

U-Net: Convolutional Networks for Biomedical Image Segmentation

Olaf Ronneberger Philipp Fischer and Thomas Brox

김성민

CONTENTS

01 FCN

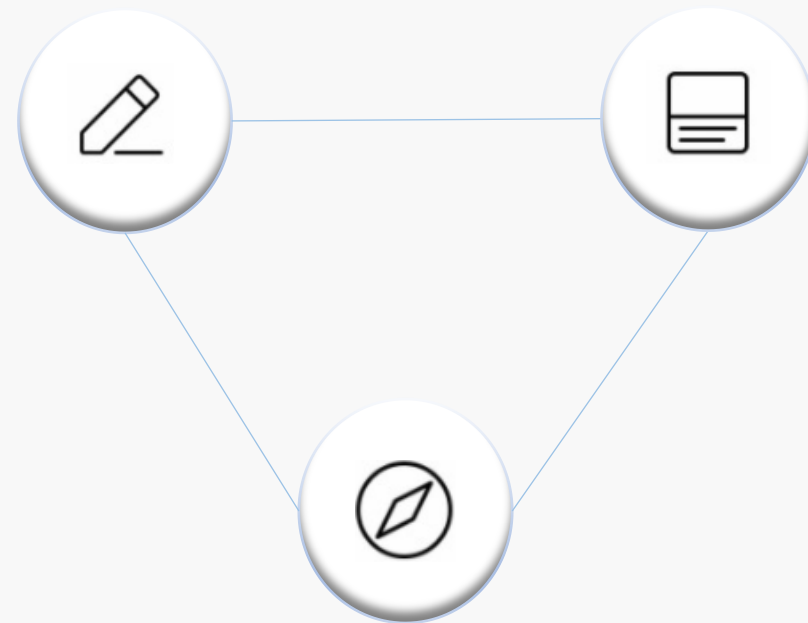
02 Introduce

03 Architecture

04 Training

05 Experiment

06 Conclusion



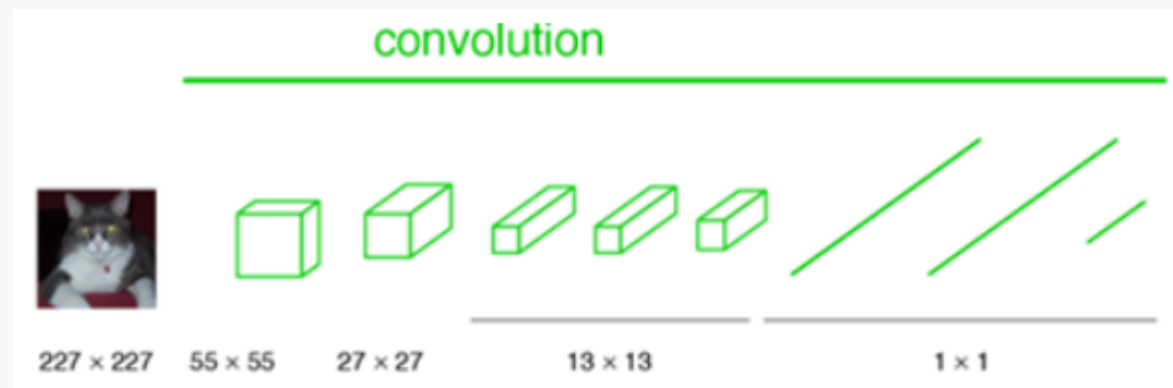
CONTENTS

01 FCN



Image Segmentation

01 FCN

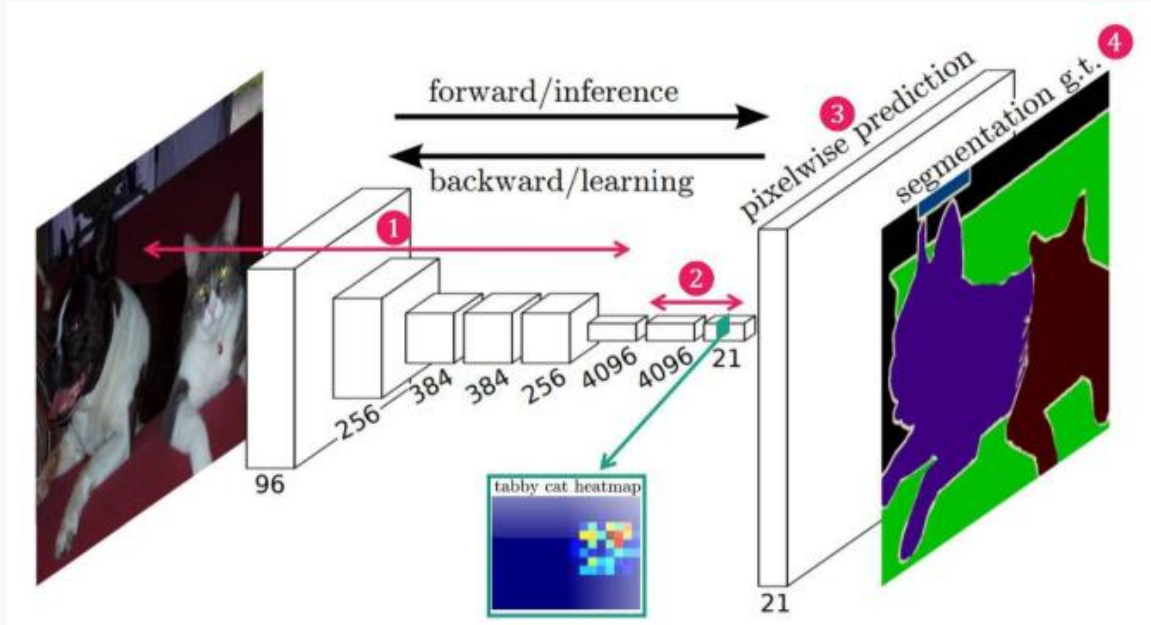


FCN은 분류에서 성능을 검증된 기존의 네트워크(AlexNet, VGGNet, GoogLeNet)를 이용한다.

하지만 Fully Connected Layer는 픽셀의 위치 정보를 사라지게 해서 segmentation에 적합하지 않다.

→ 마지막 FCL들을 1x1 Convolution Layer로 교체 : 위치 정보 등 2차원 정보의 손실을 줄임

01 FCN



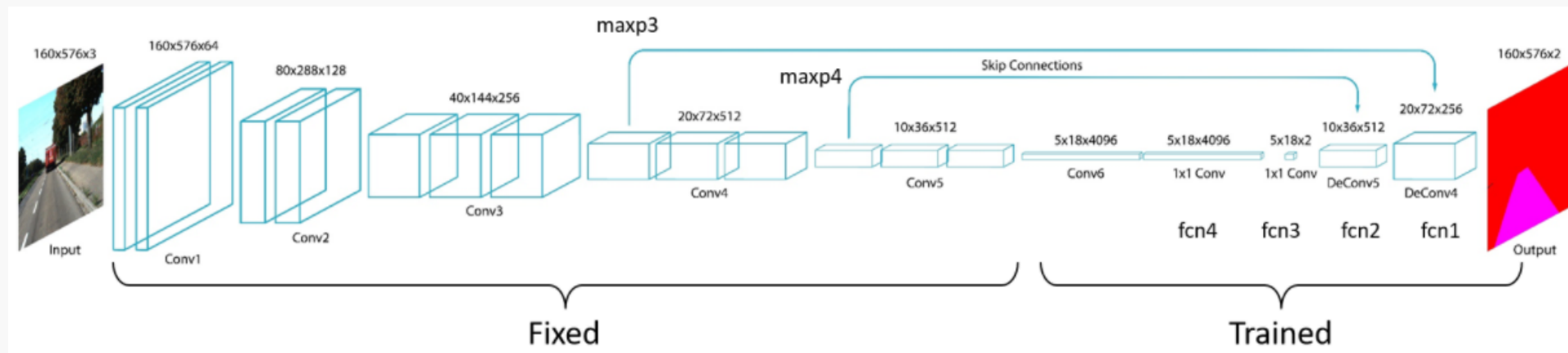
Convolution 층들을 거치고 나서 얻게 된 heatmap은 대략적인 정보(Class의 스코어와 위치 정보)만 가지고 있으며 그 수는 훈련된 Class의 수와 동일하다.

고양이 Class에 대한 heatmap이면 고양이가 있는 위치의 픽셀 값이 높고 강아지 Class에 대한 heatmap이면 강아지가 있는 위치의 픽셀 값이 높다.

대략적인 heatmap들의 크기를 원래 이미지의 크기로 복원할 필요가 있다.

01 FCN

이미지 크기를 복원하기 위해 CAE와 같은 구조가 사용되며 이미지 크기를 복원하는 Up-Sampling 시 transposed convolution layer를 사용



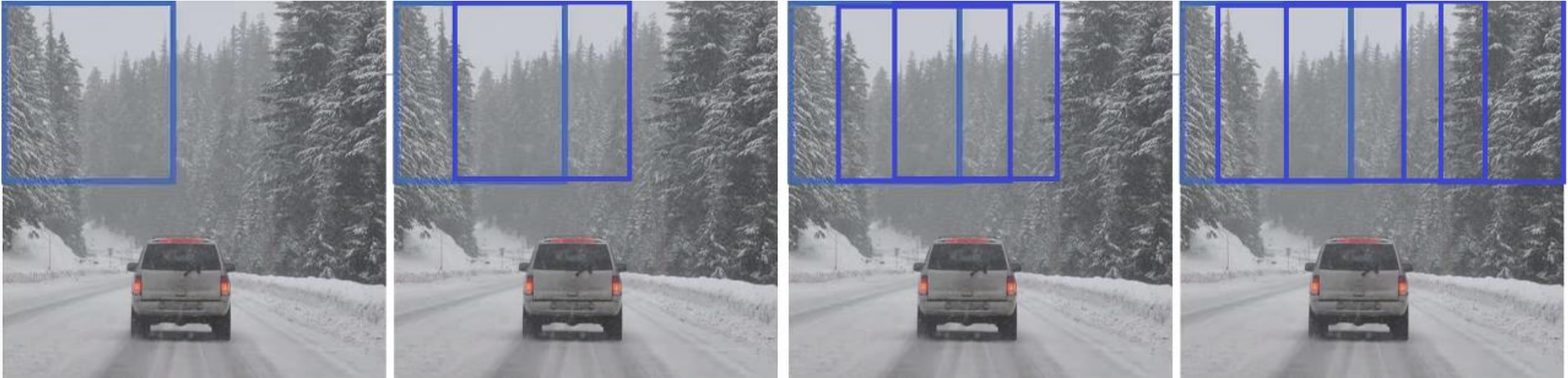
CAE와 다른 점은 Skip Connection을 이용해 공간 정보를 전달한다는 것이다.

02 Introduce

논문에서 많이 사용되는 용어들

1. Patch : 이미지 인식 단위
2. Context : 이웃한 픽셀들 간의 관계, 이미지의 문맥
3. Localization : 이미지에서 각각의 픽셀 자체의 정보

02 Introduce

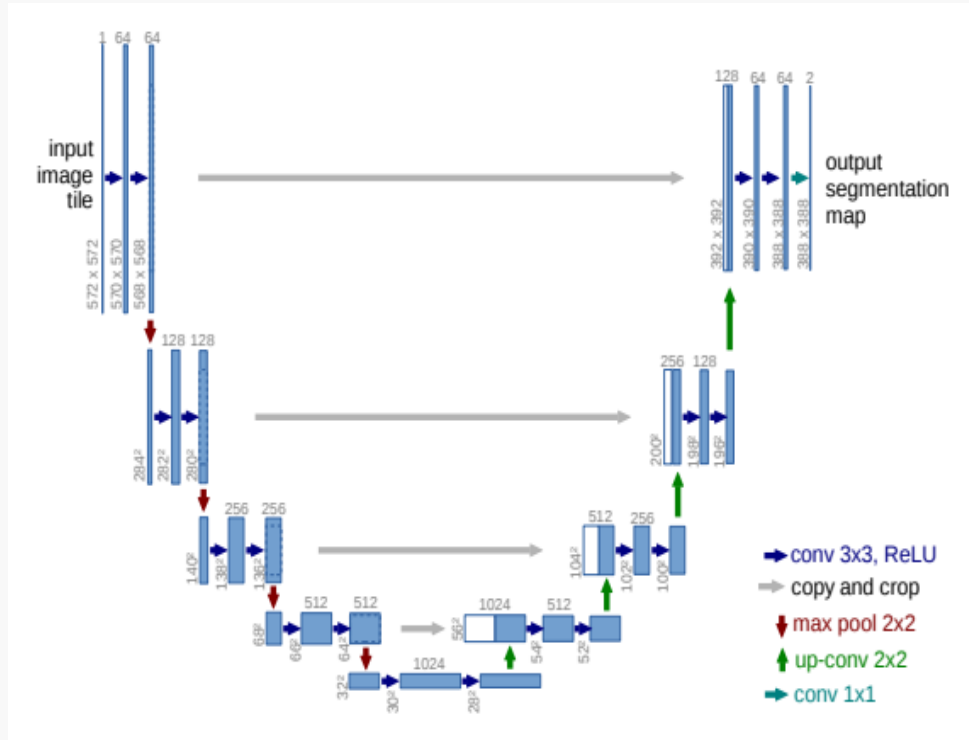


Sliding window

Patch에 겹치는 부분이 많아 학습 속도가 매우 느리며, trading-off (context, localization) 문제가 존재

02 Introduce

U-Net



1. FCN을 조금 변형해서 U자형의 구조로 네트워크 생성
2. FCN과 비교해서 Upsampling(오른쪽 부분)과 Concatenate에서 차이가 난다.
3. Fully connected layer가 없어서 Segmentation 맵을 얻을 수 있다.
4. 출력 이미지가 입력 이미지보다 줄어드는 현상 발생 (overlap-tile strategy 사용)
5. 적은 학습 이미지로도 학습이 가능하도록 Data Augmentation 사용

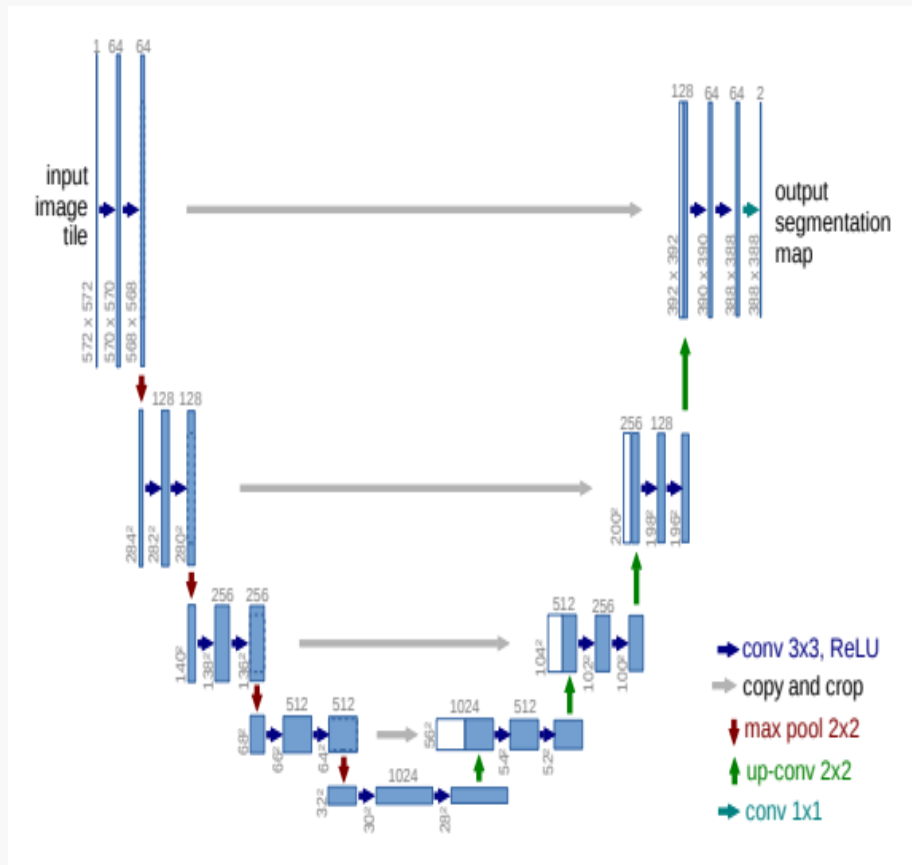
03 Network Architecture

Contracting path

1. 전형적인 CNN 구조 (without fully connected layer)
2. 두 번의 3x3 컨볼루션 (unpadded) 후 ReLU 사용
3. 한 번의 2x2 max 풀링 (stride 2)
4. 다운 샘플링 시 feature channel이 2배로 늘어난다.
5. 이미지 사이즈는 줄어듦

Expanding path

1. Max 풀링 레이어 대신 UP Conv. Layer 사용
2. 두 번의 컨볼루션 후 ReLU 사용
3. Contraction Path에서 추출된 feature map을 가져와 사용
4. 마지막에 1x1 컨볼루션으로 원하는 채널 수로 줄임
5. 이미지 사이즈가 커지고 채널 수가 반씩 감소



04 Training

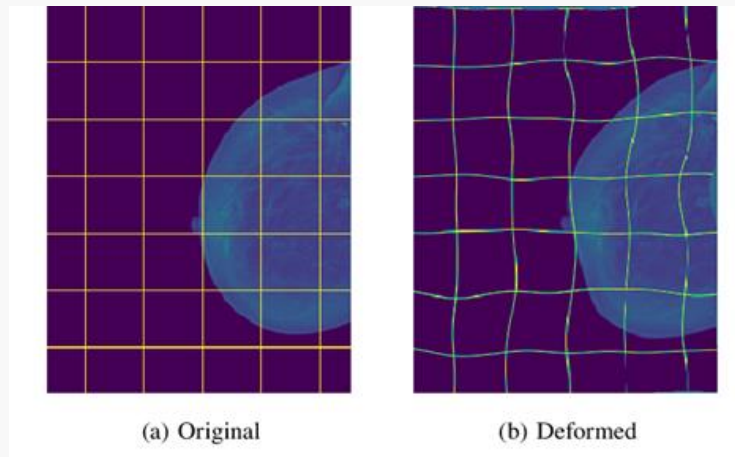
Patch 사이즈를 크게, 배치 사이즈는 줄여 GPU 효율 UP

배치 사이즈가 작으면 학습에 있어 최적화가 잘 안되는 문제 발생

→ 모멘텀을 0.99로 크게 설정해서 이전 데이터들의 값이 더 많이 반영되도록 함

적은 이미지 수를 보완하기 위한 Data Augmentation

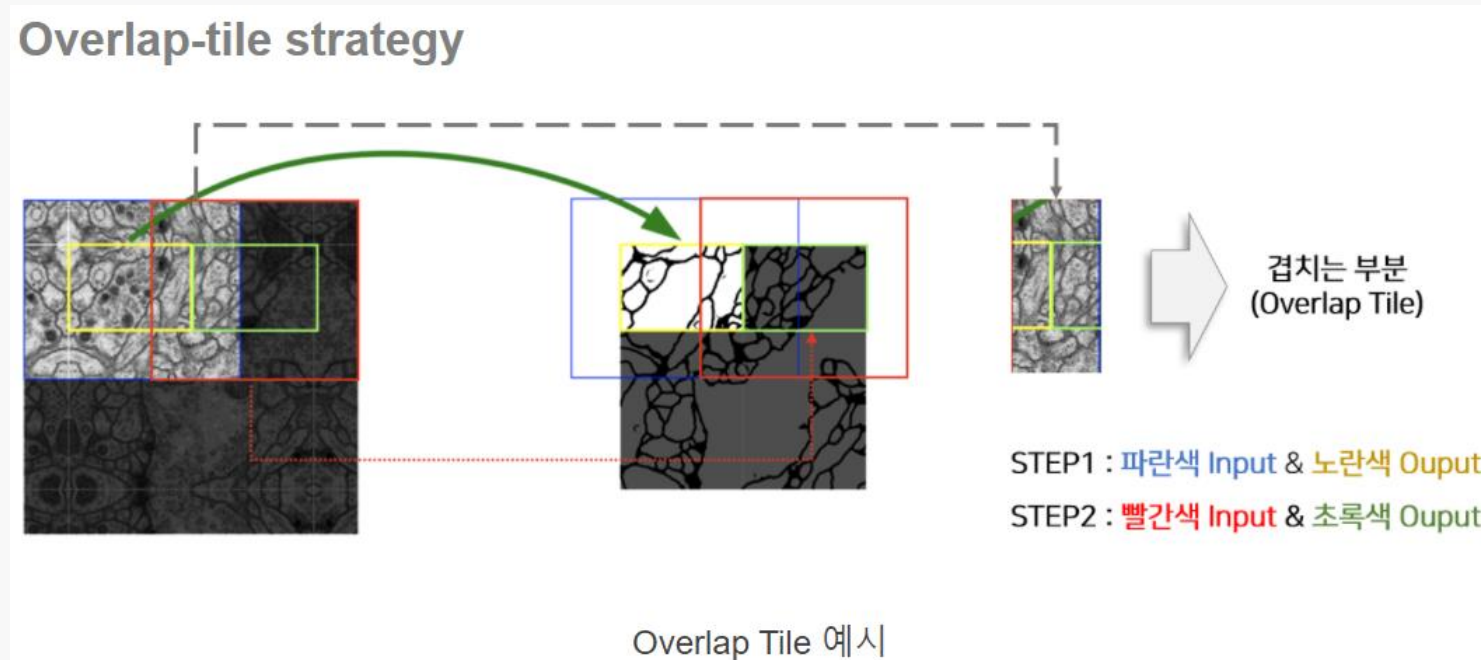
기존의 방법이 아닌 Elastic Deformation 사용 (랜덤 변위 벡터 생성, 2중 보간 이용)



04 Training

Overlap-tile strategy

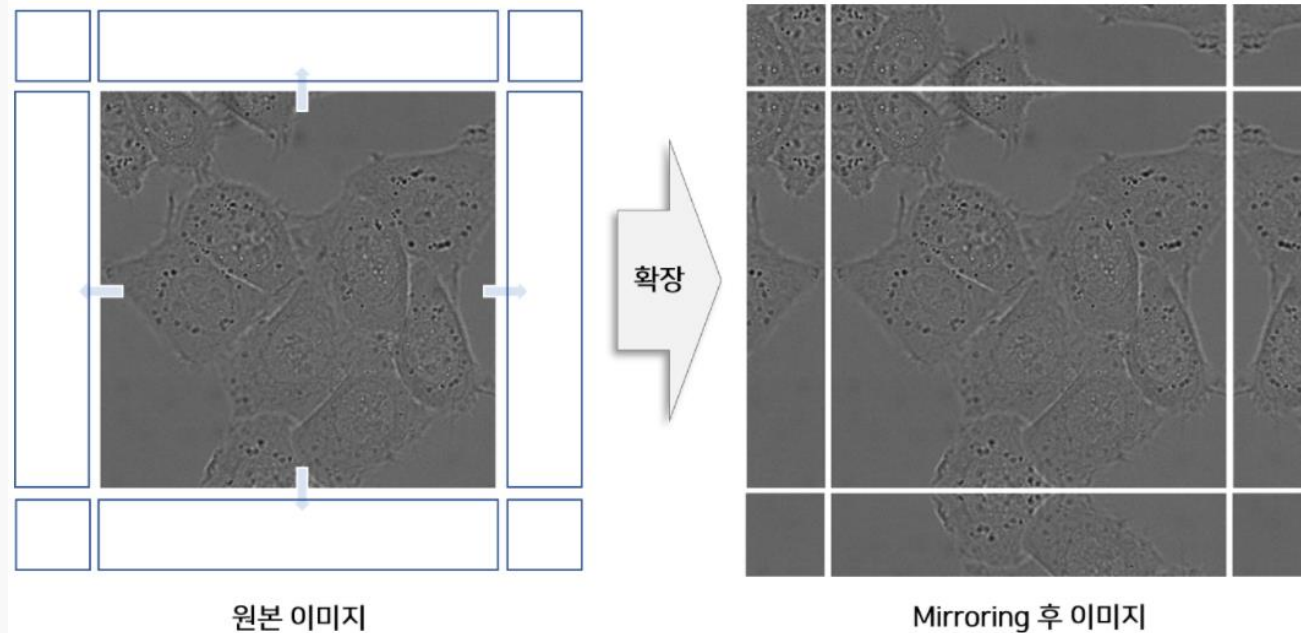
Convolution layer에서 패딩을 하지 않아 출력 이미지의 크기가 입력 이미지의 크기보다 작아지는 발생한다. 이를 해결하기 위해 Overlap-tile strategy를 사용한다.



겹치는 부분이 존재하도록 이미지를 자르고 Segmentation하기 때문에 Overlap-strategy라고 논문에서 지칭한다.

04 Training

Mirroring Extrapolate



이미지의 경계에 위치한 이미지를 복사하고 좌우 반전을 통해 Mirror 이미지를 생성한 후 원본 이미지의 주변에 붙여 Input으로 사용한다.

이미지 출처 (전 페이지와 본 페이지 이미지) : <https://joungeekim.github.io/2020/09/28/paper-review/#:~:text=Overlap%2Dtile%20strategy%20%3A%20%ED%81%B0%20%EC%9D%B4%EB%AF%B8%EC%A7%80,%EC%9D%98%20Input%EC%9C%BC%EB%A1%9C%20%ED%99%9C%EC%9A%A9%ED%95%A9%EB%8B%88%EB%8B%A4.&text=Data%20Augmentation%20%3A%20%EC%A0%81%EC%9D%80%20%EB%8D%B0%EC%9D%B4%ED%84%B0%EB%A1%9C,%EC%A6%9D%EA%B0%95%20%EB%B0%A9%EB%B2%95%EC%9D%84%20%ED%99%9C%EC%9A%A9%ED%95%A9%EB%8B%88%EB%8B%A4.>

04 Training

Soft-max:

$$p_k(x) = \exp(a_k(x)) / \sum_{k'=1}^K \exp(a_{k'}(x))$$

특정 채널 k 에서 픽셀 x 의 소프트 맥스를 구하고 이를 이용해 교차 엔트로피 손실 함수를 구한다.

Cross-Entropy loss function:

$$E = - \sum_{x \in \Omega} w(x) \log(p_{l(x)}(x))$$

여기서 $w(x)$ 가 추가된다. $w(x)$ 는 학습 시 경계에 해당하는 픽셀을 잘 학습하게 한다.

04 Training

$$w(x) = w_c(x) + w_0 \exp\left(-\frac{(d_1(x) + d_2(x))^2}{2\sigma^2}\right)$$

해당 픽셀 위치에 해당하는 클래스의 빈도에 따라 결정

픽셀로부터 가까운 cell border의 거리로 구할 수 있음
→ 픽셀 위치에 따라 가중치를 달리 부여한다는 의도

$W(x)$ 는 픽셀 x 와 경계의 거리가 가까우면 큰 값을 가지게 된다. 따라서 해당 픽셀의 Loss 비중이 커지게 되어 학습 시 경계에 해당하는 픽셀을 잘 학습하게 된다.

05 Experiment

1. EM segmentation Challenge

학습 데이터는 30개의 이미지 세트이며 512x512 픽셀로 이루어져 있고, 주석이 달린 segmentation 맵(세포와 세포막)이 제공된다.

Table 1. Ranking on the EM segmentation challenge [14] (march 6th, 2015), sorted by warping error.

Rank	Group name	Warping Error	Rand Error	Pixel Error
	** human values **	0.000005	0.0021	0.0010
1.	u-net	0.000353	0.0382	0.0611
2.	DIVE-SCI	0.000355	0.0305	0.0584
3.	IDSIA [1]	0.000420	0.0504	0.0613
4.	DIVE	0.000430	0.0545	0.0582
⋮				
10.	IDSIA-SCI	0.000653	0.0189	0.1027

Warping Error

픽셀 보다는 객체에 초점을 맞추어 객체 간의 위상 오류를 측정

Rand Error

두 데이터 클러스터링 간의 유사성을 나타내는 척도이다.

Pixel Error

모델이 예측한 픽셀의 라벨이 정답 데이터로 제공한 라벨과 일치 여부와 관련된 에러

<https://ashm8206.github.io/2018/04/08/Segmentation-Metrics.html>

05 Experiment

2. ISBI cell tracking challenge : PhC-U373

35개의 부분적 주석이 달린 학습 이미지 세트

3. ISBI cell tracking challenge : DIC-HeLa data set

20개의 부분적 주석이 달린 학습 이미지 세트

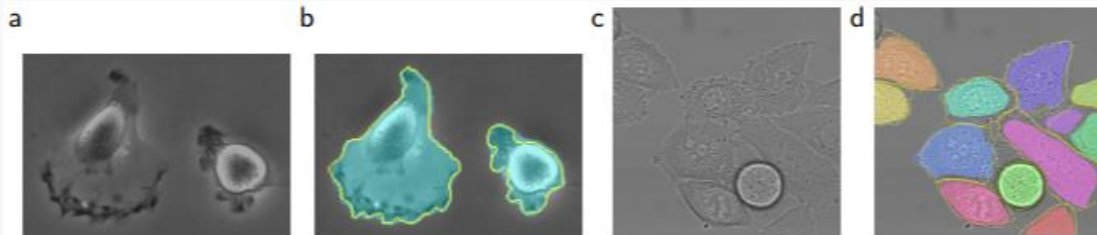


Fig. 4. Result on the ISBI cell tracking challenge. (a) part of an input image of the “PhC-U373” data set. (b) Segmentation result (cyan mask) with manual ground truth (yellow border) (c) input image of the “DIC-HeLa” data set. (d) Segmentation result (random colored masks) with manual ground truth (yellow border).

Table 2. Segmentation results (IOU) on the ISBI cell tracking challenge 2015.

Name	PhC-U373	DIC-HeLa
IMCB-SG (2014)	0.2669	0.2935
KTH-SE (2014)	0.7953	0.4607
HOUS-US (2014)	0.5323	-
second-best 2015	0.83	0.46
u-net (2015)	0.9203	0.7756

06 Conclusion

U-Net은 서로 다른 Biomedical segmentation 과제를 해결하는 데 있어 좋은 성능을 보여주었다.

Elastic Augmentation 덕분에 U-Net은 아주 적은 이미지로 학습을 진행할 수 있었고, 많은 시간이 필요하지도 않았다.

U-Net이 다른 과제에도 쉽게 적용될 것이라고 확신한다.

THANK YOU -

경청해주셔서 감사합니다.