nnU-Net:
Self-adapting Framework for U-Net-Based Medical Image Segmentation

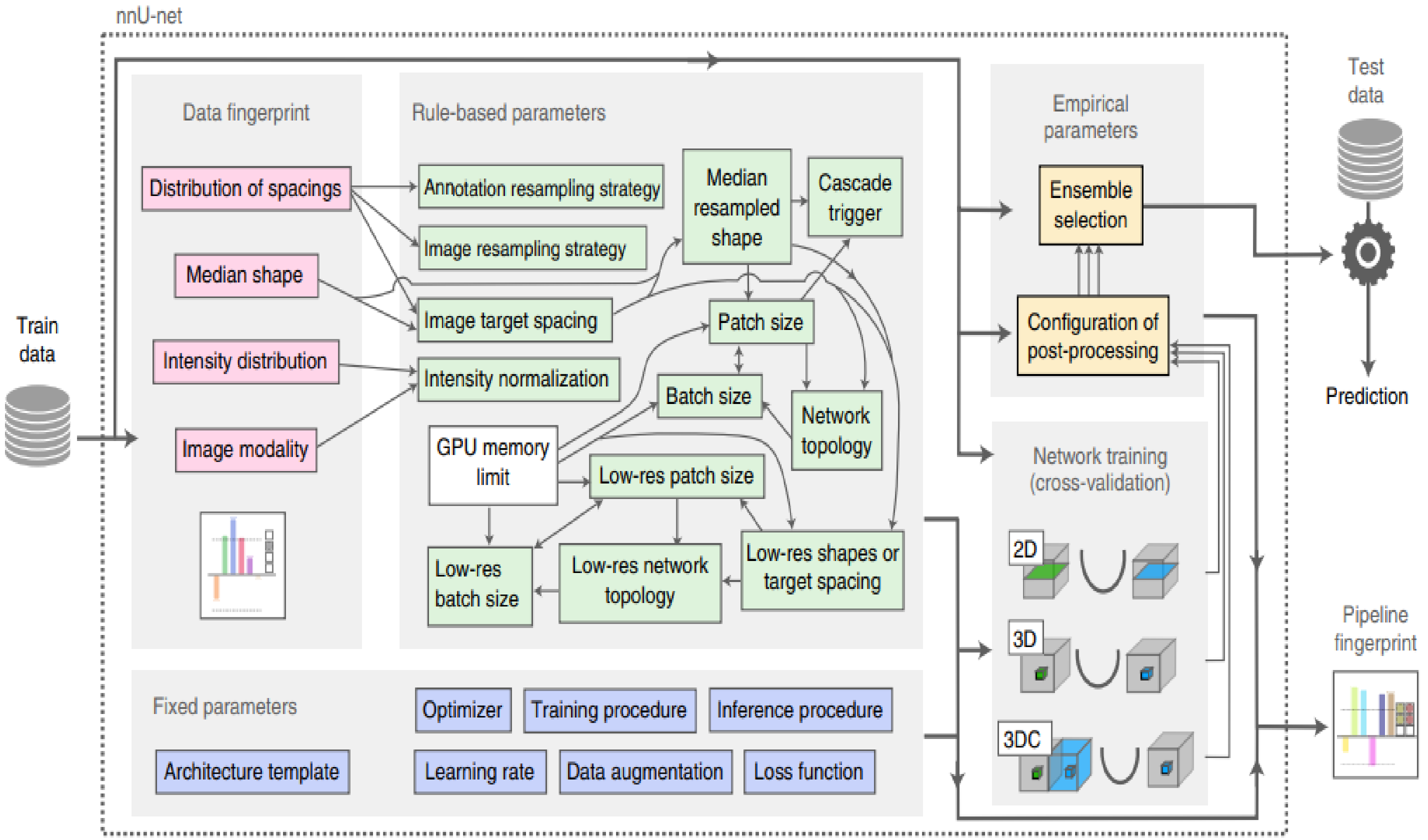
2022.12.09

논문 리뷰

배성훈

nnU-Net: Self-adapting Framework for U-Net-Based Medical Image Segmentation

- **Research Background**:
  - 문제: medical domain은 데이터 셋 부족, 라벨 전문성 부족, 많은 리소스 필요의 문제를 가짐
  - 또한, 저자는 model architecture보다 non-architectural aspects(preprocessing, training, inference, post-processing)이 성능에 더 많은 영향을 준다는 것을 인식
  - 해결: 주어진 데이터에 맞춰 자동적으로 architecture를 적합하는 nnUNet(no-new-UNet)을 제시

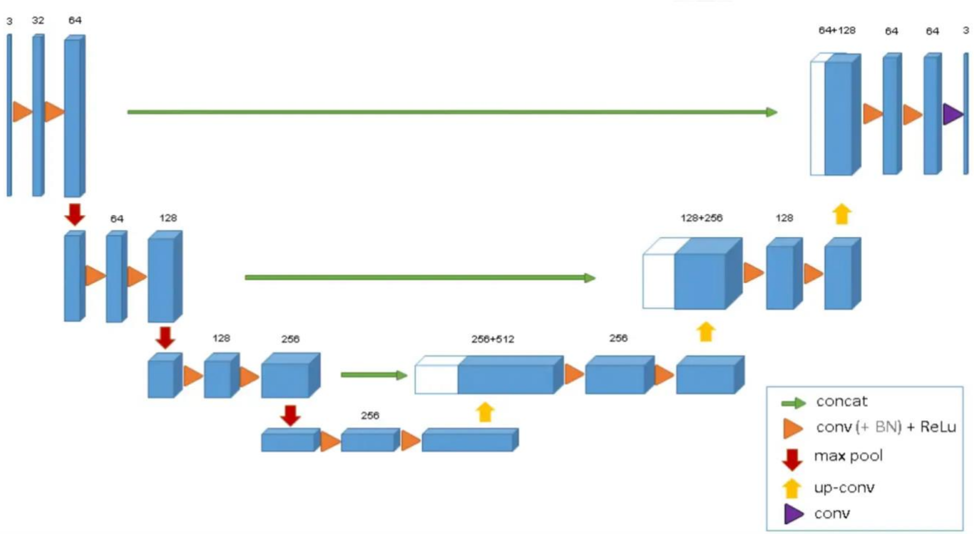


Design choice	Required input	Automated (fixed, rule-based or empirical) configuration derived by distilling expert knowledge (more details in online methods)
Learning rate	–	Poly learning rate schedule (initial, 0.01)
Loss function	–	Dice and cross-entropy
Architecture template	–	Encoder–decoder with skip-connection ('U-Net-like') and instance normalization, leaky ReLU, deep super-vision (topology-adapted in inferred parameters)
Optimizer	–	SGD with Nesterov momentum ($\mu = 0.99$)
Data augmentation	–	Rotations, scaling, Gaussian noise, Gaussian blur, brightness, contrast, simulation of low resolution, gamma correction and mirroring
Training procedure	–	1,000 epochs × 250 minibatches, foreground oversampling
Inference procedure	–	Sliding window with half-patch size overlap, Gaussian patch center weighting
Intensity normalization	Modality, intensity distribution	If CT, global dataset percentile clipping & z score with global foreground mean and s.d. Otherwise, z score with per image mean and s.d.
Image resampling strategy	Distribution of spacings	If anisotropic, in-plane with third-order spline, out-of-plane with nearest neighbor Otherwise, third-order spline
Annotation resampling strategy	Distribution of spacings	Convert to one-hot encoding → If anisotropic, in-plane with linear interpolation, out-of-plane with nearest neighbor Otherwise, linear interpolation
Image target spacing	Distribution of spacings	If anisotropic, lowest resolution axis tenth percentile, other axes median. Otherwise, median spacing for each axis. (computed based on spacings found in training cases)
Network topology, patch size, batch size	Median resampled shape, target spacing, GPU memory limit	Initialize the patch size to median image shape and iteratively reduce it while adapting the network topology accordingly until the network can be trained with a batch size of at least 2 given GPU memory constraints. for details see online methods.
Trigger of 3D U-Net cascade	Median resampled image size, patch size	Yes, if patch size of the 3D full resolution U-Net covers less than 12.5% of the median resampled image shape
Configuration of low-resolution 3D U-Net	Low-res target spacing or image shapes, GPU memory limit	Iteratively increase target spacing while reconfiguring patch size, network topology and batch size (as described above) until the configured patch size covers 25% of the median image shape. For details, see online methods.
Configuration of post-processing	Full set of training data and annotations	Treating all foreground classes as one; does all-but-largest-component-suppression increase cross-validation performance? Yes, apply; reiterate for individual classes No, do not apply; reiterate for individual foreground classes
Ensemble selection	Full set of training data and annotations	From 2D U-Net, 3D U-Net or 3D cascade, choose the best model (or combination of two) according to cross-validation performance

nnU-Net: Self-adapting Framework for U-Net-Based Medical Image Segmentation

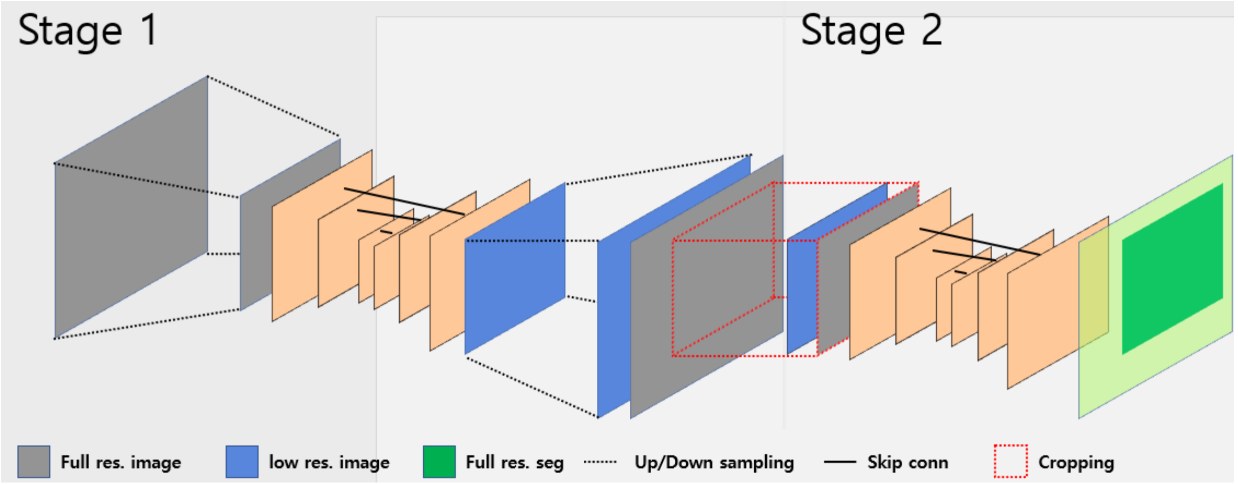
- **Method**:
  - Architecture: 무시될 정도로 거의 수정하지 않고 original 2D, 3D UNet을 사용
  - ReLU -> Leaky ReLU로, Batch Norm -> Instance Norm으로 변환
  - 3D U-Net의 단점을 해결하기 위해, large image size (Liver)는 U-Net cascade model로 학습

3D U-Net



- Good for smaller images (Brain Tumour, Hippocampus)
- Bae for large images (Liver) -> limited field of view

U-Net Cascade



Stage 1: downsampling -> upsampling <training>
Stage 2: seg concatenated one-hot encoding full res image and repeat stage 1

nnU-Net: Self-adapting Framework for U-Net-Based Medical Image Segmentation

- **Method**:
  - Input patch size와 axis당 pooling operation 수는 각 데이터 세트에 대해 자동으로 조정
  - 메모리 제약사항 때문에, input patch volume은 128x128x128을 넘기지 않고 input patch size를 데이터셋 voxel의 중간 값으로 설정

		2D U-Net	3D U-Net	3D U-Net lowres
BrainTumour	median patient shape	169x138	138x169x138	-
	input patch size	192x160	128x128x128	-
	batch size	89	2	-
	num pool per axis	5, 5	5, 5, 5	-
Heart	median patient shape	320x232	115x320x232	58x160x116
	input patch size	320x256	80x192x128	64x160x128
	batch size	33	2	2
	num pool per axis	6, 6	4, 5, 5	4, 5, 5
Liver	median patient shape	512x512	482x512x512	121x128x128
	input patch size	512x512	128x128x128	128x128x128
	batch size	10	2	2
	num pool per axis	6, 6	5, 5, 5	5, 5, 5
Hippocampus	median patient shape	50x35	36x50x35	-
	input patch size	56x40	40x56x40	-
	batch size	366	9	-
	num pool per axis	3, 3	3, 3, 3	-
Prostate	median patient shape	320x319	20x320x319	-
	input patch size	320x320	20x192x192	-
	batch size	26	4	-
	num pool per axis	6, 6	2, 5, 5	-
Lung	median patient shape	512x512	252x512x512	126x256x256
	input patch size	512x512	112x128x128	112x128x128
	batch size	10	2	2
	num pool per axis	6, 6	4, 5, 5	4, 5, 5
Pancreas	median patient shape	512x512	96x512x512	96x256x256
	input patch size	512x512	96x160x128	96x160x128
	batch size	10	2	2
	num pool per axis	6, 6	4, 5, 5	4, 5, 5

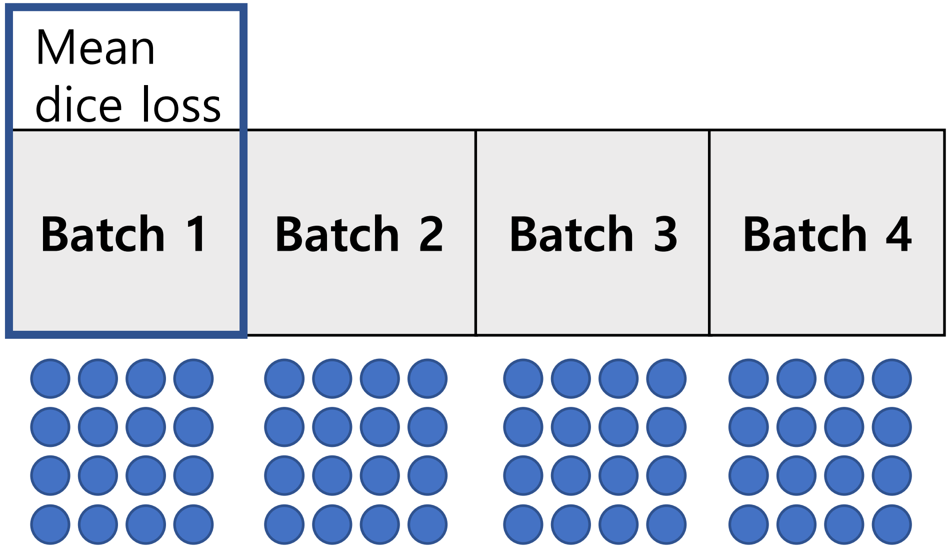
nnU-Net: Self-adapting Framework for U-Net-Based Medical Image Segmentation

- **Method**: Preprocessing
  - **Cropping**: 계산 부담을 줄이기 위해, 모든 데이터는 0이 아닌 값의 영역으로 자름
  - **Resampling**: CNNs이 자체적으로 voxel spacing을 이해할 수 없기 때문에 데이터 셋의 중간 voxel spacing으로 resampling 함
  - **U-Net cascade**: resample된 data의 중간 shape이 input patch보다 4배 일때 결정된다.
  - **Normalization**: CT의 경우, [0.5,99.5]로 clipping, MRI는 z-score normalization
  - Cropping으로 데이터 세트(복셀 단위)에서 환자의 평균 크기가 ¼이상 감소하는 경우, 정규화는 0이 아닌 요소의 마스크 내에서만 수행, 마스크 외부의 모든 값은 0으로 처리

nnU-Net: Self-adapting Framework for U-Net-Based Medical Image Segmentation

- **Method**: Training procedure
  - **Combination of dice and cross-entropy loss**: $L_{total} = L_{dice} + L_{Cross\ Entropy}$
  - 배치 내 샘플마다 dice loss 계산 후 batch 샘플들의 mean dice loss 계산
    u= softmax output, v= Ground truth segmentation map의 one-hot encoding
    u, v는 IxK shape이고, $i \in I$는 $number\ of\ pixels\ in\ training\ patch/batch$, $\mathrm{k} \in K$는 classes

In U-Net cascade & 3D U-Net



● Sample마다 dice loss 계산

$$L_{dc} = -\frac{2}{|K|} \sum_{k \in K} \frac{\sum_{i \in I} u_i^k v_i^k}{\sum_{i \in I} u_i^k + \sum_{i \in I} v_i^k}$$

nnU-Net: Self-adapting Framework for U-Net-Based Medical Image Segmentation

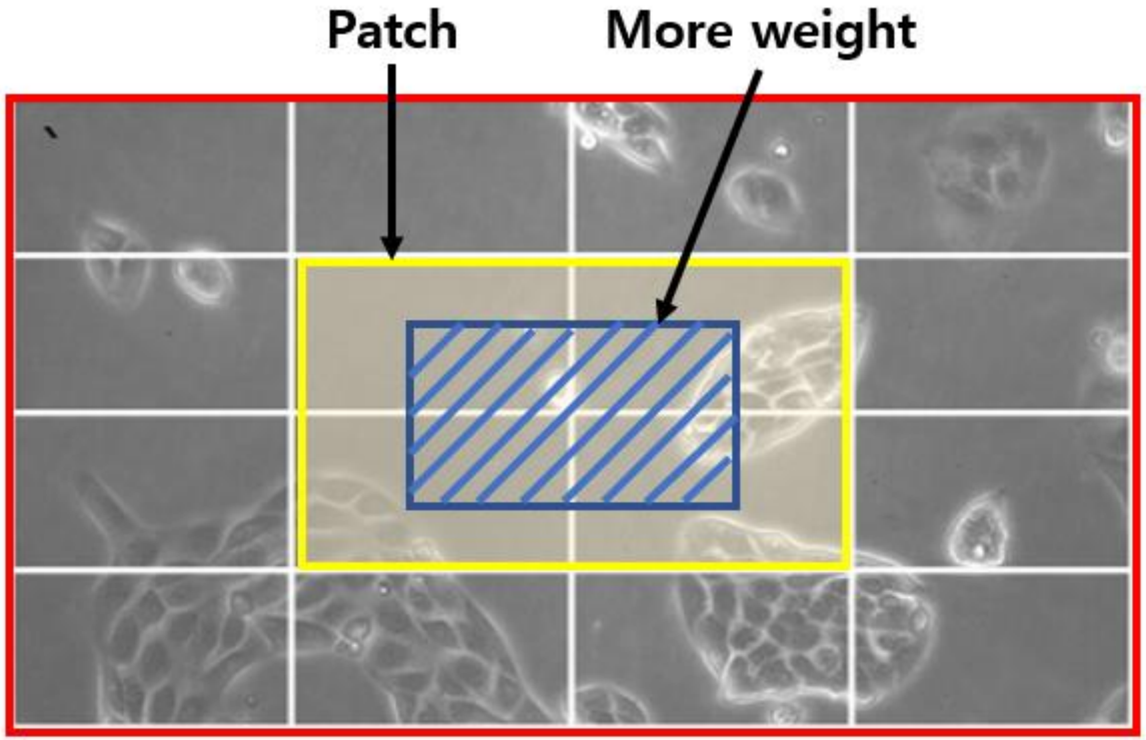
- **Method**: Training procedure
  - **Adam optimizer**
  - Initial learning rate: $3\text{x}10^{-4}$
  - 1 epoch = 250 iteration
  - Batch size = 2
  - Training loss가 30 epochs 동안 $5\text{x}10^{-3}$보다 개선되지 못하면 learning rate를 factor 5만큼 감소

  - **Data Augmentation**
  - 제한된 data 학습 시 NN은 overfitting 예방을 위해 data augmentation 수행
  - 2D, 3D U-Net 별 다른 data augmentation
  - 3D Unet, $Maximum\ edge\ length\ of\ the\ input\ patch\ size > Shortest\ edge\ length =$ 2배 이상인 경우 차원 별 data augmentation 비효율적 -> 2D data augmentation 적용

  - **Patch sampling**
  - Network 학습 안정성 증가를 위해, batch 샘플들 중 3rd 이상의 샘플에 적어도 하나 이상의 random foreground class 포함

nnU-Net: Self-adapting Framework for U-Net-Based Medical Image Segmentation

- **Method**: Inference
  - 패치 경계에서 정확성 감소를 해결하기 위해 복셀이 가운데에 위치할수록 높은 가중치
  - test case에서, 모델의 robustness를 향상시키기 위해 training set을 cross-validation해서 얻은 5개의 network를 앙상블로 사용
    -> 가장 큰 mean forground dice score를 달성하는 모델을 **자동으로 선정**해줌



nnU-Net: Self-adapting Framework for U-Net-Based Medical Image Segmentation

- **Experiments**:
  - Mean dice scores for the proposed models
  - Five-fold cross-validation으로 진행
  - Brain-Trumor를 제외하고 전부 높은 성능을 보임

	BrainTumour			Heart	Liver		Hippoc.		Prostate		Lung	Pancreas	
label	1	2	3	1	1	2	1	2	1	2	1	1	2
2D U-Net	78.60	58.65	77.42	91.36	94.37	53.94	88.52	86.70	61.98	84.31	52.68	74.70	35.41
3D U-Net	**80.71**	**62.22**	**79.07**	92.45	94.11	61.74	**89.87**	**88.20**	60.77	83.73	55.87	77.69	42.69
3D U-Net stage1 only (U-Net Cascade)	-	-	-	90.63	94.69	47.01	-	-	-	-	65.33	79.45	49.65
3D U-Net (U-Net Cascade)	-	-	-	92.40	95.38	58.49	-	-	-	-	**66.85**	**79.30**	**52.12**
ensemble 2D U-Net+ 3D U-Net	80.79	61.72	79.16	**92.70**	94.30	60.24	89.78	88.09	**63.78**	**85.31**	55.96	78.26	40.46
ensemble 2D U-Net+ 3D U-Net (U-Net Cascade)	-	-	-	92.64	95.31	60.09	-	-	-	-	61.18	78.79	45.46
ensemble 3D U-Net+ 3D U-Net (U-Net Cascade)	-	-	-	92.63	**95.43**	**61.82**	-	-	-	-	65.16	79.70	49.14
test set	67.71	**47.73**	**68.16**	**92.77**	**95.24**	**73.71**	**90.37**	**88.95**	**75.81**	**89.59**	**69.20**	**79.53**	**52.27**

nnU-Net: Self-adapting Framework for U-Net-Based Medical Image Segmentation

- **Summary**:
  - 다양한 메디컬 이미지 competitio에서 좋은 성적을 거둔 아키텍쳐와 파라미터 분석
  - 전처리 프로세스와 학습 플랜에 대한 방향성 정립
  - 다양한 태스크에 적용 가능하도록 모듈 래핑
  - +Customizing한 것보다 크게 성능이 뒤쳐지지 않는다.

  - 하지만 독립 API가 아니기 때문에 customizing해서 연구하기에는 어려운 부분을 보인다.

nnU-Net: Self-adapting Framework for U-Net-Based Medical Image Segmentation

- **Summary**:
  - Code
  - 최초의 input data가 nnUNet에 적합한 형태인지 확인하기 위해 plan_and preprocess 과정을 거친다.
  - **Plan_and_Preprocess**
    - 데이터의 무결성을 검사하고(데이터가 학습에 적합한 형태인지),
검증된 데이터로부터 distribution of intensity, voxels spacing, median shape, image modality (CT or MRI) 추출
    - 추출한 데이터는 3개의 U-Net(2D, 3D, cascade 3D)를 생성하는데 사용
    - 이때 이 3개의 UNet pipeline을 만드는 여러 파라미터 및 Strategies를 Plans 파일에 기입
    - 이를 추후 학습시 nnUNetTrainer에 활용한다. (즉, 모델 생성을 위한 파라미터들 정리)