Robotic Surgery AI Challenge

TEAM - 한우정과 아이들

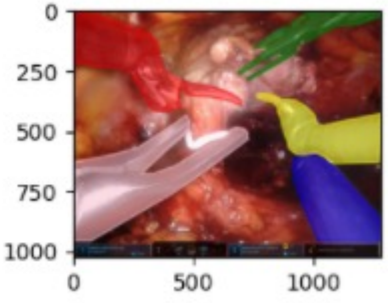
한우정과 아이들  0.577

Contents

- **Key Components of Deep Learning**
  - Data
  - Model
  - Loss
  - Algorithm
- **Further works**

Data

Train Dataset : 12,408건
중복되지 않은 사진 : 10,792장
- 한 이미지 내에 다양한 클래스가 포함 (max: 5)

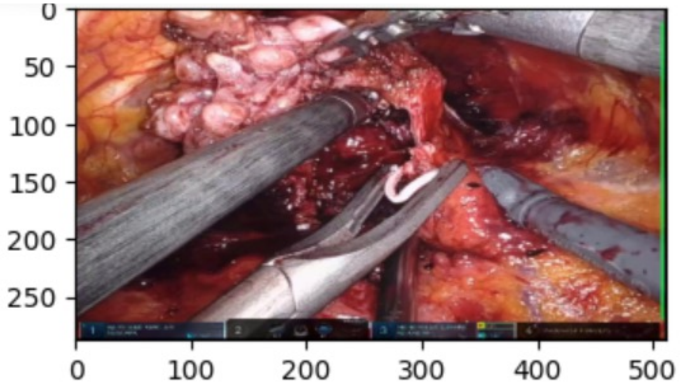


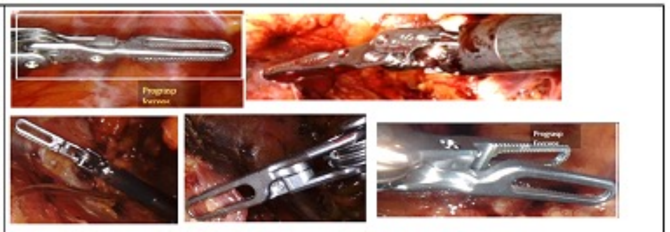
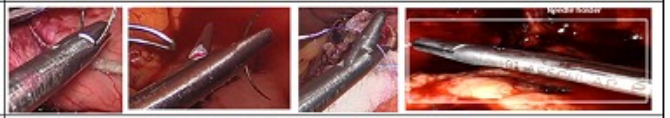
- 사진마다 크기가 다 다름.

Height, Width = {(720, 1280), (1072, 1912), (240, 352), (1024, 1280), (1076, 1912), (1074, 1912), (1080, 1920)}

Height : Width = 9:16 비율이 많다. → **Resize (288, 512)**

Image shape : (288, 512, 3)



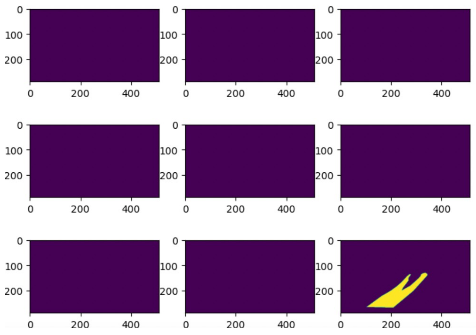
로봇수술도구	1. Prograsp	
	2. Large needle driveer	
	3. Bipolar Maryland forceps	
	4. Curved Scissors	
복강경수술도구	1. Suction irrigator	
	2. Grasping forceps	
	3. Laparoscopic needle holder	
	4. Metal clip applier	
	5. Polymer clip applier	

Data

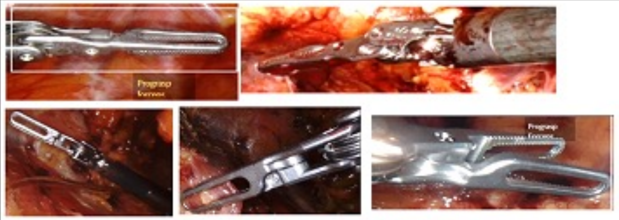
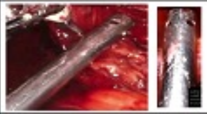
Train Dataset : 12,408건
중복되지 않은 사진 : 10,792장
- 한 이미지 내에 다양한 클래스가 포함 (max: 5)

- 라벨 데이터: 도구마다 n개의 포인트들 (json)

이미지 기준으로 **모든 도구의**
Pixel-wise Segmentation이
가능하도록 **9 channel** 의
2D 데이터 생성.



Label shape : (288, 512, 9)

로봇수술도구	1. Prograsp	
	2. Large needle driveer	
	3. Bipolar Maryland forceps	
	4. Curved Scissors	
복강경수술도구	1. Suction irrigator	
	2. Grasping forceps	
	3. Laparoscopic needle holder	
	4. Metal clip applier	
	5. Polymer clip applier	

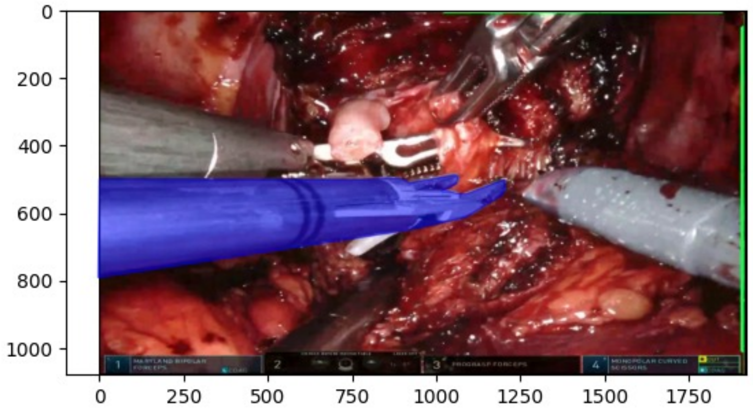
Input Data

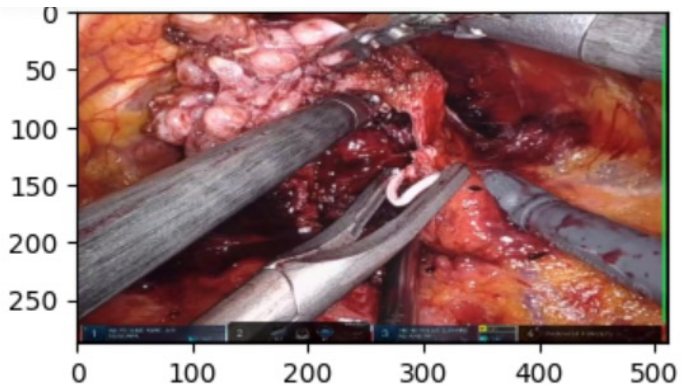
Image shape : (288, 512, 3) x 10,792

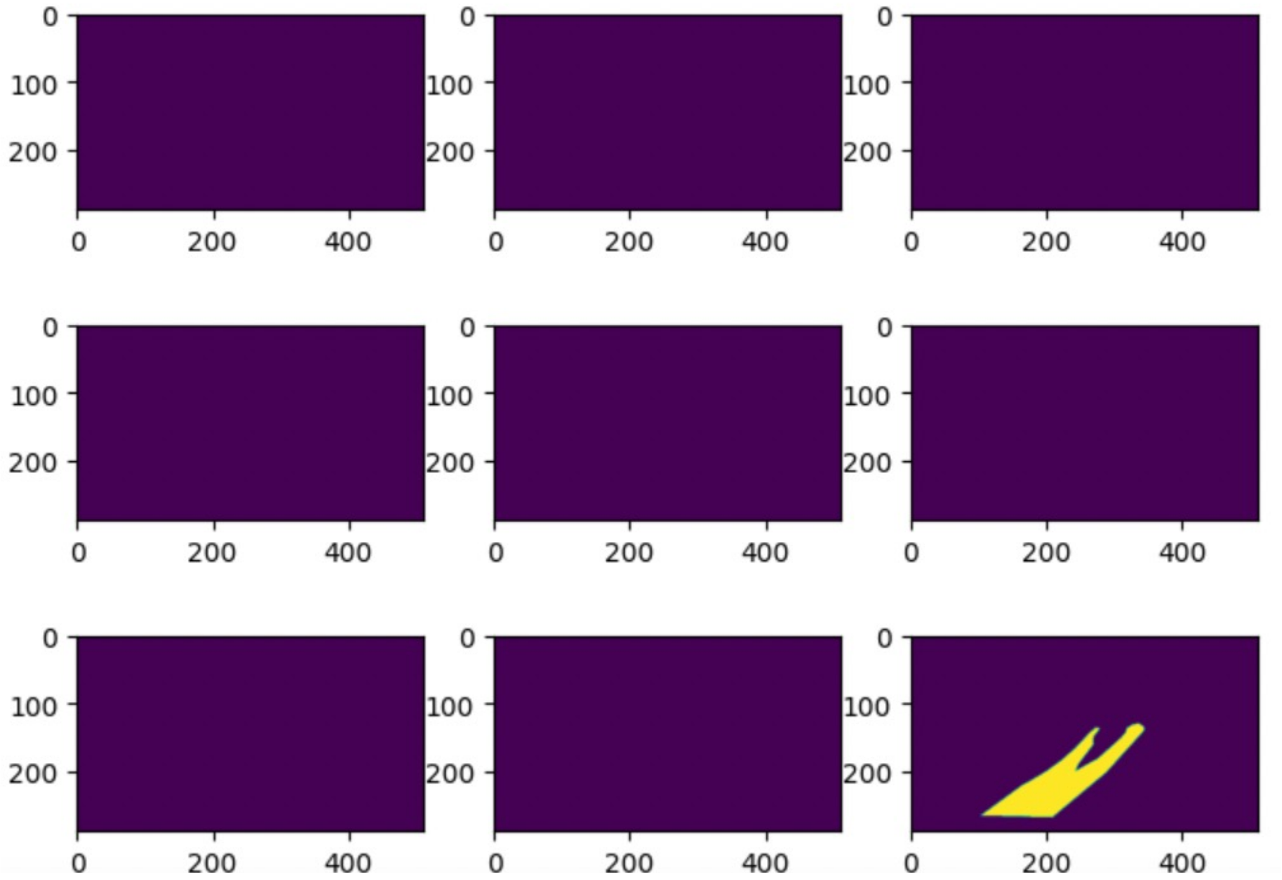
Label shape : (288, 512, 9) x 10,792

! 문제점.

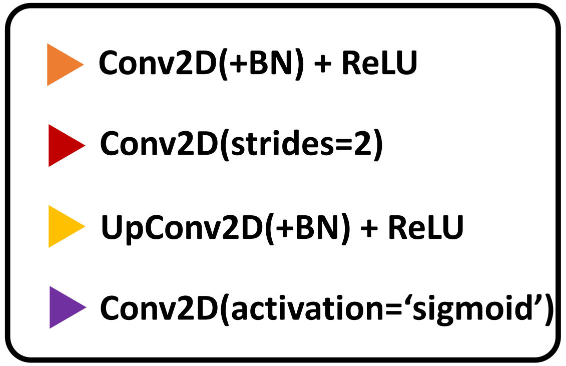
도구 기준으로 마킹을 진행했기 때문에,
이미지마다 **라벨링 값이 누락된 도구**들이
상당히 많았음.

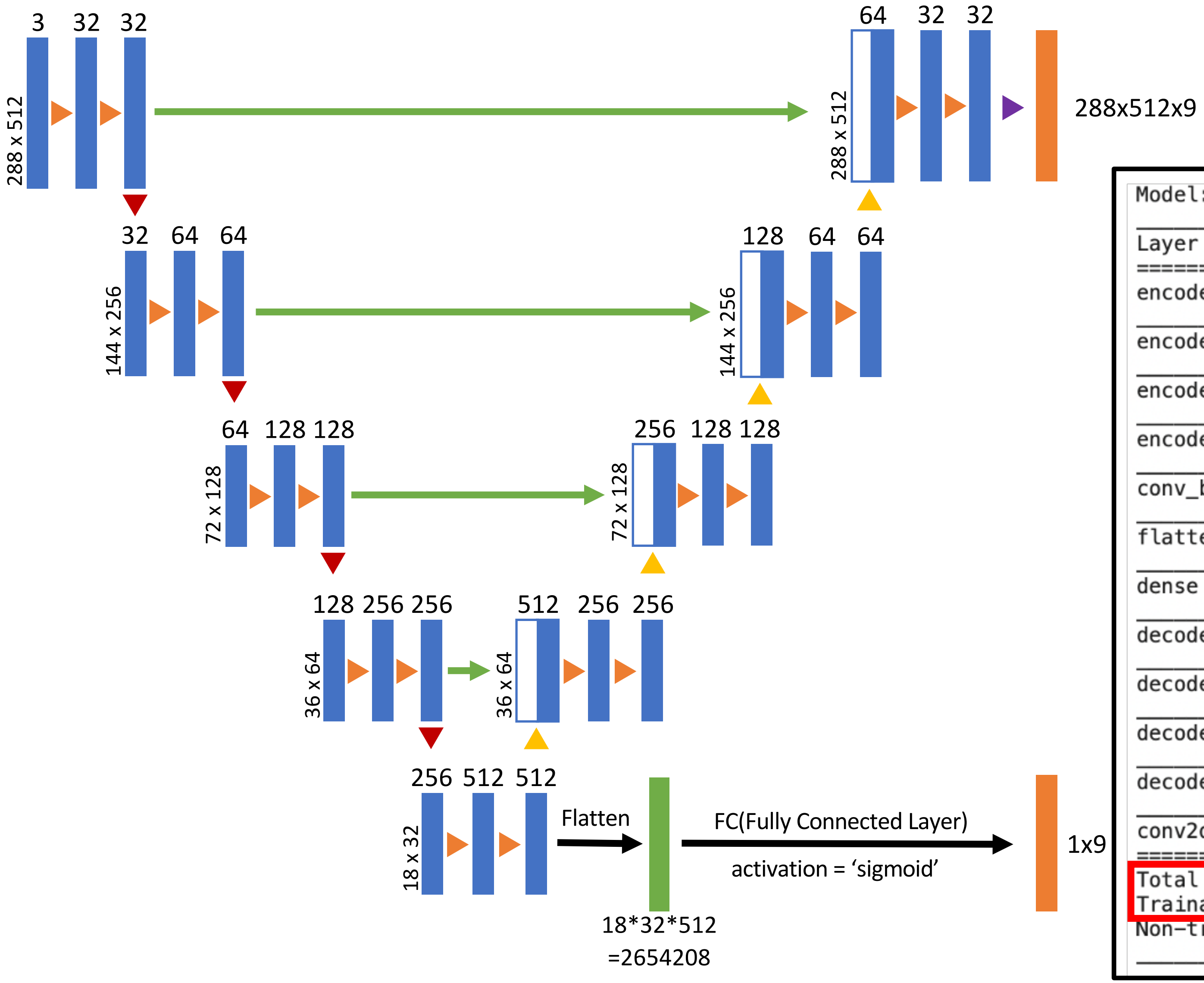






Model (U-Net based)





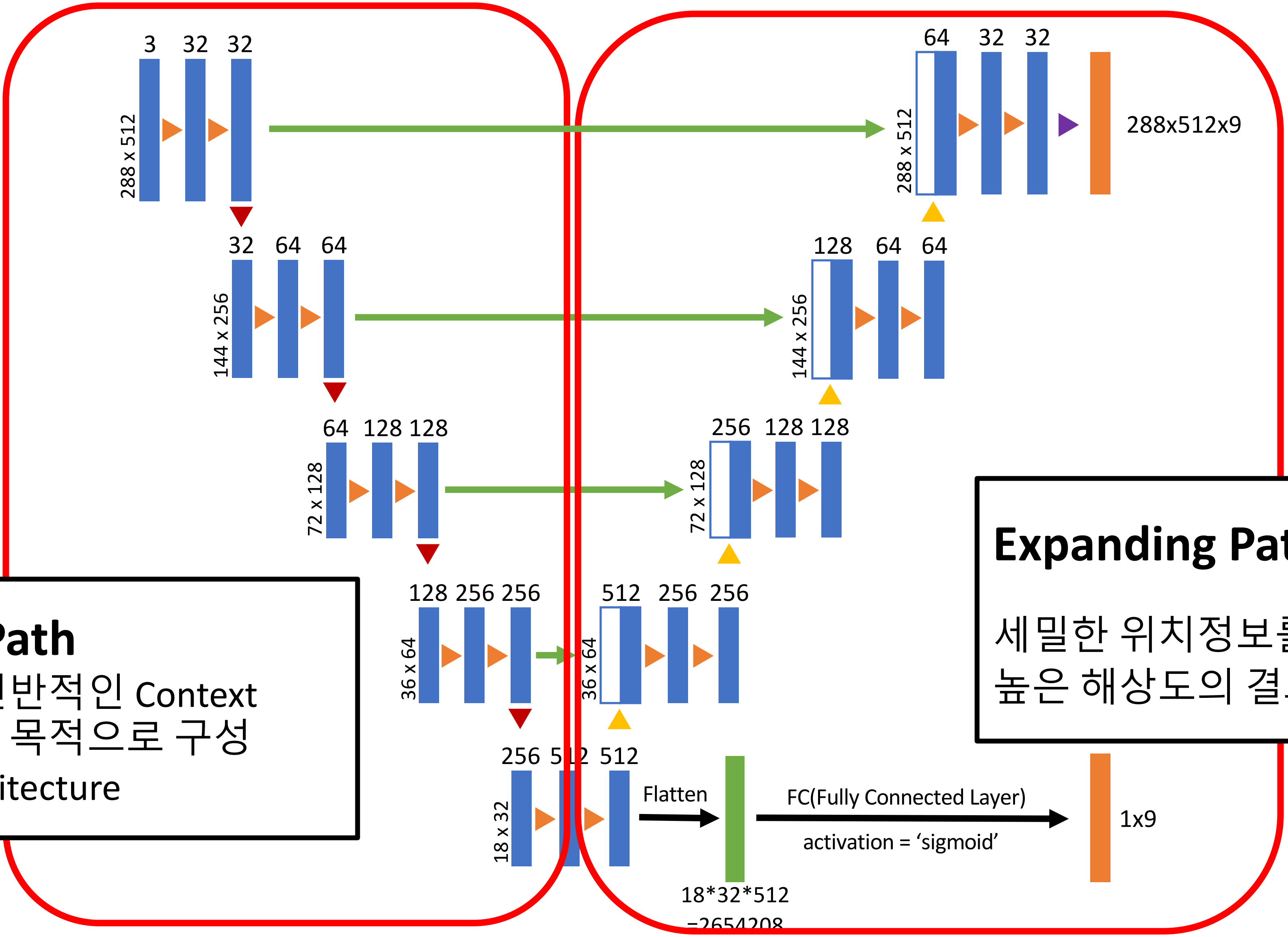
Model: "u_net"

Layer (type)	Output Shape	Param #
encoder_block (EncoderBlock)	multiple	19648
encoder_block_1 (EncoderBloc	multiple	92864
encoder_block_2 (EncoderBloc	multiple	370048
encoder_block_3 (EncoderBloc	multiple	1477376
conv_block_4 (ConvBlock)	multiple	3544064
flatten (Flatten)	multiple	0
dense (Dense)	multiple	2654217
decoder_block (DecoderBlock)	multiple	2952960
decoder_block_1 (DecoderBloc	multiple	739200
decoder_block_2 (DecoderBloc	multiple	185280
decoder_block_3 (DecoderBloc	multiple	46560
conv2d_22 (Conv2D)	multiple	297

Total params: 12,082,514
Trainable params: 12,075,666
Non-trainable params: 6,848

Params : 12M

Model (U-Net based)

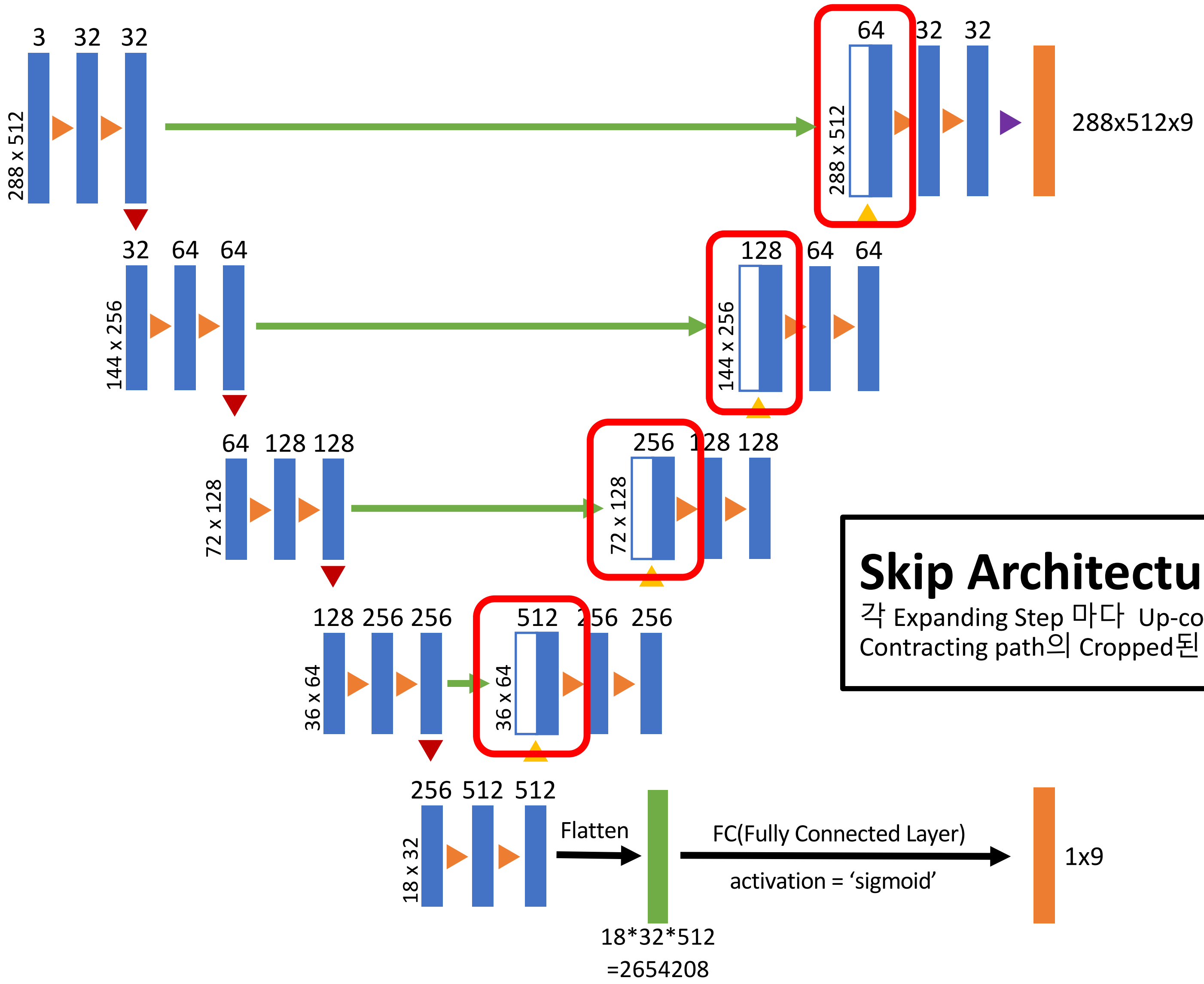


Contracting Path
입력 이미지의 전반적인 Context
정보를 얻기위한 목적으로 구성
- VGG-based Architecture

Expanding Path

세밀한 위치정보를 얻기 위해 구성
높은 해상도의 결과로 Up-Sampling

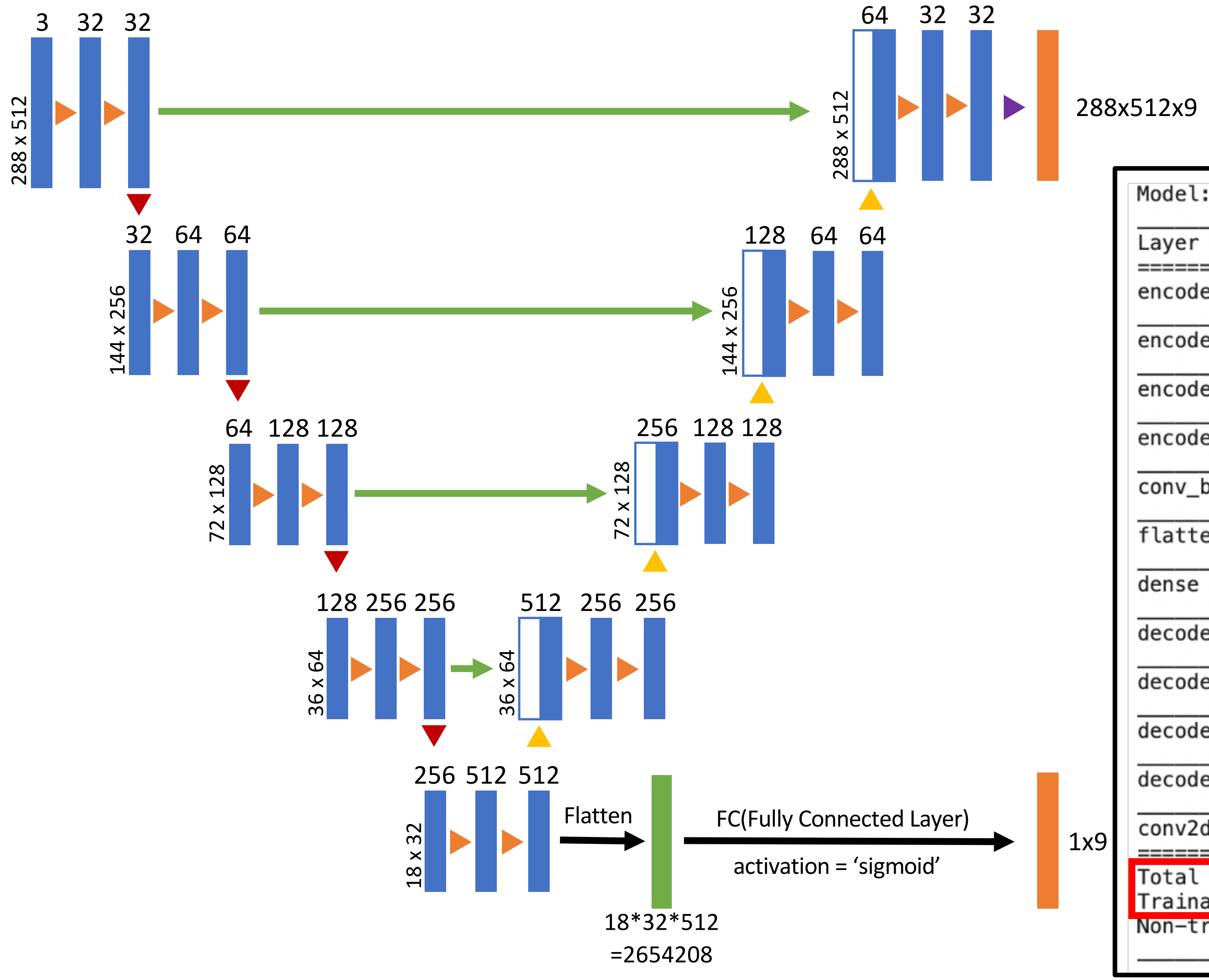
Model (U-Net based)



Skip Architecture

각 Expanding Step 마다 Up-conv 된 특징맵은
Contracting path의 Cropped된 특징맵과 Concatenate 함

Model (U-Net based)



- Conv2D(+BN) + ReLU
- Conv2D(strides=2)
- UpConv2D(+BN) + ReLU
- Conv2D(activation=‘sigmoid’)

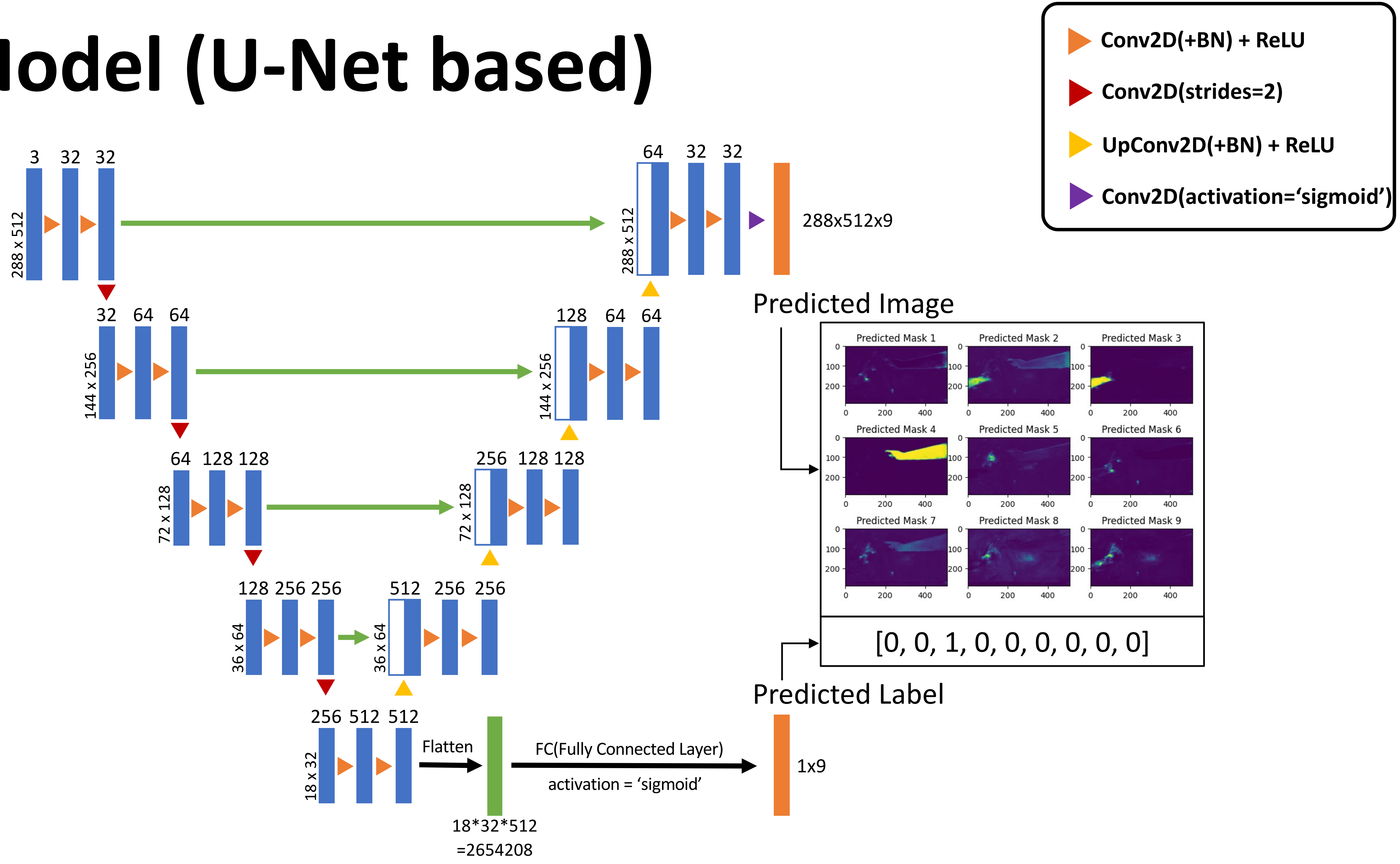
Model: "u_net"

Layer (type)	Output Shape	Param #
encoder_block (EncoderBlock)	multiple	19648
encoder_block_1 (EncoderBloc	multiple	92864
encoder_block_2 (EncoderBloc	multiple	370048
encoder_block_3 (EncoderBloc	multiple	1477376
conv_block_4 (ConvBlock)	multiple	3544064
flatten (Flatten)	multiple	0
dense (Dense)	multiple	2654217
decoder_block (DecoderBlock)	multiple	2952960
decoder_block_1 (DecoderBloc	multiple	739200
decoder_block_2 (DecoderBloc	multiple	185280
decoder_block_3 (DecoderBloc	multiple	46560
conv2d_22 (Conv2D)	multiple	297

Total params: 12,082,514
Trainable params: 12,075,666
Non-trainable params: 6,848

Params : 12M

Model (U-Net based)

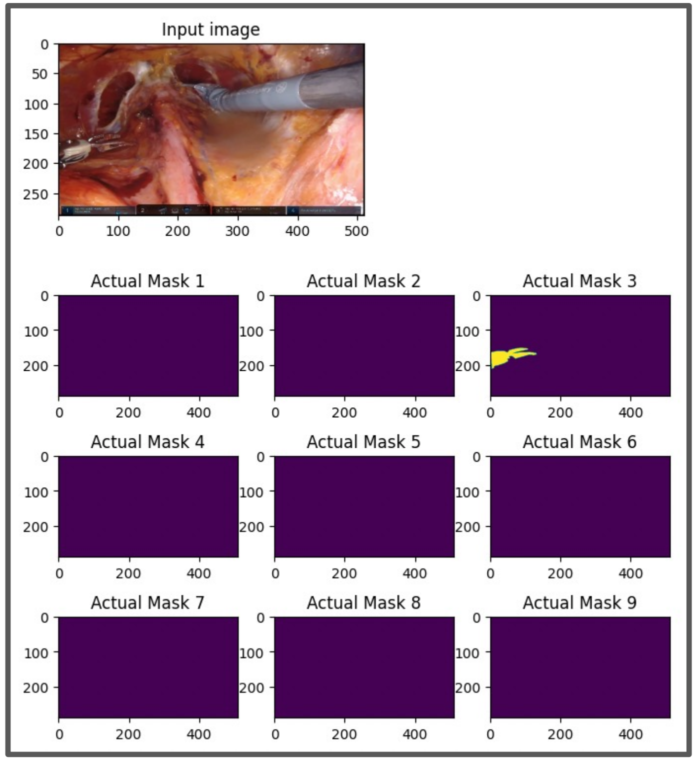


Loss

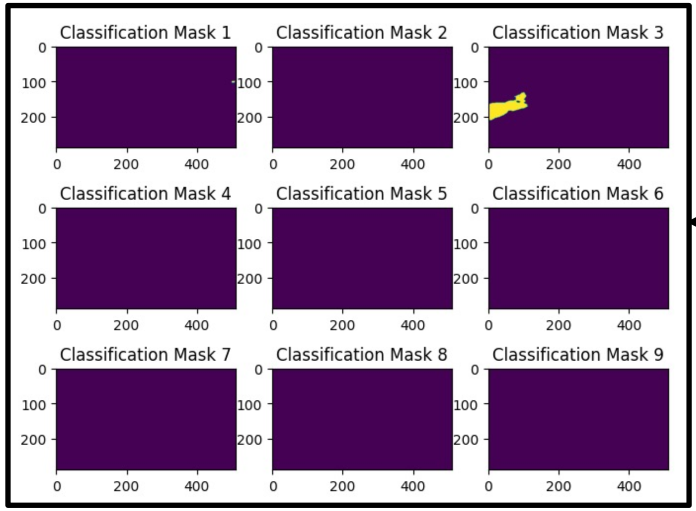
사용한
Loss Function

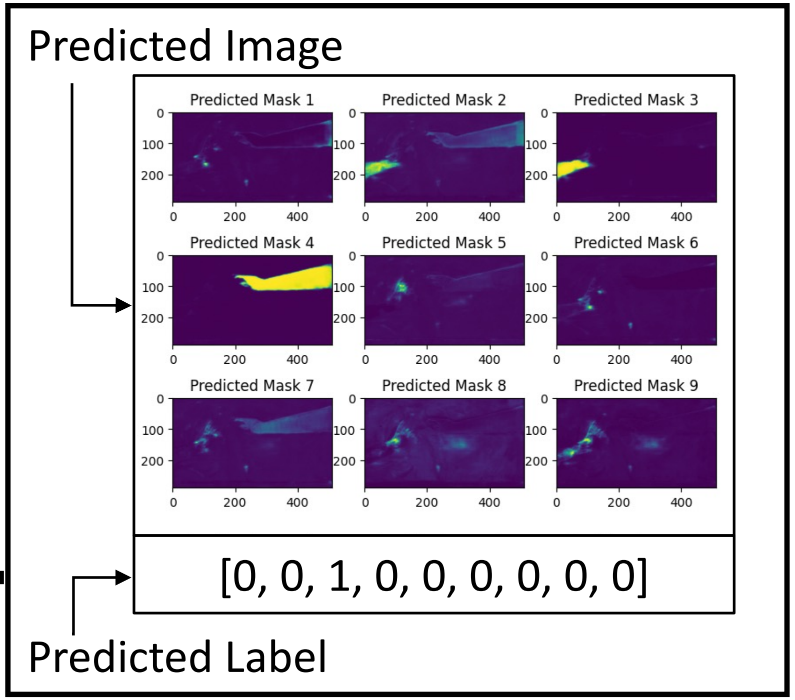
1 BCE Dice Loss
2 BCE Loss

Target



Predict

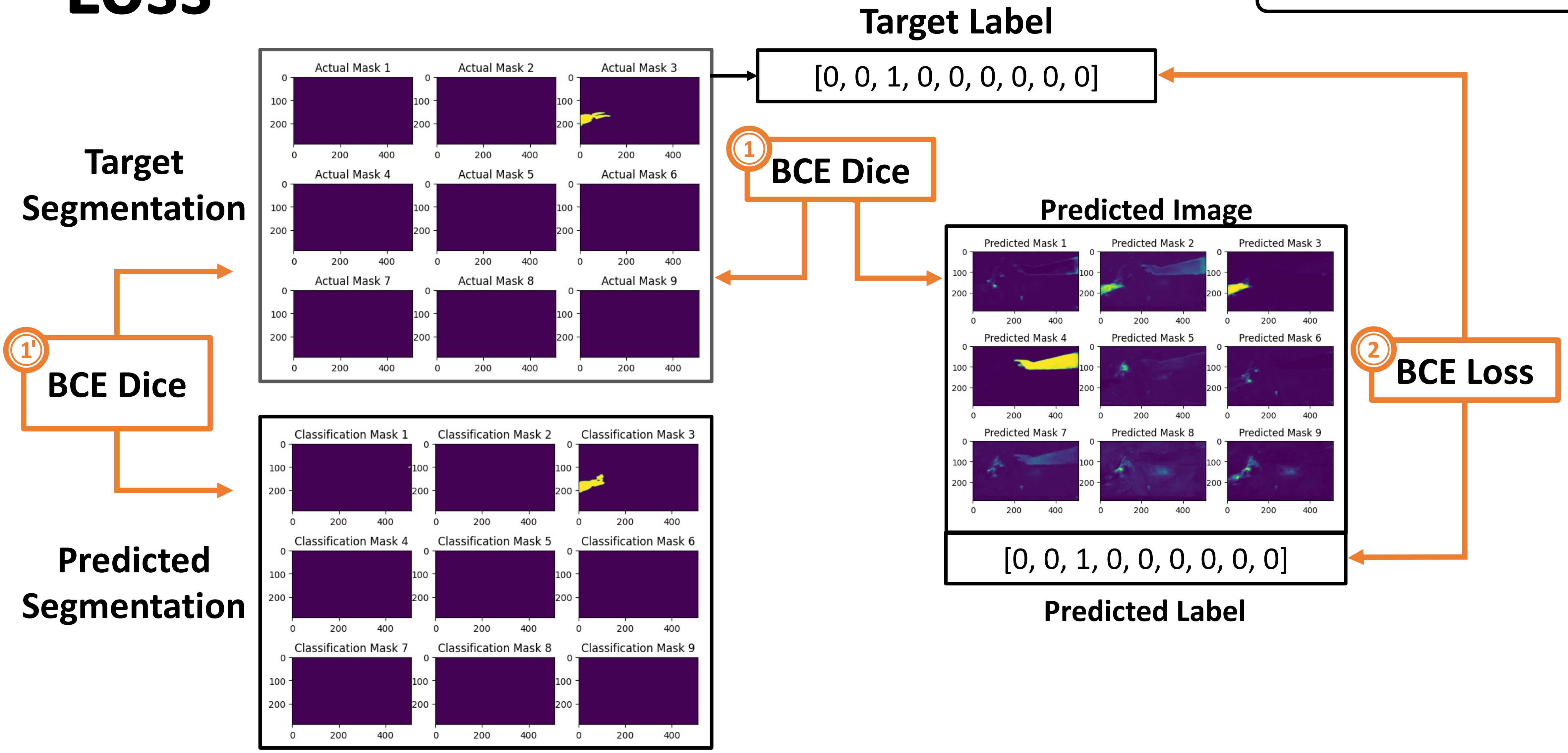




Loss

사용한
Loss Function

① BCE Dice Loss
② BCE Loss



Loss

BCE-Loss

$$J(\mathbf{w}) \;=\; \frac{1}{N}\sum_{n=1}^{N} H(p_n, q_n) \;=\; -\frac{1}{N}\sum_{n=1}^{N} \left[y_n \log \hat{y}_n + (1 - y_n) \log(1 - \hat{y}_n) \right],$$

Dice-Loss = 1– DSC

$$DSC = \frac{2|X \cap Y|}{|X| + |Y|}$$

BCE Dice Loss = BCE-Loss + Dice-Loss

Algorithm

- Training : 2 Days

- Optimizer: Adam
- Learning rate : 1e-4
- Batch size: 16
- Batch Normalization
- Dropout

https://github.com/junneidnwjddl/MAIC-RS

Further works

- Weight initialization + Early Stopping
- Data Augmentation

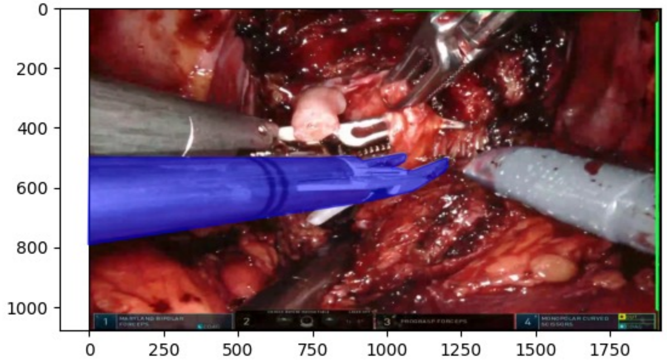
- Ablation Study
  - Model 4층 > 5층.
  - Filter size 변경

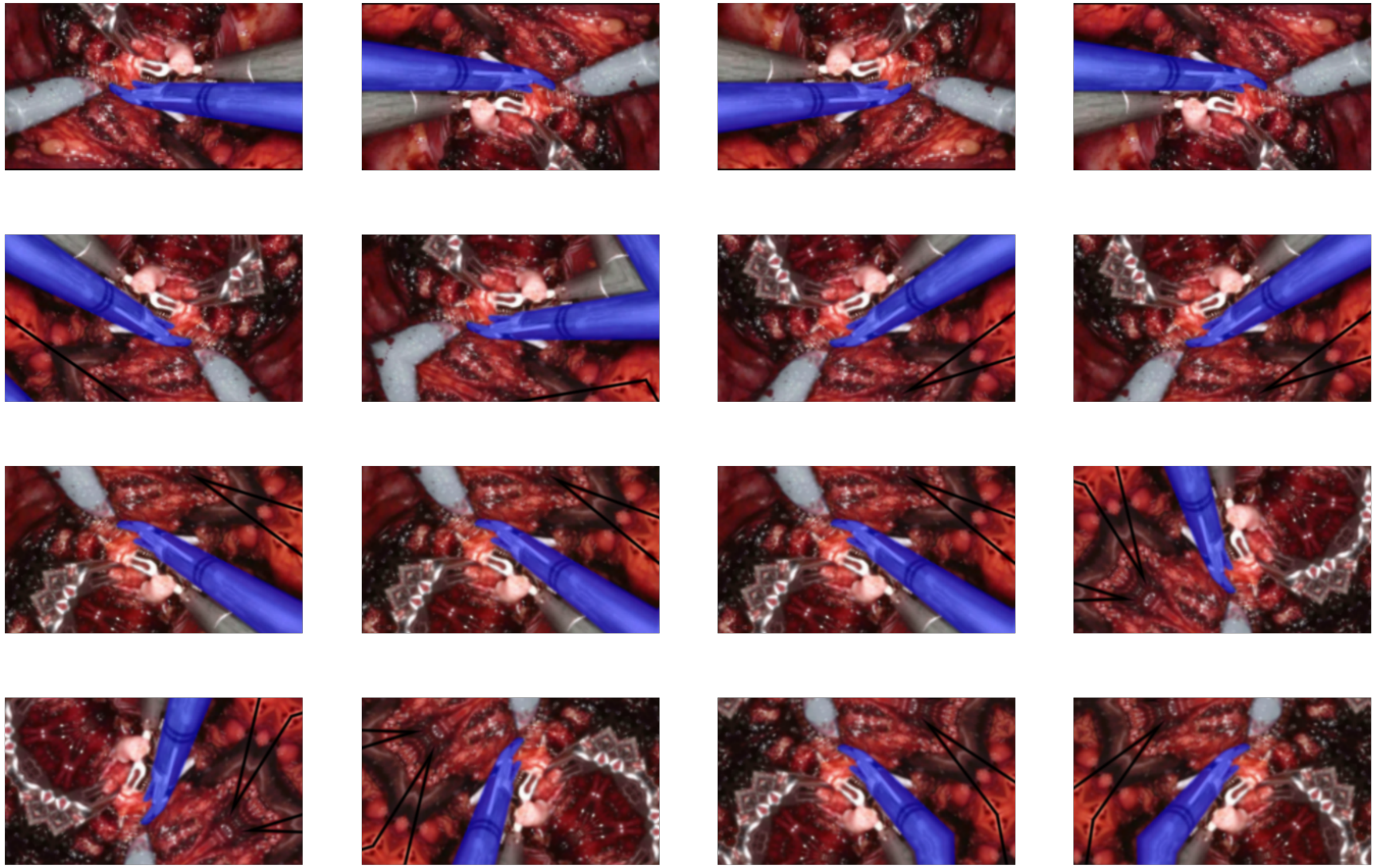
- 도구별로 Model 을 구성, 총 9개 모델을 병렬적으로 학습.

Augmentation

Train, Test 테스트 뿐만이 아니라 실제 데이터에 대한 예측을 하는 것이 목적이기 때문에,
예측 범위를 넓혀주며 한정된 Image들을 보완하기 위하여 Augmentation 진행

- Shift image
- Rotate image
- Flip image
- Hue transition





9 Models

각 도구에 민감한 모델을 각각 만들자.

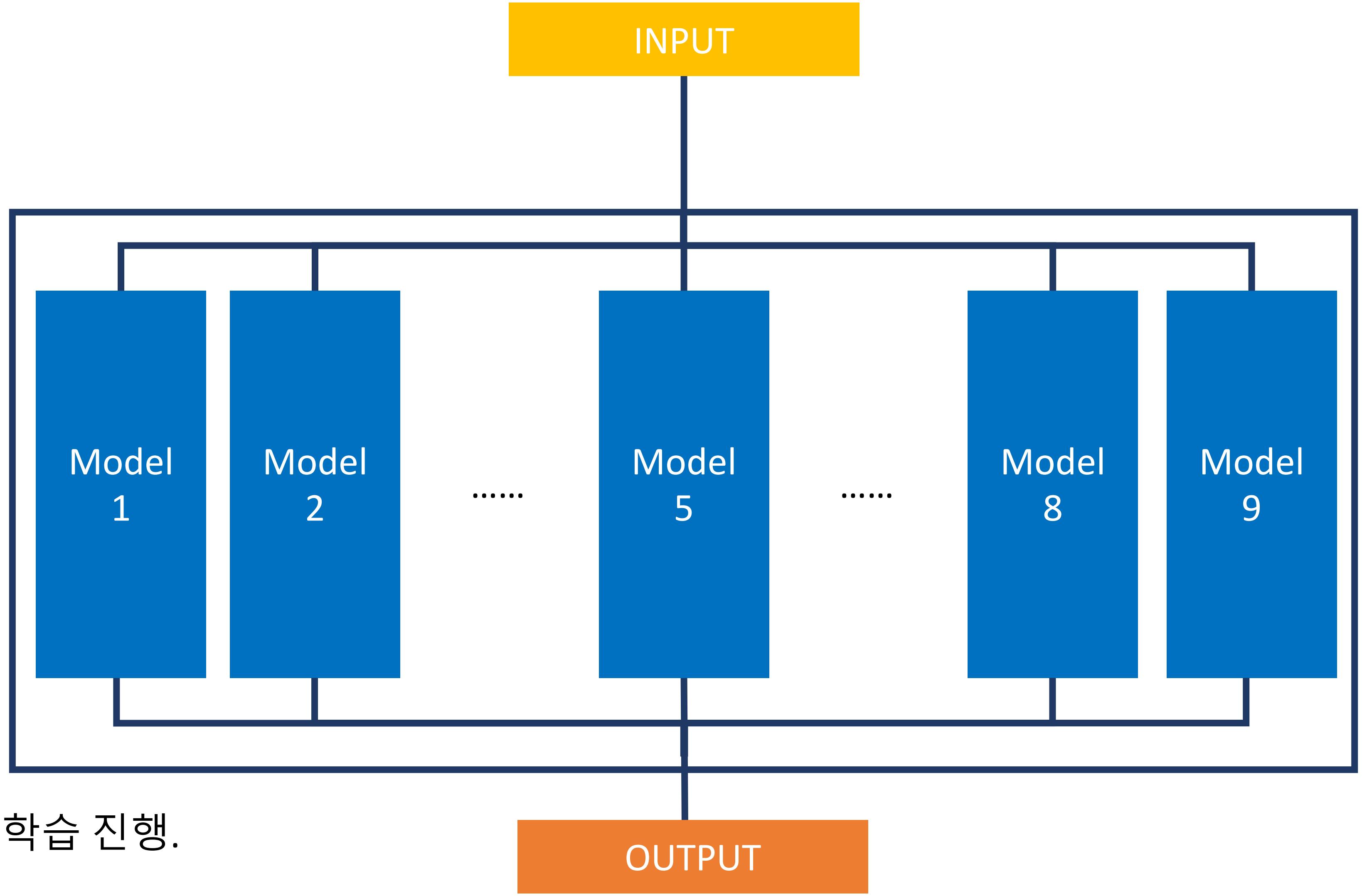
N 번 모델은 N번째 클래스 도구를
감지하는 모델.

학습 방법 -

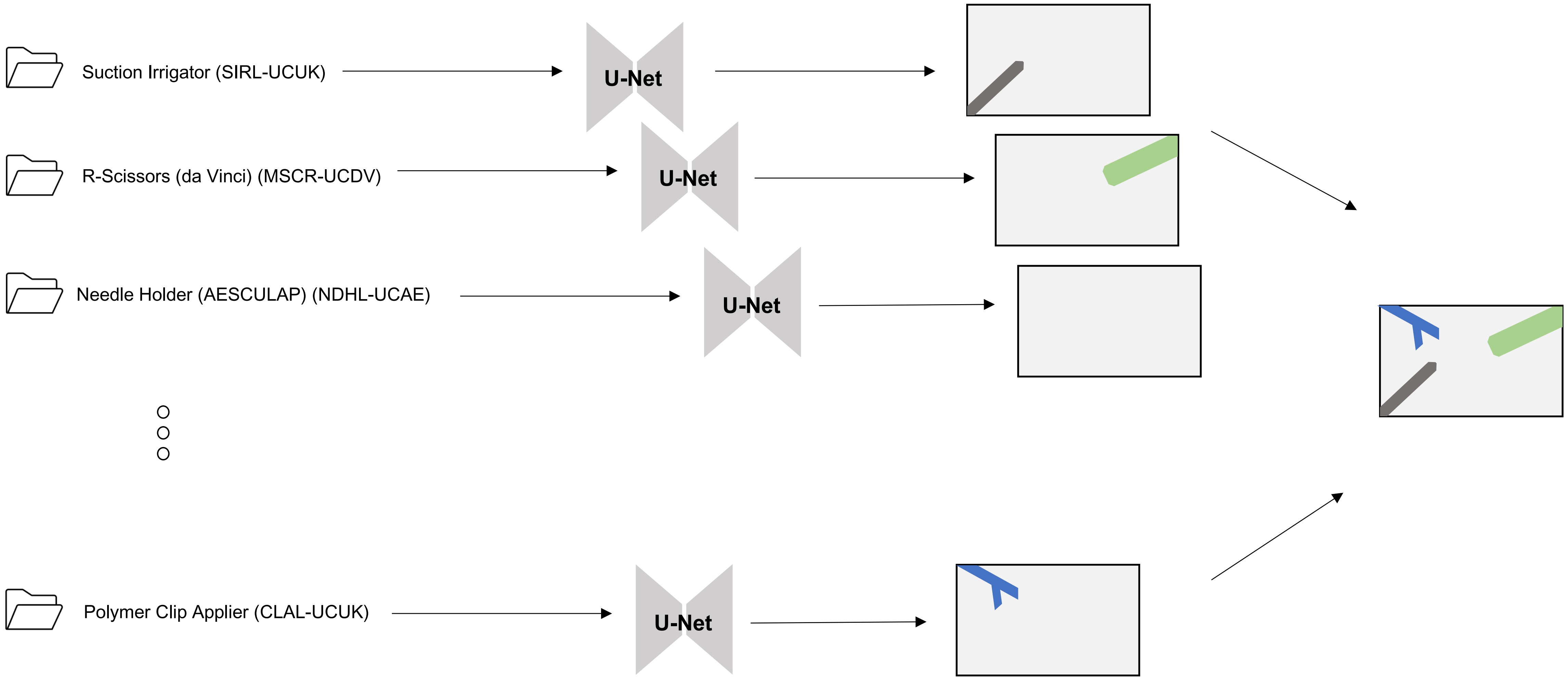
N번째 클래스의 데이터가 들어올 경우.

N번째 모델에 Positive data로 넣고,
다른 2개의 모델에 Negative data로 넣어 학습 진행.

효과 : 각 도구에 대한 세그멘테이션 성능이 월등히 높아짐.



9 Models


Suction Irrigator (SIRL-UCUK)
U-Net
R-Scissors (da Vinci) (MSCR-UCDV)
U-Net
Needle Holder (AESCULAP) (NDHL-UCAE)
U-Net
Polymer Clip Applier (CLAL-UCUK)
U-Net