



---

# Quantify Image Quality Using Neural Image Assessment

# 신경 이미지 평가를 활용한 이미지 품질 정량화

---

오영우

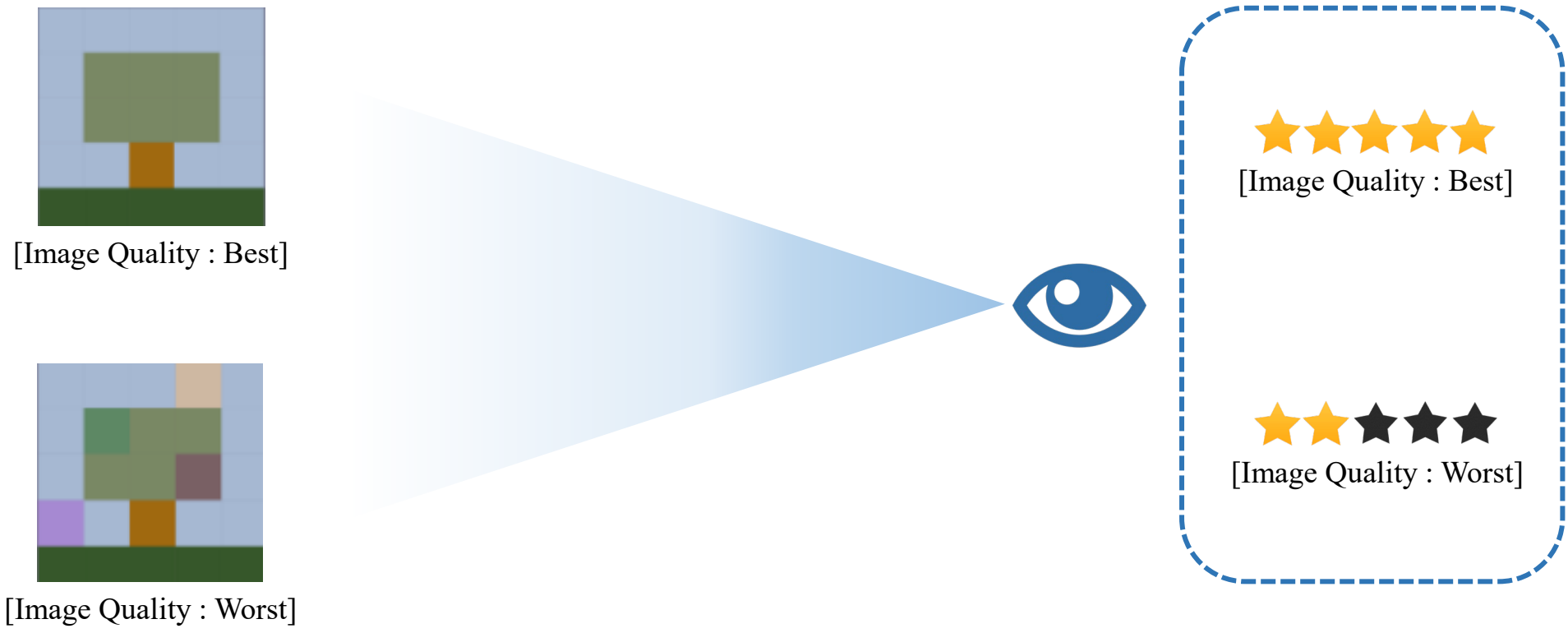
조선대학교 컴퓨터공학과 석사과정

# CONTENTS

- 01 프로젝트 목표**  
Goal of project
- 02 데이터 로드 및 평가**  
Data loading and evaluation
- 03 데이터 전처리 과정**  
Preprocess augment data
- 04 다음 발표 내용**  
Next presentation

# 1-1 프로젝트 목표 이미지 품질 평가

01 인간은 '시각'을 통해, 사물, 이미지, 영상 등에 대한 객체의 품질 평가를 수행할 수 있으나, 이는 주관적인 평가에 그치는 한계



<https://towardsdatascience.com/deep-image-quality-assessment-30ad71641fac>



- 인간은 시각을 통해, 동일한 이미지, 영상, 물체 등에 대한 품질 평가를 수행할 수 있음
- 그러나 시각을 통해 평가되는 이미지 품질은 다소 주관적인 경향을 가질 수 있음



# 1-2 프로젝트 목표 이미지 품질 평가 기법

- 01 객관적인 품질 평가를 위해 다양한 기법들이 활용되고 있으며, 가장 대표적인 PSNR (Peak-Signal-to-Noise-Ratio) 기법이 존재
- 02 MSE (Mean Square, Error)를 활용하여, 영상 화질에 대한 손실 정보를 평가하는 방식
- 03 이미지 압축률이 클수록 PSNR 값이 감소함과 동시에 이미지 왜곡 현상이 증가됨을 확인



[Original image]



[PSNR, 34.8227 dB]



[PSNR, 30.9394 dB]



[PSNR, 25.8699 dB]

<https://bskyvision.com/entry/IQA-%EC%B5%9C%EB%8C%80%EC%8B%A0%ED%98%B8%EB%8C%80%EC%9E%A1%EC%9D%8C%EB%B9%84PSNR%EC%99%80-%EC%9D%B4%EB%AF%B8%EC%A7%80-%ED%92%88%EC%A7%88>



- 이미지 품질 평가를 위해 사용되는 가장 대표적인 메트릭은 PSNR 기법은 이미지에 대한 객관적인 품질 평가를 가능케 함
- 그러나, 해당 기법은 인간 느끼는 주관적인 평가와 상반되는 경우가 빈번히 발생





# 1-3 프로젝트 목표

## PSNR 기법의 문제점

01 객관적 평가와 주관적 평가가 상반되는 결론이 도출되므로, 보다 **정량적인 평가 기법이 요구**



[PSNR, 25.27 dB]



[PSNR, 25.17 dB]



<https://towardsdatascience.com/deep-image-quality-assessment-30ad71641fac>



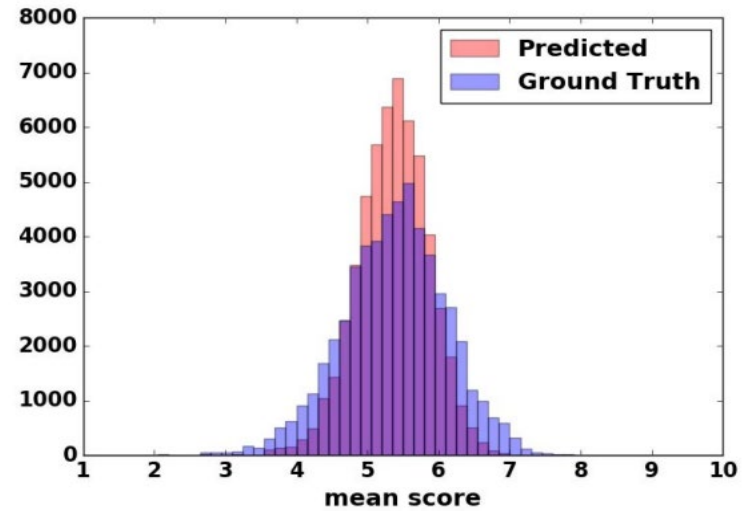
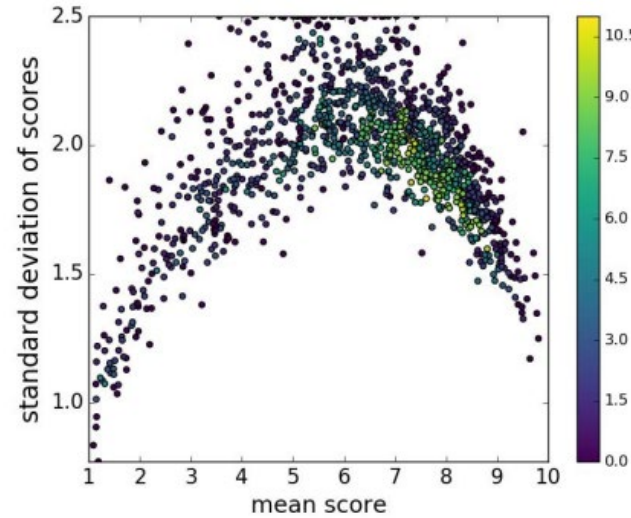
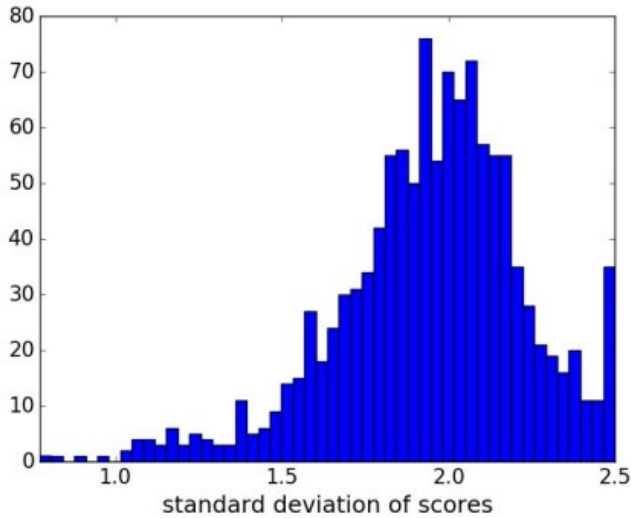
- PSNR 기법을 통해 평가된 경우, 좌측의 이미지가 높은 품질을 지녔으나, 인간이 느끼기에 화질이 크게 저하되었다고 느낌
- 우측 이미지의 경우, 좌측에 비해 높은 화질로 느껴지나, 실제 PSNR 기법으로 평가하는 경우 더 낮은 품질을 기록



# 1-4 프로젝트 목표

## NIMA & CNN 기반 품질 평가 기법

- 01 PSNR과 같은 기존 평가 기법은 인간의 주관적인 평가와의 상관성이 부족하며, 낮은 정확도
- 02 딥러닝 기반의 품질 예측 기법은 높은 정확도를 보일 수 있으나, 저/고품질과 같은 이진 분류에 의해 주관적 평가와의 상관성이 부족
- 03 NIMA-CNN 기법은 **정량적인 품질 평가를 위해, 주어진 이미지에 대해 1~10점의 척도로 평점 분포를 활용하며, 분포를 예측하는 것이 목표**  
 \* Each example in the dataset consists of an image and its ground truth (user) ratings.



[1] Talebi, Hossein, and Peyman Milanfar. "NIMA: Neural Image Assessment." IEEE Transactions on Image Processing 27, no. 8 (August 2018): 3998–4011. <https://doi.org/10.1109/TIP.2018.2831899>.  
<https://developers-kkr.googleblog.com/2018/01/introducing-nima-neural-image-assessment.html>



- PSNR, 딥러닝 (CNN) 기반의 품질 평가 방안의 경우, 인간이 체감하는 주관적 평가와 상관성이 부족
- NIMA-CNN 기법은 1~10점 사이의 인간의 주관적 평가로 구성된 평점 분포를 활용하여, 궁극적으로 분포를 예측하는 것을 목표로 함

# 2-1 데이터 로드 및 평가

## 훈련된 NIMA-CNN 기반 품질 평가 1



- NIMA 모델에 대한 평가를 위해 (3)과 같이 블러 처리를 통해, 정상적인 이미지의 품질 저하를 수행
- 사전 학습된 NIMA 모델을 이용하여, 정상 이미지와 블러 처리된 이미지에 대한 평가 결과를 진행

# 2-2 데이터 로드 및 평가

## 훈련된 NIMA-CNN 기반 품질 평가 2

04 Predict the NIMA quality score

```
[meanOriginal, stdOriginal] = predictNIMAScore(dlnet, imOriginal);  
[meanBlur, stdBlur] = predictNIMAScore(dlnet, imBlur);
```

05 Display the images and scores

```
figure  
t = tiledlayout(1,2);  
displayImageAndScoresForNIMA(t, imOriginal, meanOriginal, stdOriginal, "Original Image")  
displayImageAndScoresForNIMA(t, imBlur, meanBlur, stdBlur, "Blurred Image")
```

[Original image]

- Mean score : 7.7314
- Std Dev : 1.5516



[Blurred image]

- Mean score : 6.2639
- Std Dev : 1.7147



- predictNIMAScore 함수를 통해 측정된 평균 점수는 이미지의 품질을 나타내며, 점수의 표준 편차는 예측된 평균 점수의 신뢰 수준을 의미
- 학습된 NIMA-CNN을 통해 예측된 결과는 주관적인 시각적 평가와 일치하는 이미지 품질을 예측할 수 있음을 보임

# 2-3 데이터 로드 및 평가

## 데이터셋 로드 1

- 01 챌린지 데이터베이스인 'LIVE in the Wild Image Quality Challenge Database'를 활용한 데이터 셋 설치
- 02 1162장의 이미지와 각 이미지는 [1, 100] 기준에 대해 175명에 의해 주관적 점수를 포함
- 03 LIVE 데이터셋은 각 이미지에 대한 주관적 점수의 평균 및 표준 편차를 제공

04 Load LIVE in the Wild data (images)

```
imageData = load(fullfile(dataDir, "Data", "AllImages_release.mat"));
imageData = imageData.AllImages_release;
nImg = length(imageData);
imageList(1:7) = fullfile(dataDir, "Images", "trainingImages", imageData(1:7));
imageList(8:nImg) = fullfile(dataDir, "Images", imageData(8:end));
```

05 Load LIVE in the Wild data (scores)

```
meanData = load(fullfile(dataDir, "Data", "AllIMOS_release.mat"));
meanData = meanData.AllIMOS_release;
stdData = load(fullfile(dataDir, "Data", "AllStdDev_release.mat"));
stdData = stdData.AllStdDev_release;
```

<https://live.ece.utexas.edu/research/ChallengeDB/index.html>



- 본 예제에서는 LIVE in the Wild Image Quality Challenge Database를 활용
- 사용되는 학습 데이터셋은 각각의 이미지 데이터와 인간이 직접 이미지의 품질을 확인하여 평가한 주관적 점수로 구성

# 2-4 데이터 로드 및 평가

## 데이터셋 로드 2

06

Display a few sample images

```
figure
t = tiledlayout(1,3);
idx1 = 785;
displayImageAndScoresForNIMA(t,readimage(imds,idx1), ...
    meanData(idx1),stdData(idx1),"Image "+imageData(idx1))
idx2 = 203;
displayImageAndScoresForNIMA(t,readimage(imds,idx2), ...
    meanData(idx2),stdData(idx2),"Image "+imageData(idx2))
idx3 = 777;
displayImageAndScoresForNIMA(t,readimage(imds,idx3), ...
    meanData(idx3),stdData(idx3),"Image "+imageData(idx3))
```

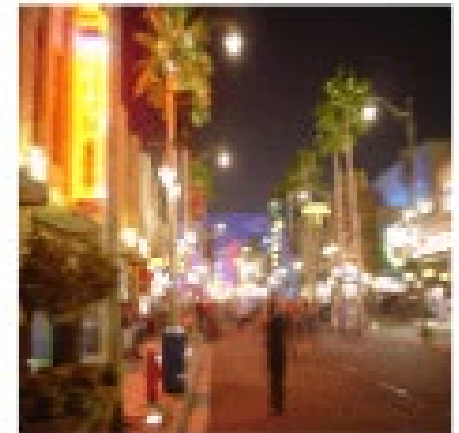
Mean score : 92.4  
Std Dev : 12.0



Mean score : 28.1  
Std Dev : 16.6



Mean score : 12.7  
Std Dev : 13.7





# 3-1 데이터 전처리 과정 이미지와 주관적 점수 결합

01 Images resizing to 256 by 256

```
rescaleSize = [256 256];  
imds = transform(imds,@(x)imresize(x,rescaleSize));
```

02 Rescales the scores to [1, 10]

```
newMaxScore = 10;  
prob = createNIMAScoreDistribution(meanData,stdData);  
cumProb = cumsum(prob,2);
```

03 Create and combine the datastores

```
probDS = arrayDatastore(cumProb',IterationDimension=2);  
dsCombined = combine(imds,probDS);  
sampleRead = preview(dsCombined)
```

sampleRead = 1x2 cell

|   | 1               | 2  |
|---|-----------------|--|
| 1 | 256x256x3 uint8 | [0.0028;0.0134;0.0452;0.1187;0.2517;0.4391;0.6450;0.8212;0.9388;1] |



- 데이터셋에 존재하는 이미지의 크기를 256 X 256으로 모두 처리하고, [1, 100] 사이의 주관적 품질 점수를 [1, 10] 사이로 스케일링을 취함
- 처리된 이미지와 주관적 품질 점수를 결합하여 NIMA-CNN 학습을 위한 학습 데이터로 활용

# 3-2 데이터 전처리 과정

## 데이터 분할 작업

04

Split dataset for training

```
numTrain = floor(0.70 * nImg);  
numVal = floor(0.15 * nImg);  
  
Idx = randperm(nImg);  
idxTrain = Idx(1:numTrain);  
idxVal = Idx(numTrain+1:numTrain+numVal);  
idxTest = Idx(numTrain+numVal+1:nImg);  
  
dsTrain = subset(dsCombined,idxTrain);  
dsVal = subset(dsCombined,idxVal);  
dsTest = subset(dsCombined,idxTest);
```

05

Augment the training data

```
inputSize = [224 224];  
dsTrain = transform(dsTrain,@(x)augmentDataForNIMA(x,inputSize));
```

[1] Talebi, Hossein, and Peyman Milanfar. "NIMA: Neural Image Assessment." IEEE Transactions on Image Processing 27, no. 8 (August 2018): 3998–4011. <https://doi.org/10.1109/TIP.2018.2831899>.



- 학습, 검증, 테스트 단계에 사용되는 데이터 분할 비율은 75 : 15: 15 비율로 구성됨
- 학습 성능 개선과 Overfitting 문제를 줄이기 위해 이미지 크기를 줄이고, 50%의 확률로 인위적인 이미지 회전을 취함



# 4-1 다음 발표 내용 학습 및 성능 분석

- 01 예제에서 사용한 MobileNet의 경우, VGG16, Inception-v2 대비 성능이 저조한 것을 확인할 수 있음
- 02 따라서, 기존 예제에서 제공하는 네트워크와 다른 CNN 모델을 활용한 성능 평가 진행 예정

| <i>Model</i>                  | <i>LCC</i><br><i>(mean)</i> | <i>SRCC</i><br><i>(mean)</i> | <i>LCC</i><br><i>(std.dev)</i> | <i>SRCC</i><br><i>(std.dev)</i> | <i>EMD</i>    |
|-------------------------------|-----------------------------|------------------------------|--------------------------------|---------------------------------|---------------|
| Kim et al. [16]               | 0.80                        | 0.80                         | –                              | –                               | –             |
| Moorthy et al. [39]           | 0.89                        | 0.88                         | –                              | –                               | –             |
| Mittal et al. [40]            | 0.92                        | 0.89                         | –                              | –                               | –             |
| Saad et al. [41]              | 0.91                        | 0.88                         | –                              | –                               | –             |
| Kottayil et al. [42]          | 0.89                        | 0.88                         | –                              | –                               | –             |
| Xu et al. [35]                | 0.96                        | 0.95                         | –                              | –                               | –             |
| Bianco et al. [7]             | 0.96                        | 0.96                         | –                              | –                               | –             |
| $\bar{NIMA}(\bar{MobileNet})$ | $\bar{0.782}$               | $\bar{0.698}$                | $\bar{0.209}$                  | $\bar{0.181}$                   | $\bar{0.105}$ |
| NIMA(VGG16)                   | 0.941                       | 0.944                        | 0.538                          | 0.557                           | 0.054         |
| NIMA(Inception-v2)            | 0.827                       | 0.750                        | 0.470                          | 0.468                           | 0.064         |

[1] Talebi, Hossein, and Peyman Milanfar. "NIMA: Neural Image Assessment." IEEE Transactions on Image Processing 27, no. 8 (August 2018): 3998–4011. <https://doi.org/10.1109/TIP.2018.2831899>.



- [1] 논문의 경우, VGG16을 적용했을 때의 성능이 가장 높게 나오는 것을 확인할 수 있으며, 네트워크마다 성능 차가 발생
- 따라서, 예제에 제공되는 MobileNet 외 다양한 CNN 모델을 활용하여, 성능 분석 및 평가를 수행할 예정



PRESENTATION END

**THANK YOU**



---

# Quantify Image Quality Using Neural Image Assessment

# 신경 이미지 평가를 활용한 이미지 품질 정량화 <sup>v2</sup>

---

오영우

조선대학교 컴퓨터공학과 석사과정

# CONTENTS v2

- 01** 데이터 증진 및 정규화  
Data Augmentation and Normalization
- 02** 네트워크 로드 및 설정  
Network Loading and Settings
- 03** 주요 매트릭 분석  
Analysis of Key Metrics
- 04** 모델별 성능 분석  
Performance Analysis for Each Model



# 1-1 데이터 증진 및 정규화 Augmentation

01

Split dataset for training

```
numTrain = floor(0.70 * nImg);  
numVal = floor(0.15 * nImg);  
  
Idx = randperm(nImg);  
idxTrain = Idx(1:numTrain);  
idxVal = Idx(numTrain+1:numTrain+numVal);  
idxTest = Idx(numTrain+numVal+1:nImg);  
  
dsTrain = subset(dsCombined,idxTrain);  
dsVal = subset(dsCombined,idxVal);  
dsTest = subset(dsCombined,idxTest);
```

02

Augment the training data

```
inputSize = [224 224];  
dsTrain = transform(dsTrain,@(x)augmentDataForNIMA(x,inputSize));
```

[1] Talebi, Hossein, and Peyman Milanfar. "NIMA: Neural Image Assessment." IEEE Transactions on Image Processing 27, no. 8 (August 2018): 3998–4011. <https://doi.org/10.1109/TIP.2018.2831899>.



- 학습, 검증, 테스트 단계에 사용되는 데이터 분할 비율은 70 : 15: 15 비율로 구성됨
- 학습 성능 개선과 Overfitting 문제를 줄이기 위해 이미지 크기를 줄이고, 50%의 확률로 인위적인 이미지 회전을 취함

# 1-2 데이터 증진 및 정규화

## Z-점수 정규화

- 01 네트워크의 입력 계층은 **훈련 이미지의 Z-점수 정규화**를 수행
- 02 Z-점수 정규화를 수행하기 위해 학습 데이터 셋 이미지에 대한 평균 및 표준편차를 계산하는 과정을 수행  
' $(X - \text{Mean}) / \text{Standard Deviation}$ ' changes the value so that some data corresponds to the standard normal distribution.

03 Input normalization

```
meanImage = zeros([inputSize 3]);
meanImageSq = zeros([inputSize 3]);
while hasdata(dsTrain)
    dat = read(dsTrain);
    img = double(dat{1});
    meanImage = meanImage + img;
    meanImageSq = meanImageSq + img.^2;
end
meanImage = meanImage/numTrain;
meanImageSq = meanImageSq/numTrain;
varImage = meanImageSq - meanImage.^2;
stdImage = sqrt(varImage);
reset(dsTrain);
```

[1] Talebi, Hossein, and Peyman Milanfar. "NIMA: Neural Image Assessment." IEEE Transactions on Image Processing 27, no. 8 (August 2018): 3998–4011. <https://doi.org/10.1109/TIP.2018.2831899>.



- 본 예제에서 활용되는 데이터셋의 경우, 이미지와 주관적 평가를 모두 포함하므로 주관적 평가에 대한 평균 및 분포 등의 정규화 과정을 수행
- Z-점수 정규화 수행을 위해 Argumentation된 학습 데이터 셋 이미지에 대한 평균과 표준편차를 계산



# 1-3 데이터 증진 및 정규화 훈련 데이터 배치

04 해당 예제의 배치 데이터 추출 방식은 'SSCB (Spatial, Spatial, Channel, Batch)' 형식을 따름

05 Batch training data

```
miniBatchSize = 128;
mbqTrain = minibatchqueue(dsTrain,MiniBatchSize=miniBatchSize, ...
    PartialMiniBatch="discard",MiniBatchFormat=["SSCB",""], ...
    DispatchInBackground=canUseGPU);
mbqVal = minibatchqueue(dsVal,MiniBatchSize=miniBatchSize, ...
    MiniBatchFormat=["SSCB",""],DispatchInBackground=canUseGPU);
```

MATLAB >> deep learning toolbox >> help >> function >> minibatchqueue

```
mbq = minibatchqueue(augimdsTrain, ...
    MiniBatchSize=miniBatchSize, ...
    MiniBatchFcn=@preprocessMiniBatch, ...
    MiniBatchFormat=[ "SSCB" "" ]);
```

<https://kr.mathworks.com/help/deeplearning/ref/dlnetwork.forward.html>

[1] Talebi, Hossein, and Peyman Milanfar. "NIMA: Neural Image Assessment." IEEE Transactions on Image Processing 27, no. 8 (August 2018): 3998–4011. <https://doi.org/10.1109/TIP.2018.2831899>.



- 본 예제에서는 SSCB 형식의 배치 데이터 추출 방식을 활용
- Batch size의 경우, 128로 지정되었으나 다양한 64, 256에 대한 배치 사이즈 조정 후 학습 했을 때의 학습 성능 평가가 요구 됨



# 2-1 네트워크 로드 및 설정 학습된 모델 호출 및 변환

01 본 예제의 경우 ImageNet [2]에서 훈련된 MobileNet-v2 CNN을 활용

02 MobileNet-v2의 각 계층은 1~10까지의 점수를 나타내며, **각 이미지에 대한 점수 확률을 예측**



[2] ImageNet. <https://www.image-net.org>.



- 네트워크의 입력 이미지는 224 X 224 픽셀이며, 이미지의 평균과 표준 편차를 이용하여 이미지 데이터에 대한 Z-점수 정규화를 수행
- (5)번의 경우, CNN 계층의 학습률 및 Drop-out 정도를 지정할 수 있으며, 본 예제에서는 각각 10배, 75%를 설정



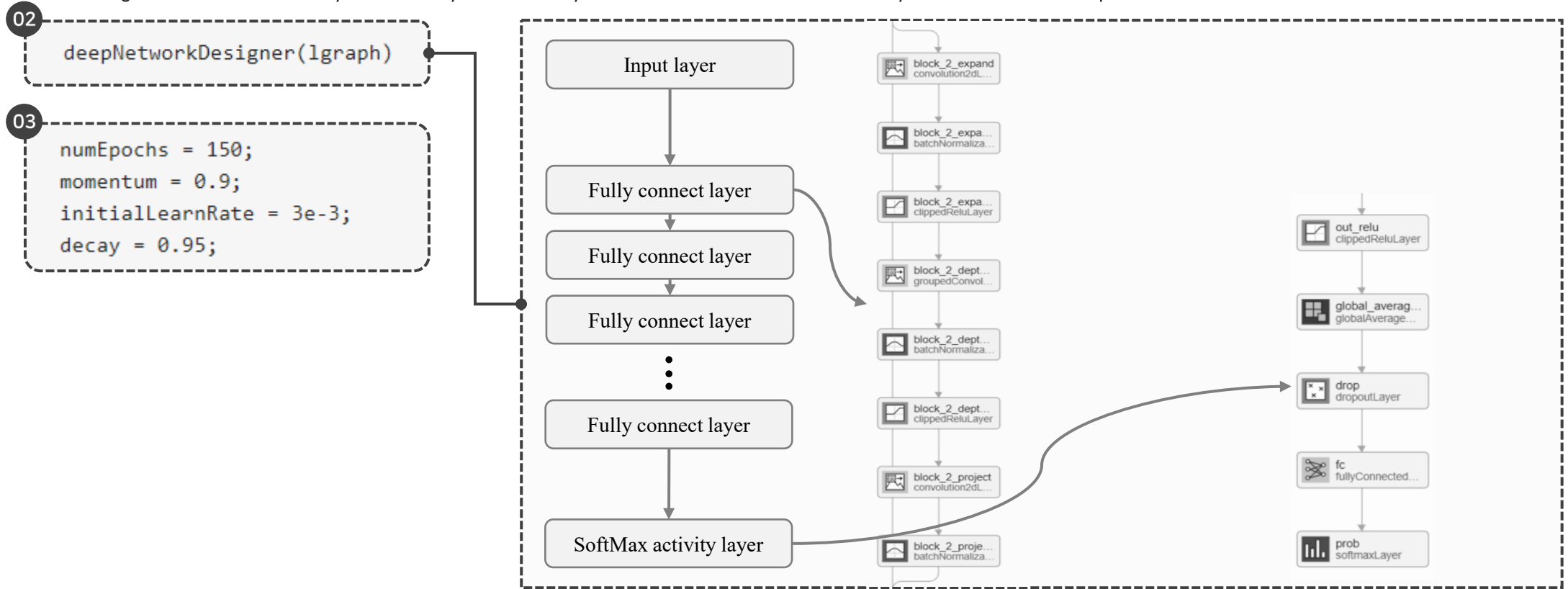


# 2-2 네트워크 로드 및 설정

## 네트워크 시각화 및 분석

01 사전 학습된 **MobileNet-v2 CNN** 및 **SGDM** (Stochastic Gradient Descent with Momentum) optimizer를 활용

The original final classification layer with a fully connected layer with 10 neurons. Add a SoftMax layer to normalize the outputs.



[2] ImageNet. <https://www.image-net.org>.



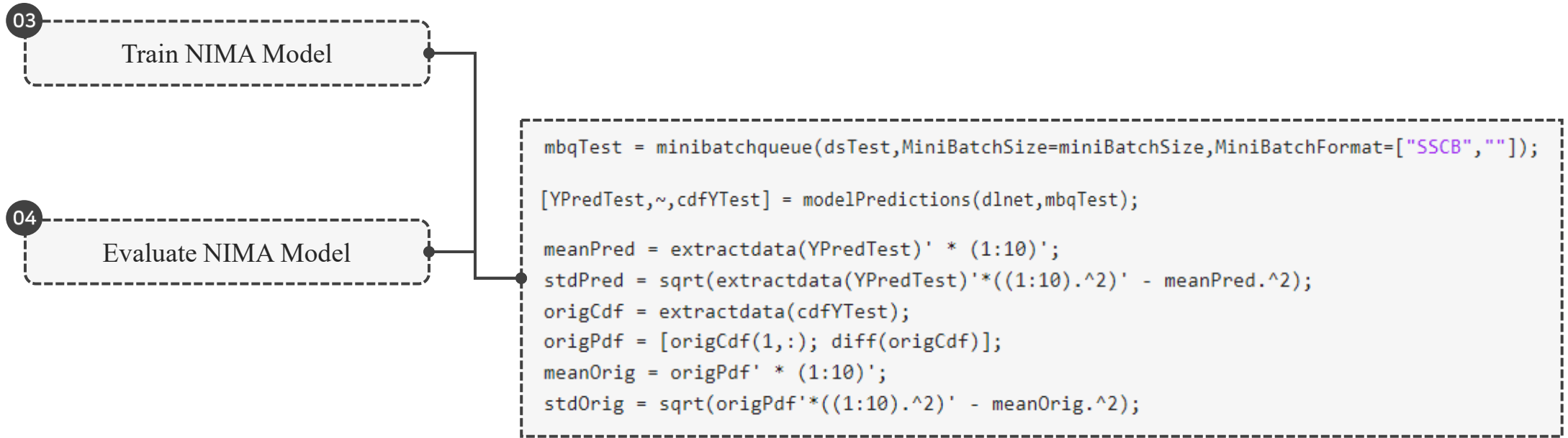
- 본 예제에 활용되는 'MobileNet-v2' 네트워크 모델은 완전 연결계층과 마지막 출력 계층에서의 SoftMax가 활용
- 각각의 연결 계층은 입력된 이미지에 대한 점수를 예측하는데 사용되며, 배치 정규화, Drop-out 등이 사용

# 2-3 네트워크 로드 및 설정 NIMA 학습 및 평가

01 NIMA 모델의 학습 과정은 일반적인 CNN 방식과 동일하게 수행되므로 생략

NIMA의 학습 과정은 일반적인 CNN과 동일하게 학습을 수행하며, 이미지의 특징을 학습 및 예측하는 것이 아닌, 각 이미지 별 점수에 대한 분포를 예측하는 것에 중점적

02 본 예제는 기존 CNN 기반의 품질 평가 기법에서 수행되는 방식이 아닌 3가지 주요 메트릭 (EMD, ACC, CC)을 통한 모델의 학습 성능을 판단



[1] Talebi, Hossein, and Peyman Milanfar. "NIMA: Neural Image Assessment." IEEE Transactions on Image Processing 27, no. 8 (August 2018): 3998–4011. <https://doi.org/10.1109/TIP.2018.2831899>.

[3] Sandler, Mark, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. "MobileNetV2: Inverted Residuals and Linear Bottlenecks." In 2018 IEEE/CVF, 4510–20. Salt Lake City, UT.

[5] Hou, Le, Chen-Ping Yu, and Dimitris Samaras. "Squared Earth Mover's Distance-Based Loss for Training Deep Neural Networks." Preprint, submitted November 30, 2016. <https://arxiv.org/abs/1611.05916>.



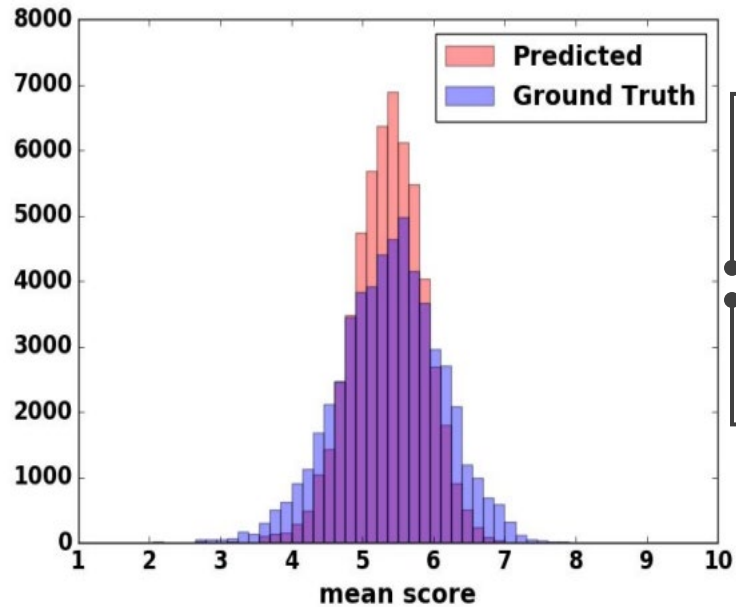
- NIMA 모델의 학습 과정은 일반적인 CNN 방식과 동일하게 처리되므로 설명을 생략
- 딥러닝 기반의 이미지 품질 평가 방식과 달리 3가지 주요 메트릭인 EMD, ACC, CC 등을 통해 모델의 학습 성능을 판단



# 3-1 주요 메트릭 분석

## EMD (Earth Mover's Distance)

- 01 EMD의 사전적 정의, ‘확률 분포 A를 확률 분포 B로 변환 할 때의 필요한 일의 총량을 의미’
- 02 즉, 인간이 각 이미지에 대해 정의한 주관적 평가들의 분포와 모델이 예측한 평가의 분포가 얼마나 최소화 되는가에 집중하는 성능 지표



03 본 논문은 딥러닝 기반의 이미지 품질 평가의 객관화와 주관적 평가 사이의 합리적인 정량적 품질 평가를 목적으로 함

04 해당 메트릭은 모델에서 예측하는 평가 (Predicted)와 실제 사람의 주관적 평가 (Ground Truth) 사이의 분포를 분석하기 위해 활용

[1] Talebi, Hossein, and Peyman Milanfar. "NIMA: Neural Image Assessment." IEEE Transactions on Image Processing 27, no. 8 (August 2018): 3998–4011. <https://doi.org/10.1109/TIP.2018.2831899>.

[3] Sandler, Mark, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. "MobileNetV2: Inverted Residuals and Linear Bottlenecks." In 2018 IEEE/CVF, 4510–20. Salt Lake City, UT.

[5] Hou, Le, Chen-Ping Yu, and Dimitris Samaras. "Squared Earth Mover's Distance-Based Loss for Training Deep Neural Networks." Preprint, submitted November 30, 2016. <https://arxiv.org/abs/1611.05916>.



- 본 논문의 주요 목적은 실제 사람이 평가하는 주관적 점수와 모델의 예측하는 객관적 점수 간 합리적인 평가를 가능케 하는 것
- 주관적 평가 (Ground Truth)와 모델 예측 점수 (Predicted) 간 EMD 값이 낮을수록 모델이 인간의 주관 평가와의 높은 상관성을 달성

# 3-2 주요 메트릭 분석

## Classification Acc.

01 본 예제는 딥러닝 기반 이미지 품질 평가 기법과 동일하게 Binary 분류 정확도 (고/저품질)를 성능 지표로 채택

02 '품질\_임계값' 변수를 활용하여 고/저품질 분류를 수행

03 Binary classification accuracy

```
qualityThreshold = 5;  
binaryPred = meanPred > qualityThreshold;  
binaryOrig = meanOrig > qualityThreshold;  
binaryAccuracy = 100 * sum(binaryPred==binaryOrig)/length(binaryPred)
```

04 평가 점수는 [1, 10] 사이의 값으로 정규화 되었으므로, 중앙 값인 '5'를 임계 값으로 설정



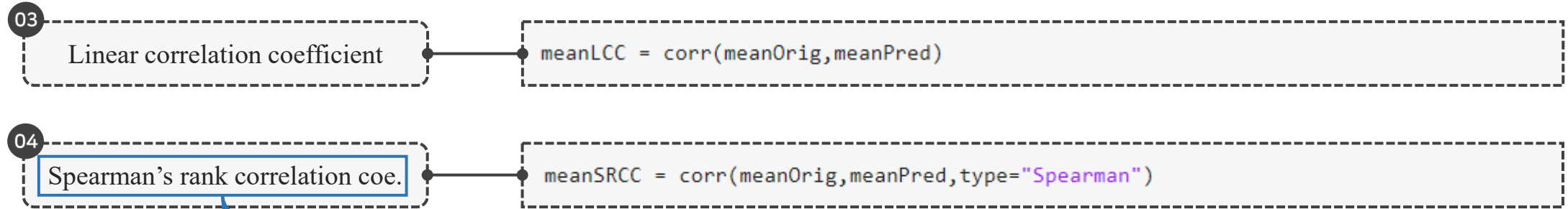
- 고/저품질 분류의 성능을 확인하기 위한 분류 정확도를 채택
- 임계 값 설정을 통해, 두 분류에 대한 정확도를 분석하였으나, '5'로 선정하는 것이 정확한 설계인지 의문



# 3-3 주요 메트릭 분석

## Correlation Coefficients

- 01 본 논문의 주요 목표는 주관적 평가와 모델이 예측한 평가 사이의 차이를 최소화하는 것
- 02 모델에서 **예측한 평가 점수와 실제 사람이 측정한 주관적 평가 간의 상관성**이 높을수록 상관 계수 값이 크게 나타남



- 05 상관성 분석 기법들은 일반적으로 동일한 상관 평가를 달성할 수 있으나, 주체의 관계가 다소 차이
- 06 Pearson (Linear) benchmarks *linear* relationship
- 07 Spearmen benchmarks *monotonic* relationship  
통계-상관성 분석은 다양한 상관 분석 기법을 통해 결정하는 것이 좋음

<https://www.surveymonkey.com/market-research/resources/pearson-correlation-vs-spearman-correlation/>  
<https://stats.stackexchange.com/questions/8071/how-to-choose-between-pearson-and-spearman-correlation>  
<https://www.statisticssolutions.com/free-resources/directory-of-statistical-analyses/correlation-pearson-kendall-spearman/>



- 주관적 평가와 모델이 예측한 평가 사이의 값이 최소화하는 것이 본 연구의 주요 목표
- 따라서, 주관적 평가와 예측한 평가 간의 상관성이 높을수록 모델의 성능이 타 기법 대비 높음을 시사할 수 있음



# 4-1 모델별 성능 분석

## MobileNet vs VGG-16

- 01 본 예제에 활용된 'MobileNet-v2'와 논문에서 언급한 'VGG-16' 간의 학습 성능을 분석 (두 모델은 ImageNet [2]에서 사전 학습을 완료)
- 02 실제 학습 결과, 논문에서 언급한 바와 같이 'VGG-16'가 가장 높은 성능을 기록할 수 있음을 확인

| Metrics             | MobileNet-v2 | VGG-16        |
|---------------------|--------------|---------------|
| EMD                 | 0.0643       | <b>0.8133</b> |
| Classification acc. | 76.245       | <b>84.734</b> |
| LCC                 | 0.6970       | <b>0.8523</b> |
| SRCC                | 0.6755       | <b>0.8244</b> |

| Model                | LCC (mean) | SRCC (mean) | LCC (std.dev) | SRCC (std.dev) | EMD   |
|----------------------|------------|-------------|---------------|----------------|-------|
| Kim et al. [16]      | 0.80       | 0.80        | -             | -              | -     |
| Moorthy et al. [39]  | 0.89       | 0.88        | -             | -              | -     |
| Mittal et al. [40]   | 0.92       | 0.89        | -             | -              | -     |
| Saad et al. [41]     | 0.91       | 0.88        | -             | -              | -     |
| Kottayil et al. [42] | 0.89       | 0.88        | -             | -              | -     |
| Xu et al. [35]       | 0.96       | 0.95        | -             | -              | -     |
| Bianco et al. [7]    | 0.96       | 0.96        | -             | -              | -     |
| NIMA(MobileNet)      | 0.782      | 0.698       | 0.209         | 0.181          | 0.105 |
| NIMA(VGG16)          | 0.941      | 0.944       | 0.538         | 0.557          | 0.054 |
| NIMA(Inception-v2)   | 0.827      | 0.750       | 0.470         | 0.468          | 0.064 |

[1] Talebi, Hossein, and Peyman Milanfar. "NIMA: Neural Image Assessment." IEEE Transactions on Image Processing 27, no. 8 (August 2018): 3998–4011. <https://doi.org/10.1109/TIP.2018.2831899>.

[3] Sandler, Mark, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. "MobileNetV2: Inverted Residuals and Linear Bottlenecks." In 2018 IEEE/CVF, 4510–20. Salt Lake City, UT.

[5] Hou, Le, Chen-Ping Yu, and Dimitris Samaras. "Squared Earth Mover's Distance-Based Loss for Training Deep Neural Networks." Preprint, submitted November 30, 2016. <https://arxiv.org/abs/1611.05916>.




- 예제와 논문에 활용된 'MobileNet-v2', 'VGG-16' 간 학습 성능 분석을 수행
- 논문에 언급한 바와 같이 'VGG-16' 네트워크 모델이 예제에 활용된 'MobileNet-v2' 대비 높은 성능을 제공할 수 있음을 확인



# THANK YOU

---

 010-2983-9390

 snlyoungwoo@gmail.com



# Chapter 10.

## Morphological Operations (형태학적연산) v1

오영우

조선대학교 컴퓨터공학과 석사과정





# CONTENTS

**01** **챕터 소개**  
Introduction of chapter

**02** **용어 및 정의 설명**  
Explanation of term

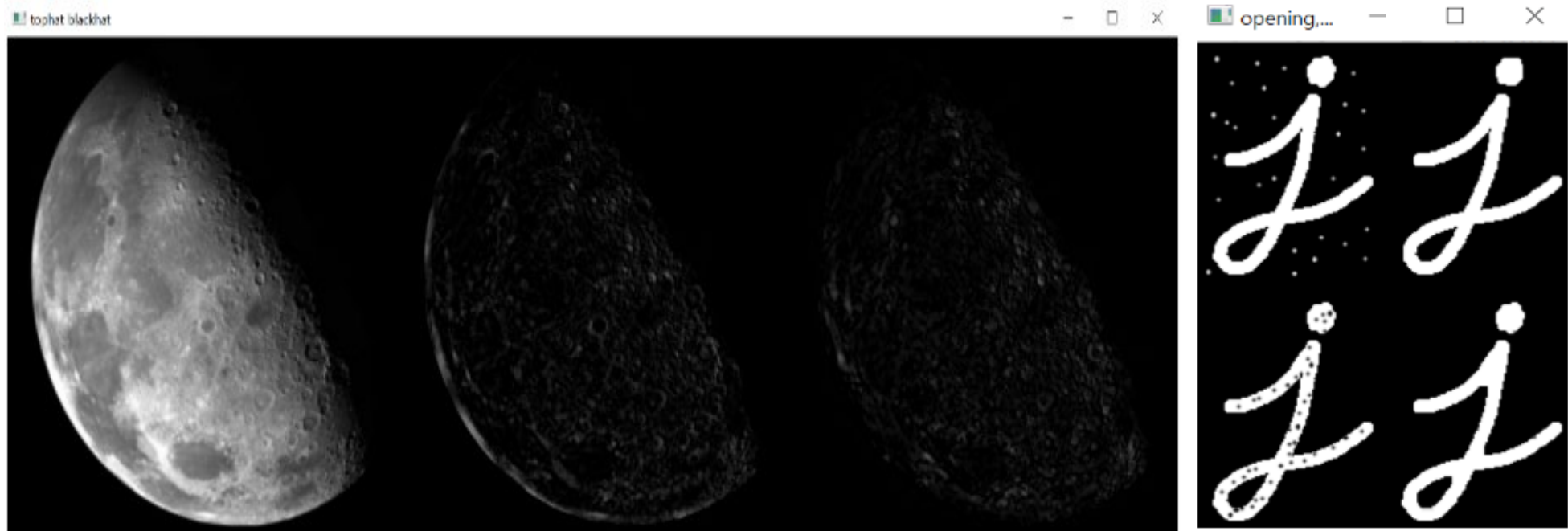
**03** **다음 발표 내용**  
Next presentation



# 1-1

## 챕터 소개 챕터의 주요 구성

- 01 Image Processing Toolbox 모폴로지 함수에 대해 학습
- 02 모폴로지 함수의 주요 기능은 '대비 향상', '노이즈 제거', '골격화', '채우기 및 분할' 등의 이미지 처리 작업을 지원

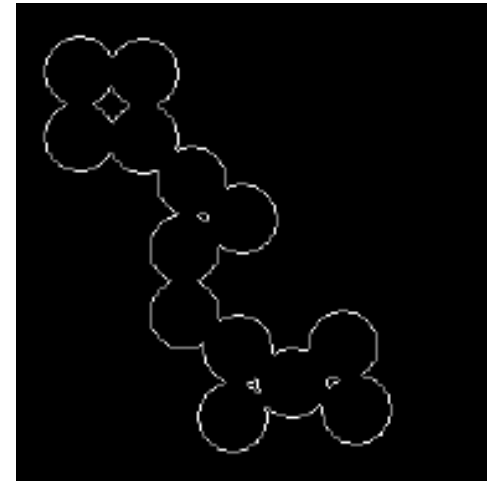
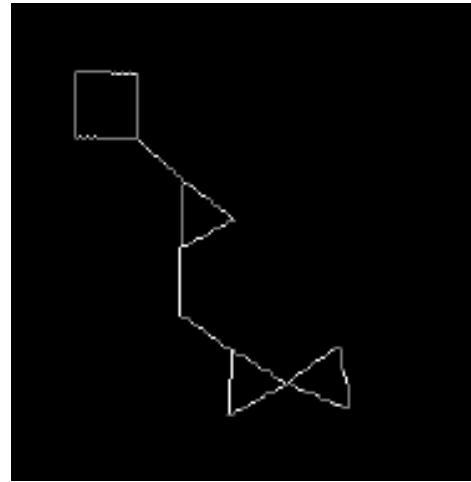
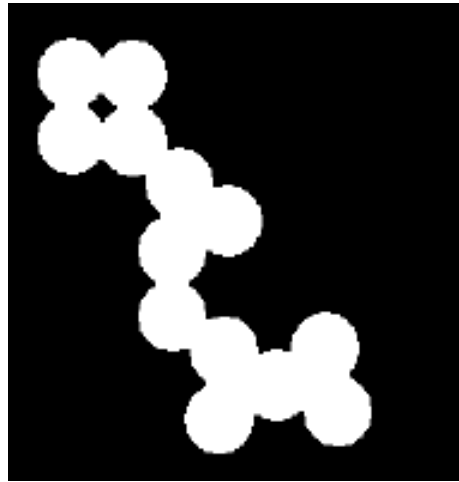
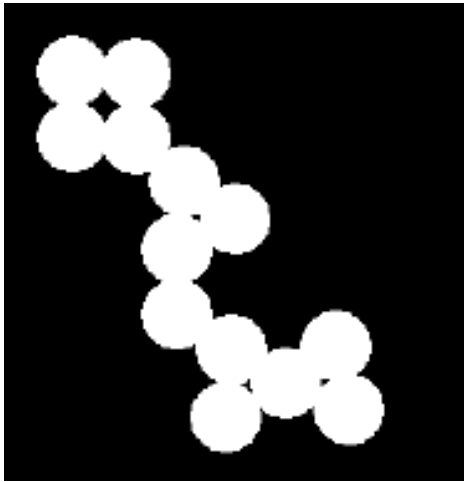


- 형태학적 연산은 학습 이미지 데이터에 대한 노이즈 제거, 분할, 채우기 등의 다양한 전처리 기능을 제공
- 팽창 연산을 적용한 이미지에 대해 침식 연산을 적용한 이미지를 빼는 등의 그레디언트(Gradient) 연산을 포함

# 1-2

## 챕터 소개 주요 학습 내용

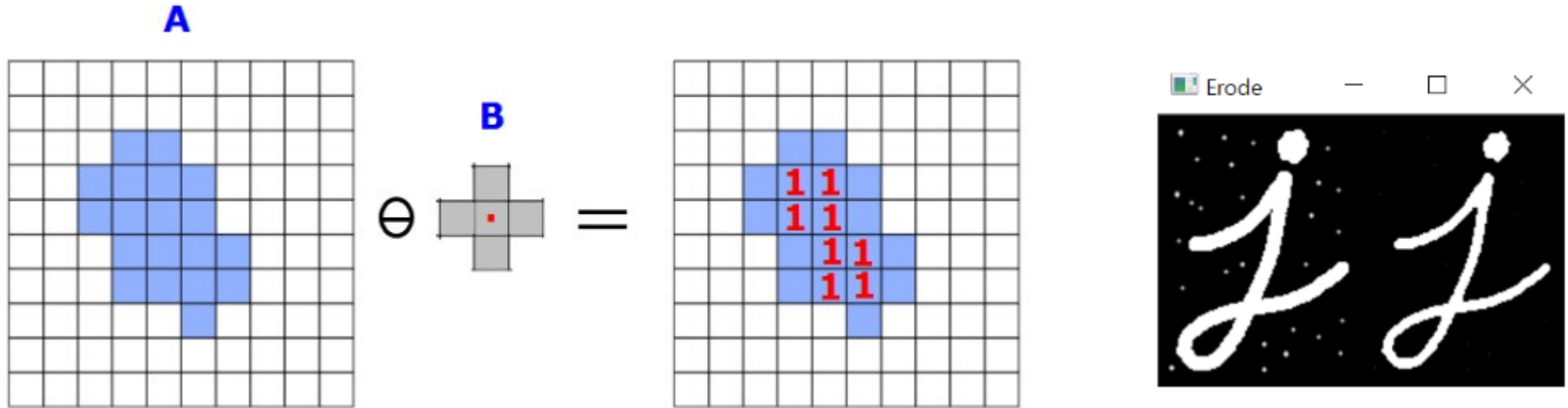
- 01 Morphology(형태학)는 '모양'을 기반으로 이미지를 처리하는 광범위한 이미지 처리 기법을 포함함
- 02 본 장에서 다루는 '형태학적 연산'은 구조 요소를 입력 이미지에 적용하여 동일한 이미지 크기의 출력 이미지를 생성
- 03 이때, 모폴로지 연산(형태학적 연산)의 출력 이미지는 입력 이미지의 해당 픽셀과 이웃 픽셀의 비교를 기반으로 수행됨



- 학습 이미지셋에 대해 외곽을 추출하거나, 노이즈를 추가, 빈 공간을 채워 넣는 등의 여러 기능을 지원하는 함수를 포함
- 그러나, 해당 기법은 인간 느끼는 주관적인 평가와 상반되는 경우가 빈번히 발생

# 2-1 용어 및 정의 설명 Erosion (침식)

- 01 침식(Erosion)이란 말 그대로 형태를 깎아 내는 것으로 이미지를 깎아 내는 연산을 뜻함
- 02 침식 연산 수행을 위해 구조화 요소 커널 (Structuring element kernel)이라는 0과 1로 구성된 커널이 요구됨



<http://blog.daum.net/shksjy/314>

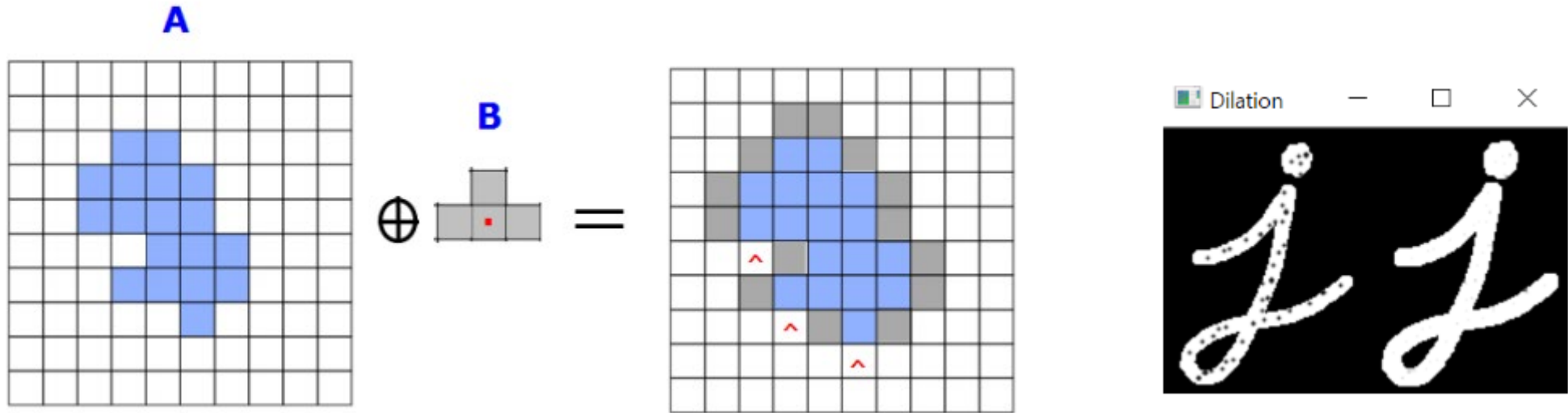
<https://bkshin.tistory.com/m/entry/OpenCV-19-%EB%AA%A8%ED%8F%B4%EB%A1%9C%EC%A7%80Morphology-%EC%97%B0%EC%82%B0-%EC%B9%A8%EC%8B%9D-%ED%8C%BD%EC%B0%BD-%EC%97%B4%EB%A6%BC-%EB%8B%AB%ED%9E%98-%EA%B7%B8%EB%A0%88%EB%94%94%EC%96%B8%ED%8A%B8-%ED%83%91%ED%96%87-%EB%B8%94%EB%9E%99%ED%96%87>



- 침식 연산은 큰 물체의 주변을 깎는 기능을 수행하며, 작은 물체에 대해 제거를 수행하므로써 노이즈 제거에 탁월한 성능을 제공
- 실제 침식 연산 결과, 노이즈가 제거되었으며, 글씨가 전반적으로 가늘어졌음을 확인할 수 있음

# 2-2 용어 및 정의 설명 Dilation (팽창)

- 01 팽창(Dilation)은 침식\* 과 반대로 물체의 주변을 확장을 수행하는 연산을 의미함
- 02 팽창은 침식과 반대로 구조화 요소 커널과 완전히 겹치지 않으면 1로 변경하며, 구조화 요소 커널의 형태는 십자, 일자 등 다양하게 존재



<http://blog.daum.net/shksjy/314>

<https://bkshin.tistory.com/m/entry/OpenCV-19-%EB%AA%A8%ED%8F%B4%EB%A1%9C%EC%A7%80Morphology-%EC%97%B0%EC%82%B0-%EC%B9%A8%EC%8B%9D-%ED%8C%BD%EC%B0%BD-%EC%97%B4%EB%A6%BC-%EB%8B%AB%ED%9E%98-%EA%B7%B8%EB%A0%88%EB%94%94%EC%96%B8%ED%8A%B8-%ED%83%91%ED%96%87-%EB%B8%94%EB%9E%99%ED%96%87>



- 팽창 연산은 침식 연산과 같이 노이즈 제거에 탁월한 성능을 제공
- 실제 팽창 연산 결과, 'J' 글씨가 더 팽창되긴 했으나 글씨 안에 존재하는 노이즈가 제거된 것을 확인할 수 있음



# 3-1 다음 발표 내용 모폴로지 연산과 구조적 요소

- “Types of Morphological Operations” on page 10-2
- “Structuring Elements” on page 10-9
- “Border Padding for Morphology” on page 10-13
- “Morphological Reconstruction” on page 10-14
- “Find Image Peaks and Valleys” on page 10-21
- “Pixel Connectivity” on page 10-27
- “Lookup Table Operations” on page 10-30
- “Dilate an Image to Enlarge a Shape” on page 10-32
- “Remove Thin Lines Using Erosion” on page 10-36
- “Use Morphological Opening to Extract Large Image Features” on page 10-38
- “Flood-Fill Operations” on page 10-42
- “Detect Cell Using Edge Detection and Morphology” on page 10-45
- “Granulometry of Snowflakes” on page 10-50
- “Distance Transform of a Binary Image” on page 10-55
- “Label and Measure Connected Components in a Binary Image” on page 10-57

2차 발표 범위



- 1차 발표는 간단한 챕터의 소개와 2차 발표에 포함되는 내용을 일부 확인하였음
- 2차 발표에서는 모폴로지 연산의 종류와 실행결과 그리고, 구조적 요소에 대해 다룸



PRESENTATION END

**THANK YOU**



---

# Chapter 10.

## Morphological Operations (형태학적연산) v2

---

오영우

조선대학교 컴퓨터공학과 석사과정





# CONTENTS

**01** Erosion & Dilation  
침식과 팽창

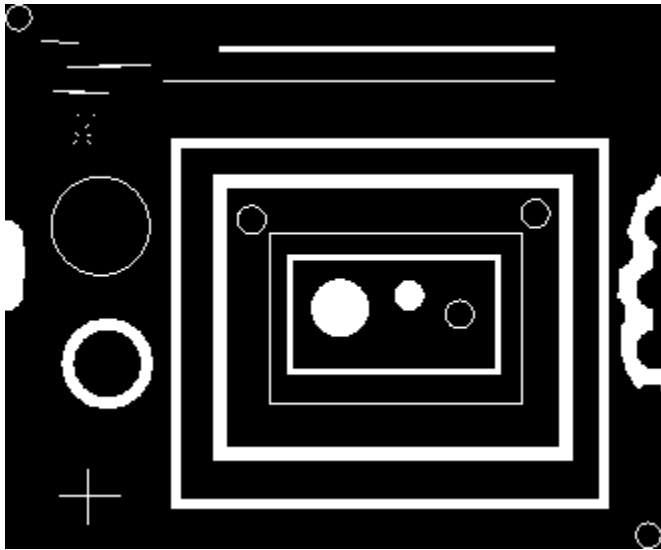
**02** Reconstruction  
형태학적 재구성

**03** Maker & Mask  
마커, 마스크 이미지

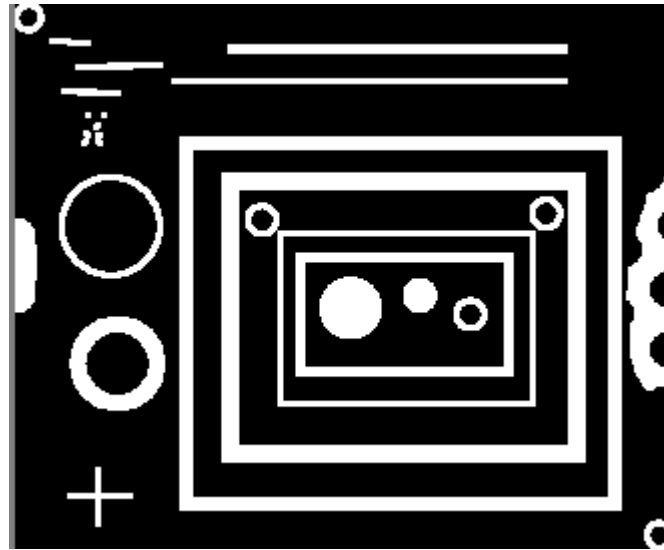
# 1-1 Erosion & Dilation

## 침식과 팽창

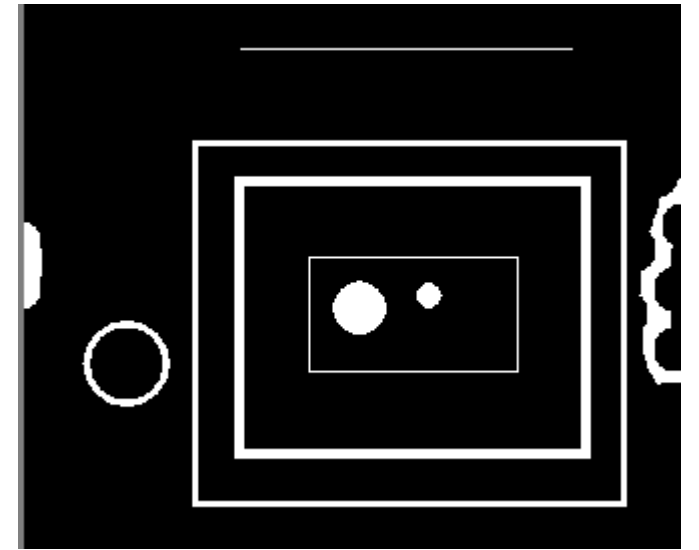
- 01 Image Processing Toolbox 모폴로지 함수 동작의 두 가지 기본 동작을 학습
- 02 침식(Erosion)이란 말 그대로 형태를 깎아 내는 것으로 이미지를 깎아 내는 연산을 뜻함
- 03 팽창(Dilation)은 침식(Erosion)과 반대로 물체의 주변을 확장을 수행하는 연산을 의미함



입력 이미지 (Original)



팽창 (Dilation)



침식 (Erosion)

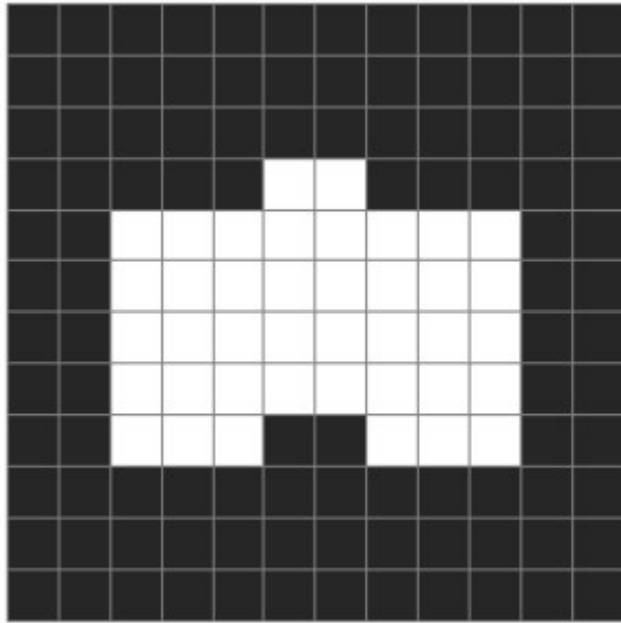


- 형태학적 (모폴로지) 연산의 기본 원칙은 ‘침식’, ‘팽창’을 활용
- 이때, 침식은 형태를 깎아내리는 등의 이미지 축소 효과를, 팽창은 물체 주변을 확장하여 이미지를 확대하는 효과를 지님

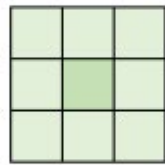
# 1-2 Erosion & Dilation

## 침식

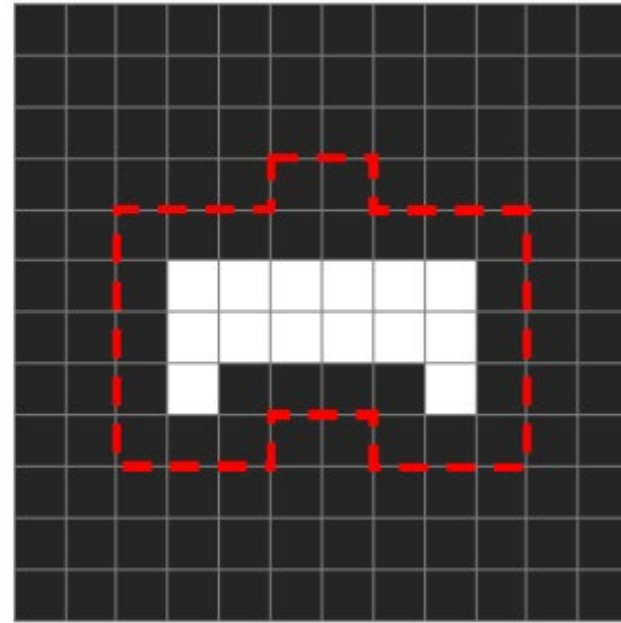
- 01 침식(Erosion)이란 말 그대로 형태를 깎아 내는 것으로 이미지를 깎아 내는 연산을 뜻함
- 02 침식 연산 수행을 위해 구조화 요소 커널 (Structuring element kernel)이라는 0과 1로 구성된 커널이 요구됨



입력 이미지 (Original)



구조화 요소 커널



침식 결과 (Erosion)

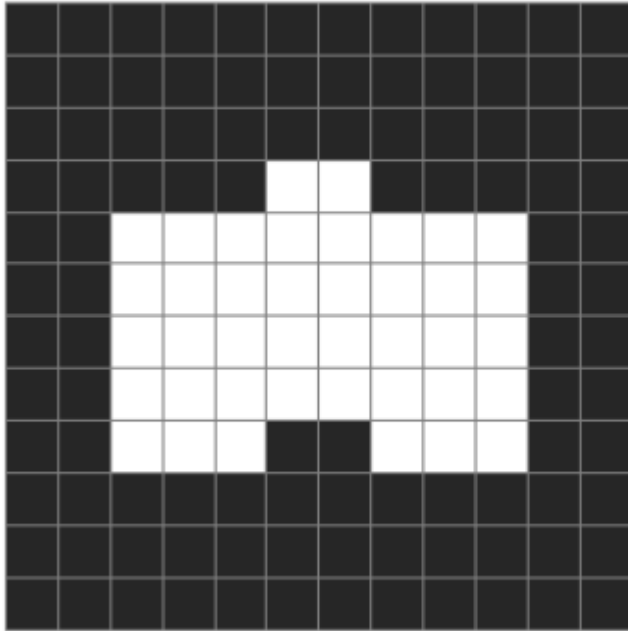


- 침식 연산은 큰 물체의 주변을 깎는 기능을 수행하며, 작은 물체에 대해 제거를 수행하므로써 노이즈 제거에 탁월한 성능을 제공
- 실제 침식 연산 결과, 노이즈가 제거되었으며, 글씨가 전반적으로 가늘어졌음을 확인할 수 있음

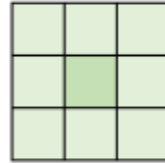
# 1-3 Erosion & Dilation

## 팽창

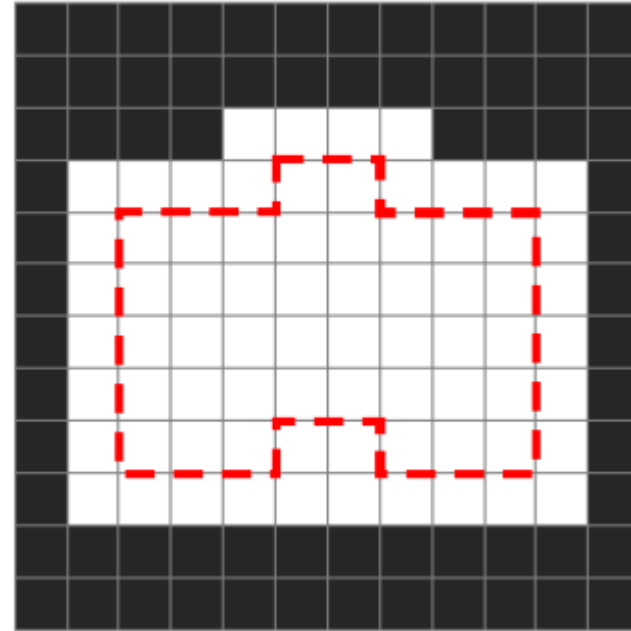
- 01 팽창(Dilation)은 침식(Erosion)과 반대로 물체의 주변을 확장을 수행하는 연산을 의미함
- 02 팽창은 침식과 반대로 구조화 요소 커널과 완전히 겹치지 않으면 1로 변경하며, 구조화 요소 커널의 형태는 십자, 일자 등 다양하게 존재



입력 이미지 (Original)



구조화 요소 커널



침식 결과 (Erosion)



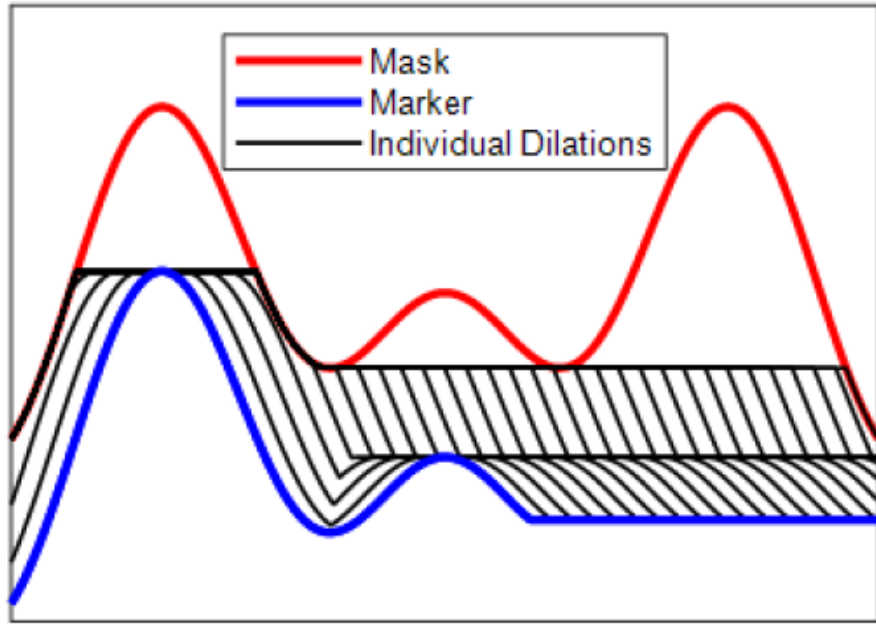
- 팽창 연산은 침식 연산과 같이 노이즈 제거에 탁월한 성능을 제공
- 실제 팽창 연산 결과, 'J' 글씨가 더 팽창되긴 했으나 글씨 안에 존재하는 노이즈가 제거된 것을 확인할 수 있음



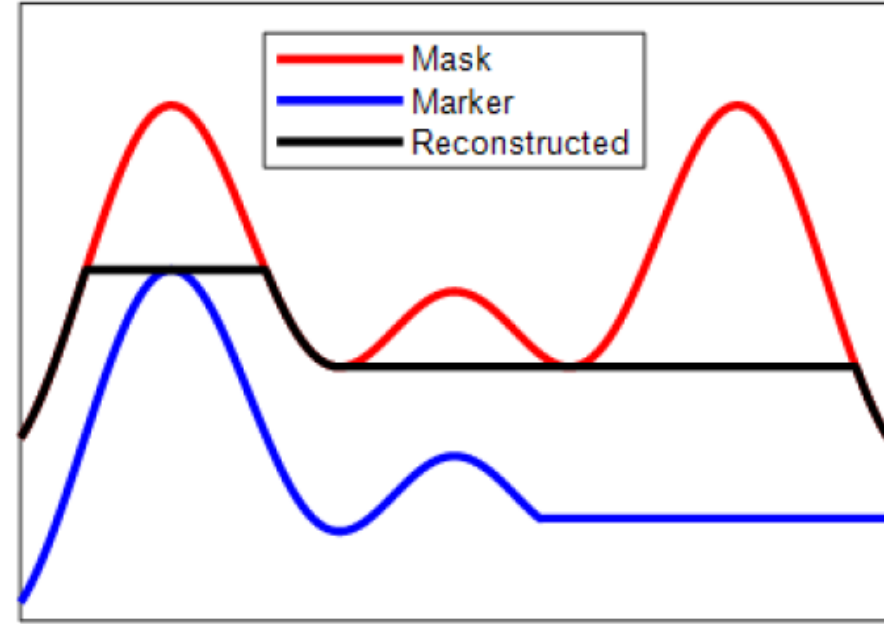
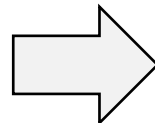
# 2-1 Reconstruction 형태학적 재구성

- 01 이미지를 향상 시키고자 하는 이미지를 'Mask'라고 하며, 'Marker'는 추출 및 강조할 영역을 표시하기 위해 사용
- 02 재구성 과정에서 마스크 이미지에 의한 제약에 의해 온전한 재구성이 불가능할 수 있음

\* 형태학적 재구성 (Reconstruction)은 지정된 마커를 시작으로 연속적인 팽창 과정에 의해 처리 됨



재구성 프로세스



재구성된 Mask 이미지



- 형태학적 재구성 (Reconstruction)은 이미지 내의 객체를 추출하거나, 표현을 강조하는 등 다양하게 활용
- 시작 (강조)되는 마커를 기준으로 연속적 팽창 프로세스를 통해 재구성된 이미지. 즉, Mask 이미지를 얻어낼 수 있음



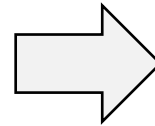
# 2-2 Reconstruction

## 마커와 마스크 이미지

- 01 이미지의 재구성은 Marker와 Mask라는 2개의 이미지에 의해 처리되며, 프로세스는 연속적 팽창 과정을 거침
- 02 즉, 재구성 과정을 위해서 Marker와 Mask라는 2개의 이미지의 사전 처리가 요구됨

|    |    |    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|----|----|
| 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| 10 | 14 | 14 | 14 | 10 | 10 | 11 | 10 | 11 | 10 |
| 10 | 14 | 14 | 14 | 10 | 10 | 10 | 11 | 10 | 10 |
| 10 | 14 | 14 | 14 | 10 | 10 | 11 | 10 | 11 | 10 |
| 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| 10 | 11 | 10 | 10 | 10 | 18 | 18 | 18 | 10 | 10 |
| 10 | 10 | 10 | 11 | 10 | 18 | 18 | 18 | 10 | 10 |
| 10 | 10 | 11 | 10 | 10 | 18 | 18 | 18 | 10 | 10 |
| 10 | 11 | 10 | 11 | 10 | 10 | 10 | 10 | 10 | 10 |
| 10 | 10 | 10 | 10 | 10 | 10 | 11 | 10 | 10 | 10 |

Mask 이미지



|   |    |    |    |   |    |    |    |   |   |
|---|----|----|----|---|----|----|----|---|---|
| 8 | 8  | 8  | 8  | 8 | 8  | 8  | 8  | 8 | 8 |
| 8 | 12 | 12 | 12 | 8 | 8  | 9  | 8  | 9 | 8 |
| 8 | 12 | 12 | 12 | 8 | 8  | 8  | 9  | 8 | 8 |
| 8 | 12 | 12 | 12 | 8 | 8  | 9  | 8  | 9 | 8 |
| 8 | 8  | 8  | 8  | 8 | 8  | 8  | 8  | 8 | 8 |
| 8 | 9  | 8  | 8  | 8 | 16 | 16 | 16 | 8 | 8 |
| 8 | 8  | 8  | 9  | 8 | 16 | 16 | 16 | 8 | 8 |
| 8 | 8  | 9  | 8  | 8 | 16 | 16 | 16 | 8 | 8 |
| 8 | 9  | 8  | 9  | 8 | 8  | 8  | 8  | 8 | 8 |
| 8 | 8  | 8  | 8  | 8 | 8  | 9  | 8  | 8 | 8 |

Marker 이미지



- 입력되는 Mask 이미지에 대한 Marker 이미지 생성은 Mask 이미지의 픽셀에 상수를 증분하여 얻을 수 있음
- 이때, 증분 된 최대 픽셀의 크기는 최초 입력된 Mask 이미지의 최대 픽셀 값을 초과할 수 없음

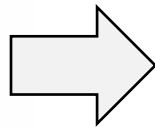


# 3-1 Marker & Mask 마커와 마스크 이미지

- 01 Marker 이미지 내 증분 과정을 통해 최소 값을 지닌 픽셀을 모두 동일한 값으로 처리
- 02 아래 예시와 같이 피크가 2개 이상인 경우, 피크 중 하나를 제거하여 최대 픽셀 값을 남기도록 함

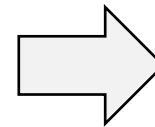
|   |    |    |    |   |    |    |    |   |   |
|---|----|----|----|---|----|----|----|---|---|
| 8 | 8  | 8  | 8  | 8 | 8  | 8  | 8  | 8 | 8 |
| 8 | 12 | 12 | 12 | 8 | 8  | 9  | 8  | 9 | 8 |
| 8 | 12 | 12 | 12 | 8 | 8  | 8  | 9  | 8 | 8 |
| 8 | 12 | 12 | 12 | 8 | 8  | 9  | 8  | 9 | 8 |
| 8 | 8  | 8  | 8  | 8 | 8  | 8  | 8  | 8 | 8 |
| 8 | 9  | 8  | 8  | 8 | 16 | 16 | 16 | 8 | 8 |
| 8 | 8  | 8  | 9  | 8 | 16 | 16 | 16 | 8 | 8 |
| 8 | 8  | 9  | 8  | 8 | 16 | 16 | 16 | 8 | 8 |
| 8 | 9  | 8  | 9  | 8 | 8  | 8  | 8  | 8 | 8 |
| 8 | 8  | 8  | 8  | 8 | 8  | 9  | 8  | 8 | 8 |

Marker 이미지의 재구성 과정



|    |    |    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|----|----|
| 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| 10 | 12 | 12 | 12 | 10 | 10 | 10 | 10 | 10 | 10 |
| 10 | 12 | 12 | 12 | 10 | 10 | 10 | 10 | 10 | 10 |
| 10 | 12 | 12 | 12 | 10 | 10 | 10 | 10 | 10 | 10 |
| 10 | 12 | 12 | 12 | 10 | 10 | 10 | 10 | 10 | 10 |
| 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| 10 | 10 | 10 | 10 | 10 | 16 | 16 | 16 | 10 | 10 |
| 10 | 10 | 10 | 10 | 10 | 16 | 16 | 16 | 10 | 10 |
| 10 | 10 | 10 | 10 | 10 | 16 | 16 | 16 | 10 | 10 |
| 10 | 10 | 10 | 10 | 10 | 16 | 16 | 16 | 10 | 10 |
| 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |

1차 재구성된 Marker 이미지



|    |    |    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|----|----|
| 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| 10 | 10 | 10 | 10 | 10 | 16 | 16 | 16 | 10 | 10 |
| 10 | 10 | 10 | 10 | 10 | 16 | 16 | 16 | 10 | 10 |
| 10 | 10 | 10 | 10 | 10 | 16 | 16 | 16 | 10 | 10 |
| 10 | 10 | 10 | 10 | 10 | 16 | 16 | 16 | 10 | 10 |
| 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |

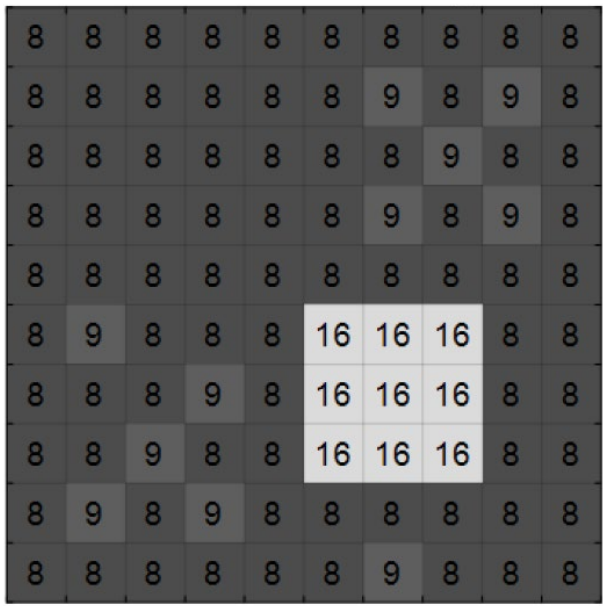
2차 재구성된 Marker 이미지



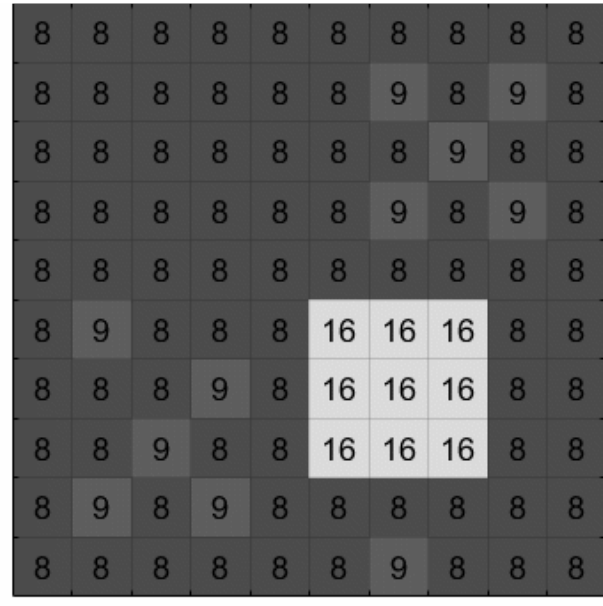
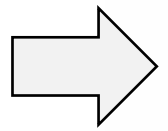
- Marker 이미지는 Mask 이미지에 대한 강조 및 표현을 표시하게 위해 사용되며 증분 과정을 통해, 이미지 내 픽셀의 최소 값을 정규화
- 증분과정을 통해 산출되는 여러 Peak 중 특정 Peak를 지우거나, 포함하여 Mask 이미지의 강조 부분을 표기할 수 있음

# 3-2 Marker & Mask 마커와 마스크 이미지

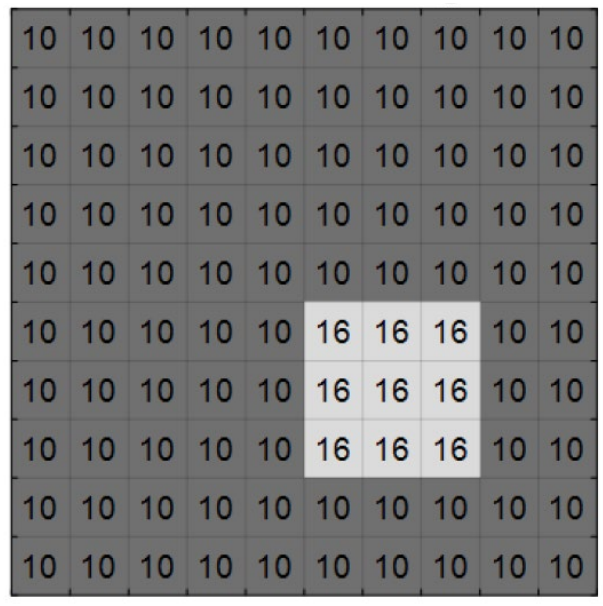
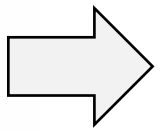
- 01 Peak가 하나만 존재하는 경우, 아래와 같이 증분 과정을 통해 Marker 이미지를 산출
- 02 이때, 재구성되는 이미지의 픽셀 값은 최대 Peak의 값을 초과할 수 없음



입력된 Marker 이미지



입력된 Marker에 대한 재구성 과정



최종 재구성된 Marker 이미지



- 형태학적 재구성 (Reconstruction)은 이미지 내의 객체를 추출하거나, 표현을 강조하는 등 다양하게 활용
- 이때, 향상 시키고자 하는 입력 이미지를 'Mask', 입력된 이미지를 추출하거나 강조할 영역을 표시하는 것을 'Marker'라고 함





PRESENTATION END

**THANK YOU**



# Chapter 10.

## Morphological Operations (형태학적연산) v3

오영우

조선대학교 컴퓨터공학과 석사과정



# CONTENTS

**01** Global & Regional  
전역 및 지역 최대, 최소값

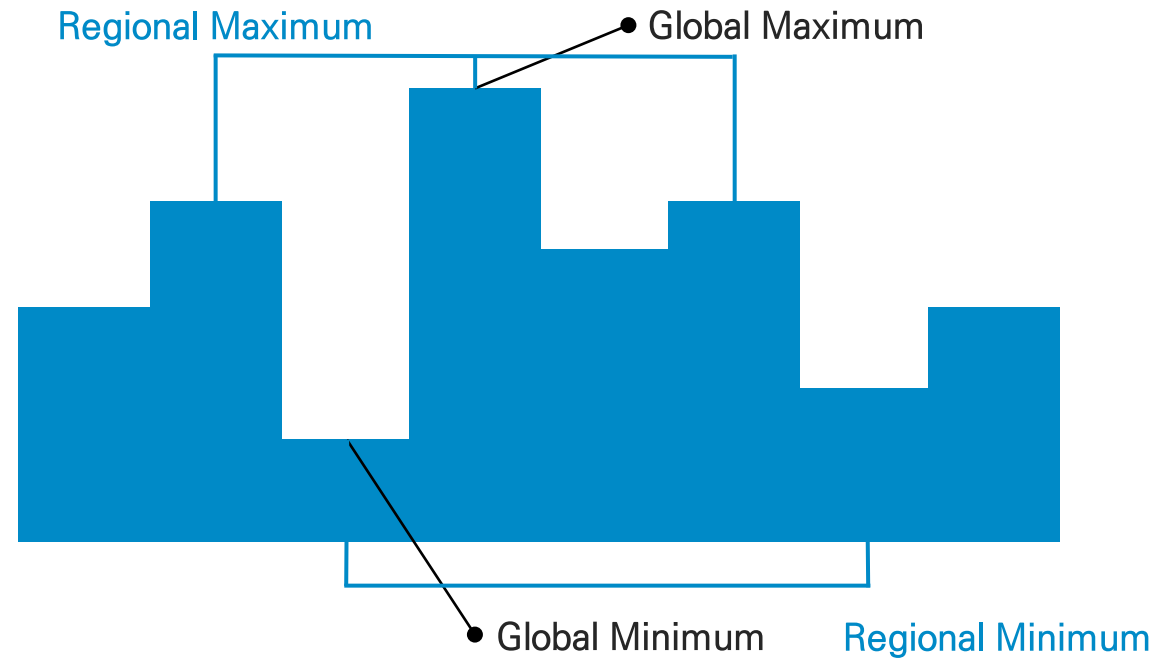
**02** Related Work  
학제간 연구: 영상처리와 무선 통신

**03** Next Session  
다음 발표 내용



# 1-1 Global & Regional 전역 및 지역의 최대, 최소

- 01 이미지 데이터의 경우, 여러 지역 최대, 최소 값이 존재하므로 전체 이미지에 대한 전역 최대, 최소 값을 찾는 것이 필요
- 02 **Image Processing Toolbox 모폴로지 함수**는 이러한 픽셀의 전역 및 지역의 최대, 최소 값을 찾도록 지원



- 이미지 및 영상 데이터의 경우, 다양한 지역 최대 및 최소 값을 지니기에, 전체 입력 이미지에 대한 최대, 최소 값을 판별하는 것이 요구 됨
- 이때, 모폴로지 함수는 모든 지역 최대, 최소 값을 찾거나, 특정 임계 값을 기반으로 값을 찾는 등의 여러 함수를 지원

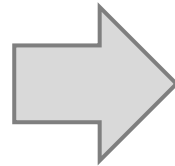
# 1-2 Global & Regional 최대 값 탐색 (1)

01 imregionalmax, imregionalmin 함수는 모든 지역의 최대 및 최소 값 탐색을 지원

\* 모든 지역 변수에 대한 값 탐색을 수행하므로, 전역과 지역에 대한 구분 판별할 수 없음

```
A = [10 10 10 10 10 10 10 10 10 10;
     10 14 14 14 10 10 11 10 11 10;
     10 14 14 14 10 10 10 11 10 10;
     10 14 14 14 10 10 11 10 11 10;
     10 10 10 10 10 10 10 10 10 10;
     10 11 10 10 10 18 18 18 10 10;
     10 10 10 11 10 18 18 18 10 10;
     10 10 11 10 10 18 18 18 10 10;
     10 11 10 11 10 10 10 10 10 10;
     10 10 10 10 10 10 11 10 10 10];
```

< 입력 이미지 >



B = imregionalmax(A)

```
B =
0 0 0 0 0 0 0 0 0 0
0 1 1 1 0 0 1 0 1 0
0 1 1 1 0 0 0 1 0 0
0 1 1 1 0 0 1 0 1 0
0 0 0 0 0 0 0 0 0 0
0 1 0 0 0 1 1 1 0 0
0 0 0 1 0 1 1 1 0 0
0 0 1 0 0 1 1 1 0 0
0 1 0 1 0 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0
```

< imregionalmax 함수 실행 결과 >

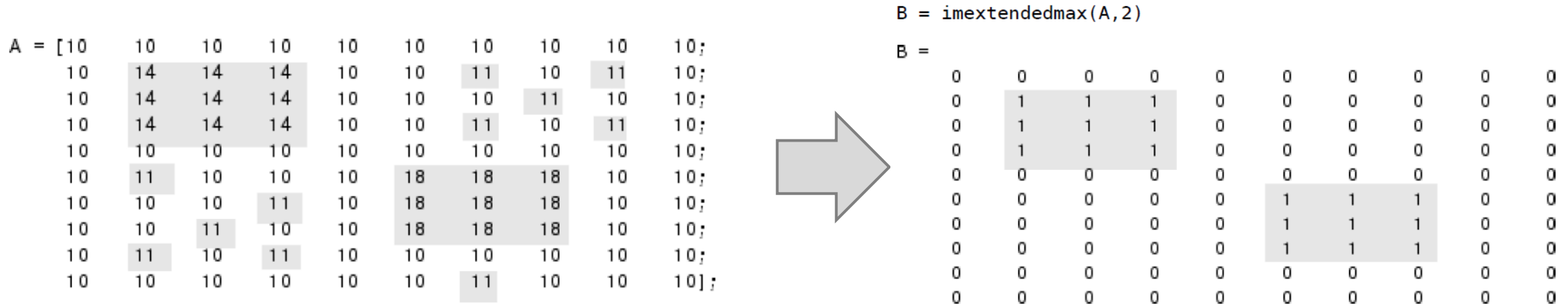


- 모든 지역에 대한 최대 값 탐색과 특정 임계 값 기반의 최대 값 탐색을 지원
- 이때, 탐색된 결과는 0과 1로 표현되며, 0은 최대 값이 아닌 값을 의미하고, 1은 모든 지역 내 최대 값을 나타냄

# 1-3 Global & Regional 최대 값 탐색 (2)

01 imextendedmax, imextendedmin 함수는 특정 임계 값을 기준으로 지역의 최대 및 최소 값 탐색을 수행

\* 인접 픽셀의 값을 허용(임계 값)하여, 전체 지역 내 값 중 조건에 부합하는 최대 값을 판별할 수 있음



< 입력 이미지 >

< imextendedmax함수 실행 결과 >



- 모든 지역에 대한 최대 값 탐색하는 과정에서 임계 값을 설정하여, 인접 픽셀이 모두 최대 값인 픽셀을 탐색할 수 있음
- 그러나, 전역 최대 및 최소 값에 대한 탐색은 별도의 작업이 요구됨



# 2-1 Related Work

## 학제간 연구 : 영상 처리와 무선 통신

```
function [capacityOfSubsetMax, Ex_H_sel]=AS_Exhaustive(Nr,Nt,Lr,SNR,H,fullAntenna)

% Record the maximum channel capacity of all subset when the number of antennas selected is Lr
capacityOfSubsetMax=0;
Ex_H_sel = [];

% The number of antennas selected is Lr with a total of nchoosek (Nr,Lr) subset.
for k=1:nchoosek(Nr,Lr)
    % Subset
    indexOfChannel = fullAntenna(k,:);

    % Selected channel matrix H
    Ex_H_sel=H(indexOfChannel,:);

    % Capacity of subset
    capacityOfSubset=log2(det(eye(Nt)+SNR/Nt*(Ex_H_sel'*Ex_H_sel)));

    % Maximum channel capacity among all subsets
    if(capacityOfSubset>capacityOfSubsetMax)
        capacityOfSubsetMax=capacityOfSubset;
    end
end
```



- MIMO 시스템에서의 모든 안테나를 사용하는 것은 RF chain 생성 및 전력 소모 등의 문제를 야기
- 기존 접근 방식의 경우, 가능한 모든 안테나 부 집합을 생성하고, 해당 부집합에 따라 결정되는 채널 용량을 최대화



# 2-2 Related Work

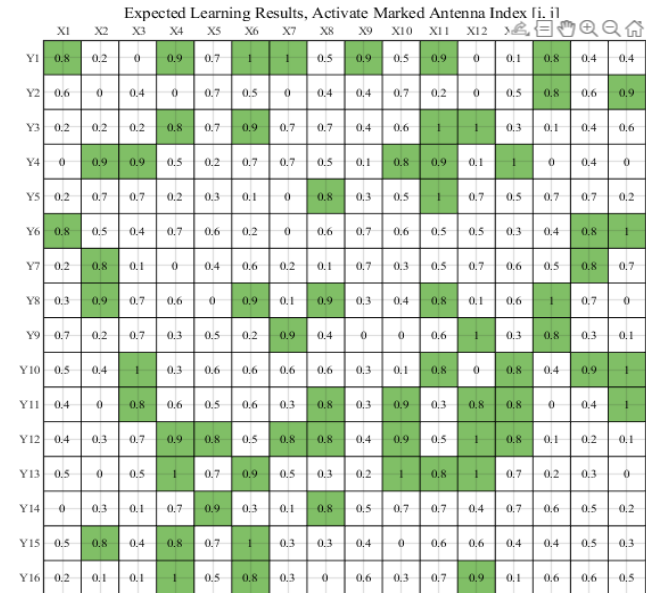
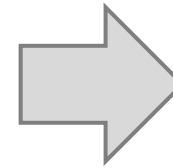
## 영상 처리 기반 접근 방식

01 Max-min 정규화 기법을 통해 처리된 채널 행렬은 다음과 같이 하나의 학습 이미지로 사용할 수 있음

$$h_{ij} \leftarrow \frac{h_{ij} - \min\{H_i\}}{\max\{H_i\} - \min\{H_i\}}, \text{ for all } i, j \quad (3)$$

|        |        |        |        |        |        |        |        |
|--------|--------|--------|--------|--------|--------|--------|--------|
| 1.0000 | 0.9137 | 0.2252 | 0.2349 | 0.0000 | 0.7049 | 0.8066 | 0.5195 |
| 0.7220 | 0.5300 | 1.0000 | 0.9615 | 0.8863 | 0.0000 | 0.2767 | 0.7799 |
| 0.4215 | 1.0000 | 0.9847 | 0.6378 | 0.8566 | 0.0000 | 0.4372 | 0.7287 |
| 1.0000 | 0.3264 | 0.5441 | 0.1959 | 0.8721 | 0.0000 | 0.5292 | 0.5606 |
| 0.0000 | 0.4948 | 1.0000 | 0.6168 | 0.1889 | 0.7438 | 0.3409 | 0.7681 |
| 0.0000 | 0.7865 | 1.0000 | 0.5667 | 0.4338 | 0.5453 | 0.9044 | 0.7055 |
| 0.2248 | 0.0142 | 0.7559 | 0.4305 | 0.5064 | 0.2774 | 1.0000 | 0.0000 |
| 0.6323 | 1.0000 | 0.5391 | 0.2704 | 0.0000 | 0.1969 | 0.3514 | 0.7501 |

< 정규화 된 채널 행렬의 vector (0~1) >



< 생성된 채널 행렬 이미지 >

\* 실제 논문에 발췌된 내용은 아니며, 이해를 돕기 위한 이미지

Jiaxin Cai, Yan Li, Ying Hu, "Deep convolutional neural network based antenna selection in multiple-input multiple-output system," Proc. SPIE 10710, Young Scientists Forum 2017.

Cai J-x, Zhong R, Li Y, "Antenna selection for multiple-input multiple-output systems based on deep convolutional neural networks." PLoS ONE 14(5): e0215672.



- 기존의 접근 방식은 채널 행렬을 고려하지 않고, 조합을 이용하여 최적의 채널 용량을 생성하는 안테나 부집합에 초점
- 반면, 제안하는 방식은 송신 및 수신 과정에서 생성되는 채널 행렬(입력 데이터)에서 최적의 안테나 부 집합 패턴을 학습하는 방식





# 2-3 Related Work

## 채널행렬에 대한 최적 안테나 집합 생성

- 01 본 논문의 주요 방식은 Exhaustive 탐색과 동일하지만, 채널 행렬(입력 이미지)를 통해 최적의 안테나 부 집합(패턴)을 학습
- 02 이는 곧, CNN 정확도가 증가함에 따라 최적의 성능을 제공하는 Exhaustive searching 기법과 동일한 성능을 제공할 수 있음

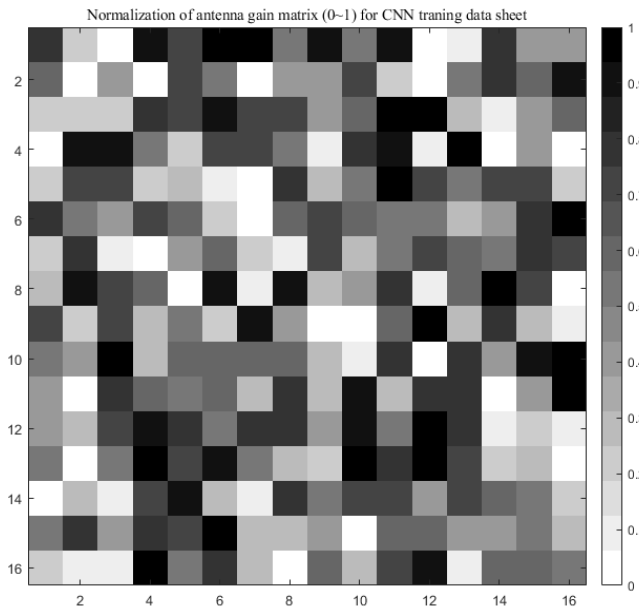


Table 1. Examples of selected antenna indices and their corresponding classes.

| Selected antenna index                 | Corresponding class |
|--|---------------------|
| $(1, 2, \dots, N_s - 1, N_s)$          | 1                   |
| $(1, 2, \dots, N_s - 1, N_s + 1)$      | 2                   |
| ...                                    | ...                 |
| $(1, 2, \dots, N_s - 1, N_r)$          | $L = r - s + 1$     |
| $(1, 2, \dots, N_s - 2, N_s, N_s + 1)$ | $L = r - s + 2$     |
| $(1, 2, \dots, N_s - 2, N_s, N_s + 2)$ | $L = r - s + 3$     |
| ...                                    | ...                 |

Jiaxin Cai, Yan Li, Ying Hu, "Deep convolutional neural network based antenna selection in multiple-input multiple-output system," Proc. SPIE 10710, Young Scientists Forum 2017.  
 Cai J-x, Zhong R, Li Y, "Antenna selection for multiple-input multiple-output systems based on deep convolutional neural networks." PLoS ONE 14(5): e0215672.



- 입력되는 채널행렬 이미지에 대해 최적의 안테나 부 집합 패턴을 학습하는 방식으로 접근
- CNN 학습 성능에 따라 실제 안테나 선택 기법의 성능이 변화되므로, 높은 정확도를 달성하는 것이 중요



# 2-4 Related Work

## CNN 모델 별 학습 성능 분석

- 01 실제 논문 결과에 따르면 ResNet의 정확도 가장 높으며, 반면 VGG-16 정확도가 가장 낮음
- 02 CNN 모델의 분류 정확도를 개선할 수 있다면, 모든 경우의 수에 대한 비교 없이 저-복잡도로 높은 채널 용량을 달성할 수 있음을 의미

Table 2. Comparison of antenna selection methods about classification accuracy.

| Antenna selection method | Accuracy(%) |
|--------------------------|-------------|
| ResNet                   | 79.16       |
| LeNet                    | 49.21       |
| AlexNet                  | 3.60        |
| VGG-16                   | 3.65        |
| RNN                      | 24.00       |
| LSTM                     | 60.00       |
| KNN                      | 8.29        |
| SVM                      | 22.12       |

Jiaxin Cai, Yan Li, Ying Hu, "Deep convolutional neural network based antenna selection in multiple-input multiple-output system," Proc. SPIE 10710, Young Scientists Forum 2017.  
 Cai J-x, Zhong R, Li Y, "Antenna selection for multiple-input multiple-output systems based on deep convolutional neural networks." PLoS ONE 14(5): e0215672.



- CNN 모델의 분류 정확도에 따라 MIMO 시스템의 전체 채널 용량 증가될 수 있음을 보임
- 특히, Exhaustive searching 기반 선택 기법 대비 월등히 적은 복잡도로 준-최적화된 성능을 제공할 수 있음



# 2-5 Related Work

## 제안 방식의 장단점

- 01 5G, 6G 네트워크에서 요구되는 안테나 및 전력 할당 기법 등의 **자원할당 영역에 쉽게 적용 가능한 접근 방식**
- 02 본 논문에서 제안된 **CNN 기반의 안테나 선택 기법 성능에 대한 구체적인 분석이 시행되지 않고 있음**

| PROS   | CONS  |
|--|---|
| Massive MIMO 규격에 준하는 안테나 운용에도 <b>저-복잡도</b> 로 처리가 가능    | 구체적인 타 기법과의 성능 <b>분석이 시행되지 않음</b>                     |
| 최근 제안되고 있는 강화학습 기반 할당 기법과 달리 <b>개발에 오랜 소요가 요구되지 않음</b> | 입력 데이터(채널 행렬)에 대응하는 <b>최적의 부 집합(패턴) 생성에 오랜 시간이 소요</b> |
| CNN 분류 정확도 개선된다면, 실제 채널 <b>용량 개선의 가능성이 존재</b>          | <b>Imperfect CSI</b> 상황에서의 운용이 어려움                    |

Jiaxin Cai, Yan Li, Ying Hu, "Deep convolutional neural network based antenna selection in multiple-input multiple-output system," Proc. SPIE 10710, Young Scientists Forum 2017.  
 Cai J-x, Zhong R, Li Y, "Antenna selection for multiple-input multiple-output systems based on deep convolutional neural networks." PLoS ONE 14(5): e0215672.



- 영상 처리 기법 및 CNN을 활용한 무선 네트워크의 자원 할당 문제를 쉽게 해결 할 수 있는 접근 방식이라고 판단 됨
- 그럼에도 불구하고, 이에 대한 후속 연구나 타 기법들 간 성능 분석이 시행되지 않음

# 3-1 Next Session

## 다음 발표 내용

- “Types of Morphological Operations” on page 10-2
- “Structuring Elements” on page 10-9
- “Border Padding for Morphology” on page 10-13
- “Morphological Reconstruction” on page 10-14
- “Find Image Peaks and Valleys” on page 10-21
- “Pixel Connectivity” on page 10-27
- “Lookup Table Operations” on page 10-30
- “Dilate an Image to Enlarge a Shape” on page 10-32
- “Remove Thin Lines Using Erosion” on page 10-36
- “Use Morphological Opening to Extract Large Image Features” on page 10-38
- “Flood-Fill Operations” on page 10-42
- “Detect Cell Using Edge Detection and Morphology” on page 10-45
- “Granulometry of Snowflakes” on page 10-50
- “Distance Transform of a Binary Image” on page 10-55
- “Label and Measure Connected Components in a Binary Image” on page 10-57

4차 발표 범위



PRESENTATION END

**THANK YOU**



# Chapter 10.

## Morphological Operations (형태학적연산) v4

오영우

조선대학교 컴퓨터공학과 석사과정



# CONTENTS

**01** Detect cell using edge detection and morphology  
경계 검출 및 모폴로지를 활용한 세포 검출

**02** Detection cell in MATLAB  
MATLAB 내 세포 검출 예제

**03** Next Session  
다음 발표 내용



# 1-1 Detect cell using edge detection and morphology 경계 검출 및 모폴로지를 활용한 세포 검출

- 01 경계 검출과 기본 모폴로지 (팽창, 수축)을 활용한 전립선 암 세포 검출 예제
- 02 주요 목표는 이미지 내 **완전한 형태의 세포를 검출 또는 분할** 하는 것

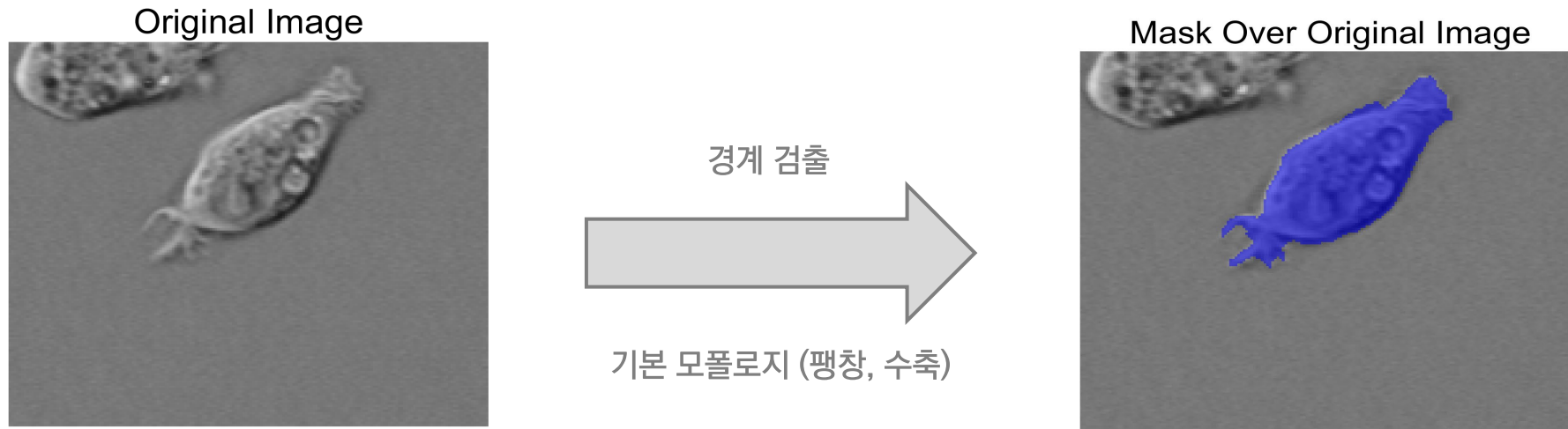


Image courtesy of Alan Partin  
Johns Hopkins University



- 경계 검출과 기본 모폴로지를 사용하여 배경과 두드러지게 대비되는 세포 객체를 영상 내에서 검출하는 대한 예제
- 검출된 세포 객체를 마스크 혹은 외각선을 표시하여, 탐지된 객체를 표기할 수 있음





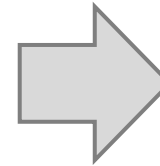
# 2-1 Detection cell in MATLAB

## 세포 이미지 호출

01 imread(), imshow(), 함수를 통해 입력 이미지를 호출 및 표시

```
I = imread('cell.tif');  
imshow(I)  
title('Original Image');  
text(size(I,2),size(I,1)+15, ...  
      'Image courtesy of Alan Partin', ...  
      'FontSize',7,'HorizontalAlignment','right');  
text(size(I,2),size(I,1)+25, ....  
      'Johns Hopkins University', ...  
      'FontSize',7,'HorizontalAlignment','right');
```

〈 입력 이미지 호출 및 표시 〉



Original Image



Image courtesy of Alan Partin  
Johns Hopkins University

〈 입력 이미지 〉



- imread(), imshow() 함수를 이용하여, 'cell.tif' 입력 이미지를 호출 및 표시
- 호출된 이미지는 각각 불완전한 세포 1개와 완전한 세포 객체를 포함하고 있음



# 2-2 Detection cell in MATLAB

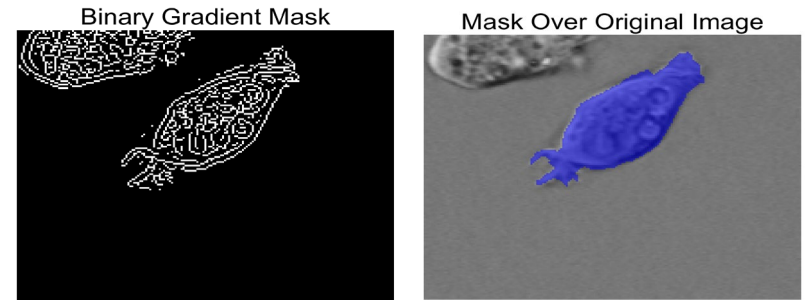
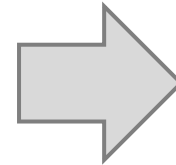
## 완전한 세포 객체 검출

- 01 Edge와 sobel 연산자를 활용하여 입력 이미지에 대한 임계 값을 계산
- 02 Binary 타입의 마스크를 얻기 위해, **edge 연산을 통해 얻어진 임계 값과 퍼지 값의 곱**으로 다시 (BW) edge 검출을 시행

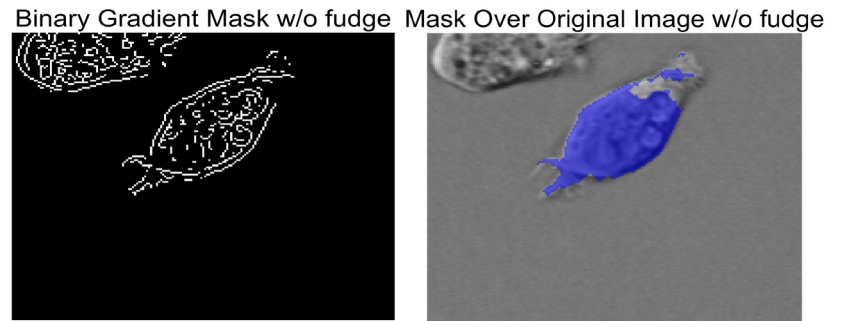
\* fudgeFactor를 사용하여 임계값이 하향 조정됨에 따라 검출할 수 있는 edge 연산의 결과가 증가

```
[~,threshold] = edge(I,'sobel');
fudgeFactor = 0.5;
BWs = edge(I,'sobel',threshold * fudgeFactor);
imshow(BWs)
title('Binary Gradient Mask')
```

< edge 연산자를 활용한 임계 값 생성 >



< fudgeFactor를 통해 생성된 이진 마스크 >



< fudgeFactor 없이 생성된 이진 마스크 >

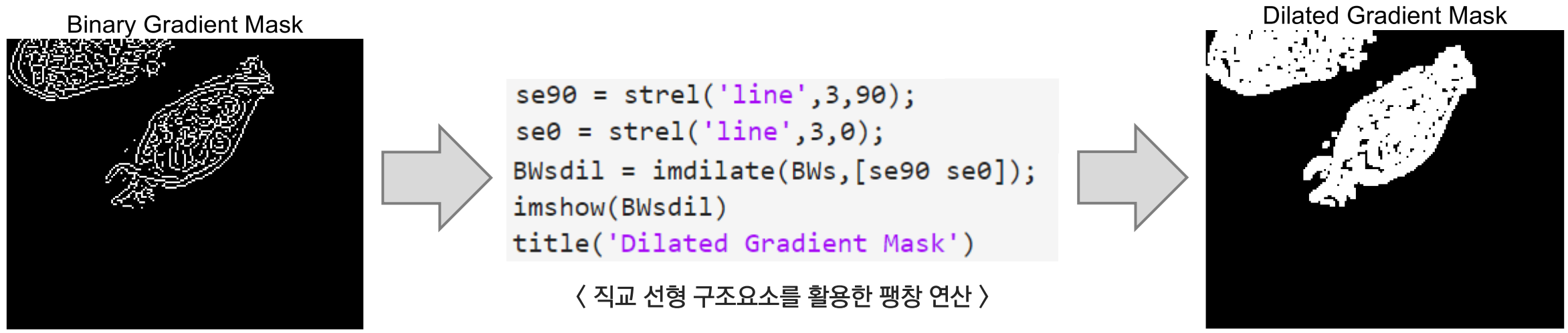


- Edge와 sobel 연산을 통해 임계 값을 계산하고, 얻어진 값에 fudgeFactor 값을 곱하여, edge 연산을 재차 실행
- 이때, fudgeFactor를 사용하여 임계 값을 하향 조정함에 따라 더 많은 edge 연산의 결과를 얻을 수 있음

# 2-3 Detection cell in MATLAB

## 입력 이미지 팽창

- 01 이진 기울기 마스크는 객체의 윤곽을 알아내기에 적합하지 않으므로 Sobel 영상에 대한 팽창을 수행
- 02 Sobel 영상에 대해 팽창을 수행하는 이유는 이진 기울기 마스크 내 객체를 둘러싼 선 간격을 제거하기 위함



- 이진 기울기 마스크 이미지에 존재하는 객체 내 선 간격을 제거하기 위해 모폴로지 연산 (팽창)을 수행
- 이때, 이미지에 대한 세로, 가로 (직교)하는 선형 구조 요소를 생성하여, 팽창을 진행

# 2-4 Detection cell in MATLAB

## 객체 내 빈 공간 제거

- 01 팽창 연산을 통해, 객체 내 선 간격을 제거하였으나, 객체 내 빈 공간이 존재하므로 이를 채우는 작업을 수행
- 02 `imfill()` 함수를 활용하여, 객체 내 빈 공간 없이 객체를 채움



- 팽창 연산을 통해 객체 내 선 간격을 제거했으나, 객체 내 빈 공간이 발생하므로 이를 채우기 위한 작업이 요구
- 이를 해결하기 위해 `imfill()` 함수를 활용하여, 세포 객체에 대해 채워 넣는 작업을 수행

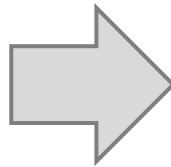


# 2-5 Detection cell in MATLAB

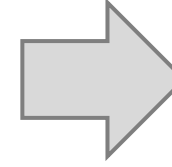
## 불완전 세포 객체 제거

- 01 Infill() 연산 이후, 테두리에 인접한 불완전 세포 객체를 제거
- 02 imclearborder() 함수를 활용하여, 대각 연결 (불완전 세포 객체)를 제거

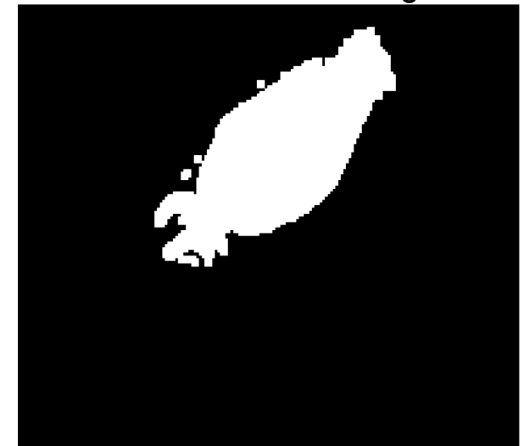
Binary Image with Filled Holes



```
Bwnobord = imclearborder(BWdfill,4);  
imshow(Bwnobord)  
title('Cleared Border Image')
```



Cleared Border Image



〈 불완전 세포 객체 제거 〉



- Imclearborder() 함수를 이용하여, 불완전 세포 객체 제거를 수행
- 그러나, 완전한 세포 객체의 테두리의 굴곡 및 불완전하므로 이에 대한 Smoothing 작업이 요구됨

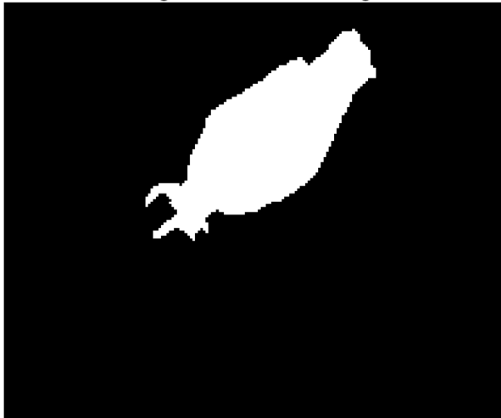


# 2-6 Detection cell in MATLAB

## 객체 평활화 및 시각화

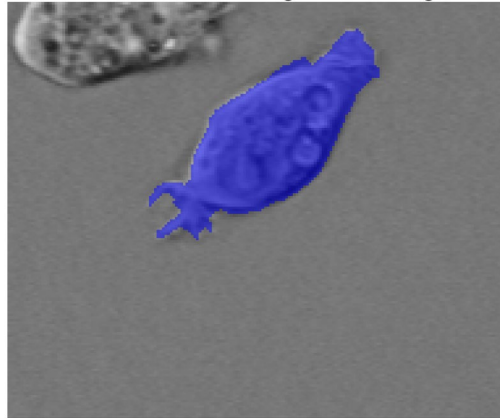
- 01 완전한 세포 객체만 남은 이미지를 생성하였으나, 해당 객체의 외각의 형태를 다듬는 과정이 요구됨
- 02 분할된 객체의 테두리에 대한 평활화 (침식) 작업을 수행한 뒤, Mask 및 외각 선 검출을 통한 결과 출력

Segmented Image



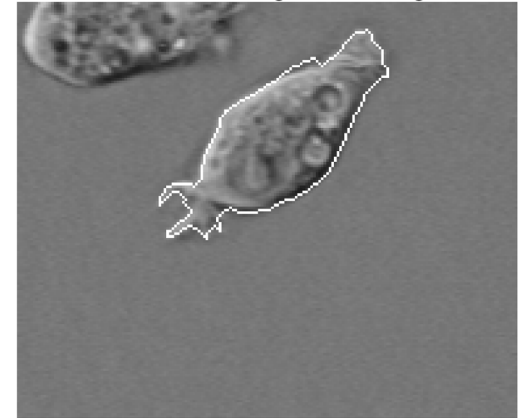
```
seD = strel('diamond',1);  
BWfinal = imerode(BWnobord,seD);  
BWfinal = imerode(BWfinal,seD);  
imshow(BWfinal)  
title('Segmented Image');
```

Mask Over Original Image



```
imshow(labeloverlay(I,BWfinal))  
title('Mask Over Original Image')
```

Outlined Original Image



```
BWoutline = bwperim(BWfinal);  
Segout = I;  
Segout(BWoutline) = 255;  
imshow(Segout)  
title('Outlined Original Image')
```



- 외각 검출 및 기본 모폴로지 (팽창, 침식) 연산을 통해, 입력된 이미지 내 완전한 세포 검출을 달성할 수 있음
- 이때, 불완전 세포 객체가 테두리에 위치한 것이 아닌, 완전한 객체와 인접했을 때의 제거에 대한 추가적인 학습이 필요함

# 3-1 Next Session

## 다음 발표 내용

- “Types of Morphological Operations” on page 10-2
- “Structuring Elements” on page 10-9
- “Border Padding for Morphology” on page 10-13
- “Morphological Reconstruction” on page 10-14
- “Find Image Peaks and Valleys” on page 10-21
- “Pixel Connectivity” on page 10-27
- “Lookup Table Operations” on page 10-30
- “Dilate an Image to Enlarge a Shape” on page 10-32
- “Remove Thin Lines Using Erosion” on page 10-36
- “Use Morphological Opening to Extract Large Image Features” on page 10-38
- “Flood-Fill Operations” on page 10-42
- “Detect Cell Using Edge Detection and Morphology” on page 10-45
- “Granulometry of Snowflakes” on page 10-50
- “Distance Transform of a Binary Image” on page 10-55
- “Label and Measure Connected Components in a Binary Image” on page 10-57

5차 발표 범위



PRESENTATION END

**THANK YOU**





# Chapter 10.

## Morphological Operations (형태학적연산) v5

오영우

조선대학교 컴퓨터공학과 석사과정



# CONTENTS

**01** Granulometry of Snowflakes  
눈송이의 입도 측정

**02** Performance analysis  
성능 분석

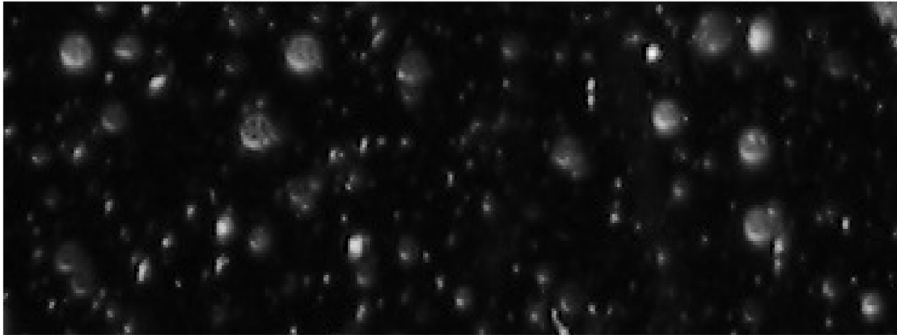
**03** Conclusion  
결론



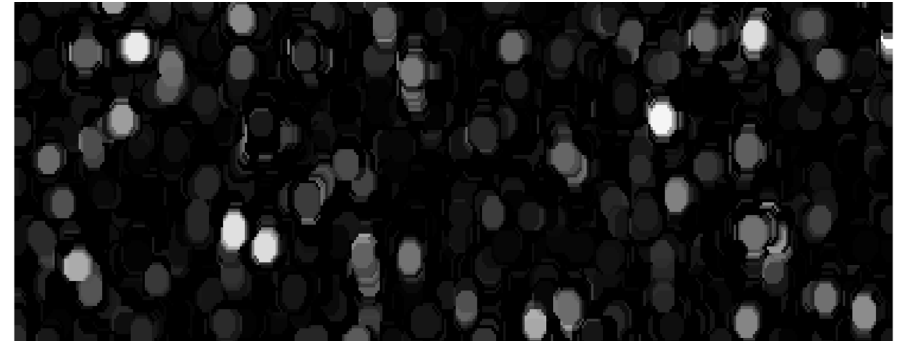
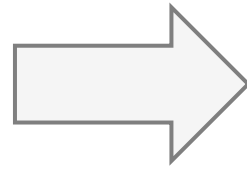
# 1-1 Granulometry of Snowflakes

## 눈송이의 입도 측정

- 01 '입도'는 특정 입자의 크기 분포의 정도를 의미. 따라서, 큰 입자가 많이 존재하는 경우, '입도'가 높다고 말할 수 있음
- 02 주요 목표는 모든 눈송이 객체를 전부 탐지하여 분포를 계산하는 것이 아닌, 단일 이미지를 통해 모든 입자 크기의 분포를 측정하는 것  
\* 모든 눈송이 객체를 판별하고, 이에 대한 크기 분포를 측정하는 것은 많은 계산량이 요구됨



〈 입력된 눈송이 이미지 〉



〈 눈송이 입도 측정에 의해 처리된 이미지 결과 〉



- 이미지 내 존재하는 모든 눈송이 객체를 탐지하여, 하나하나 입자 크기의 분포를 측정하는 것은 많은 복잡도를 요구함
- 해당 예제는 계산량을 줄임과 동시에 단일 이미지만으로 모든 입자 크기의 분포를 측정하는 것을 목표로 함



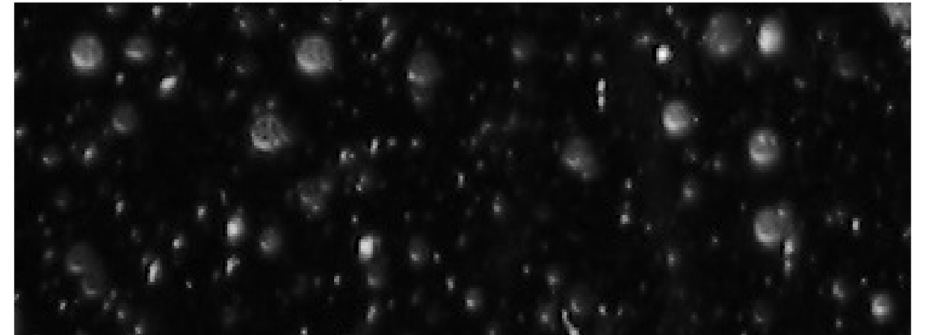
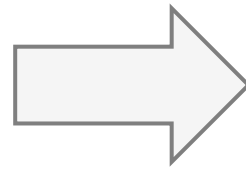
# 1-2 Granulometry of Snowflakes

## 눈송이 이미지 호출

- 01 imread(), imshow(), 함수를 통해 입력 이미지를 호출 및 표시
- 02 호출된 이미지의 눈송이 객체는 명확하지 않으므로, 이미지의 대비 향상 및 강도를 조정해야 함

```
I = imread("snowflakes.png");  
imshow(I)  
title('Input date: snowflakes')
```

〈 입력 이미지 호출 및 표시 〉



〈 호출된 눈송이 이미지 〉



- 호출된 이미지들의 경우, 눈송이 객체가 화질에 의해 명확하지 않음
- 이를 처리하기 앞서, 입력된 이미지의 대비 및 강도를 조정하여 입력된 눈송이 이미지를 처리해야 함



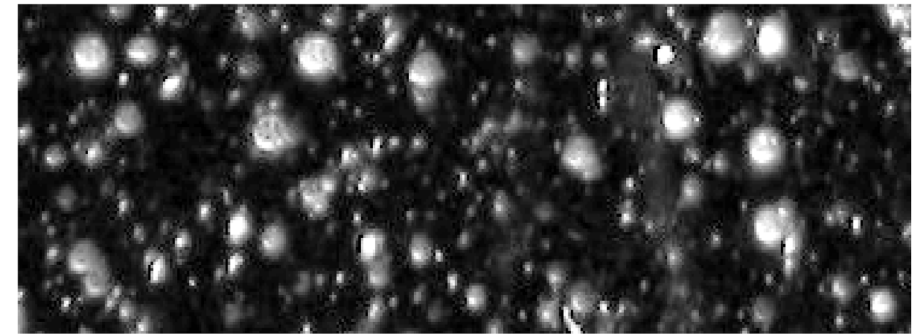
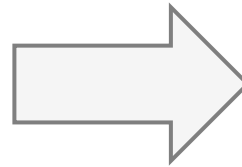
# 1-3 Granulometry of Snowflakes

## 이미지 대비 최대화

- 01 `Adapthisteq()` 함수를 이용하여 입력된 이미지의 히스토그램을 균등화 처리
- 02 균등화된 이미지에 `imadjust()` 함수를 이용하여 이미지의 대비 및 강도를 재조정

\* `Adapthisteq()` 함수는 '대비 제한 적응 히스토그램 균등화' 함수를 의미

```
claheI = adapthisteq(I,"NumTiles",[10 10]);  
claheI = imadjust(claheI);  
imshow(claheI)  
title('Enhanced contrast')
```



〈 이미지 대비 및 강도 조정을 위한 코드 〉

〈 대비 조정된 눈송이 이미지 〉



- 입력된 눈송이 이미지의 대비 및 강도를 최대화하기 위한 작업이 요구 됨
- 최초 입력된 눈송이 이미지 대비 높은 강도로 처리된 것을 확인할 수 있음



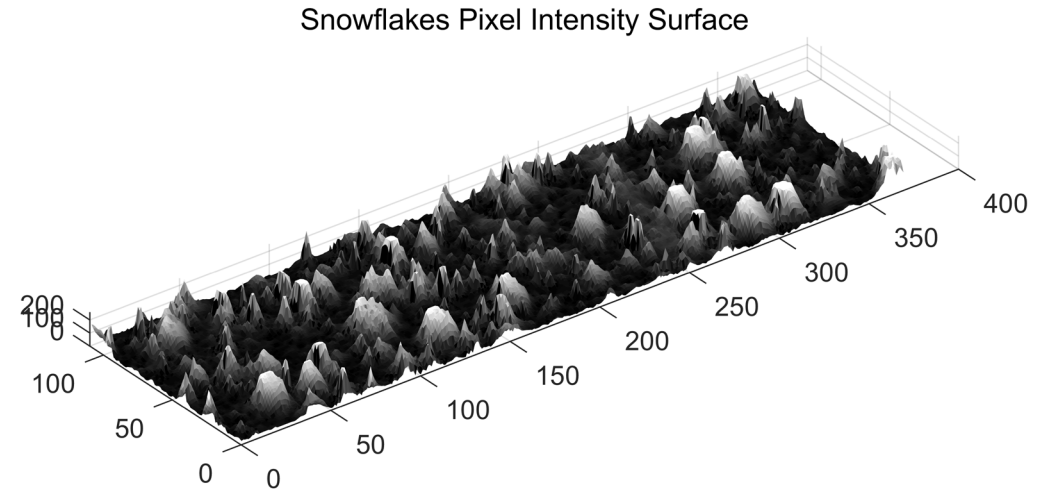
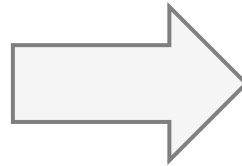
# 1-4 Granulometry of Snowflakes

## 이미지 강도를 활용한 부피 결정 (1)

- 01 모폴로지 연산에 의해 얻어지는 각 이미지 픽셀 값의 합을 통해, 이미지 표면의 강도가 계산 됨
- 02 눈송이 이미지의 표면을 강도로 표현하여, 각각의 눈송이 입자에 대한 표면적을 개념화할 수 있음

```
figure
surf(claheI, EdgeColor="none")
colormap("gray")
title("Snowflakes Pixel Intensity Surface")
daspect([1 1 15]);
```

< 이미지 강도를 활용한 Pixel 표면 부피 출력 코드 >



- 전처리된 눈송이 이미지에 대해 모폴로지 연산을 취하여, 각 이미지 내 존재하는 눈송이의 표면 강도를 계산
- 눈송이 이미지를 표면 강도로 표현하게 되면, 각각의 눈송이 입자에 대한 표면적을 개념화할 수 있음

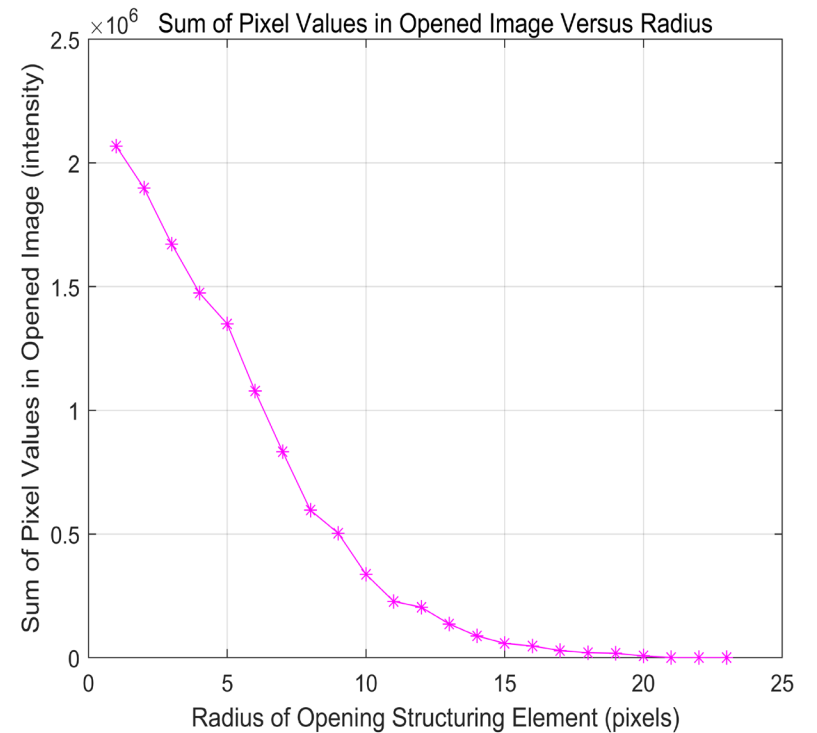


# 1-5 Granulometry of Snowflakes

## 이미지 강도를 활용한 부피 결정 (2)

- 01 구조화 요소 커널의 크기가 늘어날 때의 표면 부피가 0이 되도록 카운트의 제약을 조건화
- 02 구조화 요소에 의해 제거만큼 표면 부피가 축소되는 것을 활용하여 눈송이 객체들에 대한 부피를 결정

```
radius_range = 0:22;
intensity_volume = zeros(size(radius_range));
for counter = radius_range
    remain = imopen(claheI, strel("disk", counter));
    intensity_volume(counter + 1) = sum(remain(:));
end
figure
plot(intensity_volume, "m - *")
grid on
title("Sum of Pixel Values in Opened Image Versus Radius")
xlabel("Radius of Opening Structuring Element (pixels)")
ylabel("Sum of Pixel Values in Opened Image (intensity)")
```



- 구조화 요소 커널이 점진적으로 증가하며, 눈송이 이미지의 표면을 계산하게 됨
- 이때, 구조화 요소 커널에 의해 제거된 만큼의 부피를 계산할 수 있게 됨으로 큰 구조화 요소 커널에 제거된 영역일 수록 표면 부피가 감소하게 됨



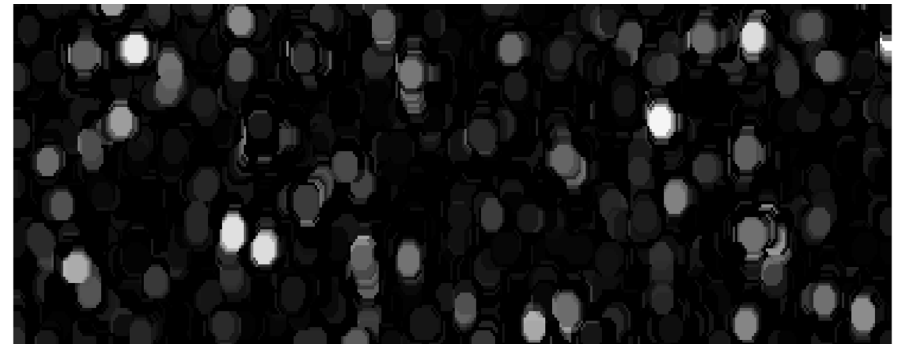
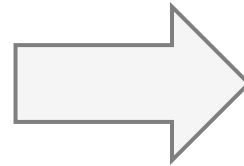
# 1-6 Detection cell in MATLAB

## 특정 입도 이상을 지닌 눈송이 강조

- 01 MATLAB에서 제공하는 도함수 기능의 Diff()를 이용하여, 눈송이 입도의 최소 및 최대 반지름을 획득
- 02 획득된 최소 반지름 (입도) 이상의 눈송이 객체를 강조 표시

```
open5 = imopen(cIaheI, strel("disk", 5));  
open6 = imopen(cIaheI, strel("disk", 6));  
rad5 = open5 - open6;  
imshow(rad5, [])  
title('Emphasize snowflakes having a particular radius')
```

〈 최소 반지름(입도)를 이용한 눈송이 객체 강조 코드 〉



〈 특정 입도 이상을 지닌 눈송이 이미지 처리 결과 〉



- 도함수를 활용하여, 눈송이의 입도를 계산하고 이에 따른 최소 입도 및 최대 입도를 지닌 눈송이 객체를 탐지
- 최소 입도 이상의 입도를 지는 눈송이를 강조하는 작업을 수행

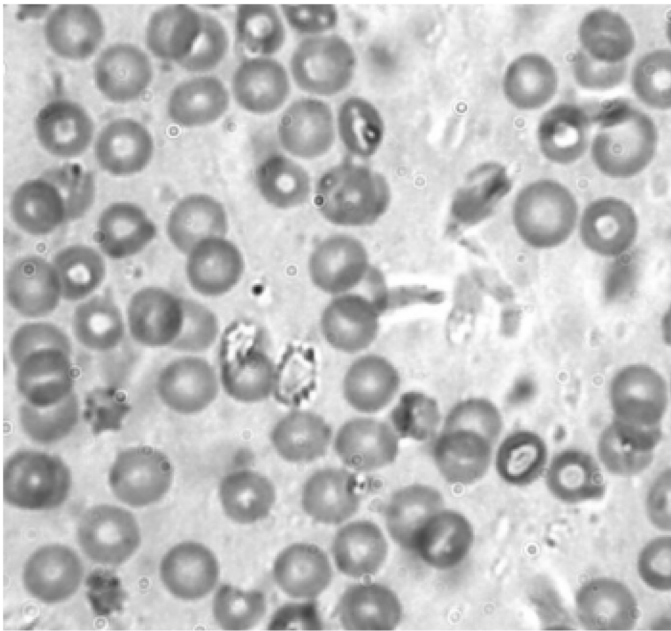




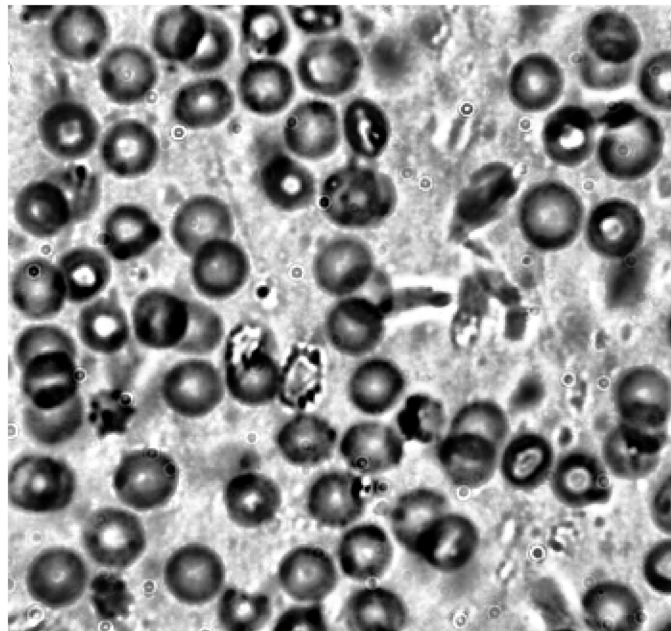
# 2-1 Performance analysis

## 세포 이미지에 대한 입도 분석 (1)

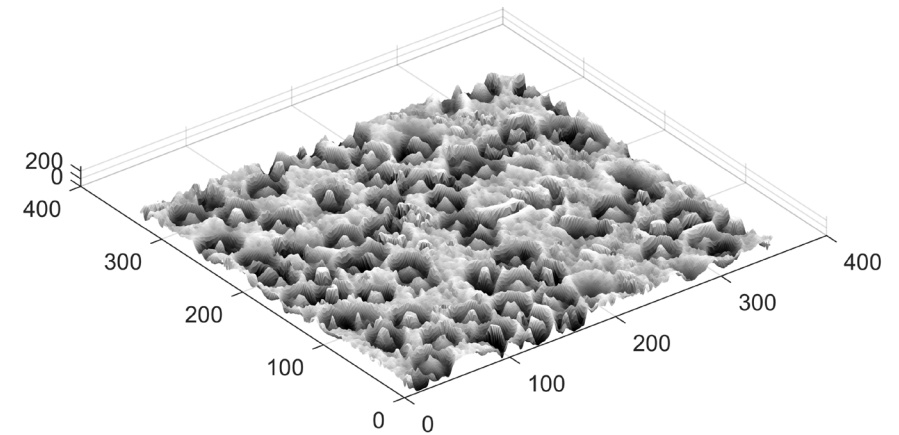
- 01 구글에 검색되는 세포(cell) 이미지에 대한 입도 측정 및 분석을 시행
- 02 RGB 값을 포함한 3차원 이미지를 1차원으로 변환하여 입력 이미지 데이터를 생성



〈 입도 분석을 위한 세포 이미지 〉



〈 대비 및 강도가 재적용된 세포 이미지 〉



〈 이미지 강도를 활용한 Pixel 표면 부피 출력 〉



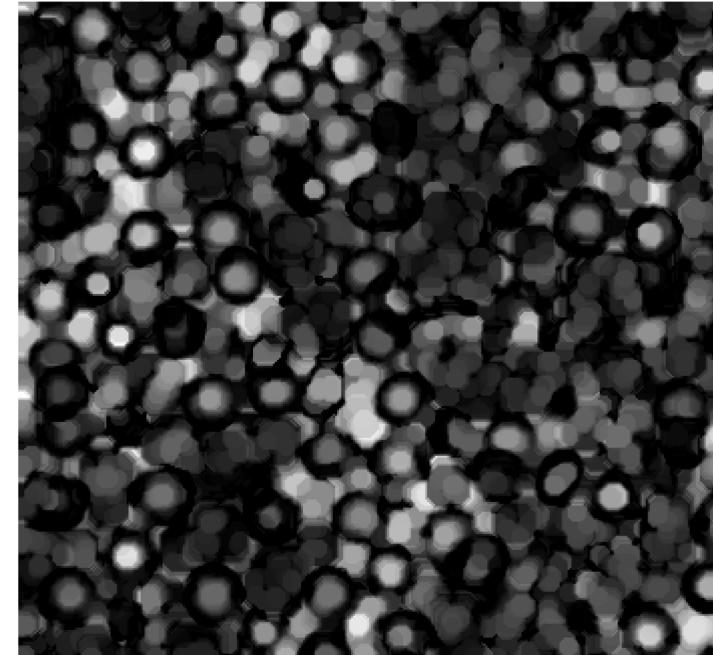
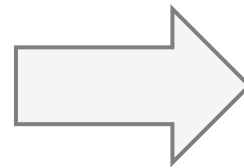
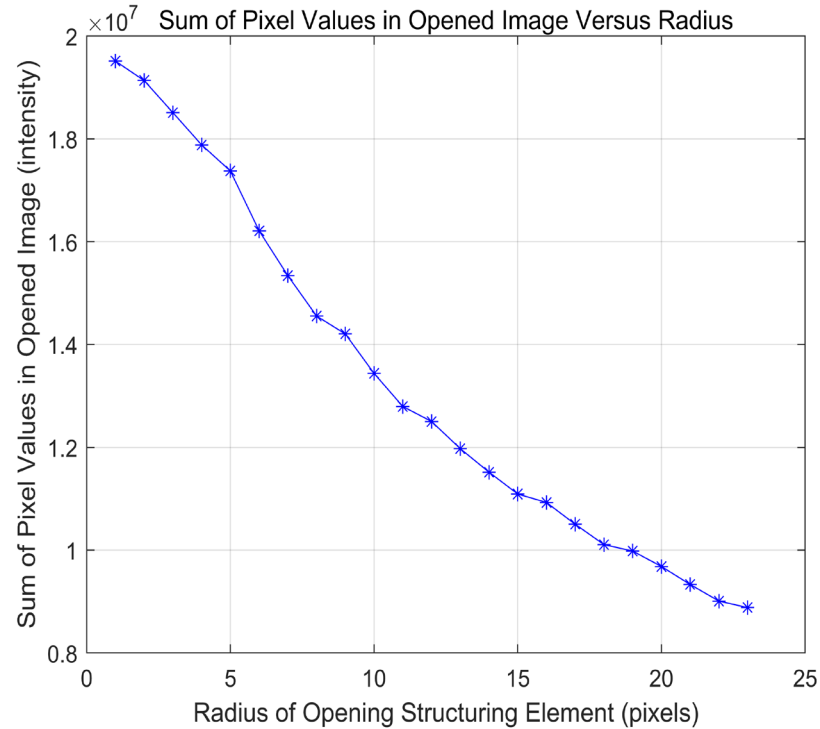
- 실험결과를 통해, 눈송이 및 세포 이미지 모두에 적용이 가능한 것을 확인할 수 있음
- 이때, 세포 이미지의 경우 RGB 값을 포함함으로 1차원으로 처리하기 위한 `rgb2gray()` 함수를 활용



# 2-2 Performance analysis

## 세포 이미지에 대한 입도 분석 (2)

01 눈송이의 입도 측정 예제는 세포 이미지에 동일하게 적용될 수 있음을 확인



- 눈송이 예제에서 활용한 입도 측정기법은 세포 및 타 현미경으로 관찰된 소형 이미지 모두에 적용 가능
- 이때, 도함수를 활용한 최대 및 최소 입도에 따라 강조되는 결과가 다소 상이하게 출력되는 경향을 보임



PRESENTATION END

**THANK YOU**