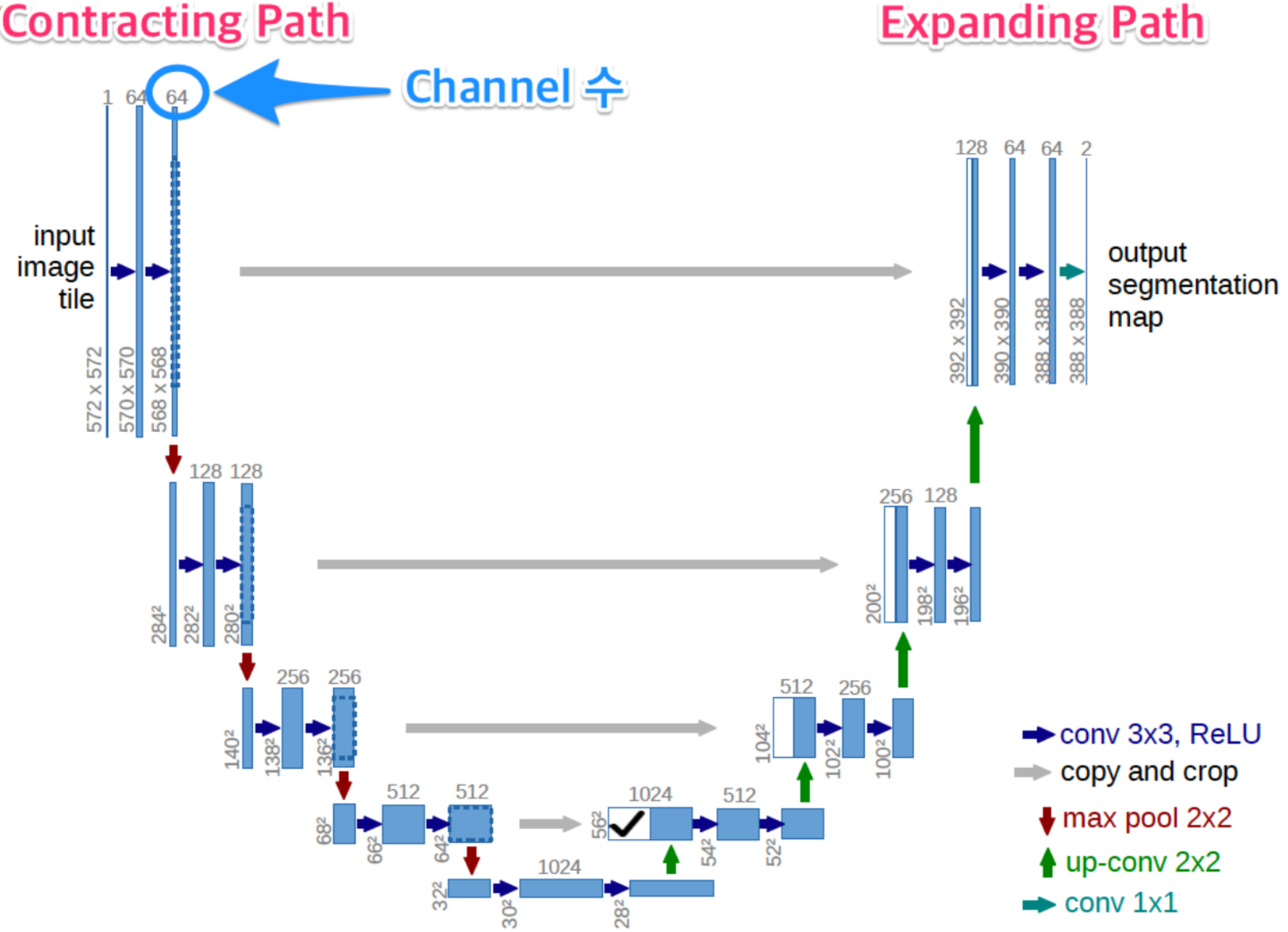
```
#preprocess the mask
        mask[mask >= 2] = 0
        mask[mask != 0 ] = 1
```

```
if len(raw.shape) == 2:
        raw = np.stack((raw,)*3, axis=-1)
```

```
y_true_f = K.flatten(y_true)
y_pred_f = K.flatten(y_pred)
y_true_f = K.cast(y_true_f, 'float32')
y_pred_f = K.cast(y_pred_f, 'float32')

intersection = K.sum(y_true_f * y_pred_f)
return (2. * intersection + smooth) / (K.sum(y_true_f) + K.sum(y_pred_f) + smooth)
```


Contracting Path
Expanding Path
Channel 수
input image tile
output segmentation map
conv 3x3, ReLU
copy and crop
max pool 2x2
up-conv 2x2
conv 1x1