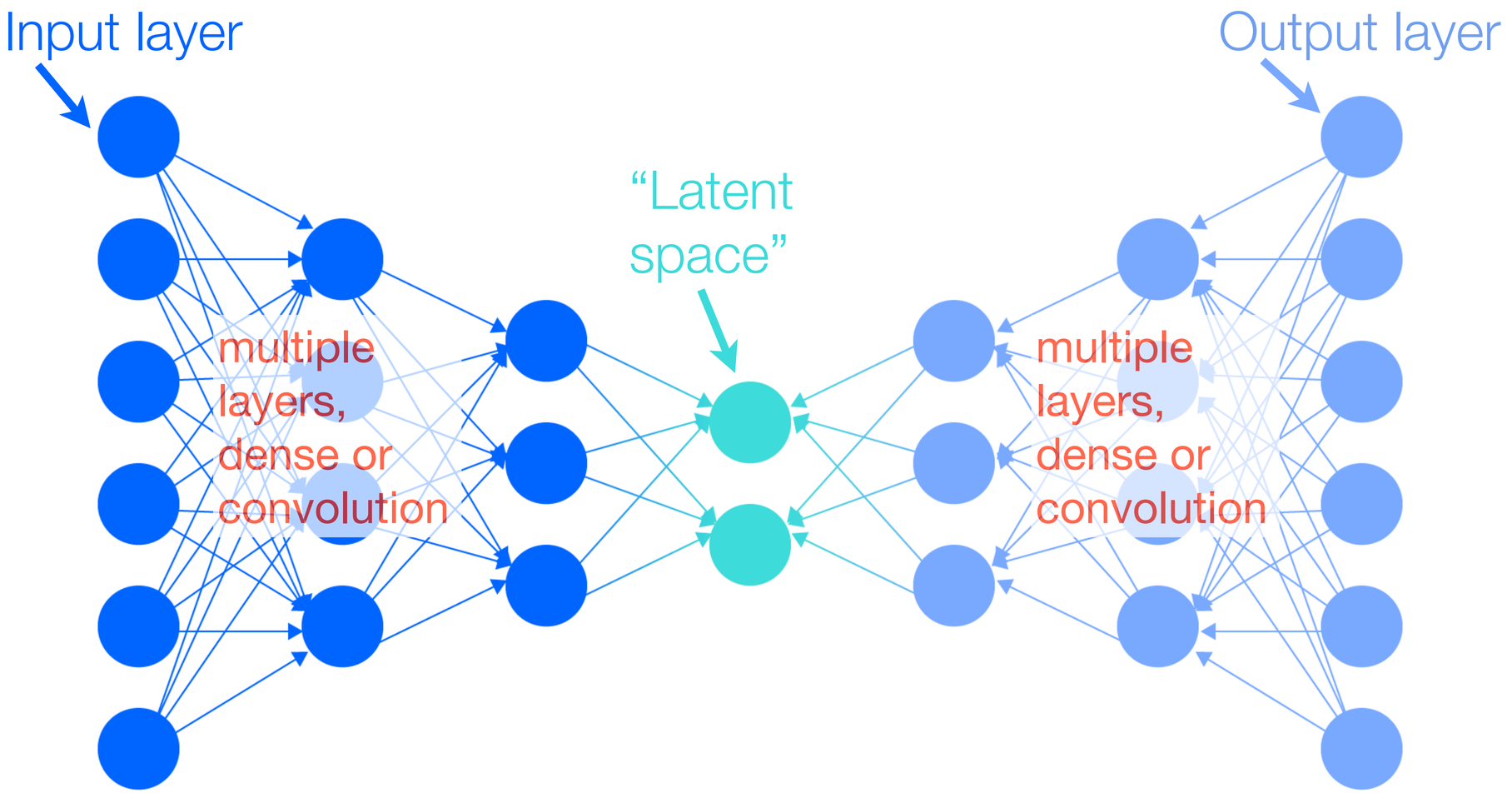
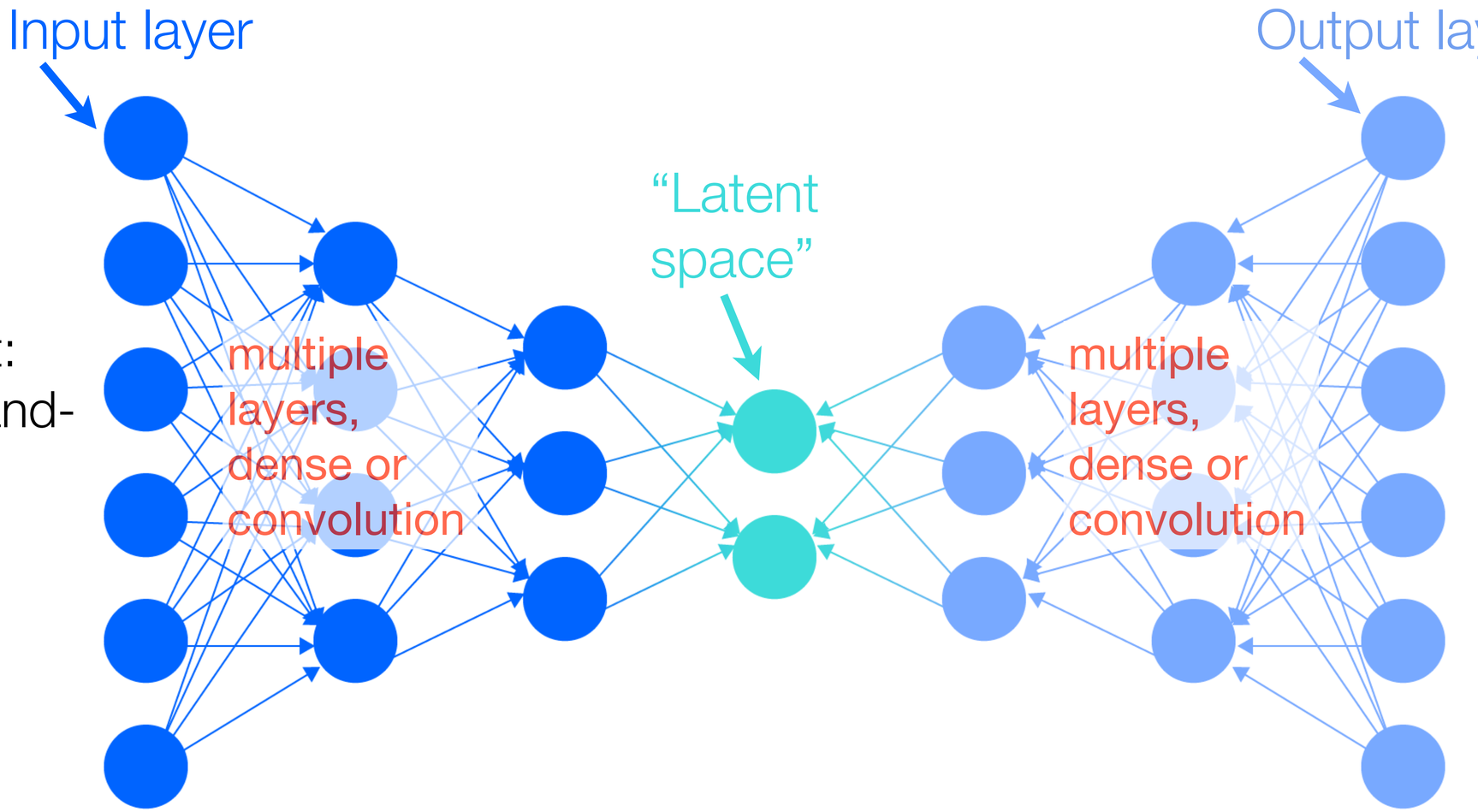
Autoencoders



https://www.ibm.com/think/topics/variational-autoencoder

https://medium.com/@divakar1591/autoencoders-6fab1a9a9f9c

Autoencoders



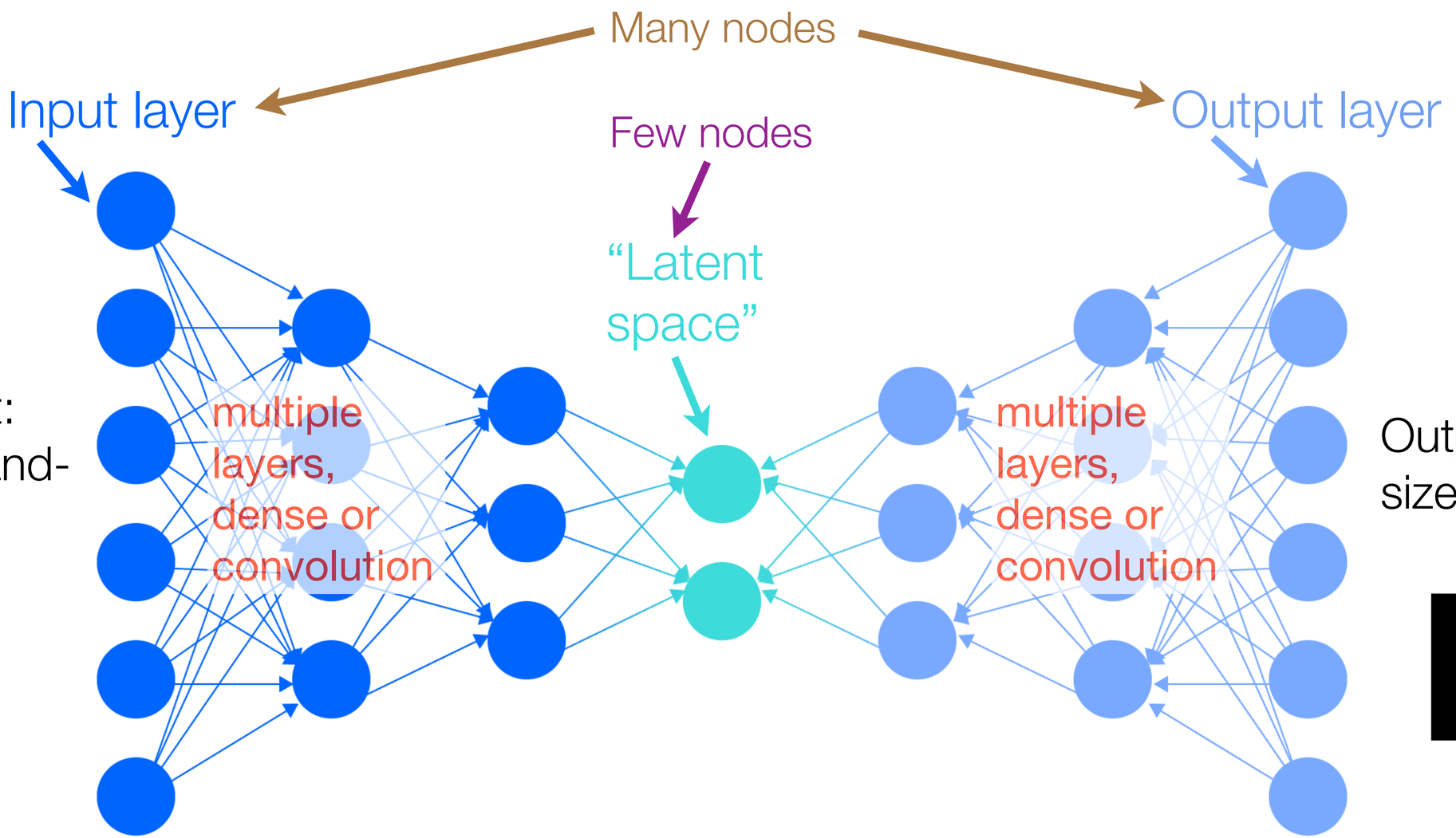
Example input: 28x28-784 hand-written digits.

Output: same size as inputs

https://www.ibm.com/think/topics/variational-autoencoder

https://medium.com/@divakar1591/autoencoders-6fab1a9a9f9c

Autoencoders



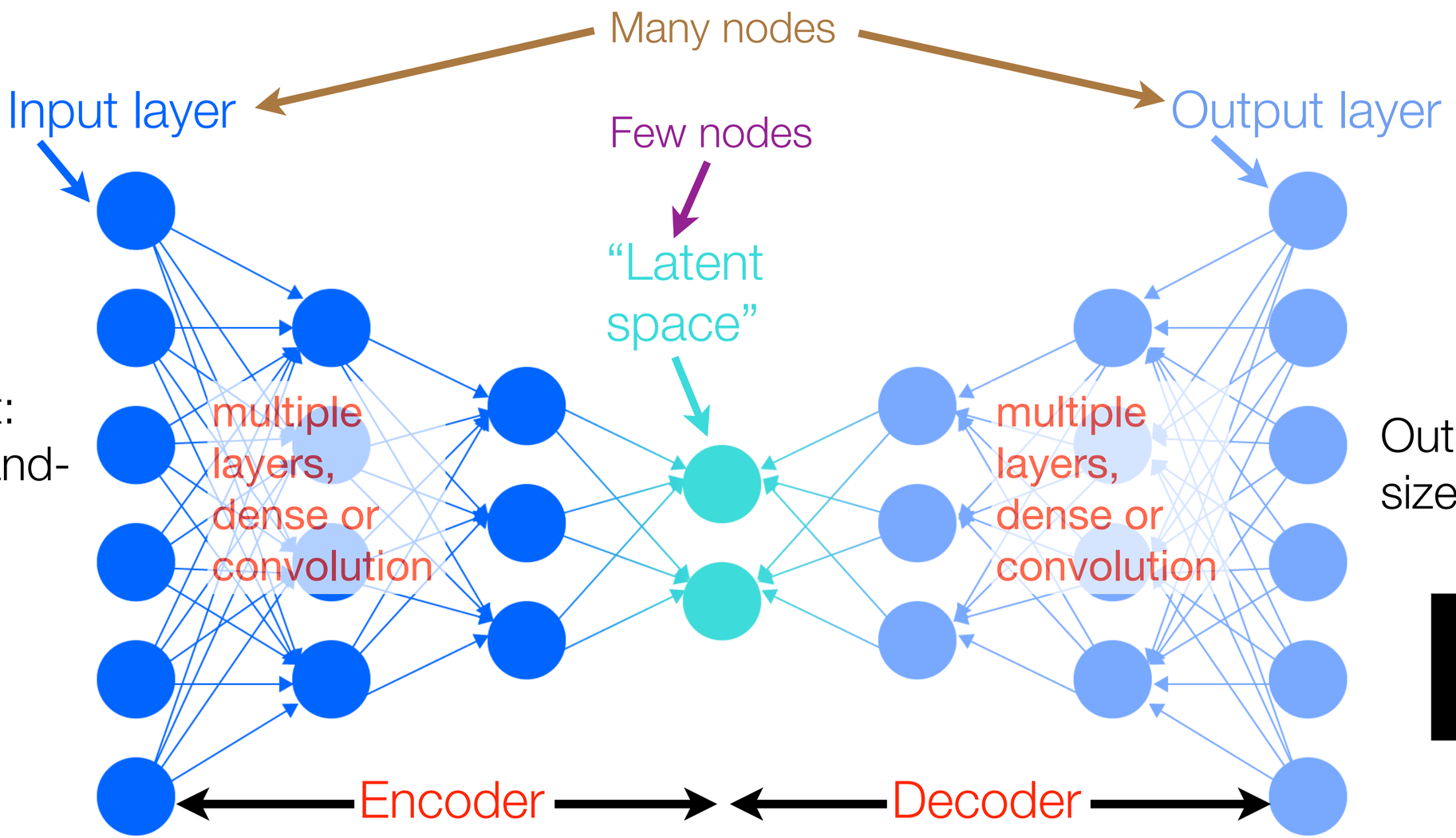
Example input: 28x28-784 hand-written digits.

Output: same size as inputs

https://www.ibm.com/think/topics/variational-autoencoder

https://medium.com/@divakar1591/autoencoders-6fab1a9a9f9c

Autoencoders



Example input: 28x28-784 hand-written digits.

Output: same size as inputs

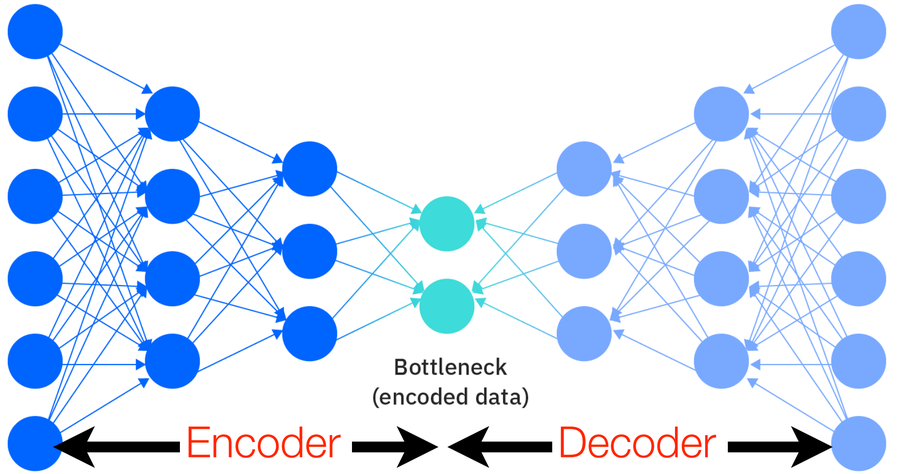
https://www.ibm.com/think/topics/variational-autoencoder

https://medium.com/@divakar1591/autoencoders-6fab1a9a9f9c

Autoencoder's goals

Example uses:

- Compression/dimension reduction: The decoder can reproduce the original image from a small number of values in the latent space. Like PCA, but nonlinear.

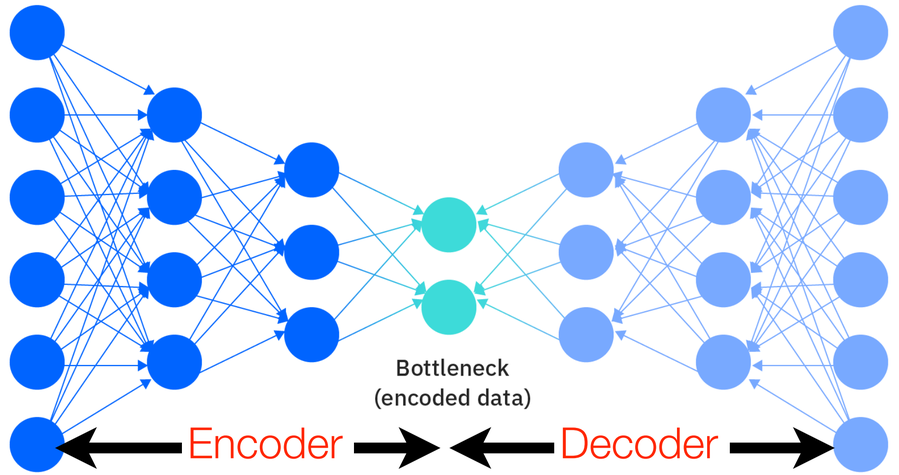


https://medium.com/@divakar1591/autoencoders-6fab1a9a9f9c

Autoencoder's goals

Example uses:

- Compression/dimension reduction: The decoder can reproduce the original image from a small number of values in the latent space. Like PCA, but nonlinear.
- Clustering: latent space is a small representation of large images. similar large images will have a similar latent space, and the latent space can be used to cluster the images.

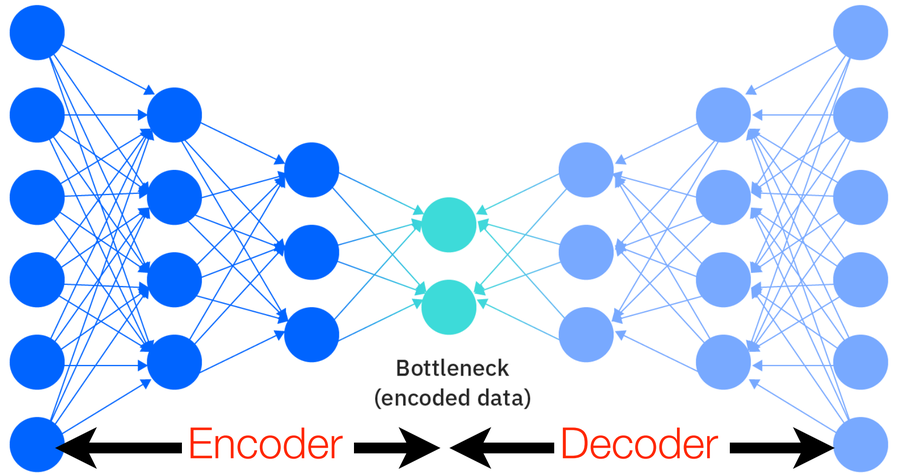


https://medium.com/@divakar1591/autoencoders-6fab1a9a9f9c

Autoencoder's goals

Example uses:

- Compression/dimension reduction: The decoder can reproduce the original image from a small number of values in the latent space. Like PCA, but nonlinear.
- Clustering: latent space is a small representation of large images. similar large images will have a similar latent space, and the latent space can be used to cluster the images.
- Noise reduction.

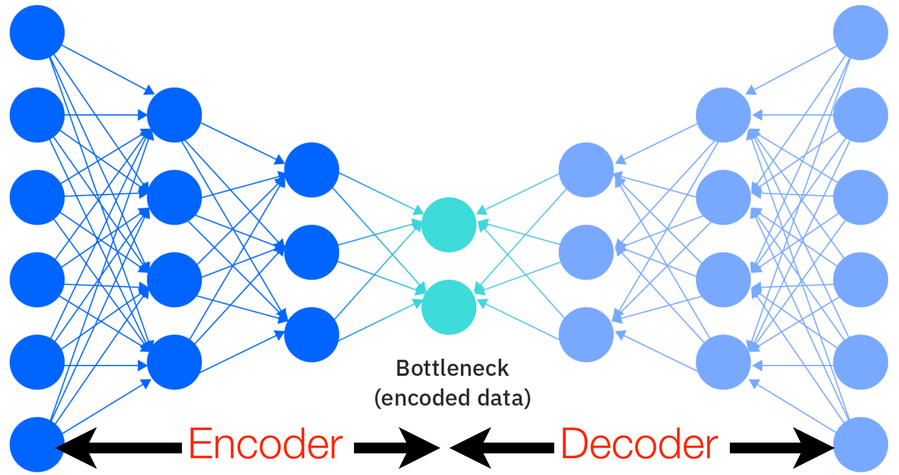


https://medium.com/@divakar1591/autoencoders-6fab1a9a9f9c

Autoencoder's goals

Example uses:

- Compression/dimension reduction: The decoder can reproduce the original image from a small number of values in the latent space. Like PCA, but nonlinear.
- Clustering: latent space is a small representation of large images. similar large images will have a similar latent space, and the latent space can be used to cluster the images.
- Noise reduction.
- U-Net!

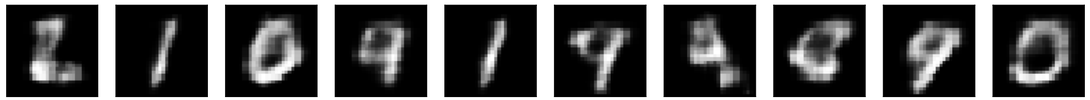


https://medium.com/@divakar1591/autoencoders-6fab1a9a9f9c

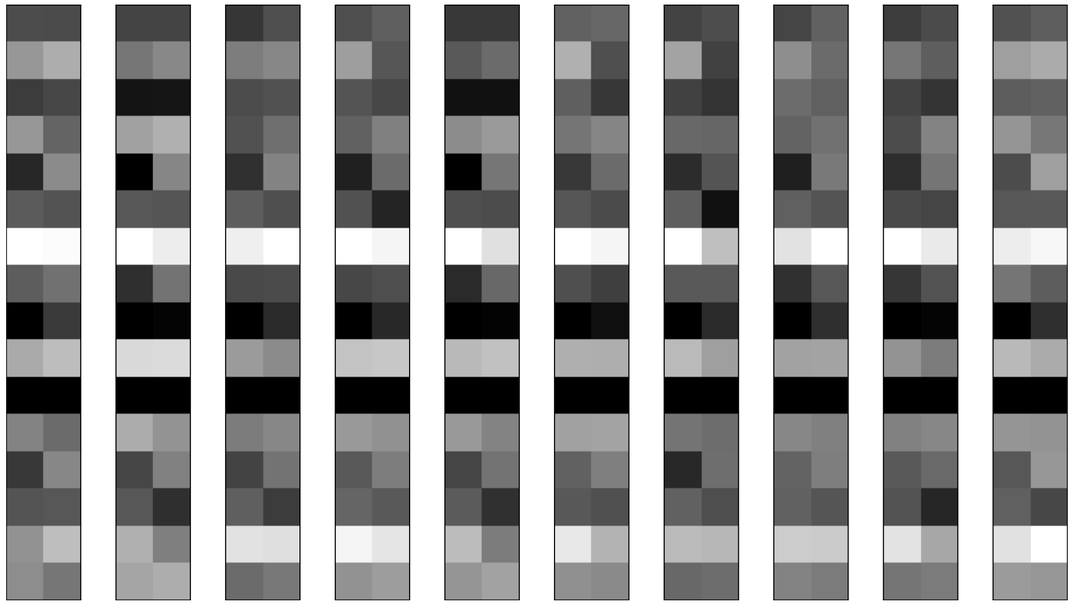
Autoencoder's training

Feed the network an image, and require the output to be as similar to the input as possible while passing through a very small latent space:

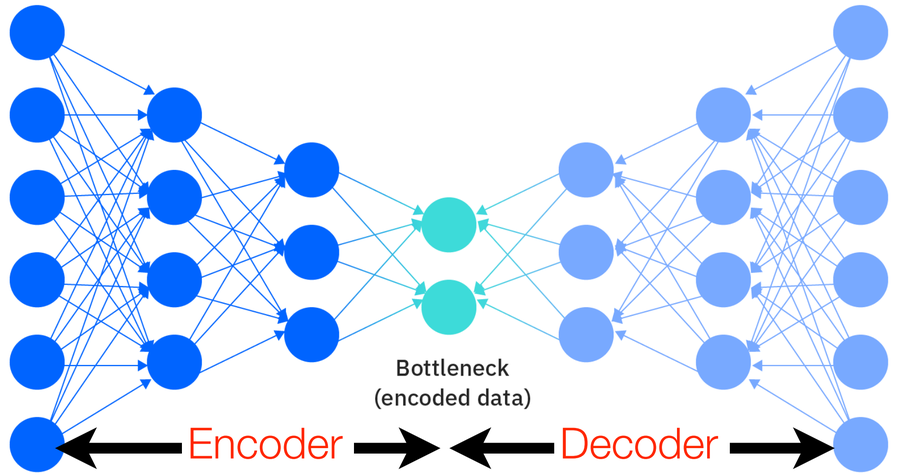
inputs (28x28=784)



outputs (28x28=784)



latent space (32)

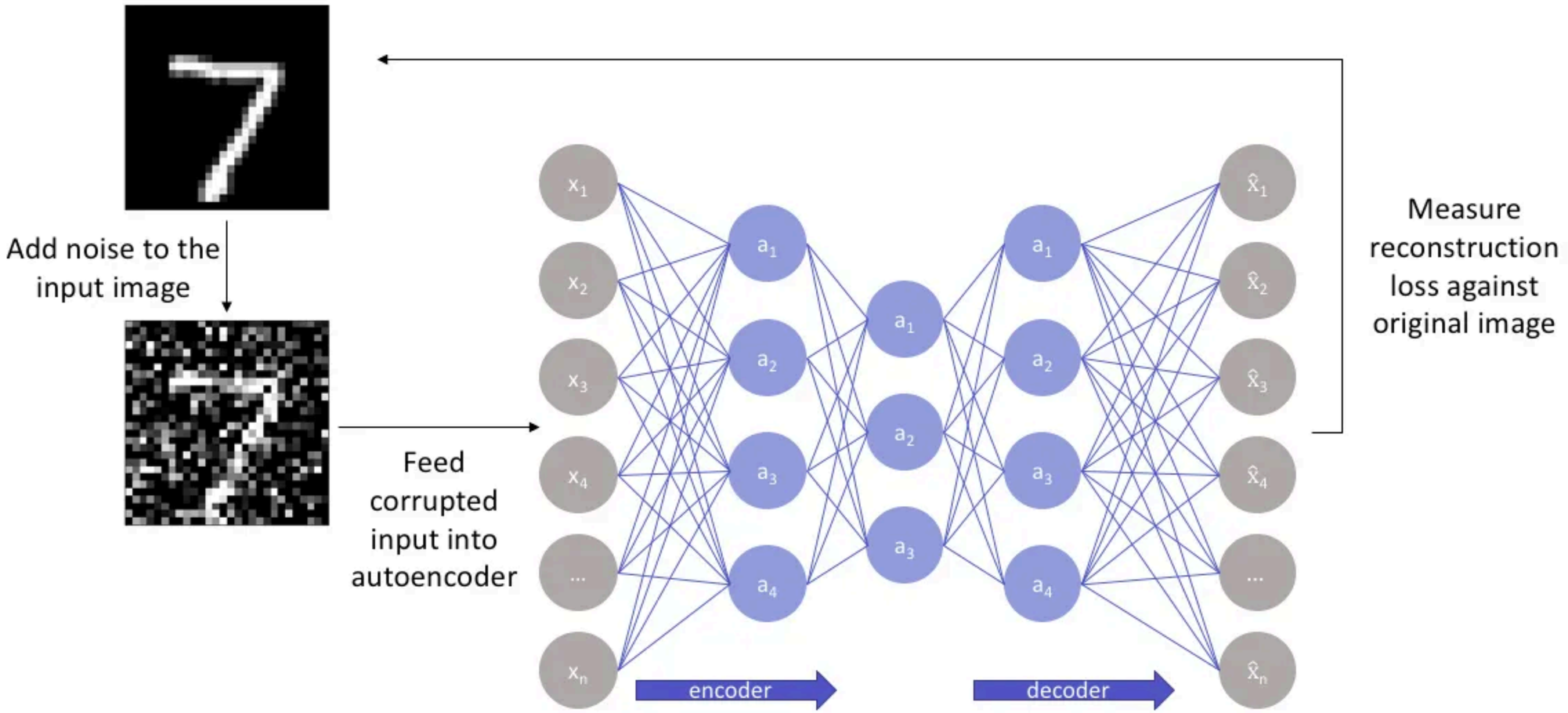


Code from: https://blog.keras.io/building-autoencoders-in-keras.html

https://medium.com/@divakar1591/autoencoders-6fab1a9a9f9c

Autoencoder's application: noise reduction

"The denoising autoencoder gets rid of noise by learning a representation of the input where the noise can be filtered out easily."



https://medium.com/@divakar1591/autoencoders-6fab1a9a9f9c

How does the decoder work: Up-convolution

input

a	b
c	d

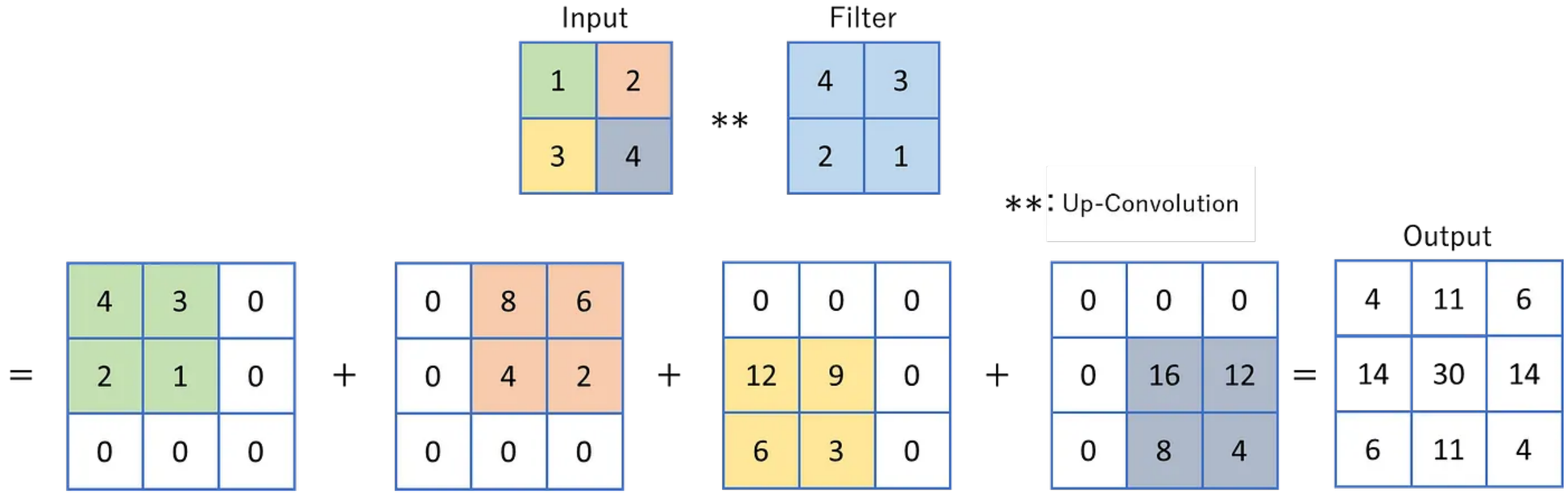
duplicate and expand

a	a	b	b
a	a	b	b
c	c	d	d
c	c	d	d

apply 2x2 convolution kernel

A	B	C
D	E	F
G	H	I

How does the decoder work: Up-convolution



"Up-convolutions: **expand & duplicate each element from the input feature map to the size of the filter**. **The filter is then applied over each of these expanded regions.**

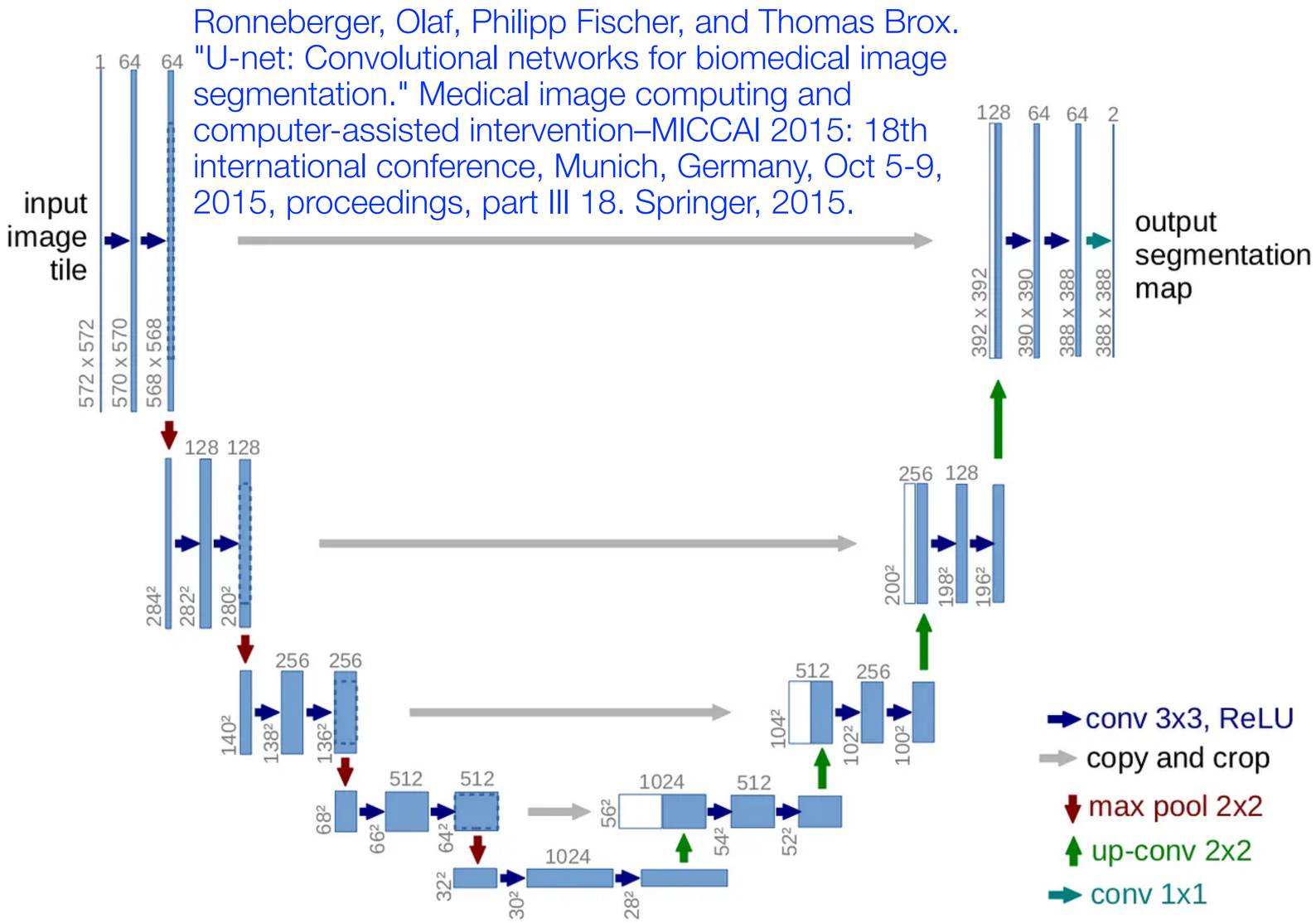
For example, the expanded green input is composed of four 1s. Likewise, the expanded red, yellow and grey regions are initially filled with just 2s, 3s and 4s, respectively. The filter is applied over each of these regions & the results are summed to form the output feature map.

Example: the spatial dimensions are doubled, if a **2x2 filter** is used with a **stride of 2**."

Alejandro Ito Aramendia https://medium.com/@alejandro.itoaramendia/decoding-the-u-net-a-complete-guide-810b1c6d56d8

A (very) popular use of autoencoders: U-Net

"The creation of the U-Net was a ground breaking discovery in the realm of **image segmentation**, a field focused on locating objects and boundaries within an image. This novel architecture proved to carry immense value in the analysis of biomedical images."

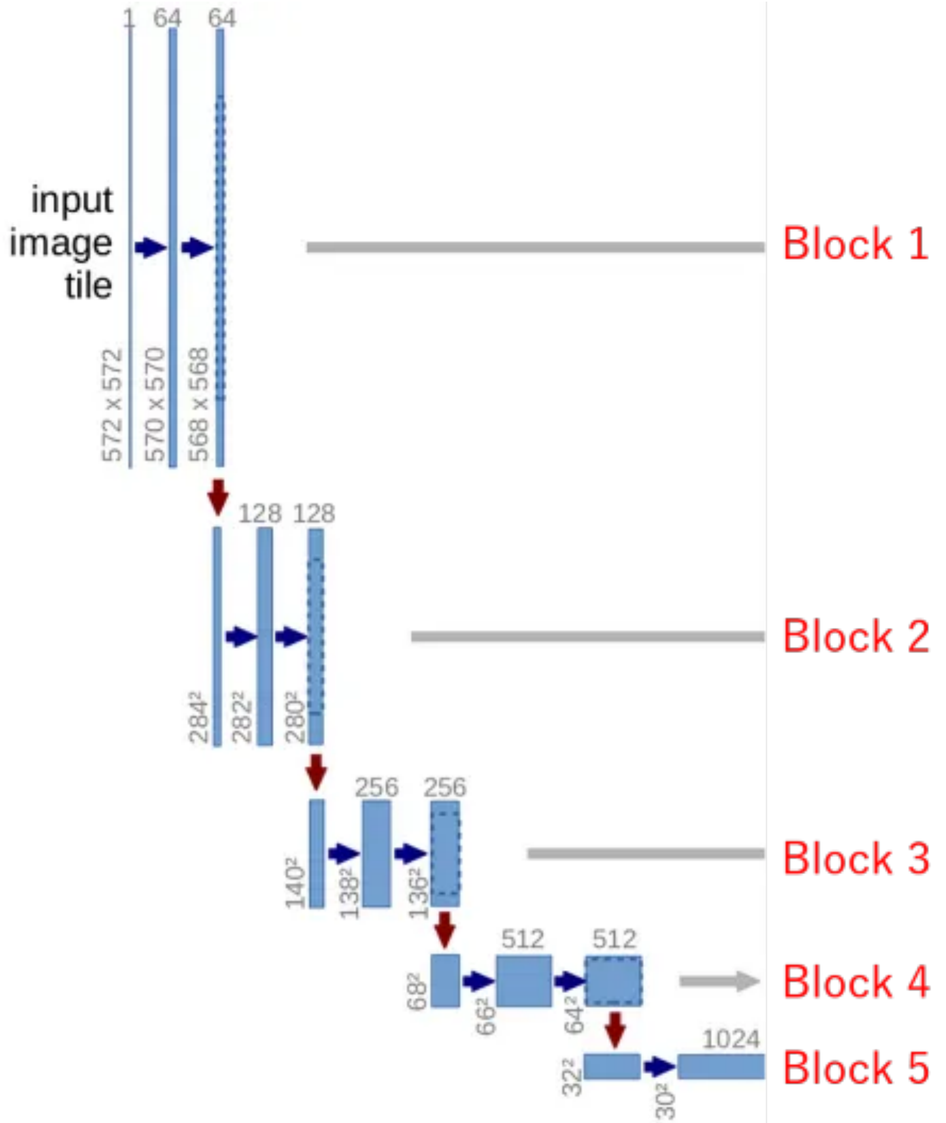


Alejandro Ito Aramendia https://medium.com/@alejandro.itoaramendia/decoding-the-u-net-a-complete-guide-810b1c6d56d8

U-Net: Contracting Path

“**Block 1:**

- Input image: 572^2 is fed into the U-Net. This input consists of 1 grayscale image channel.
- Two 3x3 convolution+ReLU unpadded layers. # of channels is set to 64 to capture higher level features.
- A 2x2 max pooling layer w/ a stride of 2 is then applied. This downsamples the feature map to half its size, 284^2.



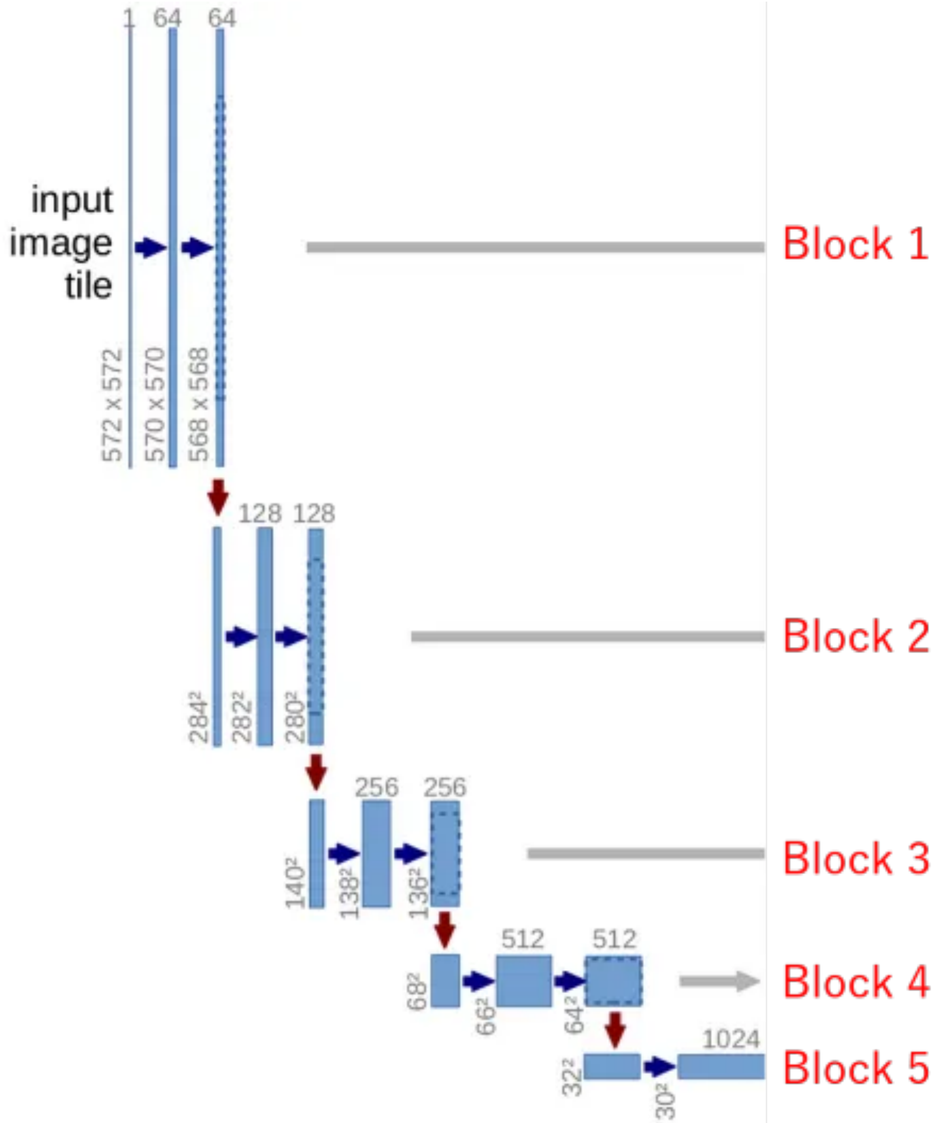
Alejandro Ito Aramendia https://medium.com/@alejandro.itoaramendia/decoding-the-u-net-a-complete-guide-810b1c6d56d8

U-Net: Contracting Path

"**Block 1:**

- Input image: 572^2 is fed into the U-Net. This input consists of 1 grayscale image channel.
- Two 3x3 convolution+ReLU unpadded layers. # of channels is set to 64 to capture higher level features.
- A 2x2 max pooling layer w/ a stride of 2 is then applied. This downsamples the feature map to half its size, 284^2.

Block 2, 3, 4: similar



Alejandro Ito Aramendia https://medium.com/@alejandro.itoaramendia/decoding-the-u-net-a-complete-guide-810b1c6d56d8

U-Net: Contracting Path

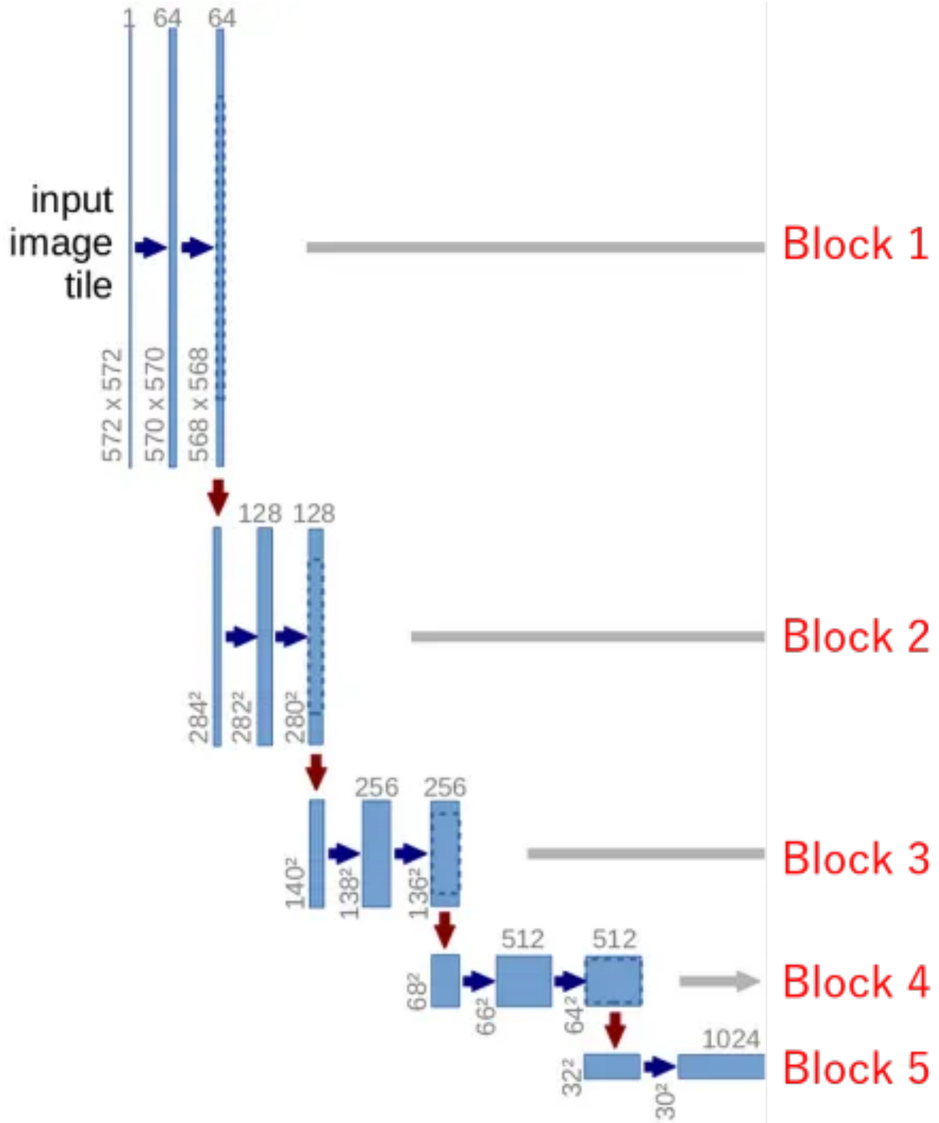
“**Block 1:**

- Input image: 572^2 is fed into the U-Net. This input consists of 1 grayscale image channel.
- Two 3x3 convolution+ReLU unpadded layers. # of channels is set to 64 to capture higher level features.
- A 2x2 max pooling layer w/ a stride of 2 is then applied. This downsamples the feature map to half its size, 284^2.

Block 2, 3, 4: similar

Block 5:

- In the final block, the # of feature channels reach 1024 after being doubled at each block.
- Also contains 3x3 convolution + ReLU unpadded layer.”

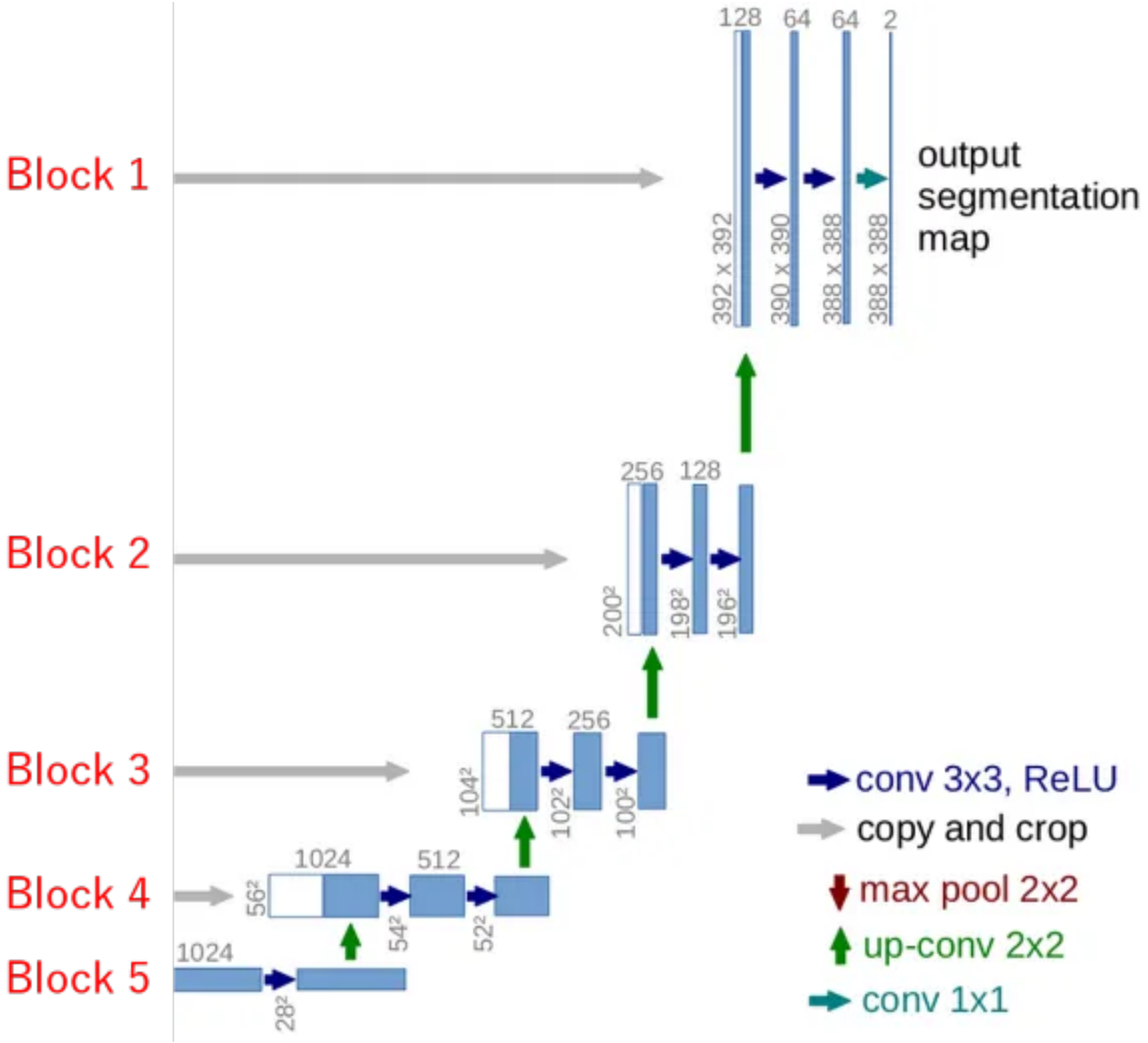


Alejandro Ito Aramendia https://medium.com/@alejandro.itoaramendia/decoding-the-u-net-a-complete-guide-810b1c6d56d8

U-Net: Expanding Path

"**Block 5:**

- A second 3x3 (unpadded) convolution + ReLU layer.
- A 2x2 up-convolution increases image dimensions twofold and also halves the # of channels to 512.



Alejandro Ito Aramendia https://medium.com/@alejandro.itoaramendia/decoding-the-u-net-a-complete-guide-810b1c6d56d8

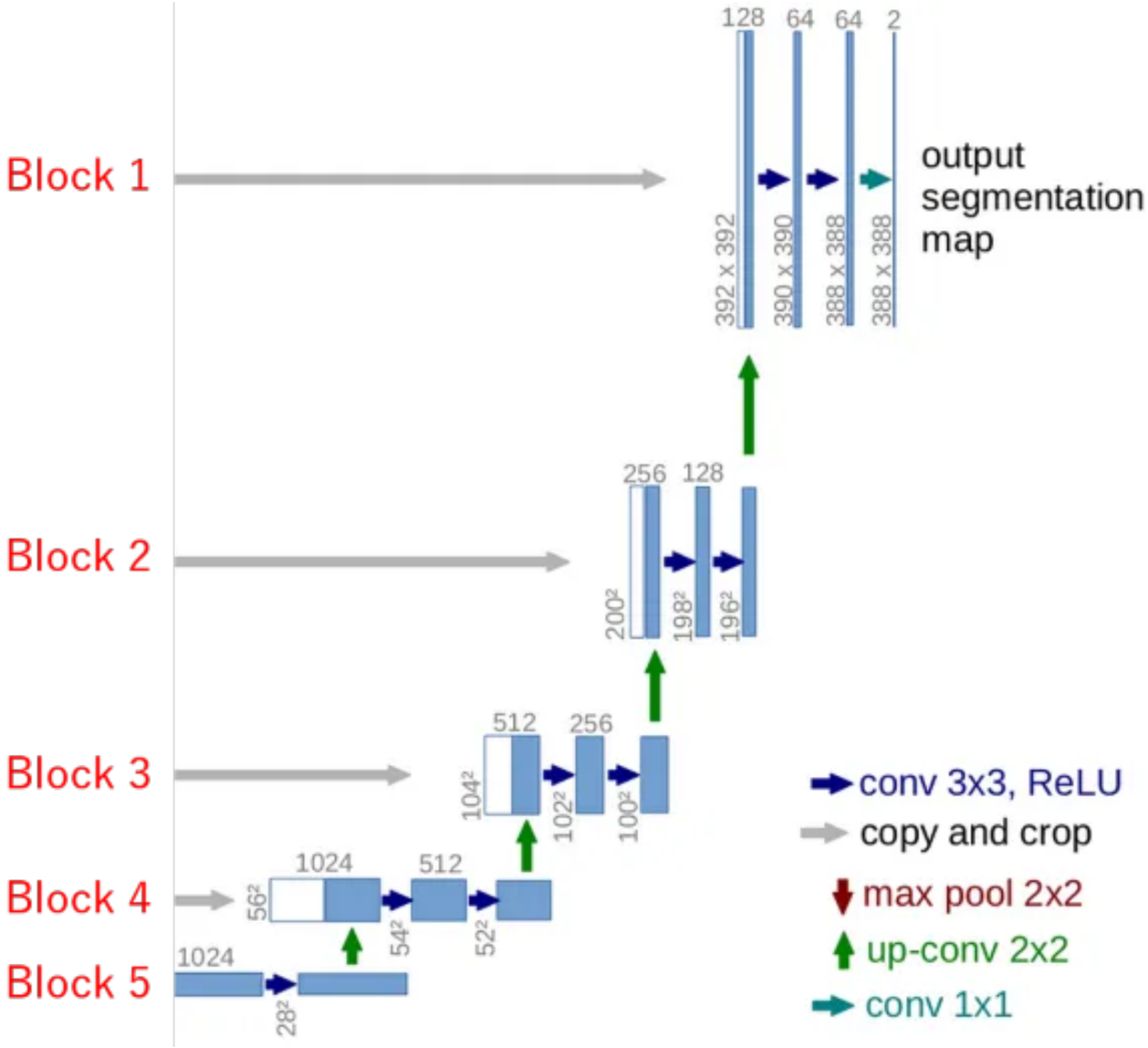
U-Net: Expanding Path

“**Block 5:**

- A second 3x3 (unpadded) convolution + ReLU layer.
- A 2x2 up-convolution increases image dimensions twofold and also halves the # of channels to 512.

Block 4:

- Skip connection: add feature map from contracting path, double feature channels to 1024. Contracting path image edges are cropped to match expanding path’s dimensions.
- Two 3x3 convolution unpadded layers are applied, each with a ReLU layer following, reducing channels to 512.
- A 2x2 up-convolution layer, upsampling the spatial dimensions x2 and also halving the # of channels to 256.



Alejandro Ito Aramendia https://medium.com/@alejandro.itoaramendia/decoding-the-u-net-a-complete-guide-810b1c6d56d8

U-Net: Expanding Path

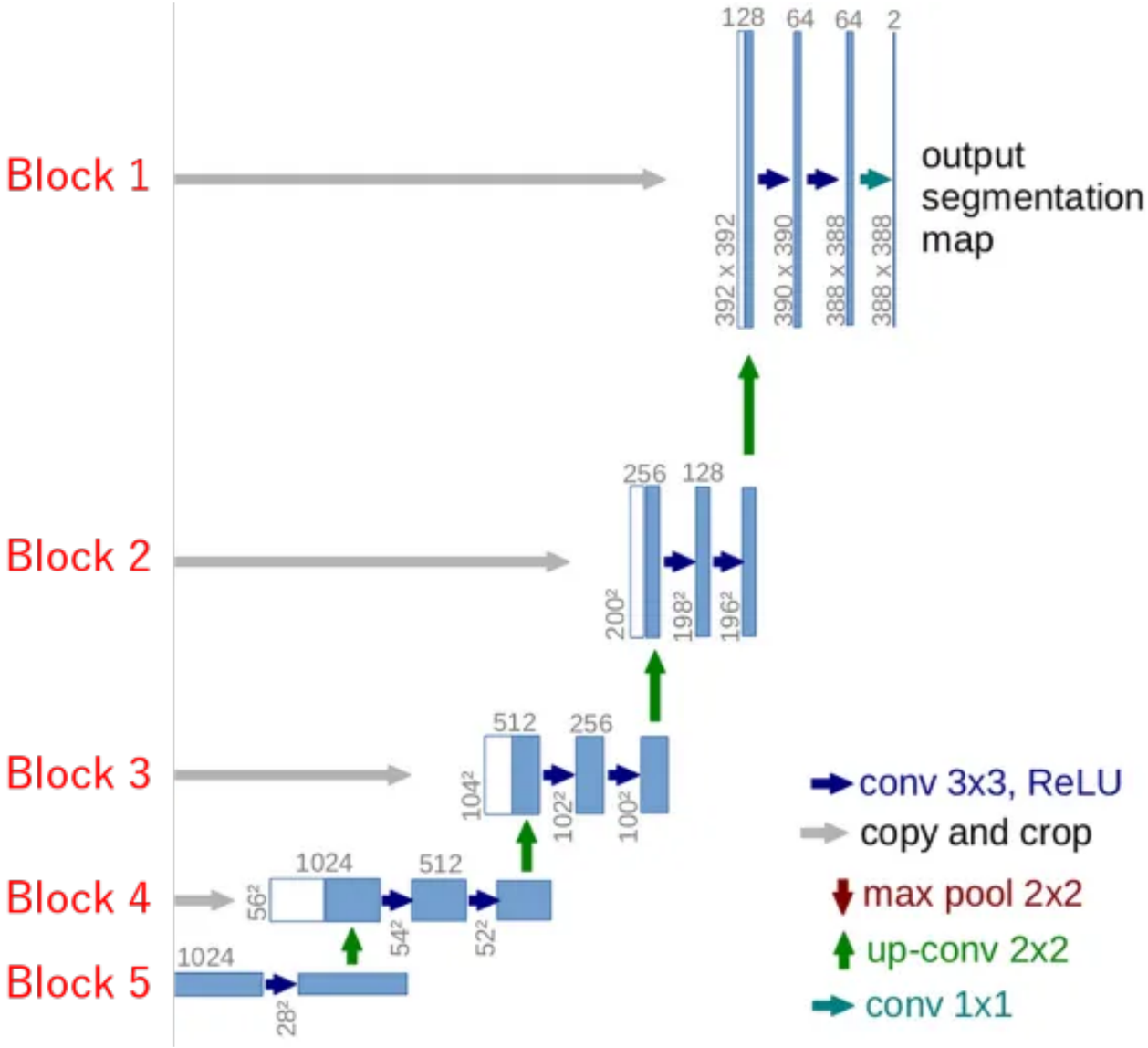
"**Block 5:**

- A second 3x3 (unpadded) convolution + ReLU layer.
- A 2x2 up-convolution increases image dimensions twofold and also halves the # of channels to 512.

Block 4:

- Skip connection: add feature map from contracting path, double feature channels to 1024. Contracting path image edges are cropped to match expanding path's dimensions.
- Two 3x3 convolution unpadded layers are applied, each with a ReLU layer following, reducing channels to 512.
- A 2x2 up-convolution layer, upsampling the spatial dimensions x2 and also halving the # of channels to 256.

Block 3, 2: similar



Alejandro Ito Aramendia https://medium.com/@alejandro.itoaramendia/decoding-the-u-net-a-complete-guide-810b1c6d56d8

U-Net: Expanding Path

"**Block 5:**

- A second 3x3 (unpadded) convolution + ReLU layer.
- A 2x2 up-convolution increases image dimensions twofold and also halves the # of channels to 512.

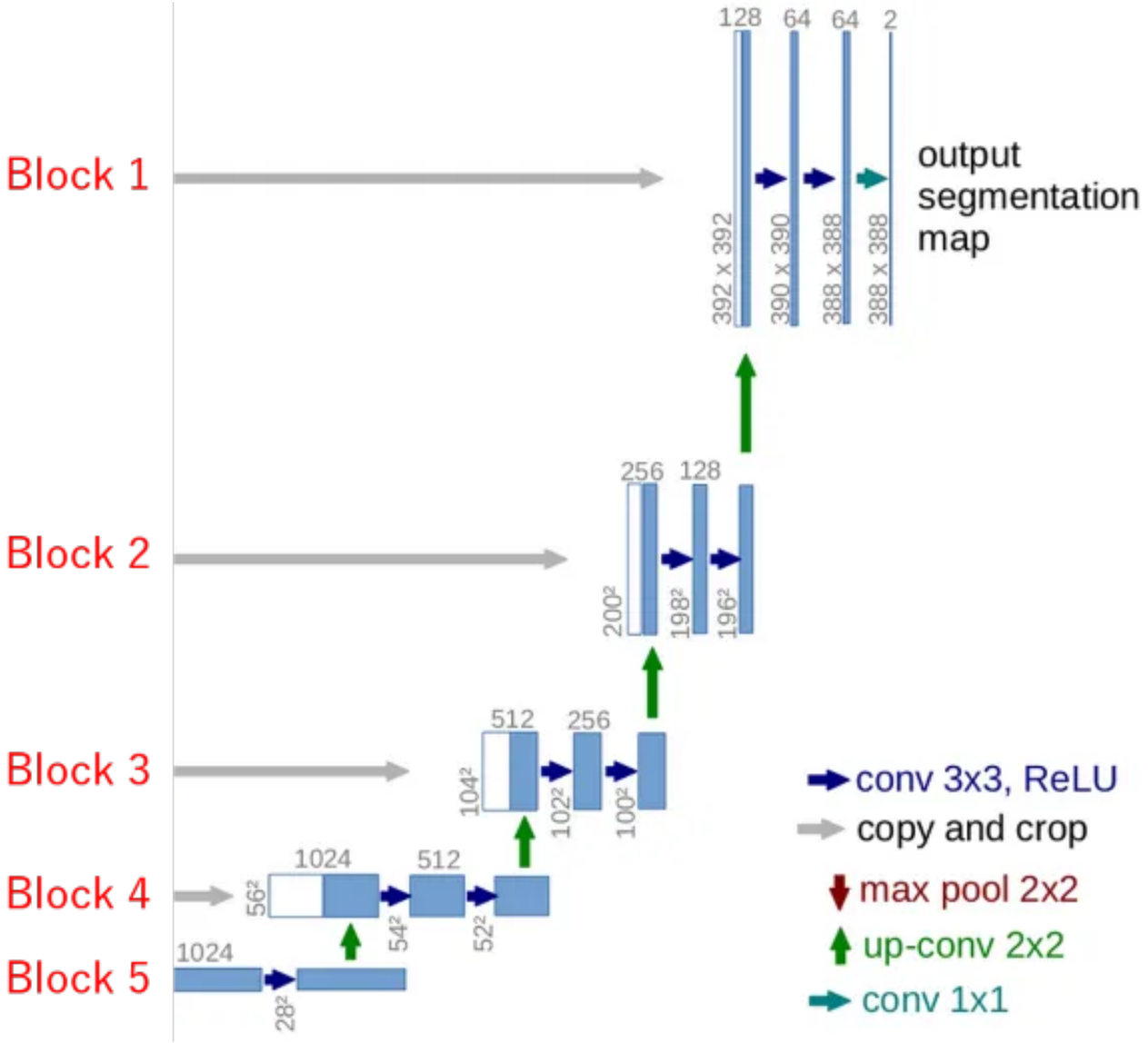
Block 4:

- Skip connection: add feature map from contracting path, double feature channels to 1024. Contracting path image edges are cropped to match expanding path's dimensions.
- Two 3x3 convolution unpadded layers are applied, each with a ReLU layer following, reducing channels to 512.
- A 2x2 up-convolution layer, upsampling the spatial dimensions x2 and also halving the # of channels to 256.

Block 3, 2: similar

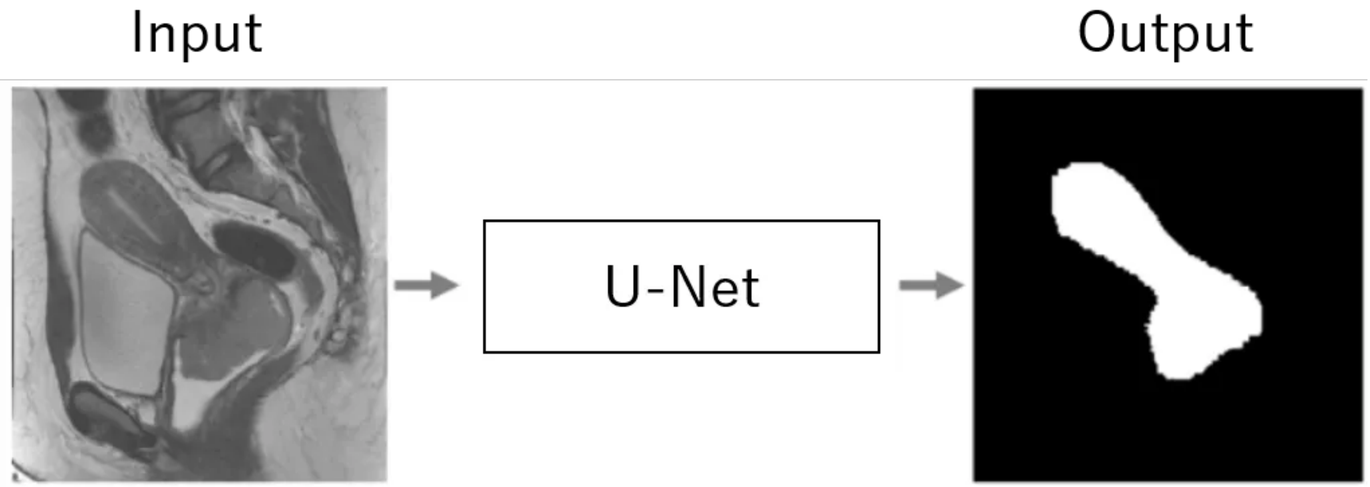
Block 1:

- Use 128 channels after concatenating skip connection.
- Two 3x3 convolution unpadded layers + ReLU, reducing # of feature channels to 64.
- Finally, a 1x1 convolution + activation layer (e.g., sigmoid for classification for medical imaging)"

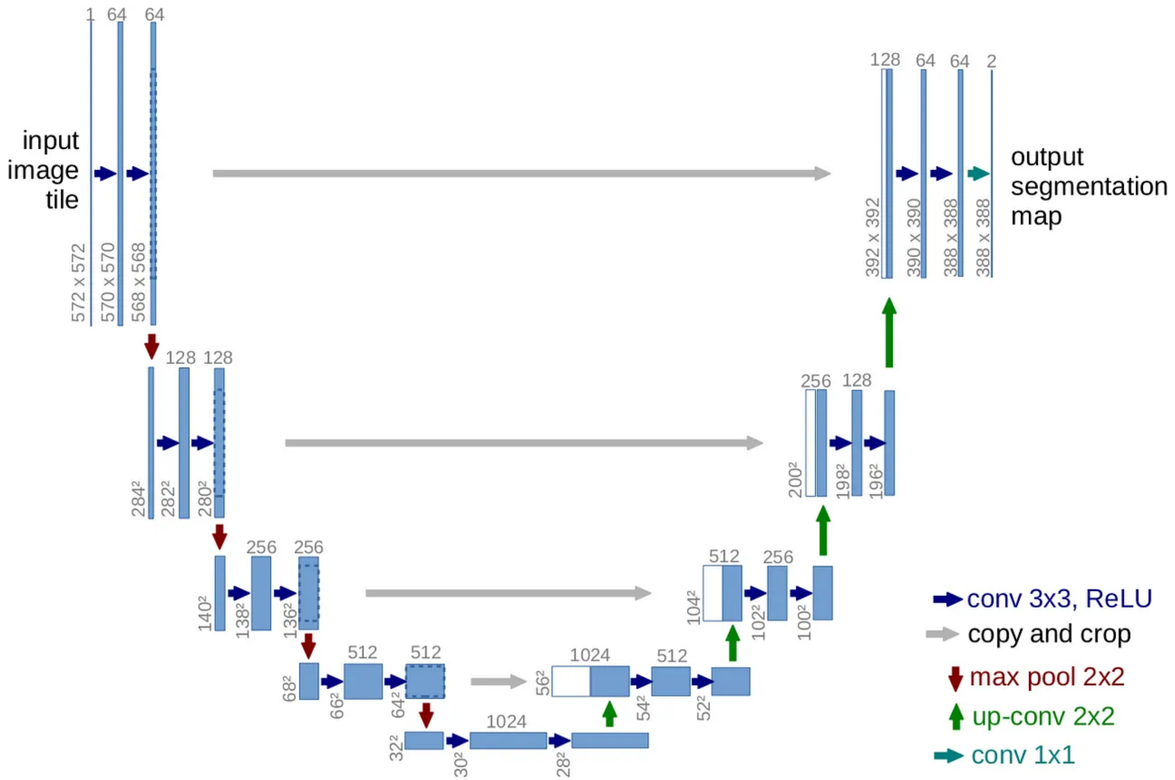


Alejandro Ito Aramendia https://medium.com/@alejandro.itoaramendia/decoding-the-u-net-a-complete-guide-810b1c6d56d8

U-Net: example use



"A possible input and output of a U-Net. In this case, a grayscale input and a single channel binary output."



Alejandro Ito Aramendia https://medium.com/@alejandro.itoaramendia/decoding-the-u-net-a-complete-guide-810b1c6d56d8

The End