U-Net 논문 리뷰

윤태우

1. U-Net 이전 모델들의 한계

2012년 AlexNet의 등장 이후 컨볼루션 레이어를 쌓아 이미지를처리하는 방식이 널리 사용되었습니다. 그러나 시간이 지날 수록 모델들의 무게가 무거워지고 그에 따라 더 많은 데이터셋이 필요해지게 되는 상황이 됐습니다.

AlexNet 등의 모델들은 주로 이미지 분류(Classification)하는 것이었습니다.

그러나 Biomedical image 같은 것들은 분류하기가 쉽지 않고, 여러개의 세포가 들어있기 때문에 픽셀별로 클래스 분류를 해야하는, Localization이 포함된 Classification이 필요했습니다.

2. U-Net 이전 모델들의 대안

이 한계들을 극복하기 위해서 나온 대안이 있습니다. Sliding-window 방식으로 훈련된 네트워크가 분류할 클래스를 예측하기 위해 픽셀과 픽셀 주변의 영역을 받아 픽셀에 담긴 정보가 어떤 객체를 나타내는건지 판단하는 방식이죠. 이렇게 되면 학습 데이터가 이미지 단위가 아닌 이미지 속 일부가 됩니다. 이를 patch라고 합니다.

그러나 이렇게 예측하게 되면 다음과 같은 단점이 있습니다.

2. U-Net 이전 모델들의 대안

1. 느립니다.

- 네트워크가 각 patch별로 분리되어 작동해야만 합니다.
- overlapping patch들 때문에 중복되는 것들이 많습니다.

2. localization 정확도와 문맥 사이의 관계 문제.

- patch들이 커질수록 localization 정확도를 낮추기 위해서 더 많은 맥스 풀링 레이어가 필요합니다.

(localization은 세부적인 특징(patch)을 잡아내는데 이 특징을 지나치게 잡는 것을 막기 위해 맥스 풀링 실시)

이 때 많은 max-pooling 레이어가 사용되면 context가 줄어들어버리는 것이죠.

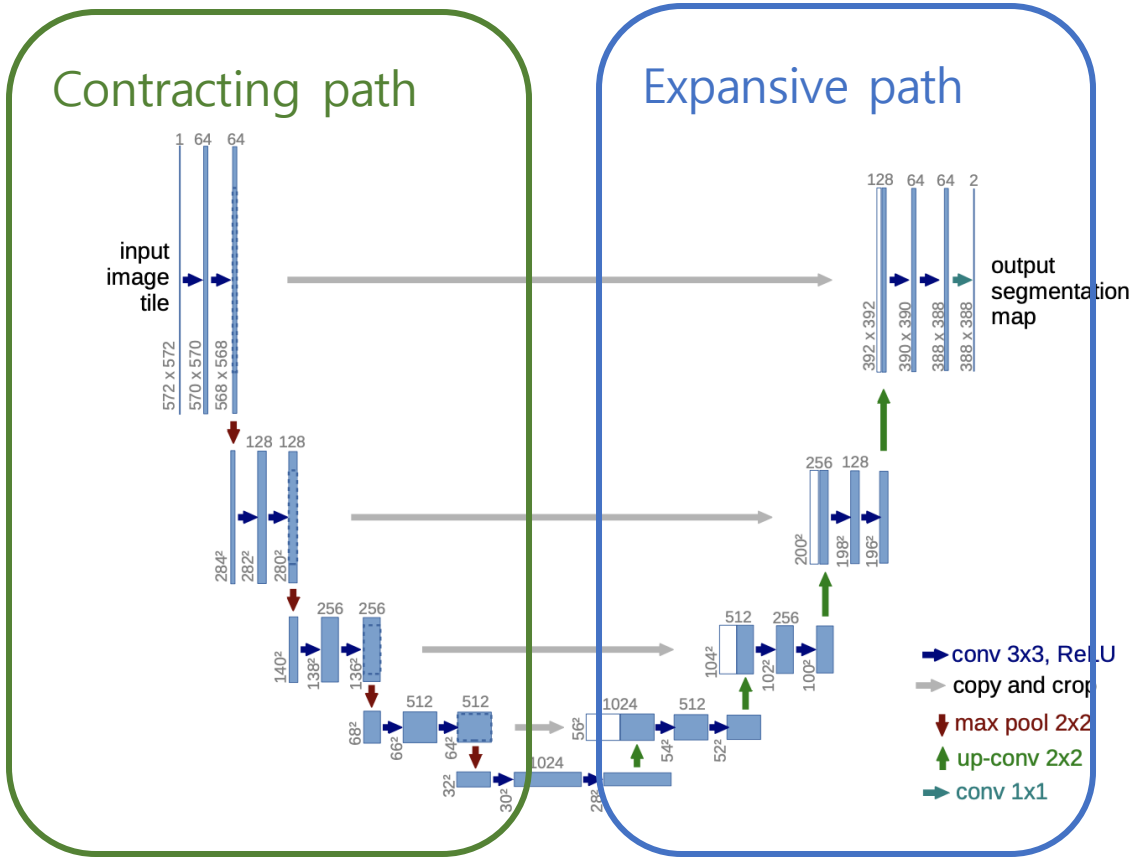
3. U-Net의 특징

1. U-Net은 fully convolutional network 구조로 만들어졌습니다.

2. Patch 되는 과정에서 트레이닝 이미지를 이미지가 아닌 데이터(이미지의 일부)로 변환되기 때문에 트레이닝 이미지의 수보다 많아집니다.
따라서 적은 데이터셋에 학습시켜도 더 좋은 성능이 나오는 모델입니다.

3. U-Net은 특성을 추출하여 점점 해상도가 줄어드는 feature map을 얻다가 upsampling 되는 CNN 레이어를 지납니다. 이 결과로 얻은 같은 해상도의 feature map과 기존의 인풋을 combine 하는 것으로 최종 아웃풋을 도출합니다.

3. U-Net의 특징

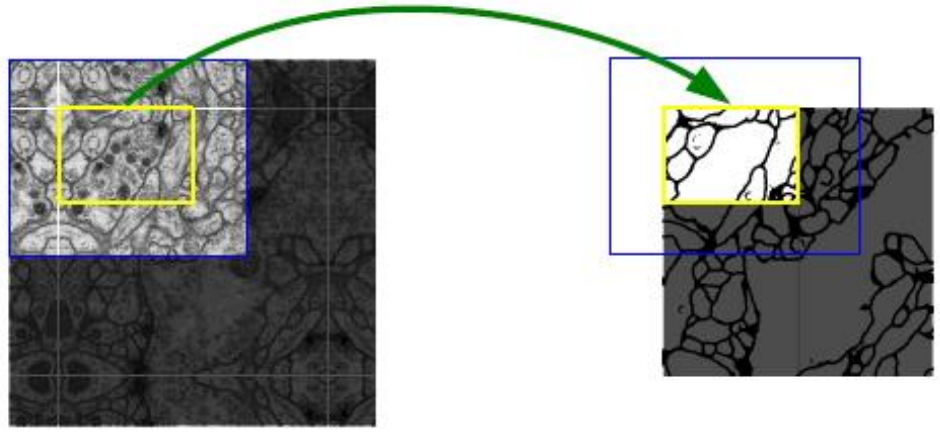


위와 같은 과정을 거치기 때문에 U-Net은 symmetric 하고 U자 모양인 모델 구조가 만들어 집니다. 옆의 그림처럼 말이죠.

4. U-Net

여러 convolution 레이어를 거쳐 upsampling 되고, 각 과정에서 왼편, 오른편의 대칭되는 구조와 combine 하기 때문에 higher resolution layer에 context information을 전파할 수 있습니다.
Fully connected layer(FCN layer)를 사용하지 않는데에다가, 각 convolution layer의 유의미한 부분만 사용 합니다(patch). segmentation map에서는 인풋 이미지에서 full context를 포함한 pixel만을 가지는 것이죠. 때문에 overlap-tile 방식을 사용해 임의의 큰 이미지의 seamless한 segmentation을 가능하게 해줍니다.
그러나 이 방식은 GPU 메모리에 의해 제한되기도 합니다.

노란 박스는 segmentation한 부분이고,
파랑색 박스가 Input이 됩니다.(overlap-tile) ➡



4. U-Net

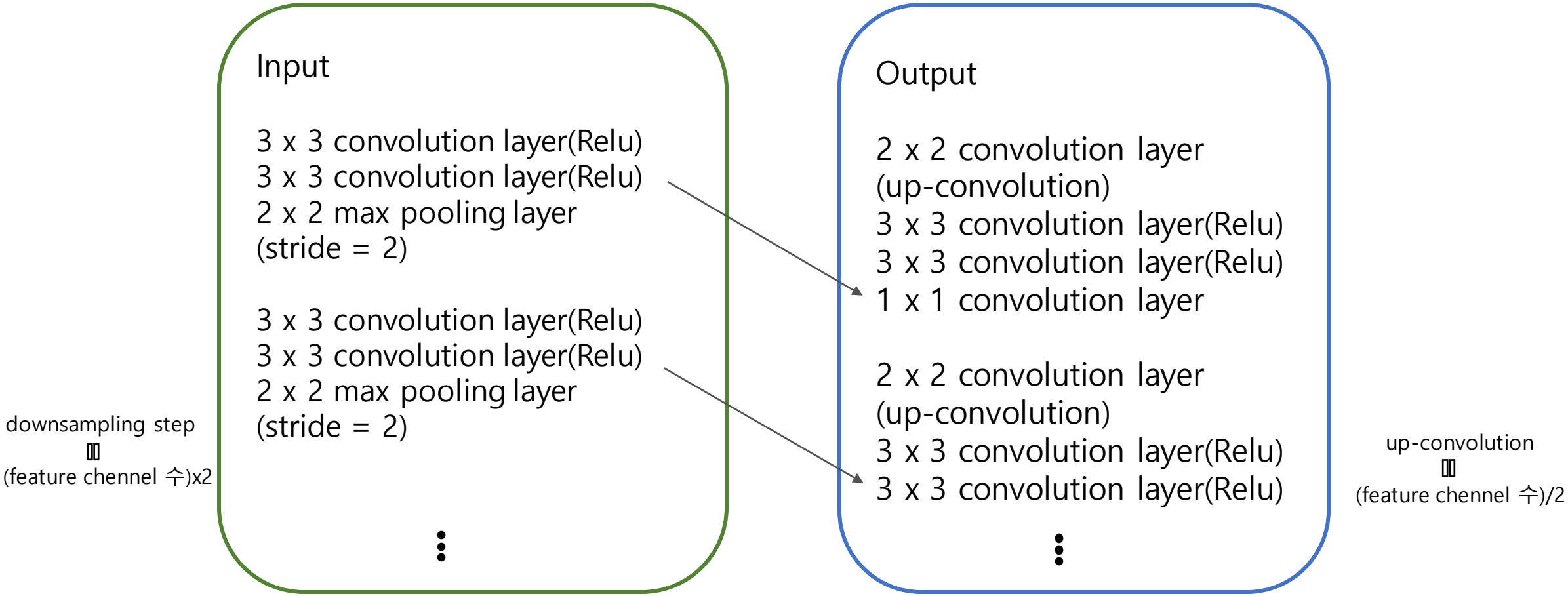
U-Net은 데이터가 매우 적은 경우, 트레이닝 이미지의 탄력적인 변형을 통해 많은 data argumentation을 했습니다. 이러한 Data augmentation는 Biomedical segmentation에서 중요합니다. data augmentation의 효과는 비지도 학습 방법에 대한 Dosovitskiy의 논문에서 자세히 알 수 있습니다.

세포에 대한 segmentation에서는 붙어있는 같은 종류의 세포들을 분류해야 합니다. U-Net은 이 세포 분류에서 가중치를 둔 loss를 사용했습니다. 붙어있는 세포들은 loss function에서 큰 가중치를 얻게됩니다.

Cell segmentation ➡ 

4. U-Net

U-Net은 다음과 같은 구조를 가집니다.



5. Training

Training은 Input 이미지와 이미지들의 대응하는 segmentation map들은 확률적 경사하강법을 이용하여 시행 했습니다. 패딩을 하지 않았기 때문에 최종 output은 input보다 작습니다.

Energy function은 크로스 엔트로피 손실함수로 최종 feature map을 최적화한 것에 pixel별 soft-max한 값입니다.

$$E = \sum_{\mathbf{x}\in\Omega} w(\mathbf{x}) \log(p_{\ell(\mathbf{x})}(\mathbf{x}))$$

- E = Energy function
- $p_{\ell(\mathbf{x})}$ = softmax 한 결과
- $w(\mathbf{x}) = w_c(\mathbf{x}) + w_0 \cdot \exp\left(-\frac{(d_1(\mathbf{x}) + d_2(\mathbf{x}))^2}{2\sigma^2}\right)$

$w_c(\mathbf{x})$: weight map to balance the class frequencies
$d_1(\mathbf{x})$: distance to the border of the nearest cell
$d_2(\mathbf{x})$: distance to the border of the second nearest cell

5. Training

Data augumantation은 데이터 변형, 이동 및 회전, gray value variation을 사용했습니다.
특히 매우 적은 이미지에 대한 elastic deformation이 핵심이 되었습니다.

Data augumentation 외에도 contracting path의 마지막에 drop-out layer를 사용했습니다.

6. Experiments

U-Net은 서로 다른 세가지의 segmentation을 시행했습니다.

첫번째 task는 전자현미경 기록들에 기반한 신경세포의 segmentation 입니다.

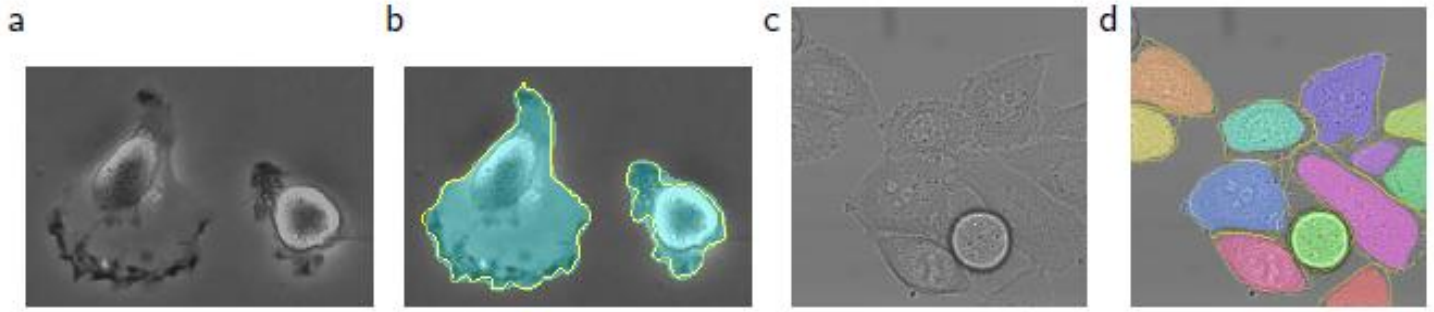
Rank	Group name	Warping Error	Rand Error	Pixel Error
	** human values **	0.000005	0.0021	0.0010
1.	u-net	**0.000353**	0.0382	0.0611
2.	DIVE-SCI	0.000355	0.0305	0.0584
3.	IDSIA [1]	0.000420	0.0504	0.0613
4.	DIVE	0.000430	0.0545	**0.0582**
⋮				
10.	IDSIA-SCI	0.000653	**0.0189**	0.1027

2. 두번째, 세번째 task는 광학현미경의 기록들에 기반한 segmentation 입니다.

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**

평가 : 각 map의 10가지 다른 level에서의 임계값과, warping error, Rand error, pixel error 계산

6. Experiments



a, c가 입력 이미지고 b, d가 ground truth segmentation map입니다.

현미경에서 얻은 이미지로 b, d 처럼 세포를 구별하는 것을 보여줍니다.

U-net는 서로 다른 dataset에 대해 각각

PhC-U373 dataset : 92%

DIC-HeLa dataset : 77.5%

의 IOU(intersection over union)를 얻었습니다.

7. Conclude

U-Net은 다양한 biomedical segmentation applications에서 매우 좋은 성능을 보여줬습니다.
저자는 elastic deformation이 포함된 Data augmentation 덕분에 적은 사이즈의 데이터셋으로 합리적인 학습 시간(NVidia Titan GPU (6 GB)에서 10시간 학습)을 가졌다고 말했습니다.