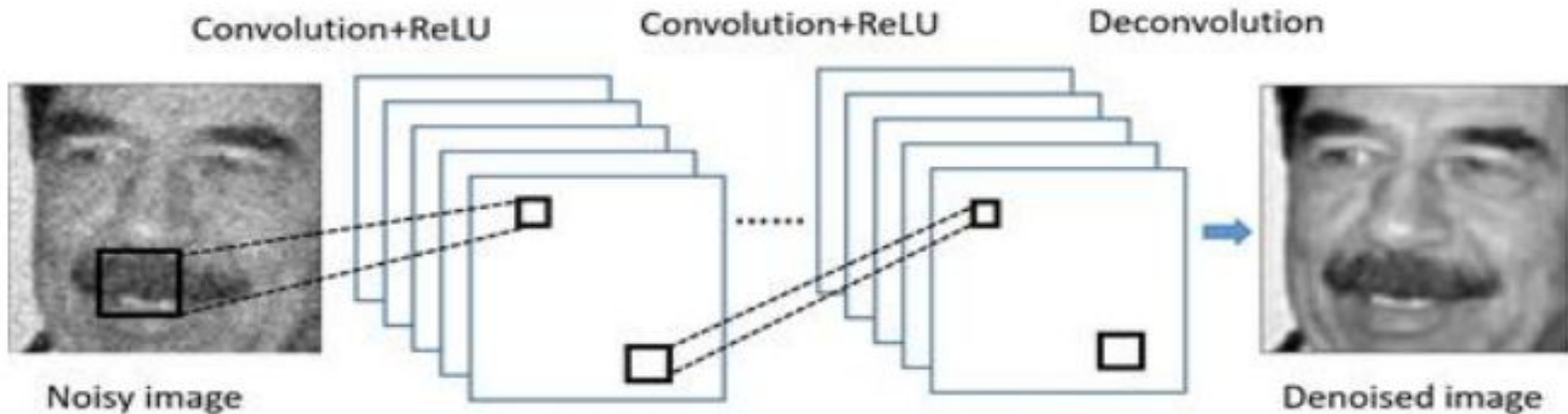


Figure 3.1 CNN architecture for denoising

- 앞선 CNN 모델과의 다른 점은 Fully-connected layer를 사용하지 않는다. 잡음 추정에는 유용하지 않기 때문이다.

3. CNN 모델을 이용한 잡음 제거

<기존 CNN 모델>

특징 추출(Feature Extraction)

$$F_1(\mathbf{y}) = \text{ReLU}(\mathbf{W}_1 * \mathbf{y} + \mathbf{b}_1),$$

\mathbf{w}, \mathbf{b} : 가중치, 바이어스
* : 컨볼루션 연산
ReLU : 활성화 함수
 \mathbf{W} : $s_1 \times s_1$ 개의 feature map

비선형 매핑(Non-linear mapping)

$$F_2(\mathbf{y}) = \text{ReLU}(\mathbf{W}_2 * F_1(\mathbf{y}) + \mathbf{b}_2),$$

재합성(Reconstruction)

$$F(\mathbf{y}) = \mathbf{W}_3 * F_2(\mathbf{y}) + \mathbf{b}_3,$$

- 매개변수는 \mathbf{w} 와 \mathbf{b} 는 \mathbf{F} 를 학습하므로 얻어진다.
- 따라서 추정값은 $(\text{원영상}(\mathbf{x}) - \mathbf{F}(\text{잡음영상}(\mathbf{y})))^2$ 값으로 사용한다. (mse)

3. CNN 모델을 이용한 잡음 제거

<패치기반 CNN 모델>

<차이점>

- 패치 추출
- 패치 aggregation
(가중평균 추가)

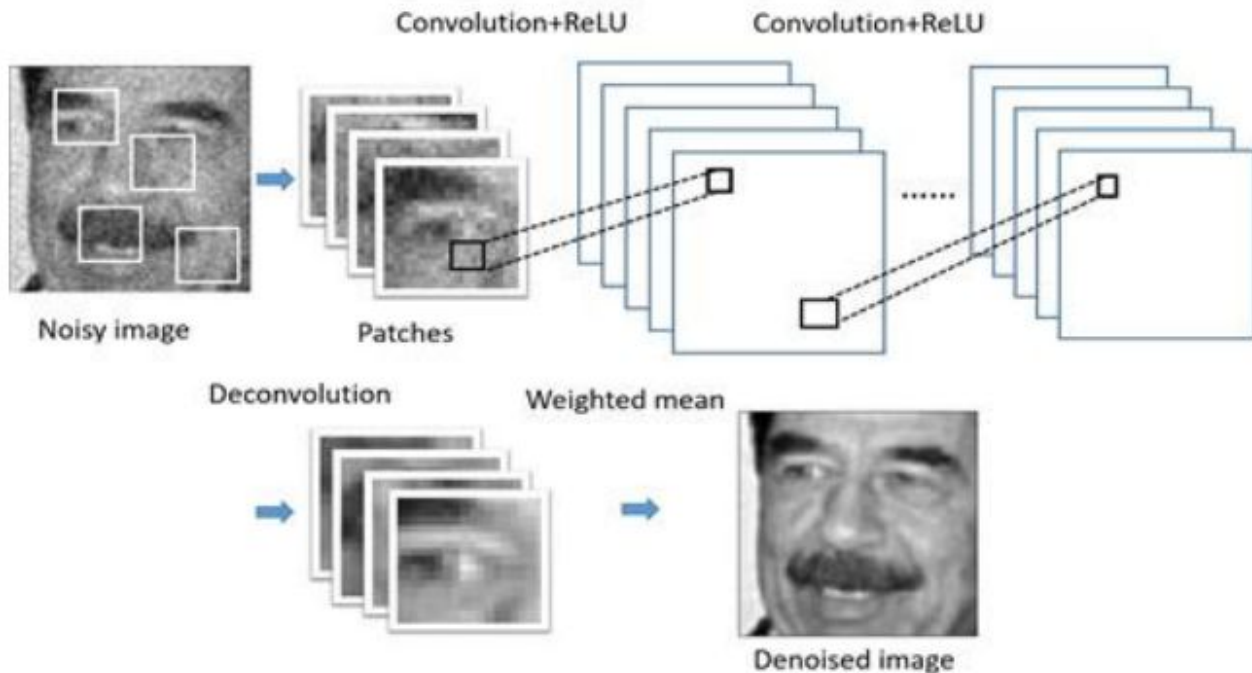


Figure 3.2 Patched-based CNN architecture for denoising

3. CNN 모델을 이용한 잡음 제거

<패치기반 CNN 모델>

Algorithm 1. Patched-based CNN denoising

Input: Training and test images with H and W , where H and W are the height and width of the image size, respectively.

Input: Patch size $p \times p$ and stride s , where $p = 20$, $s = 4$

Output: Denoised images

패치 크기 20 X 20, stride 4

패치 수행 후
이동되는 픽셀의
수

표준화

1. #Data pre-processing
2. $\mathbf{y}' = \frac{\mathbf{y}}{255}$
3. #Parameter setting of neural network with four layers
4. filter numbers, $n_1 = 64$, $n_2 = 64$, $n_3 = 128$, $n_4 = 1$ with corresponding filter sizes, $s = 3$
5. for $ih = 1$ to $(H - p)/s + 1$
6. for $iw = 1$ to $(W - p)/s + 1$
7. Extracting image patch $\mathbf{I}(ih, iw, ih + s, iw + s)$
8. #Building the model
9. $F_1(\mathbf{y}) = ReLU(\mathbf{W}_1 * \mathbf{y} + \mathbf{b}_1)$
10. $F_2(\mathbf{y}) = ReLU(\mathbf{W}_2 * F_1(\mathbf{y}) + \mathbf{b}_2)$
11. $F_3(\mathbf{y}) = ReLU(\mathbf{W}_3 * F_2(\mathbf{y}) + \mathbf{b}_3)$
12. $F(\mathbf{y}) = \mathbf{W}_4 * F_3(\mathbf{y}) + \mathbf{b}_4$

특징 추출 layer를 한 번 더 추가

3. CNN 모델을 이용한 잡음 제거

<패치기반 CNN 모델>

```
13.     #Compiling the model
14.     model.compile(optimizer=Adam, loss='mse')
15.     #Training the model
16.      $L(\Theta) = \frac{1}{N} \sum_{i=1}^N ||\mathbf{x}_i - F(\mathbf{y}_i)||^2$ 
17.     model.fit(noisy_image_train, original_image_train, batch_size=32, epochs=20)
18.     # Making predictions on noisy patches
19.     model.fit(noisy_image_test)
20. end
21. end
22.     # Aggregation of the restored patches
23. for ih = 1 to (H - p)/s + 1
24.   for iw = 1 to (W - p)/s + 1
25.     Compute the mean of overlapping denoised patches  $\mathbf{I}(ih, iw, ih + s, iw + s)$ 
    around each pixel
26.   end
27. end
```

가중 평균 구해주기

4. 성능 실험 및 논의사항

<성능 실험>

비교하고자 하는 평균값 중앙값 필터는 모두 3×3 이다.

[패치기반 CNN]

INPUT $\rightarrow (3,3,64) \rightarrow (3,3,64) \rightarrow (3,3,128) \rightarrow (3,3,1)$

[기존 CNN]

INPUT $\rightarrow (9,9,64) \rightarrow (3,3,32) \rightarrow (5,5,1)$

[데이터]

Daniel Hromada SMILEsmileD 얼굴 영상 TRAIN(2952) + TEST(738) = ALL(3690)

4. 성능 실험 및 논의사항

<가우시안 잡음>



(a) Original image



(b) Gaussian noisy image
($\sigma = 0.10$)



(c) Gaussian noisy image
($\sigma = 0.20$)

Figure 4.1 Gaussian noisy image with an original image

4. 성능 실험 및 논의사항

<가우시안 잡음>

$$\sigma = 0.10$$



(b) Gaussian noisy image
($\sigma = 0.10$)



(a) Mean



(b) Median



(c) CNN



(d) Patch-based CNN

Figure 4.3 Experiment results for Figure 4.1 (b): (a) mean filter, (b) median filter, (c) CNN, (d) patch-based CNN

4. 성능 실험 및 논의사항

<가우시안 잡음>

$$\sigma = 0.10$$

Table 4.2 Comparison of patch-based CNN with other denoising methods for Figure 4.1 (b)

Methods	MAE	PSNR
Mean	0.0310	28.088
Median	0.0388	25.818
CNN	0.0331	27.238
Patch-based CNN	0.0244	30.086

4. 성능 실험 및 논의사항

<가우시안 잡음>

$$\sigma = 0.20$$



(c) Gaussian noisy image
($\sigma = 0.20$)



(a) Mean



(b) Median



(c) CNN



(d) Patch-based CNN

Figure 4.4 Experiment results for Figure 4.1 (c) : (a) mean filter, (b) median filter, (c) CNN, (d) patch-based CNN

4. 성능 실험 및 논의사항

<가우시안 잡음>

$$\sigma = 0.20$$

Table 4.3 Comparison of patch-based CNN with other denoising methods for Figure 4.1 (c)

Methods	MAE	PSNR
Mean	0.0564	22.983
Median	0.0716	20.731
CNN	0.0467	24.463
Patch-based CNN	0.0370	26.475

4. 성능 실험 및 논의사항

<임펄스 잡음>



(a) Original image



(b) Impulse noisy image($p = 0.05$)



(c) Impulse noisy image($p = 0.10$)

Figure 4.2 Impulse noisy images with an original image

4. 성능 실험 및 논의사항

<임펄스 잡음>

$p = 0.05$



(b) Impulse noisy image($p = 0.05$)



(a) Mean



(b) Median



(c) CNN



(d) Patch-based CNN

Figure 4.5 Experiment results for Figure 4.2 (b) : (a) mean Filter, (b) median filter, (c) CNN, (d) patch-based CNN

4. 성능 실험 및 논의사항

<임펄스 잡음>

p = 0.05

Table 4.4 Comparison of patch-based CNN with other denoising methods for Figure 4.2 (b)

Methods	MAE	PSNR
Mean	0.0306	26.687
Median	0.0098	32.178
CNN	0.0377	26.111
Patch-based CNN	0.0030	46.415

4. 성능 실험 및 논의사항

<임펄스 잡음>

$p = 0.10$



(c) Impulse noisy image($p = 0.10$)



(a) Mean



(b) Median



(c) CNN



(d) Patch-based CNN

Figure 4.6 Experiment results for Figure 4.2 (c) : (a) mean filter, (b) median filter, (c) CNN, (d) patch-based CNN

4. 성능 실험 및 논의사항

<임펄스 잡음>

p = 0.10

Table 4.5 Comparison of patch-based CNN with other denoising methods for Figure 4.2 (c)

Methods	MAE	PSNR
Mean	0.0444	24.086
Median	0.0117	30.299
CNN	0.0440	24.796
Patch-based CNN	0.0043	43.462

5. 결론 및 향후 연구

- 본 논문에서는 사전에 잡음의 특성과 분포를 고려할 필요 없이 사용 가능한 방법으로 **패치기반 CNN 모형**을 이용하여 잡음을 제거하였다.
- 실험결과 패치기반 **CNN** 모형은 모든 잡음영상에서 원영상과 비슷한 영상을 얻을 수 있었다.
- 특히, 임펄스 잡음에서 **MAE** 비교를 통해 패치기반 **CNN**은 기존 **CNN**과 필터보다 **10배**이상 낮은 수치를 얻었다.
- 하지만, 결과와 다르게 패치를 사용하지 않은 **CNN** 모형에 비해 수행시간의 문제를 가지고 있다. 앞으로의 연구에서는 수행시간 개선을 해보야겠다.

[1] 영상에서 패치기반 CNN 모델을 이용한 잡음제거, 허광해, 임동훈,
한국데이터정보과학회지, 30(2), 2019

[2] 평균값 필터, 중앙값 필터

[https://m.blog.naver.com/PostView.naver?isHttpsRedirect=true&blogId=msnayana
&logNo=80111636173](https://m.blog.naver.com/PostView.naver?isHttpsRedirect=true&blogId=msnayana&logNo=80111636173)