

Computer Vision

Dr. Muhammad Tahir

DEPARTMENT OF COMPUTER SCIENCE,
FAST-NUCES, Peshawar

Course Details

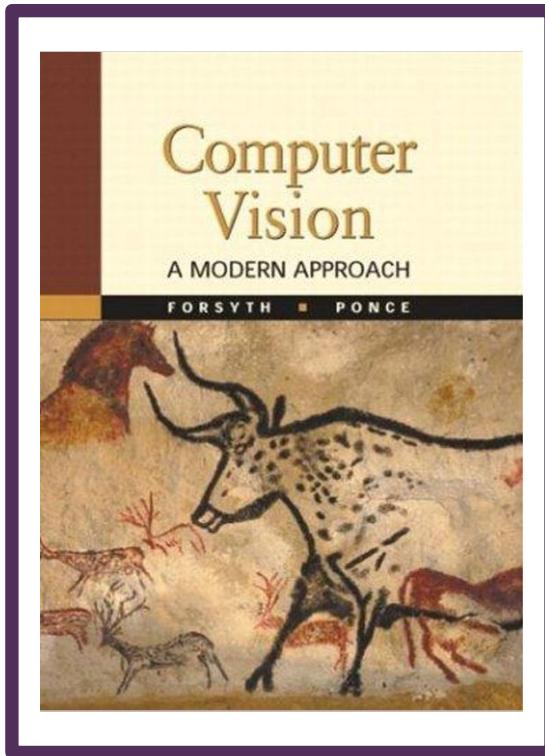
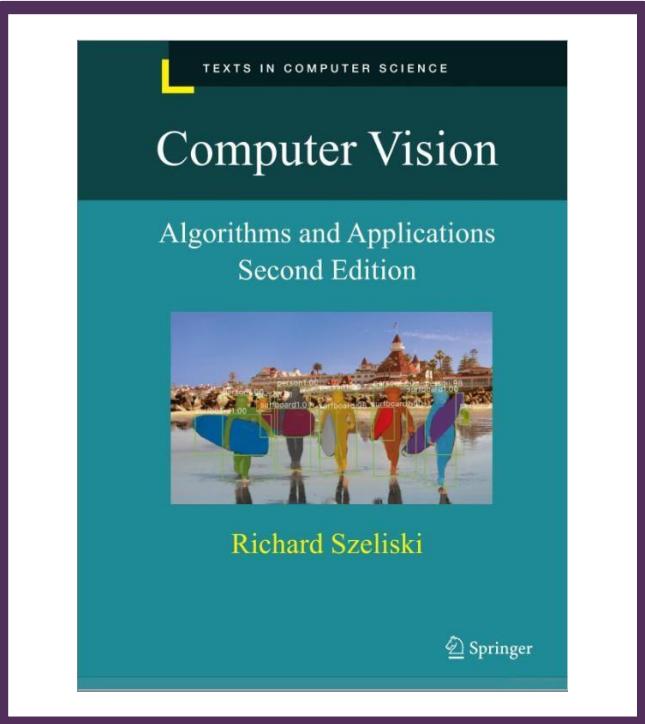
LECTURES: Monday
& Wednesday

TIMINGS:
9:30 am – 11:00 am

MY OFFICE:

OFFICE HOURS:

EMAIL: m.tahir@nu.edu.pk



References

The material in these slides are based on:

1

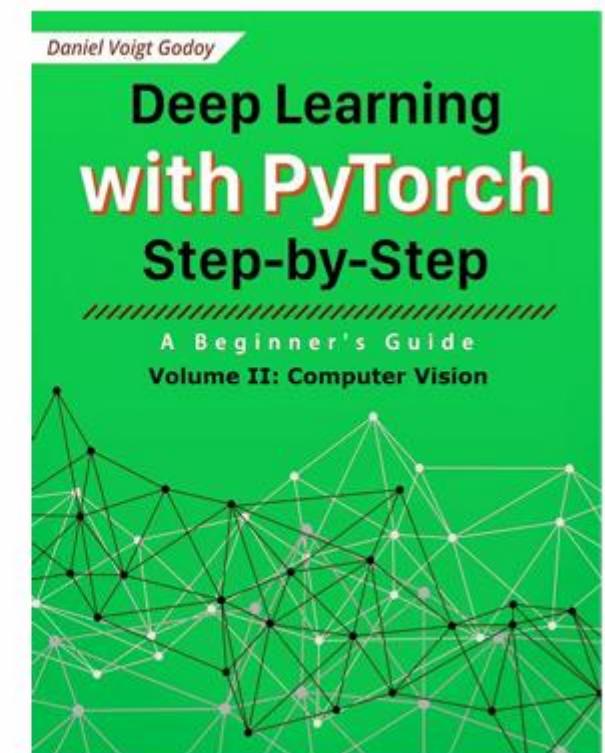
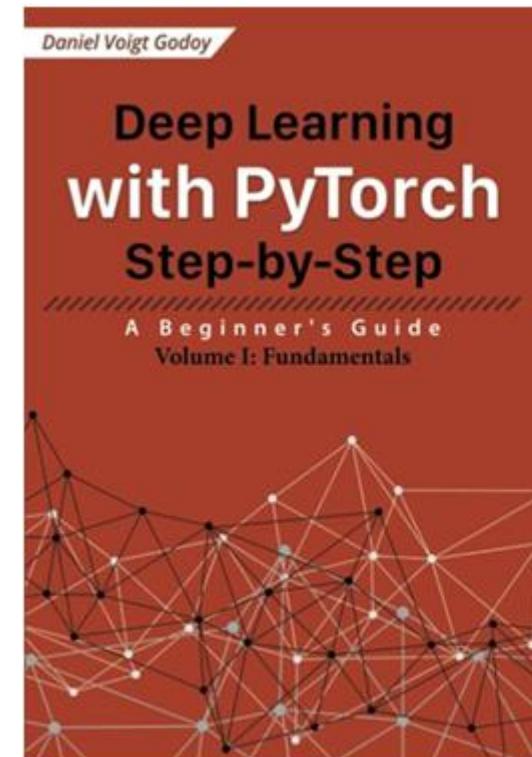
Rick Szeliski's book: [Computer Vision: Algorithms and Applications](#)

2

Forsythe and Ponce: [Computer Vision: A Modern Approach](#)

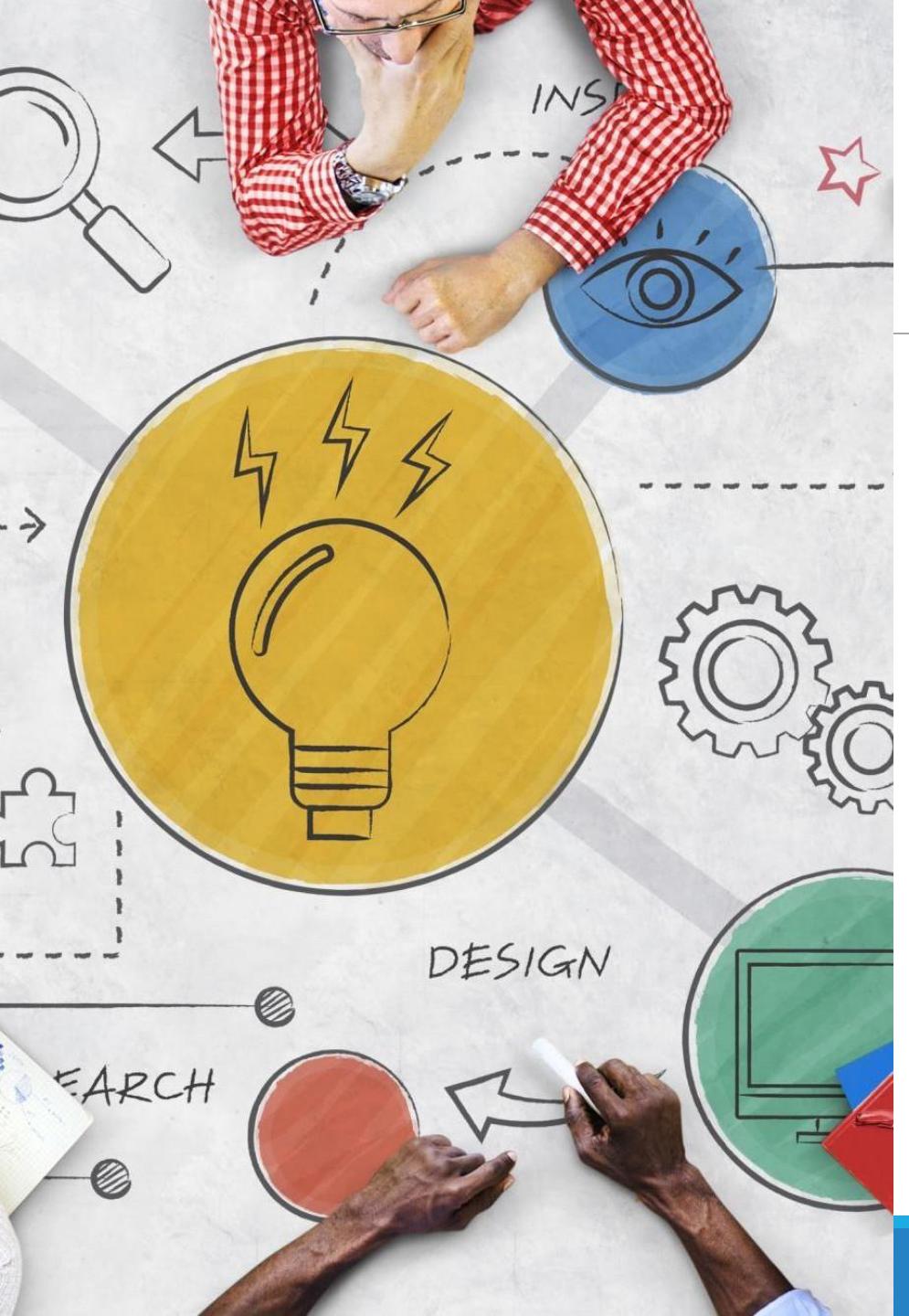
Recommended Books

Deep Learning with PyTorch Step-by-Step by Daniel Voigt Godoy



Course Learning Outcomes

No	CLO	Domain	Taxonomy Level	PLO
1	Understand the view geometry concepts, multi-scale representation, edge detection and detection of other primitives, stereo, motion and object recognition.	Cognitive		
2	Assess which methods to use for solving a given problem, and analyse the accuracy of the methods Skills	Cognitive		
3	Apply appropriate image processing methods for image filtering, image restoration, image reconstruction, segmentation, classification and representation	Cognitive		



Outline

Image Segmentation – U-Net

The Original U-Net

The Original U-Net (2015)

U-Net: Convolutional Networks for Biomedical Image Segmentation

Olaf Ronneberger, Philipp Fischer, and Thomas Brox

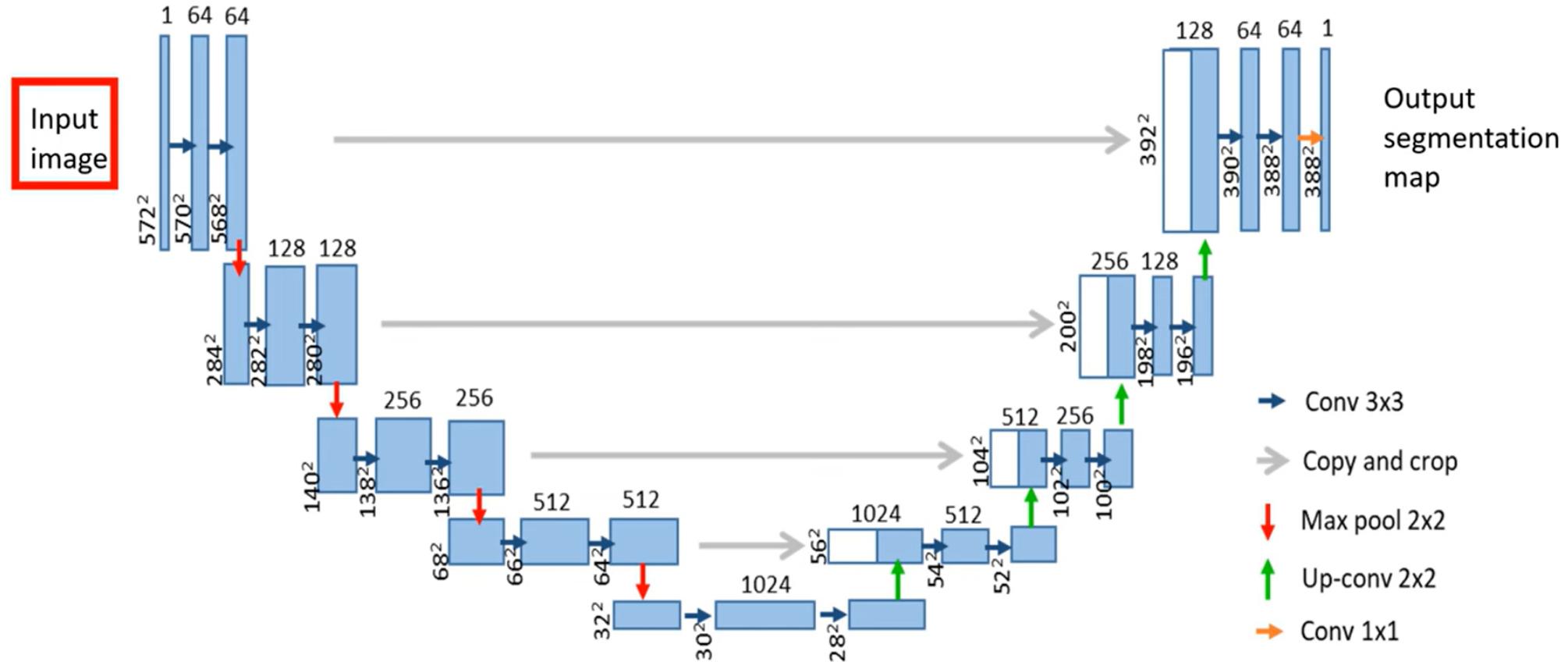
Computer Science Department and BIOSS Centre for Biological Signalling Studies,
University of Freiburg, Germany

`ronneber@informatik.uni-freiburg.de`,

WWW home page: <http://lmb.informatik.uni-freiburg.de/>

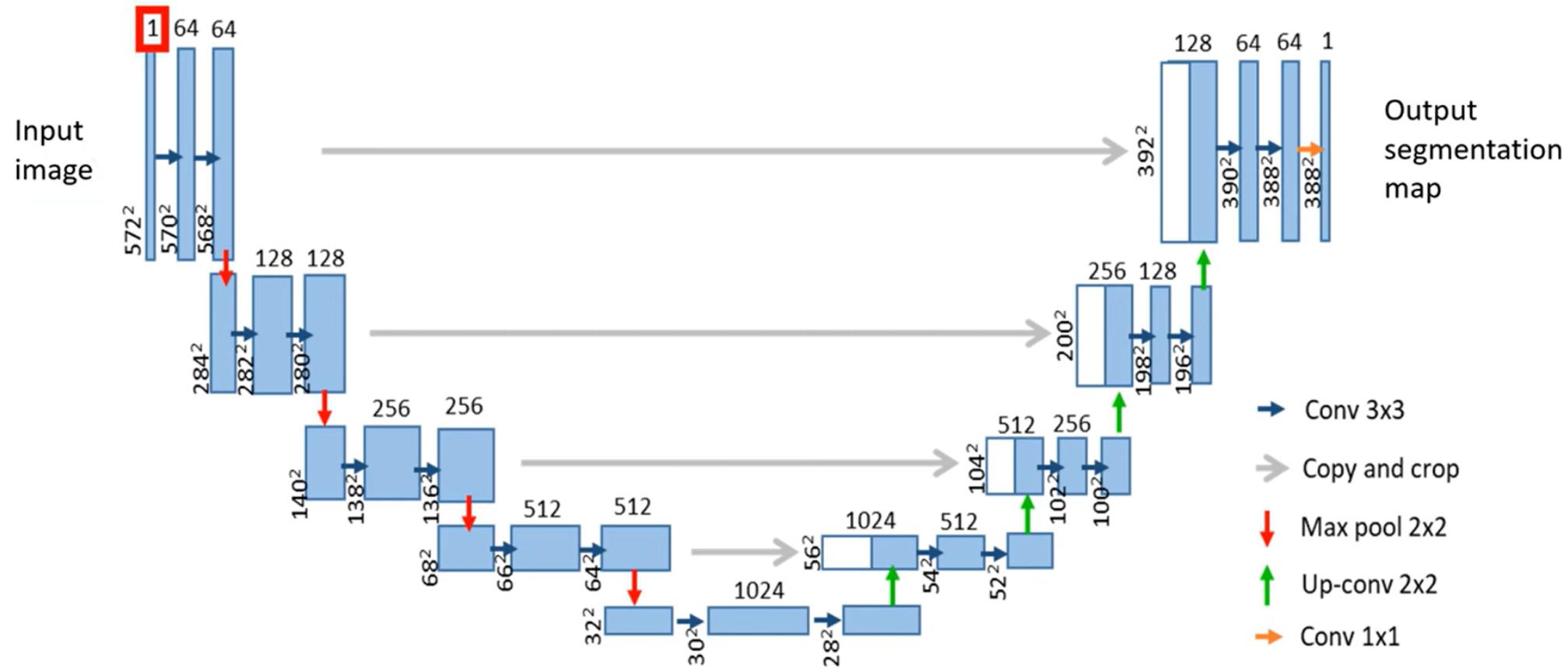
The Original U-Net

The network starts with a grayscale input image



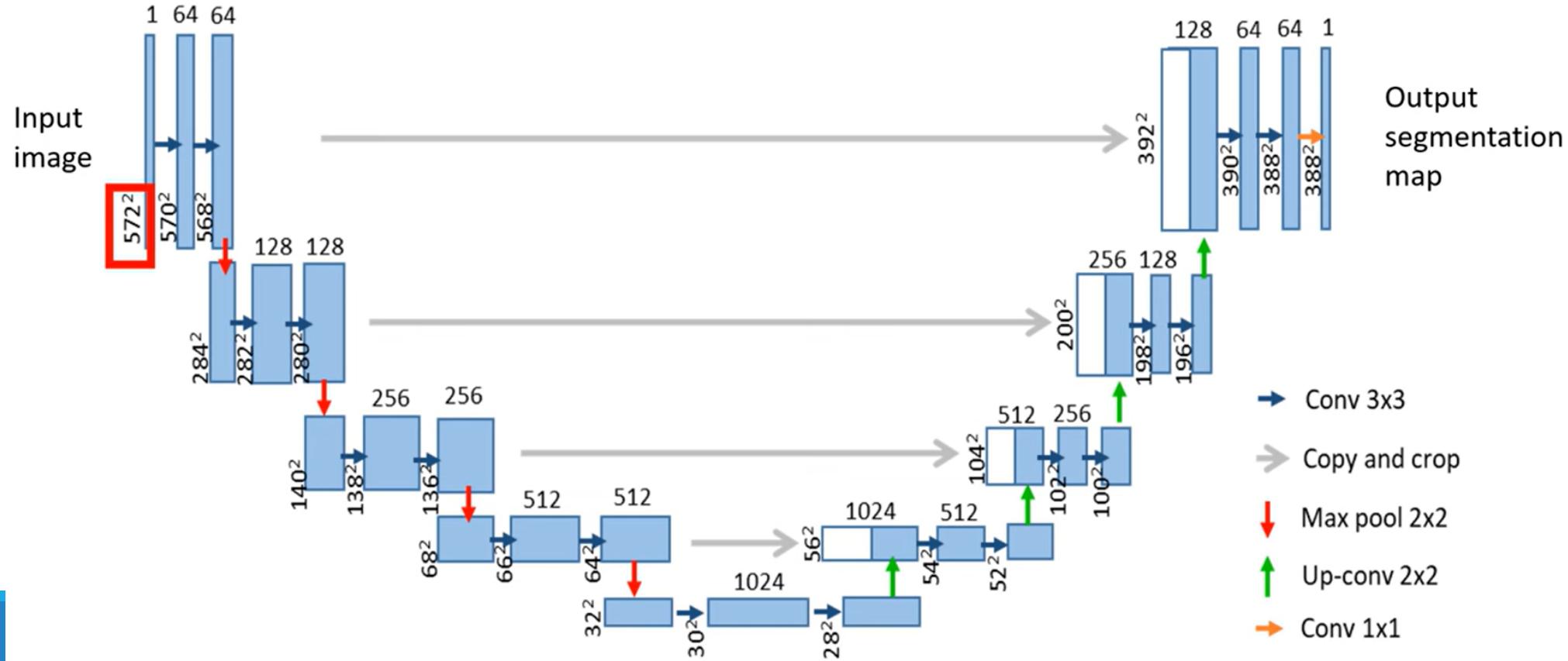
The Original U-Net

This indicates 1-input channel image



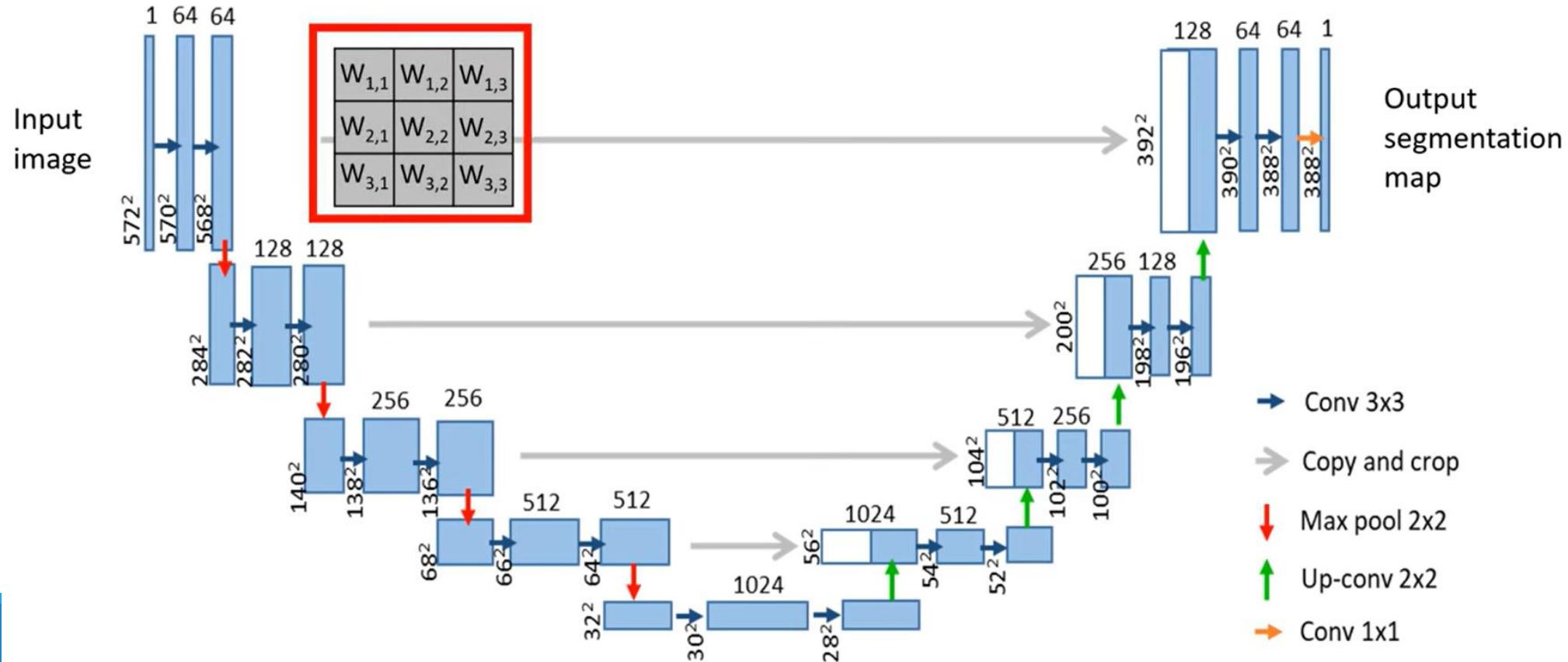
The Original U-Net

The image is rescaled to a 2D matrix of size 572×572



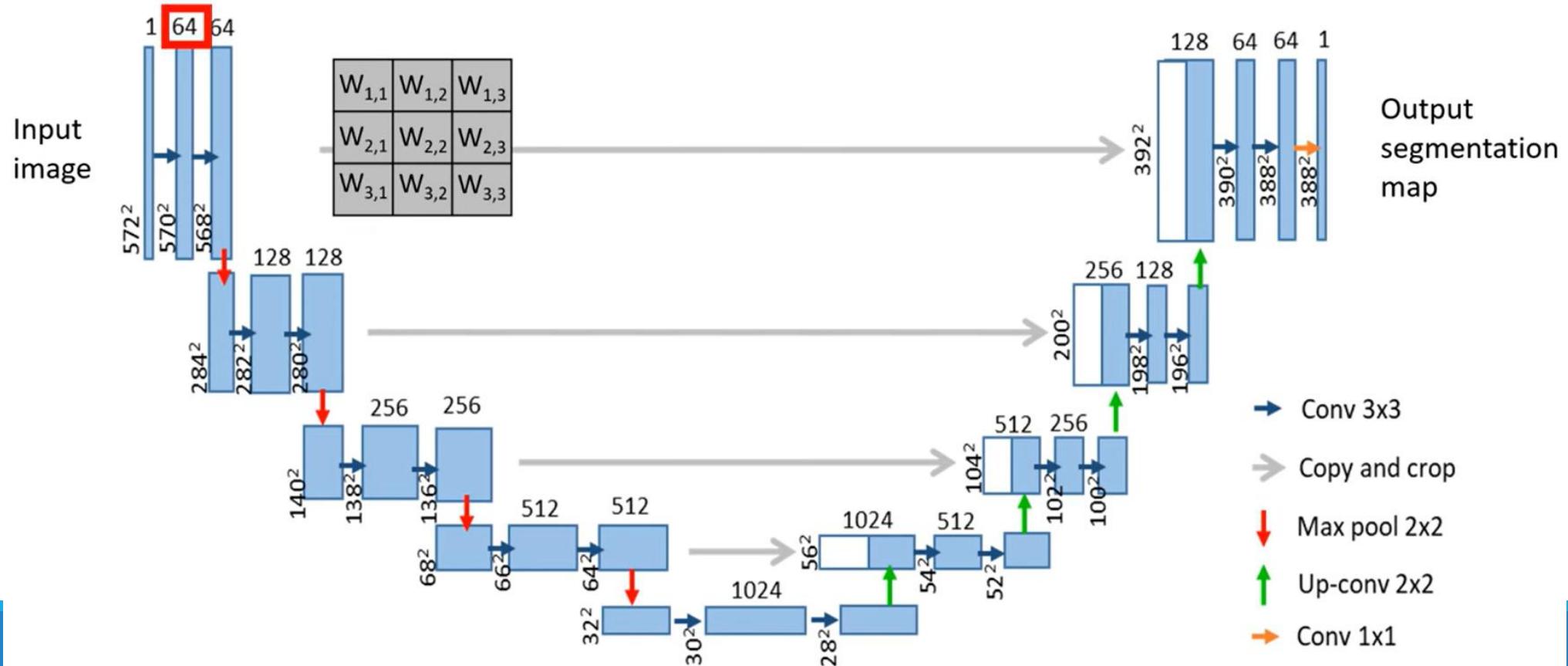
The Original U-Net

They used 64 different filters with a size of 3×3 in the first convolution step



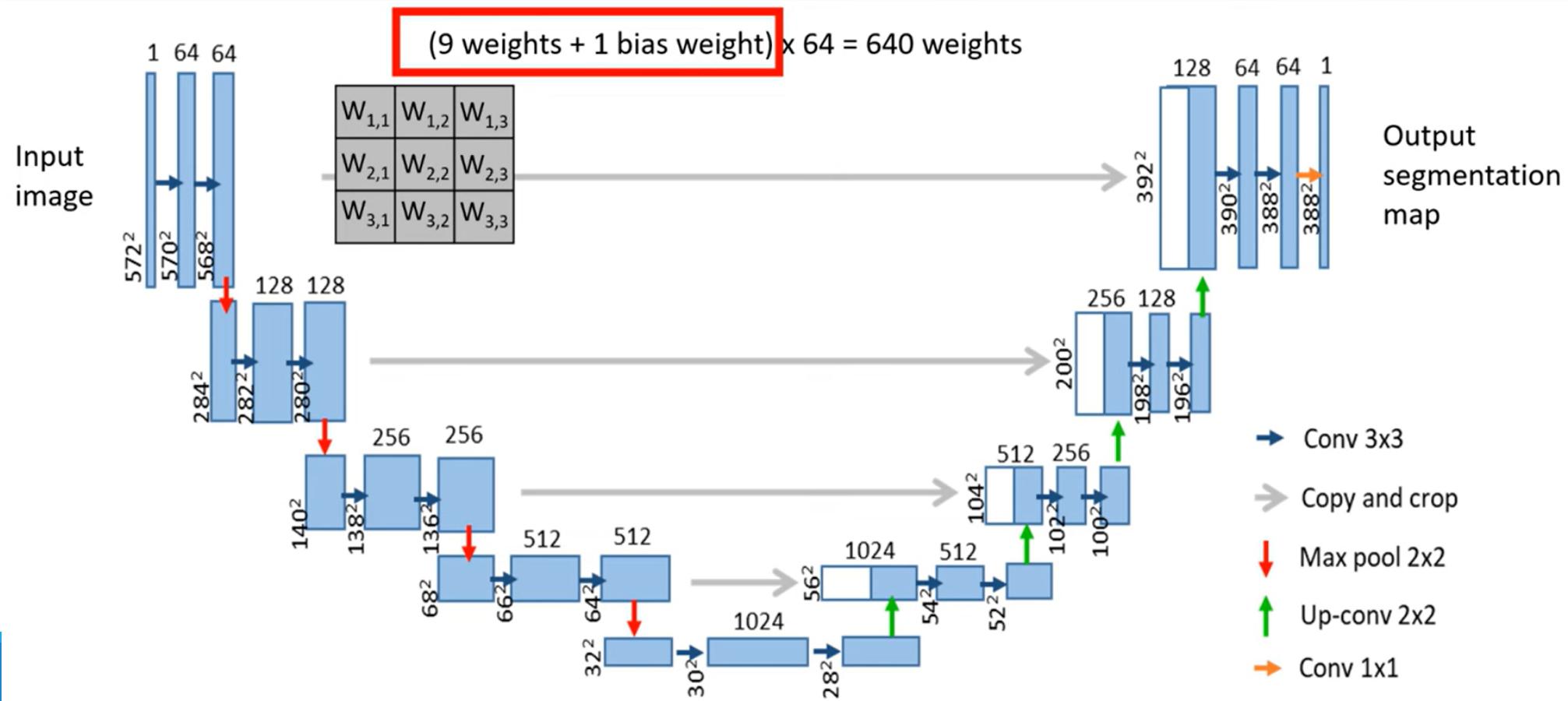
The Original U-Net

The number of output feature maps is 64 after the first convolution.



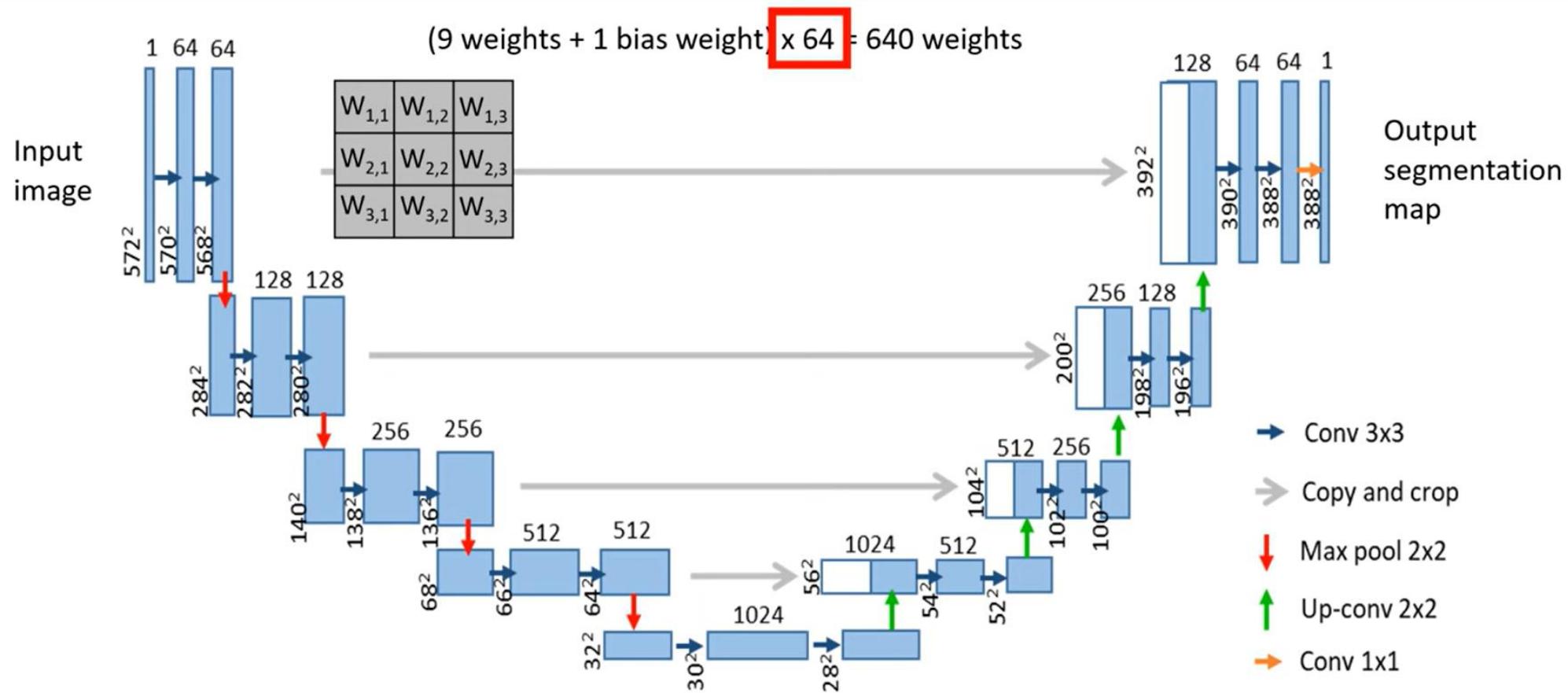
The Original U-Net

Since each filter has 9 weights + a bias weight



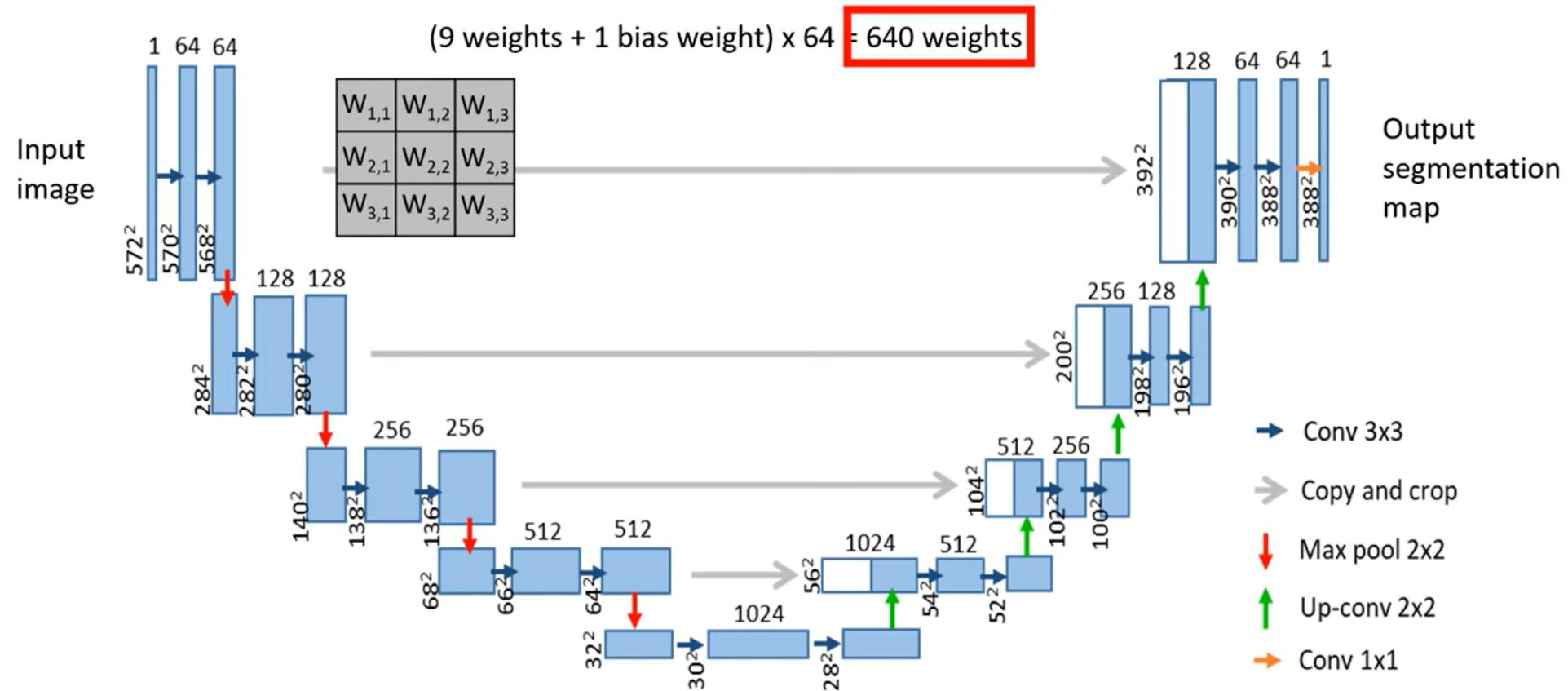
The Original U-Net

And that we have 64 such filters



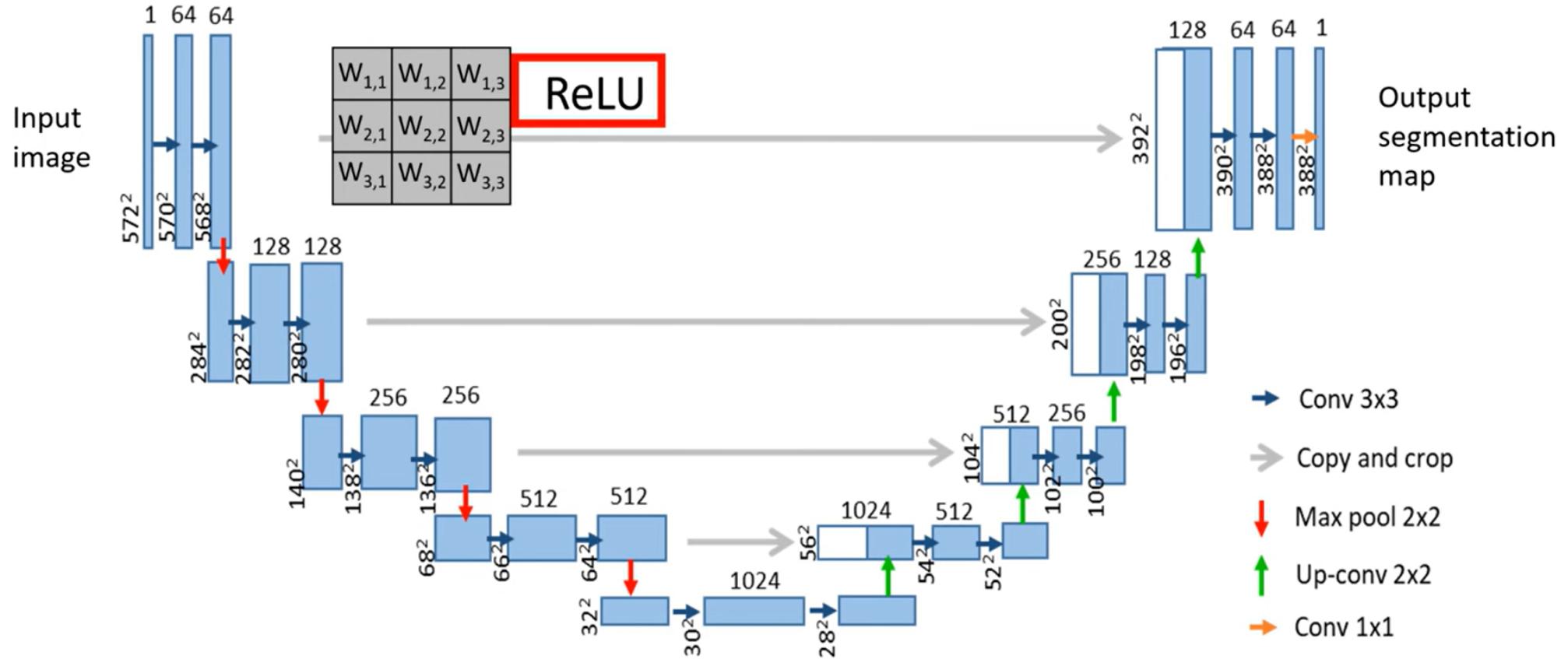
The Original U-Net

There will be **640 weights** to tune in the **first convolution** during the **training** process.



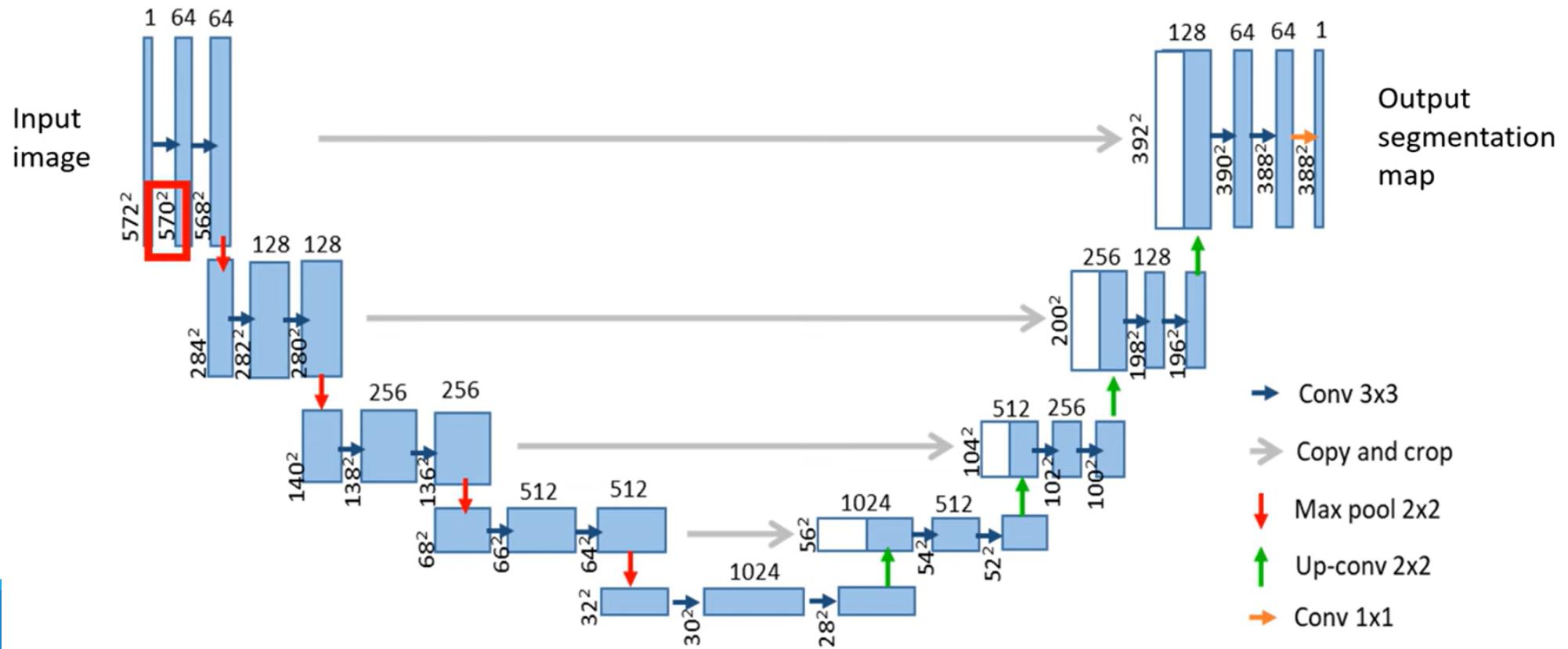
The Original U-Net

Every convolution is followed by applying the **ReLU** activation function.



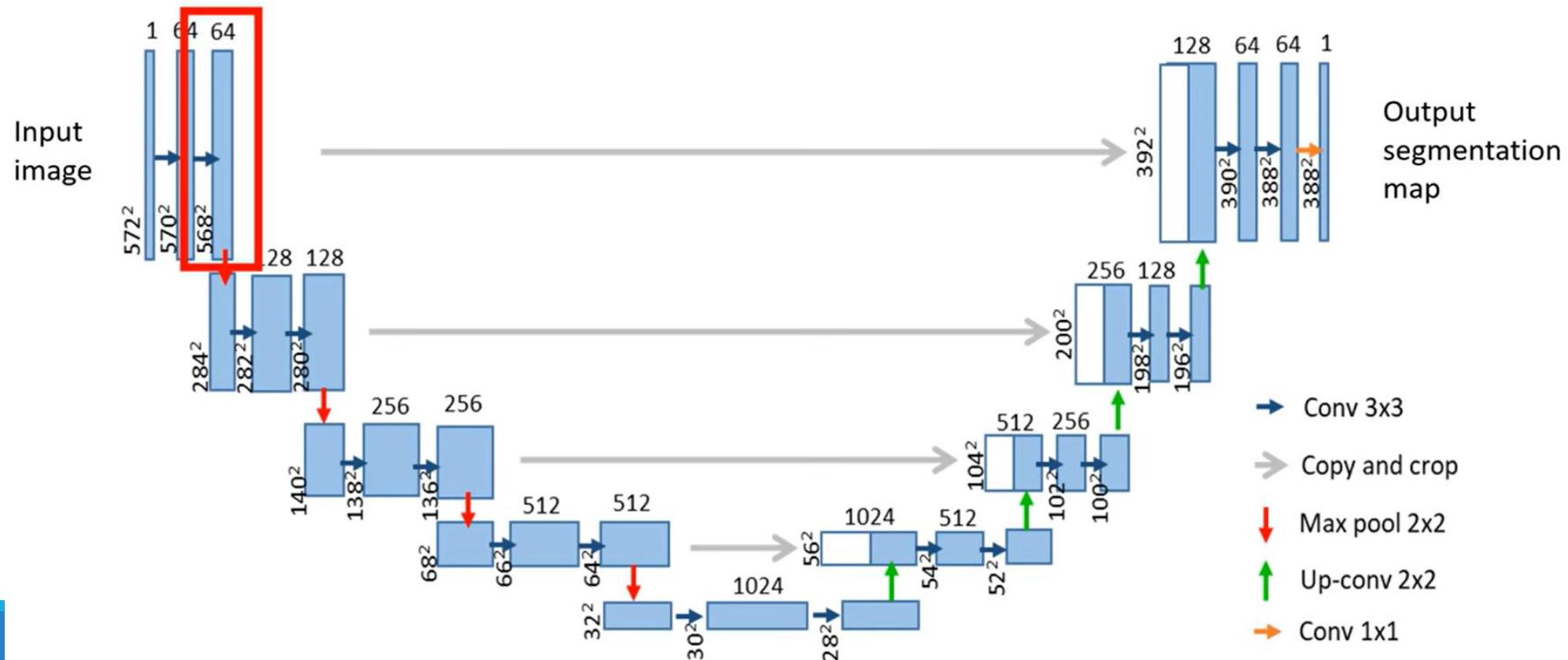
The Original U-Net

Since padding is not used in the previous convolution, the output feature maps are of size 570×570



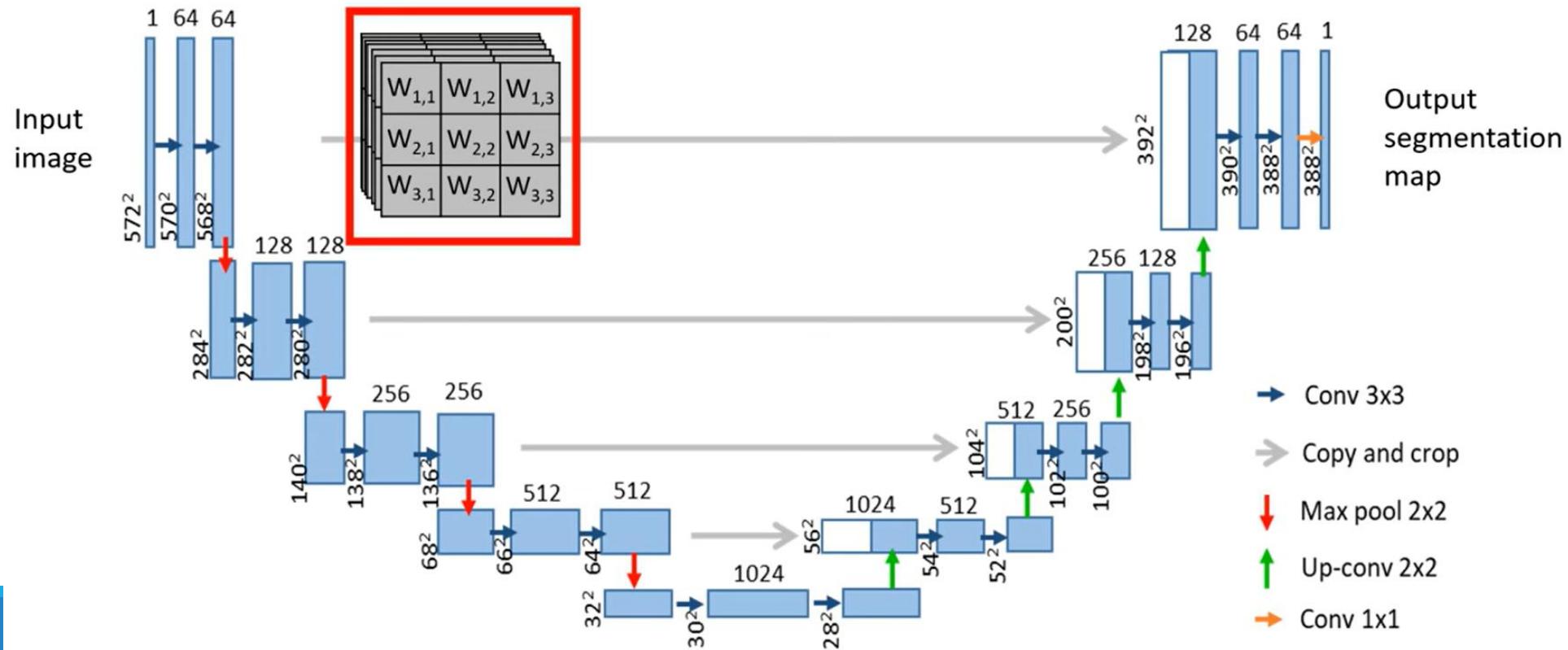
The Original U-Net

Then a second convolution of the 64 feature maps from the previous convolution



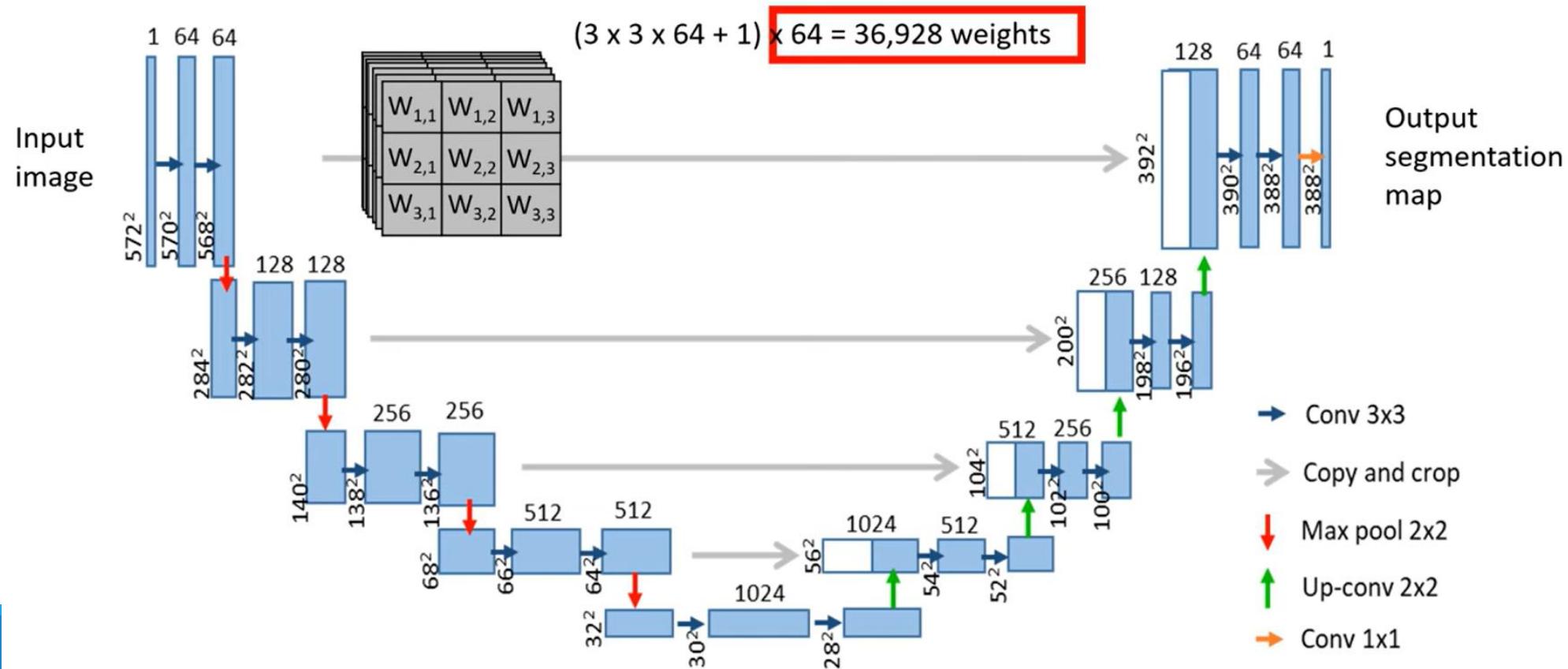
The Original U-Net

With a 3D 3×3 filter that has a depth of 64 because it will be applied on the 64 feature maps



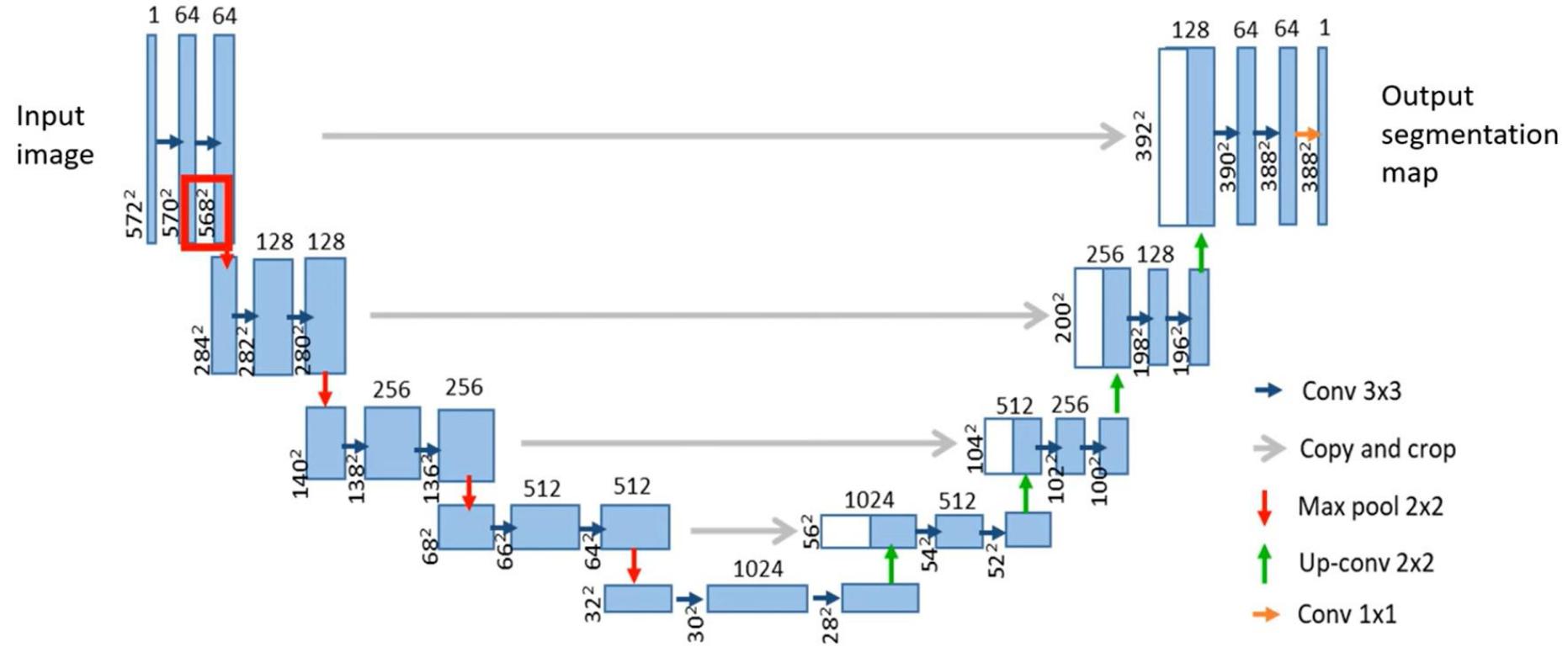
The Original U-Net

This second convolution is performed using 64 such deep filters, which output in 36,928 weights



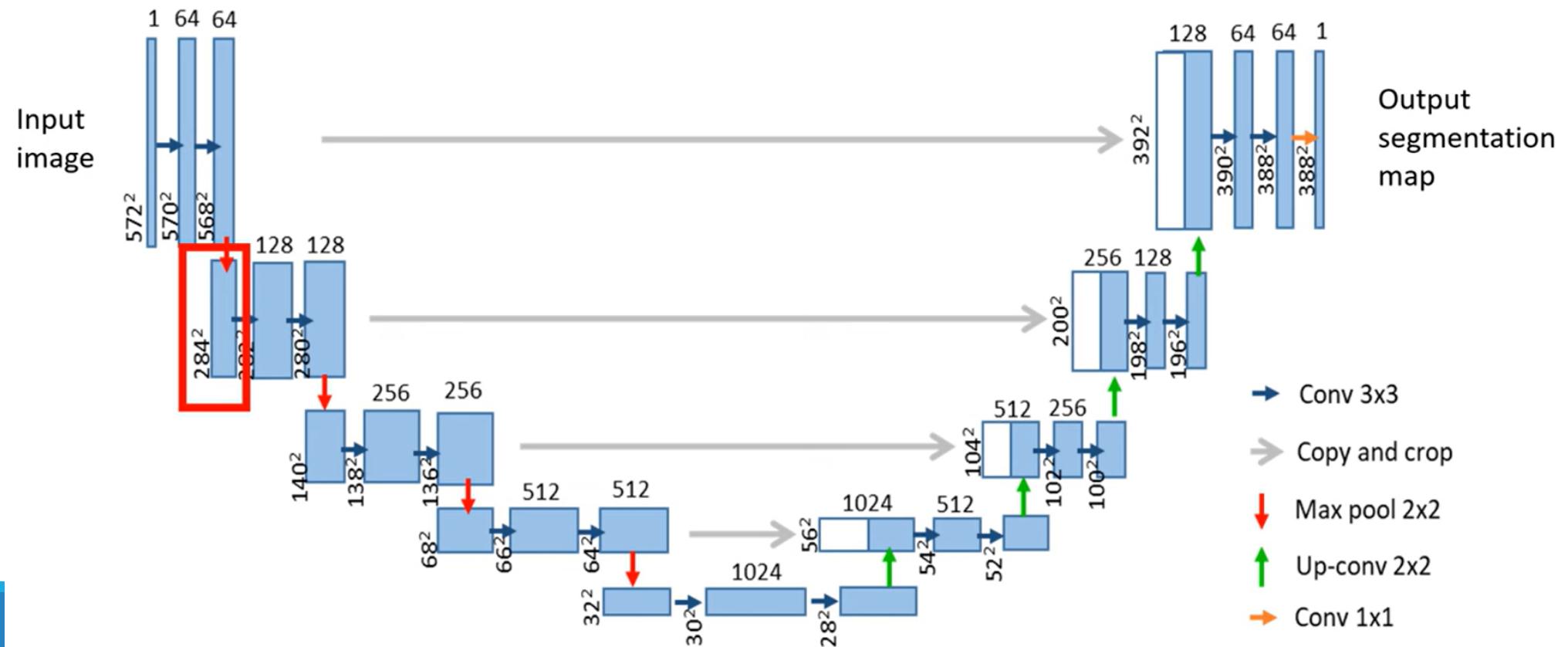
The Original U-Net

Since padding is not used in the previous convolution, the output feature maps are of size 568×568



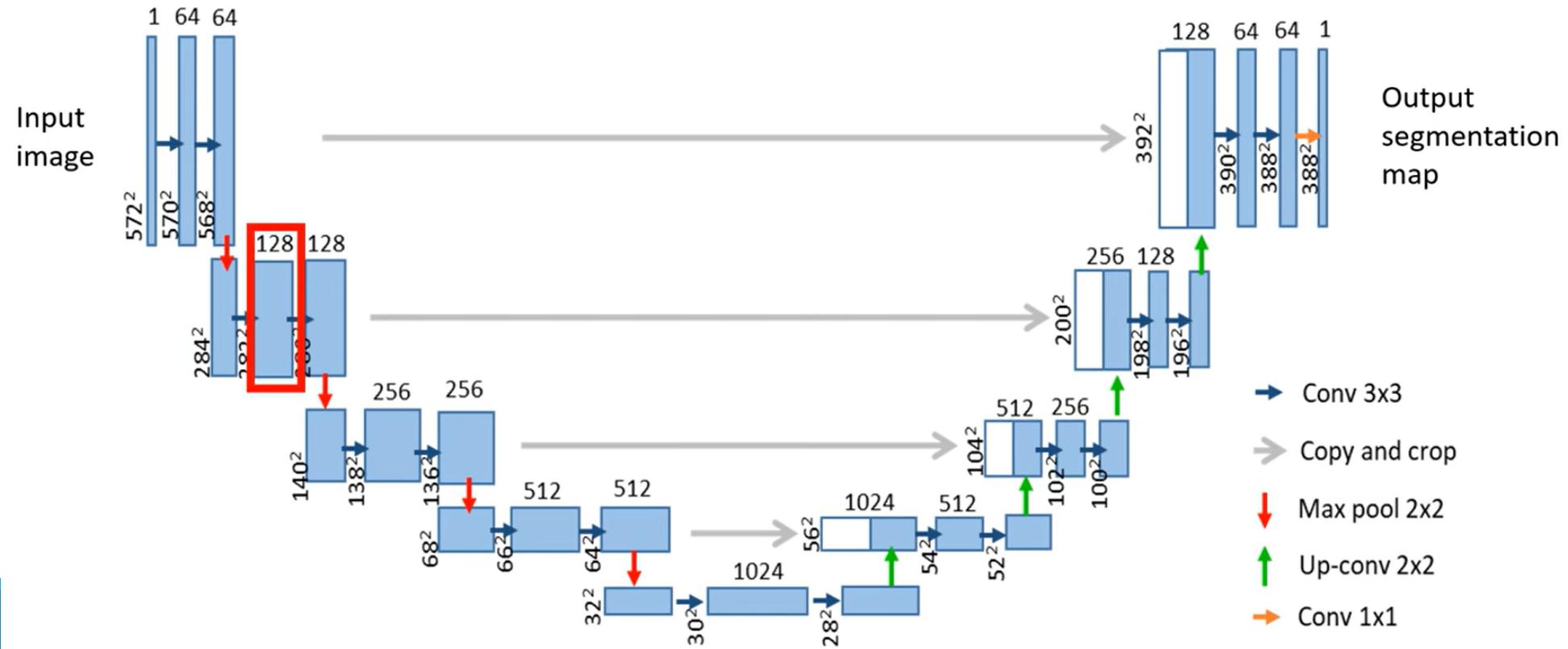
The Original U-Net

Then **MAX pooling** with a filter of size 2×2 is performed with $stride = 2$. The output is **64 pooled feature maps** with size 284×284 .



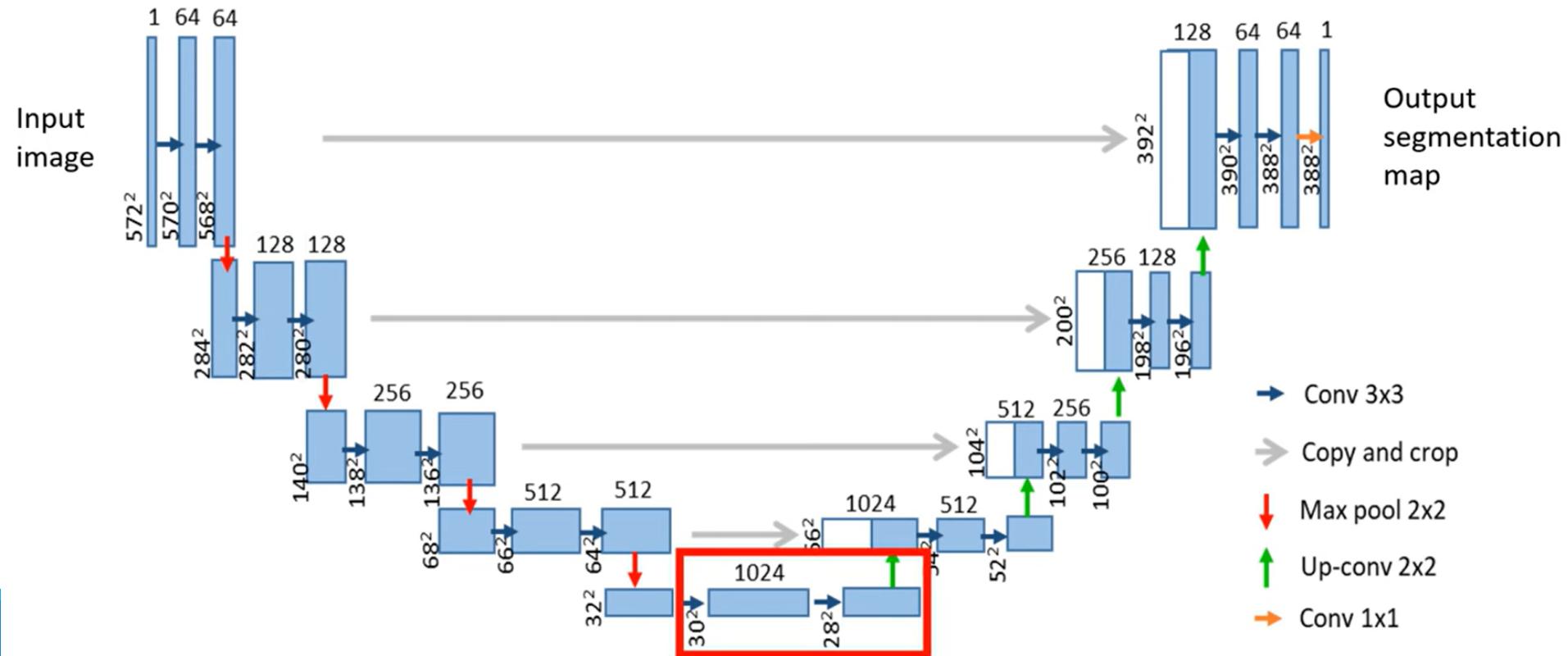
The Original U-Net

In each downsampling, number of filter is doubled, which gives 128 feature maps as output



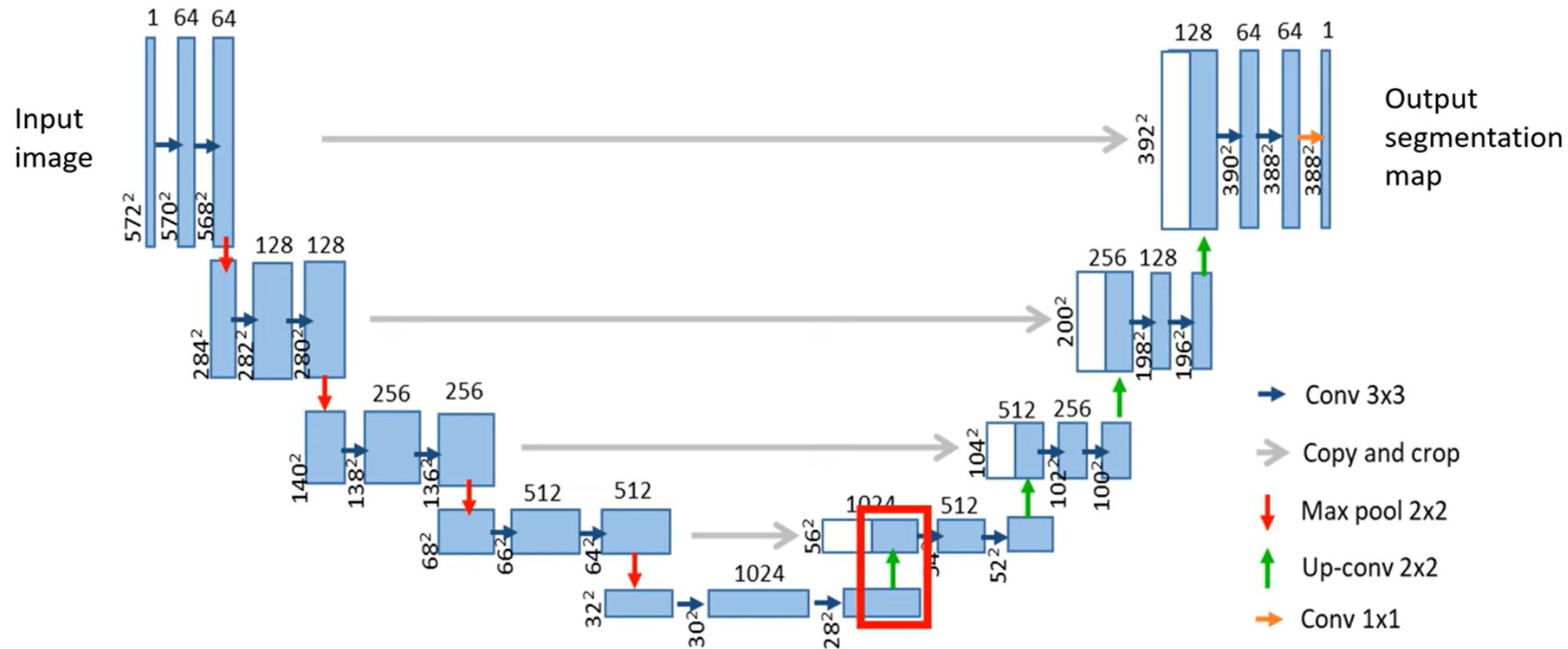
The Original U-Net

The smallest feature maps in the bottleneck have a size of 28×28



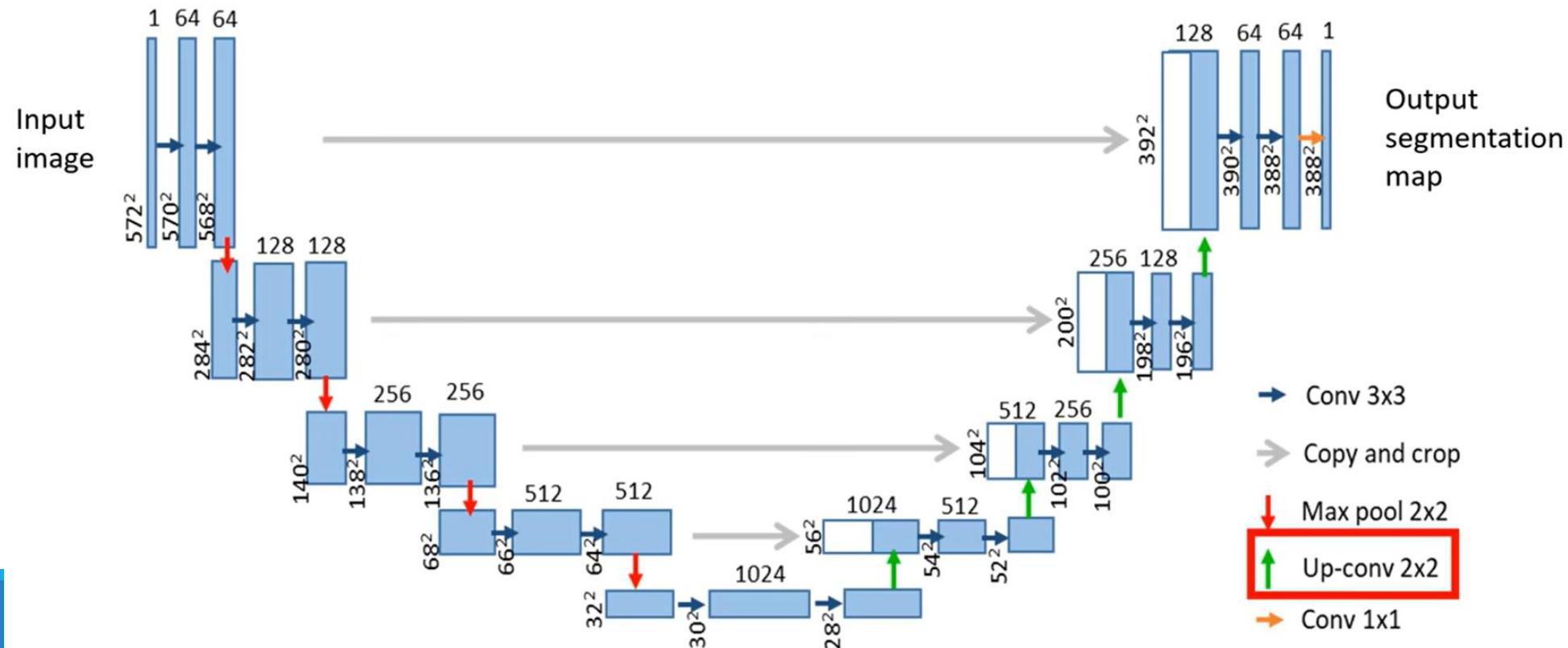
The Original U-Net

Then upscaling was performed using Transposed Convolution with 512 filters of size 2×2



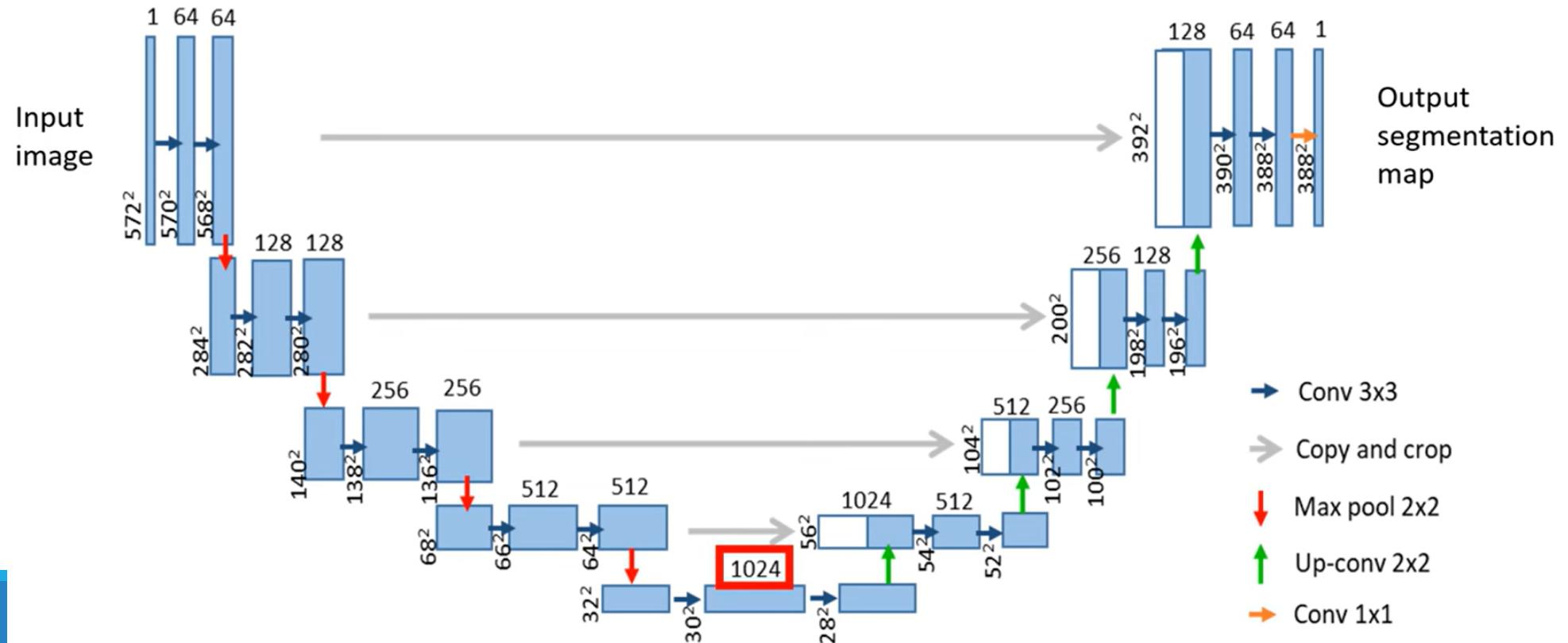
The Original U-Net

Filter size 2 × 2



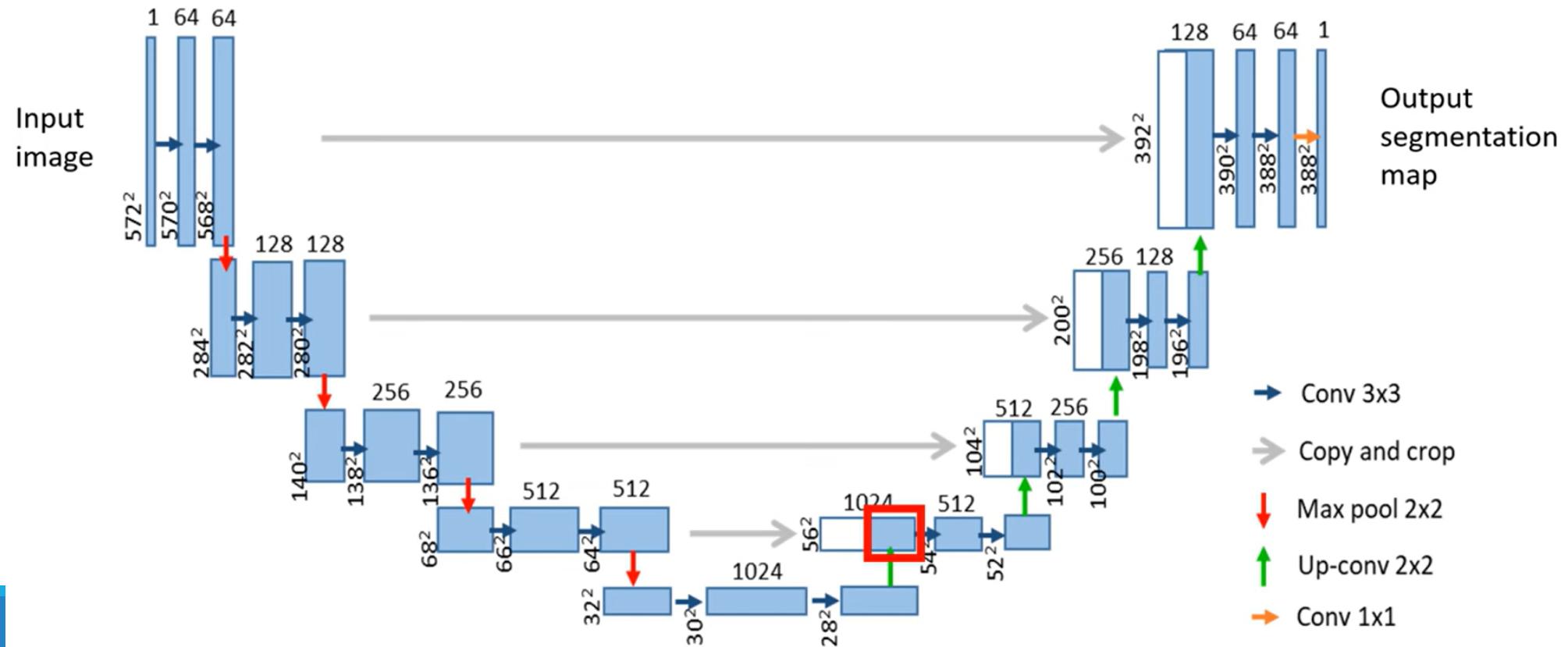
The Original U-Net

The depth of these filters is 1024



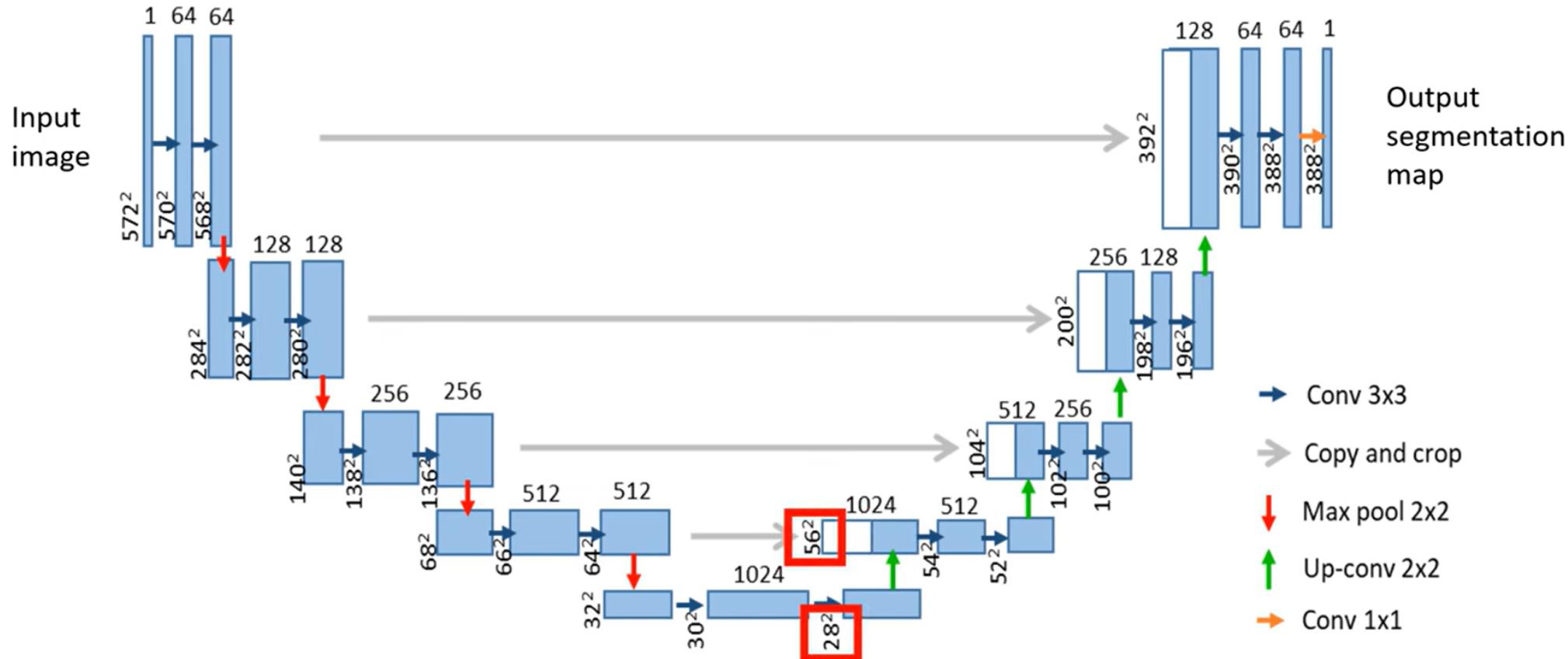
The Original U-Net

The output is 512 upsampled feature maps



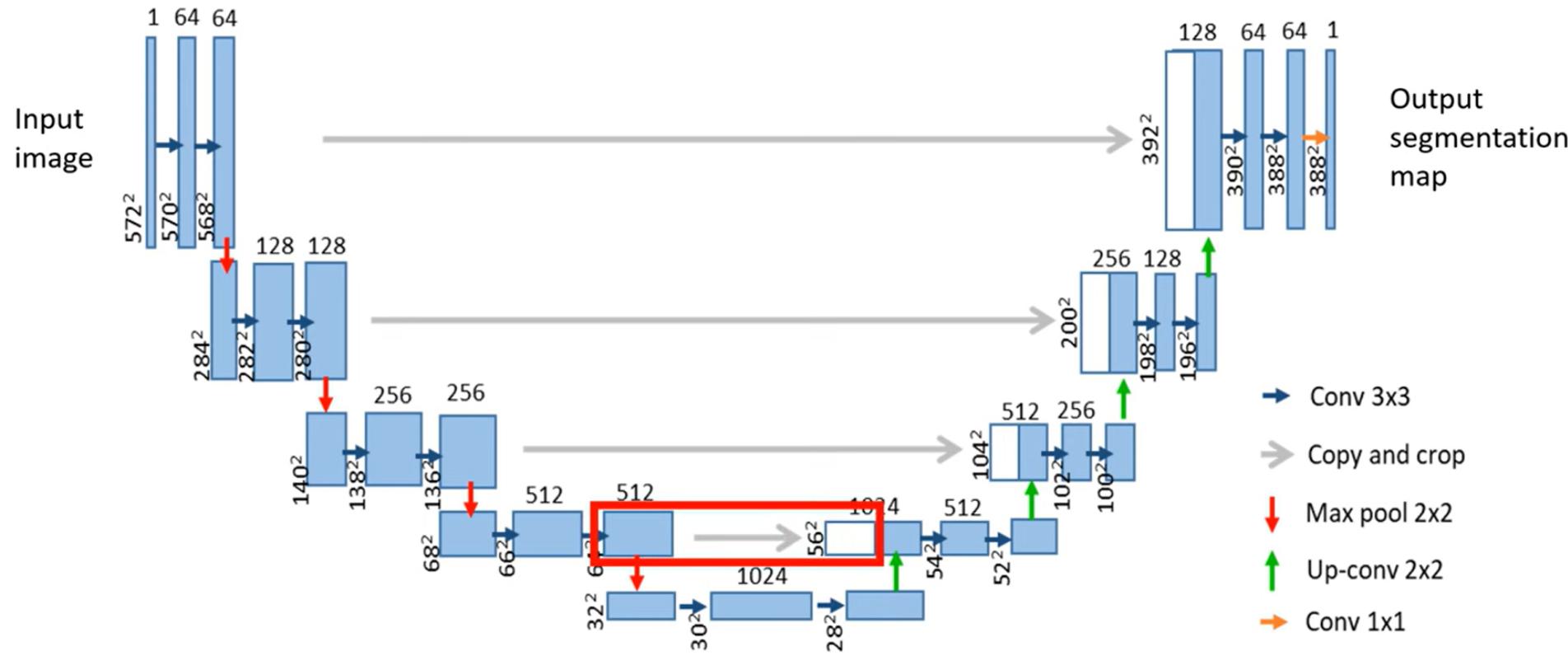
The Original U-Net

These feature maps are double in size compared to the previous feature maps



The Original U-Net

The **upsampled feature maps** were then **concatenated** with the corresponding **feature maps in the downsampling phase**



Thank you