

[Open in app](#)

Search Medium



What is Transposed Convolutional Layer?

Explained through animated gifs and [python code](#)



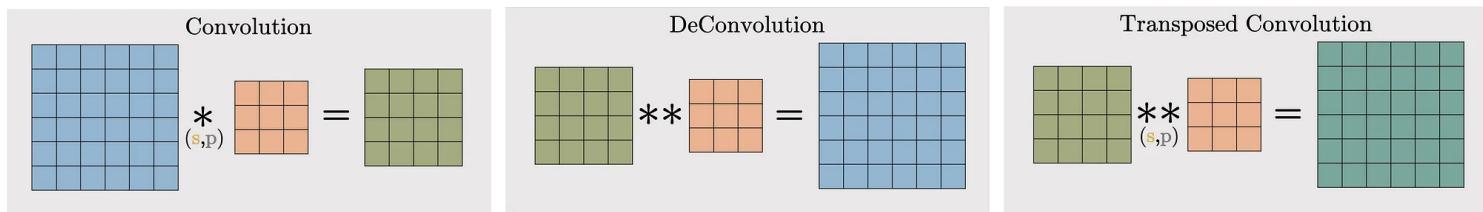
Aqeel Anwar · Follow

Published in Towards Data Science

5 min read · Mar 6, 2020

[Listen](#)[Share](#)[More](#)

The transposed Convolutional Layer is also (wrongfully) known as the Deconvolutional layer. A deconvolutional layer reverses the operation of a standard convolutional layer i.e. if the output generated through a standard convolutional layer is deconvolved, you get back the original input. The transposed convolutional layer is similar to the deconvolutional layer in the sense that the spatial dimension generated by both are the same. Transposed convolution doesn't reverse the standard convolution by values, rather by dimensions only.



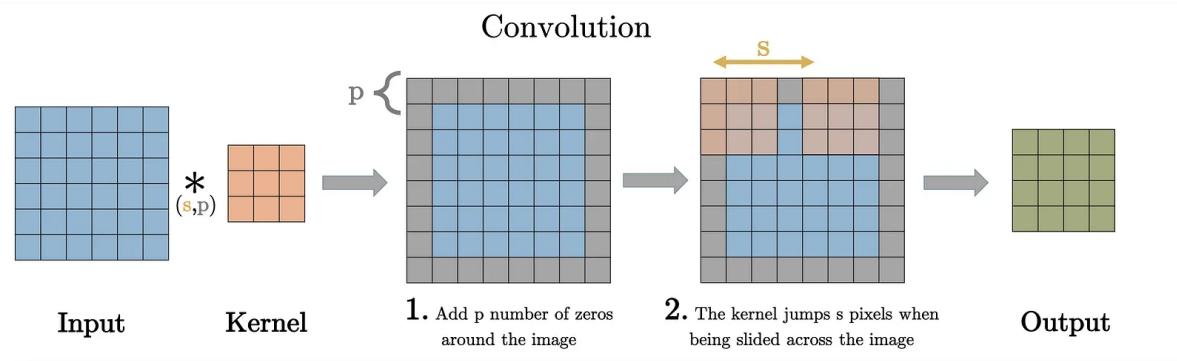
The transposed convolutional layer does exactly what a standard convolutional layer does but on a modified input feature map. Before explaining the similarity, let's first have a look at how does a standard convolutional layer works.

Standard Convolutional Layer:

A standard convolutional layer on an input of size $i \times i$ is defined by the following two parameters.

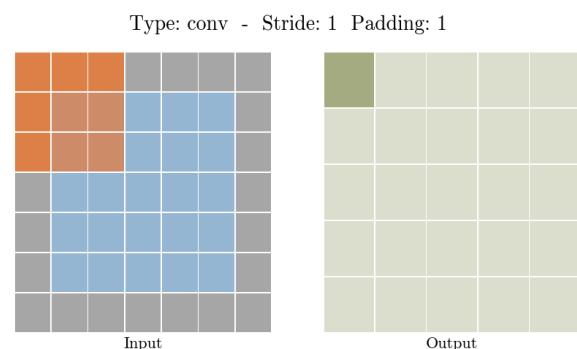
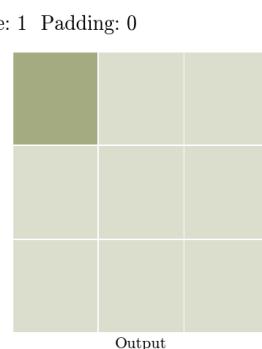
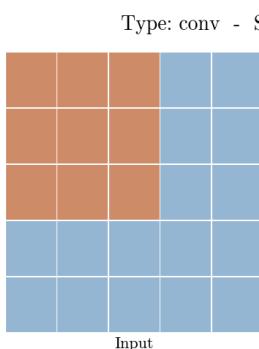
- **Padding (p):** The number of zeros padded around the original input increasing the size to $(i+2*p)x(i+2*p)$
- **Stride (s):** The amount by which the kernel is shifted when sliding across the input image.

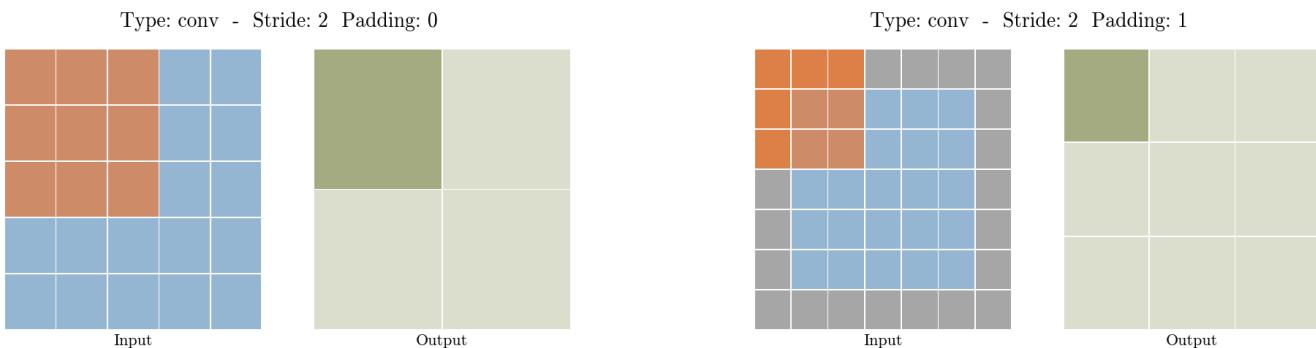
The figure below shows how a convolutional layer works as a two-step process.



In the first step, the input image is padded with zeros, while in the second step the kernel is placed on the padded input and slid across generating the output pixels as dot products of the kernel and the overlapped input region. The kernel is slid across the padded input by taking jumps of size defined by the stride. The convolutional layer usually does a down-sampling i.e. the spatial dimensions of the output are less than that of the input.

The animations below explain the working of convolutional layers for different values of stride and padding.





For a given size of the input (i), kernel (k), padding (p), and stride (s), the size of the output feature map (o) generated is given by

$$o = \frac{i + 2p - k}{s} + 1$$

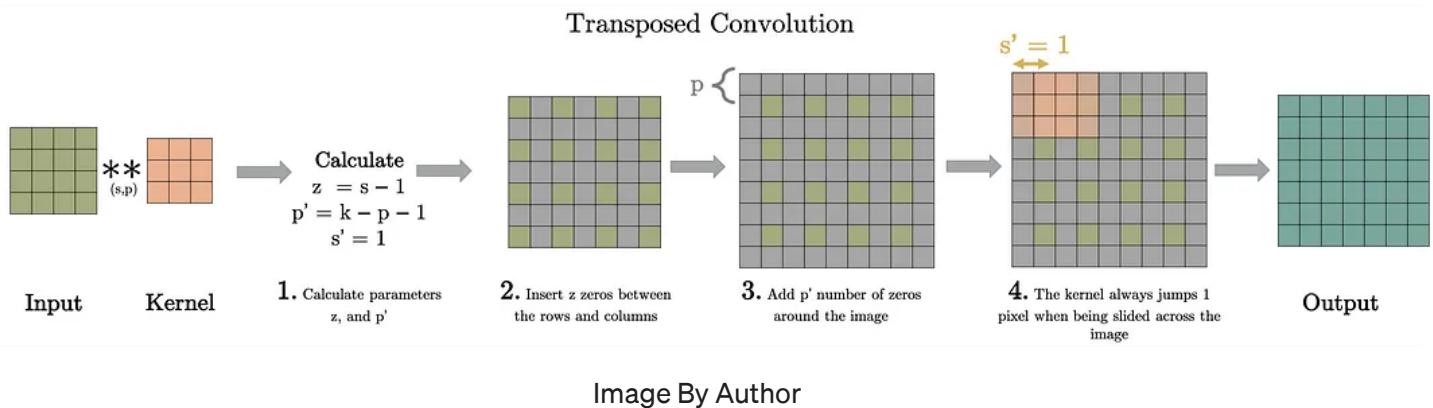
Transposed Convolutional Layer:

A transposed convolutional layer, on the other hand, is usually carried out for upsampling i.e. to generate an output feature map that has a spatial dimension greater than that of the input feature map. Just like the standard convolutional layer, the transposed convolutional layer is also defined by the padding and stride. These values of padding and stride are the one that hypothetically was carried out on the output to generate the input. i.e. if you take the output, and carry out a standard convolution with stride and padding defined, it will generate the spatial dimension same as that of the input.

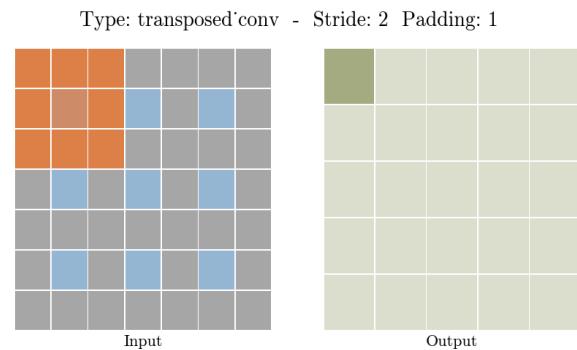
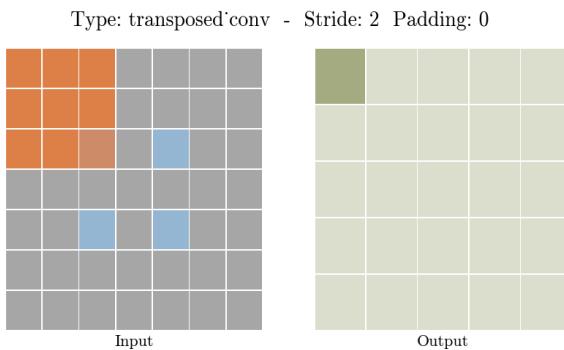
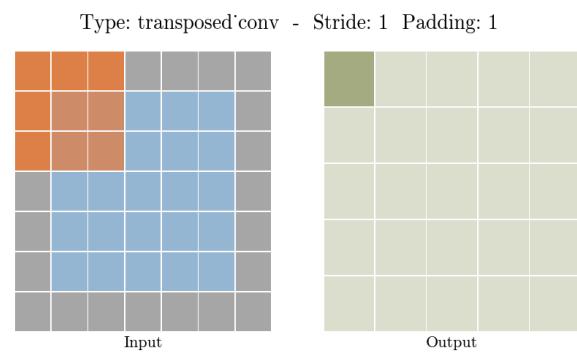
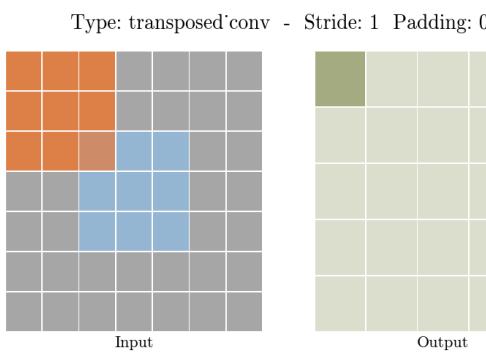
Implementing a transposed convolutional layer can be better explained as a 4 step process

- **Step 1:** Calculate new parameters z and p'
- **Step 2:** Between each row and columns of the input, insert z number of zeros. This increases the size of the input to $(2*i-1)x(2*i-1)$
- **Step 3:** Pad the modified input image with p' number of zeros
- **Step 4:** Carry out standard convolution on the image generated from step 3 with a stride length of 1

The complete steps can be seen in the figure below.



The animations below explain the working of convolutional layers for different values of stride and padding.



For a given size of the input (i), kernel (k), padding (p), and stride (s), the size of the output feature map (o) generated is given by

$$o = (i - 1) \times s + k - 2p$$

Summary:

The table below summarizes the two convolutions, standard and transposed.

Comparison					
Conv Type	Operation	Zero Insertions	Padding	Stride	Output Size
Standard	Downsampling	0	p	s	$(i+2p-k)/s + 1$
Transposed	Upsampling	$(s - 1)$	$(k-p-1)$	1	$(i-1)*s+k-2p$

- The idea behind transposed convolution is to carry out trainable upsampling
- Transposed convolutions are standard convolutions but with a modified input feature map.
- The stride and padding **do not** correspond to the number of zeros added around the image and the amount of shift in the kernel when sliding it across the input, as they would in a standard convolution operation.

Python Code:

The gifs were generated using python. The complete code can be found at

https://github.com/aqeelanwar/conv_layers_animation

Bonus:

Compact cheat sheets for this topic and many other important topics in Machine Learning can be found in the link below

Cheat Sheets for Machine Learning Interview Topics

A visual cheatsheet for ML interviews (www.cheatsheets.aqeelanwar.com)

[medium.com](https://medium.com/@aqeelanwar/cheat-sheets-for-machine-learning-interview-topics-101e0a2a2a2)

If this article was helpful to you, feel free to clap, share and respond to it. If you want to learn more about Machine Learning and Data Science, follow me @Aqeel Anwar or connect with me on [LinkedIn](#).

[Machine Learning](#)[Convolution Neural Net](#)[Convolutional Layer](#)[Towards Data Science](#)[Artificial Intelligence](#)[Follow](#)

Written by Aqeel Anwar

3.1K Followers · Writer for Towards Data Science

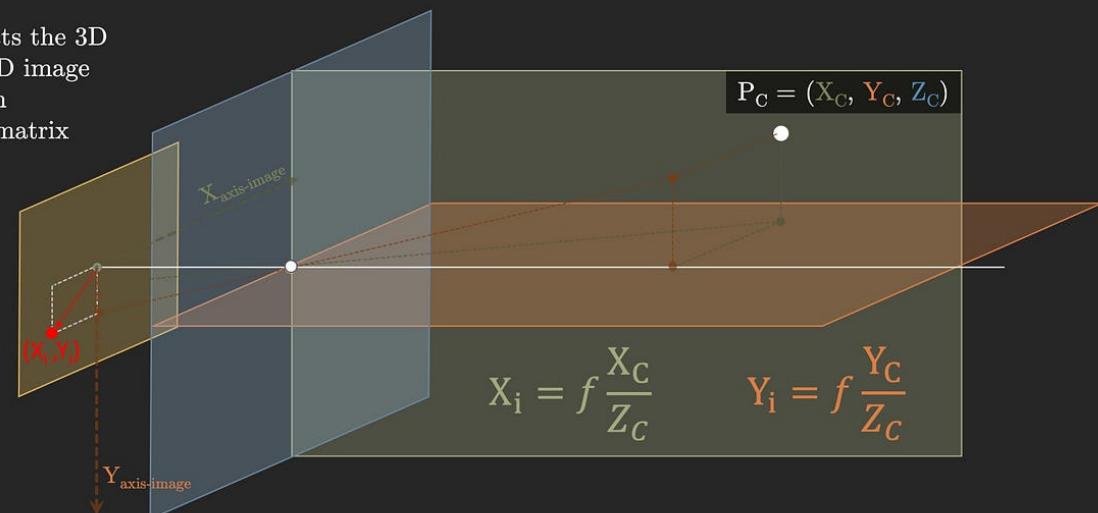
Senior ML Engineer @NVIDIA | ex-Samsung | GeorgiaTech | Writer | Researcher | Traveler | www.aqeelanwar.com | https://twitter.com/_aqeelanwar

More from Aqeel Anwar and Towards Data Science

2D Image Plane of Camera

Camera projects the 3D points to its 2D image plane. We can represent the matrix equation in homogeneous coordinates.

$$r = \sqrt{X_p^2 + Y_p^2}$$



$$\begin{pmatrix} X_i \\ Y_i \\ Z_C \end{pmatrix} = \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} X_C \\ Y_C \\ Z_C \\ 1 \end{pmatrix}$$

 Aqeel Anwar in Towards Data Science

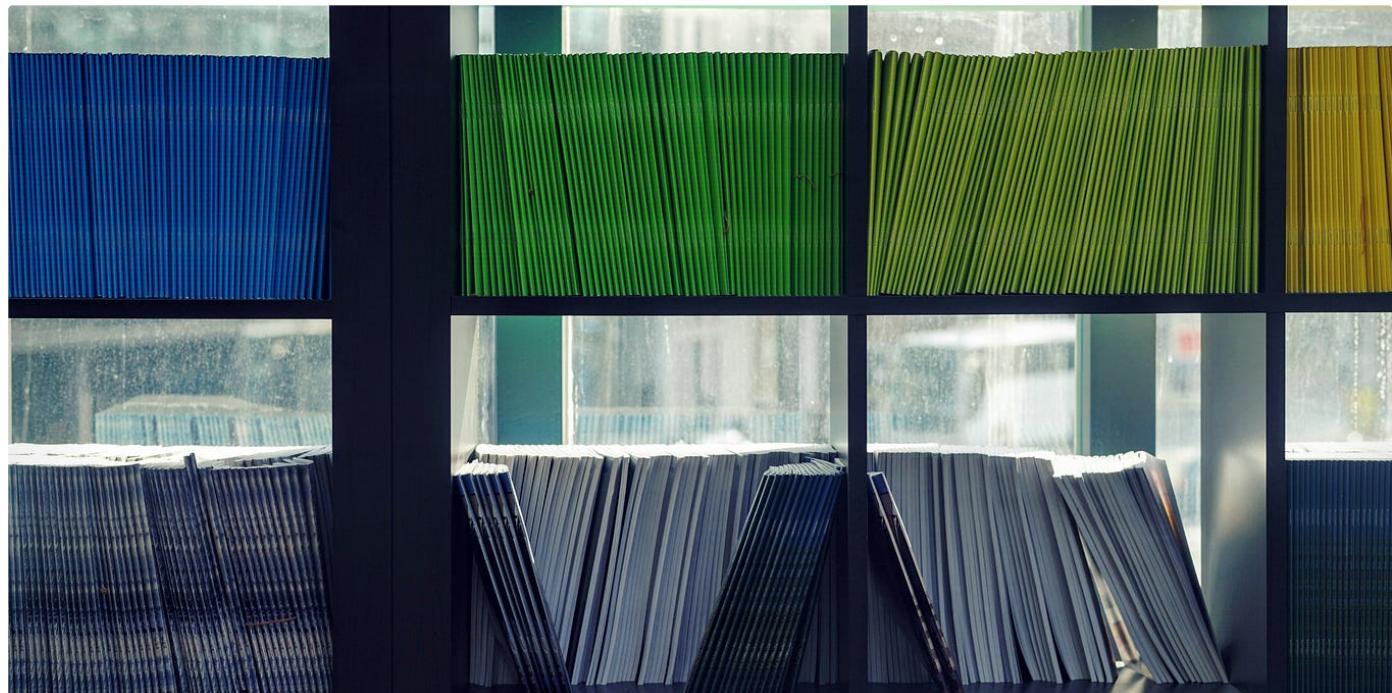
What are Intrinsic and Extrinsic Camera Parameters in Computer Vision?

A detailed explanation of intrinsic and extrinsic camera parameters with the help of visualizations.

6 min read · Mar 1, 2022

 300  3

...

 Jacob Marks, Ph.D. in Towards Data Science

How I Turned My Company's Docs into a Searchable Database with OpenAI

And how you can do the same with your docs

15 min read · Apr 25

 2.9K  39

...



 Leonie Monigatti in Towards Data Science

Getting Started with LangChain: A Beginner's Guide to Building LLM-Powered Applications

A LangChain tutorial to build anything with large language models in Python

★ · 12 min read · Apr 25

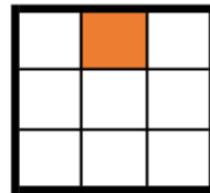
 2K

 16

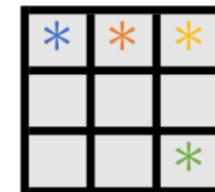


...

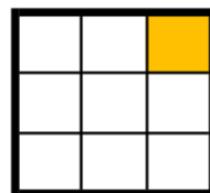
1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25



Output Feature Map of 1st conv layer



1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25



Input Feature Map of 2nd conv layer

Output Feature Map of 2nd conv layer



Aqeel Anwar in Towards Data Science

Difference between AlexNet, VGGNet, ResNet and Inception

AlexNet, VGGNet, ResNet and Inception explained

9 min read · Jun 7, 2019

694

9



...

See all from Aqeel Anwar

See all from Towards Data Science

Recommended from Medium



 Enrico Randellini

Image classification: ResNet vs EfficientNet vs EfficientNet_v2 vs Compact Convolutional...

Fine-tune and compare the latest deep neural network architectures to perform image classification tasks with PyTorch

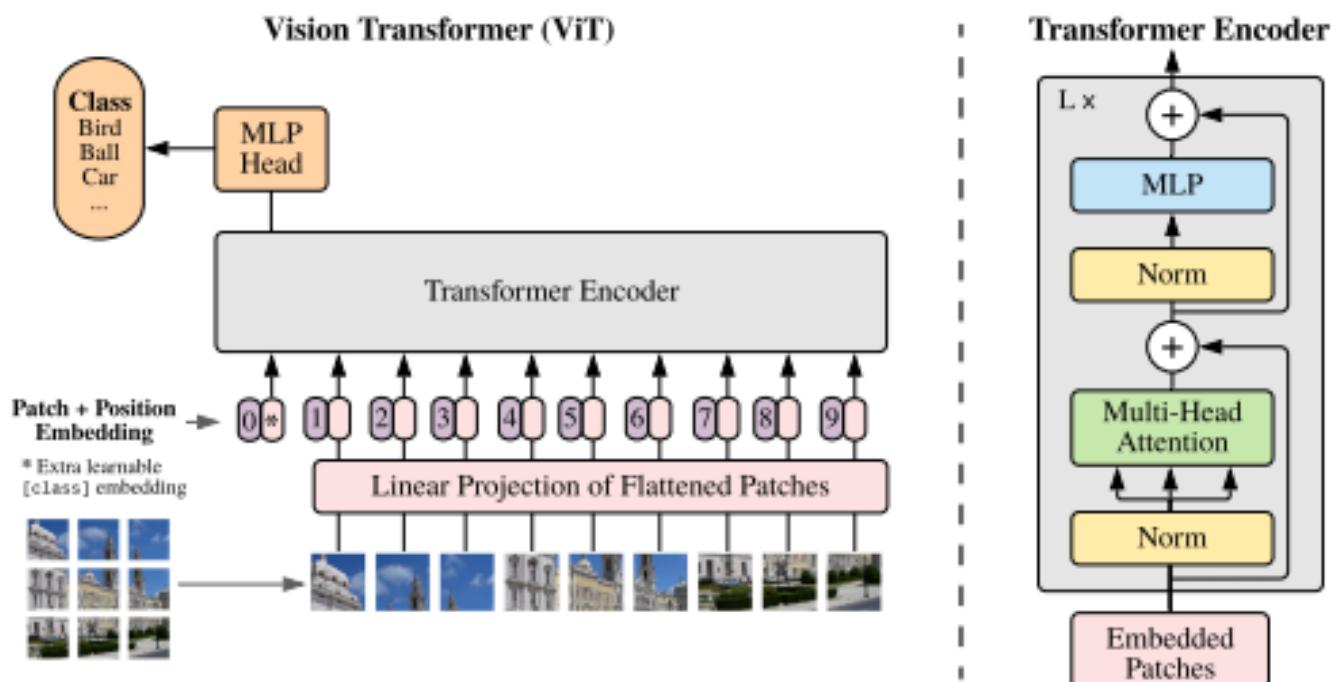
12 min read · Jan 5

 58

 3



...


 Vikram Pande

CNNs and Vision Transformers: Analysis and Comparison

Exploring the effectiveness of Vision Transformers and Convolutional Neural Networks (CNNs) in image classification tasks.

6 min read · Apr 29

 80

 3



...

Lists



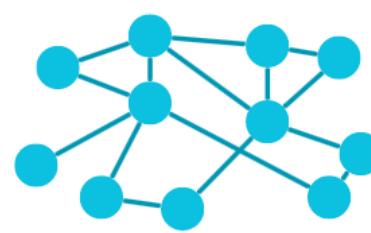
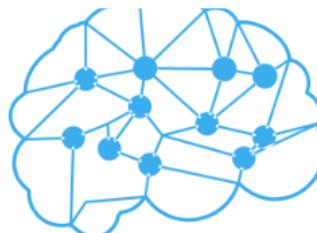
What is ChatGPT?

9 stories · 52 saves

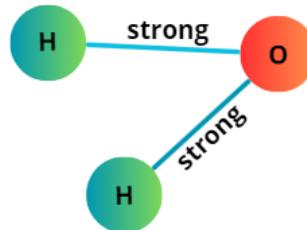
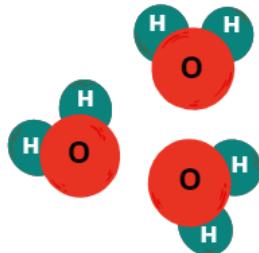


Staff Picks

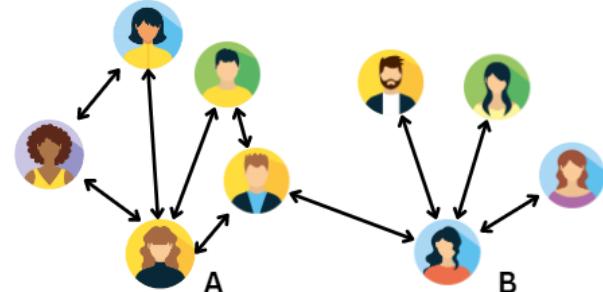
323 stories · 81 saves



Brain networks



Chemical compounds



Social networks

B Bscarleth Gtz

Introduction to Graph Neural Networks: An Illustrated Guide

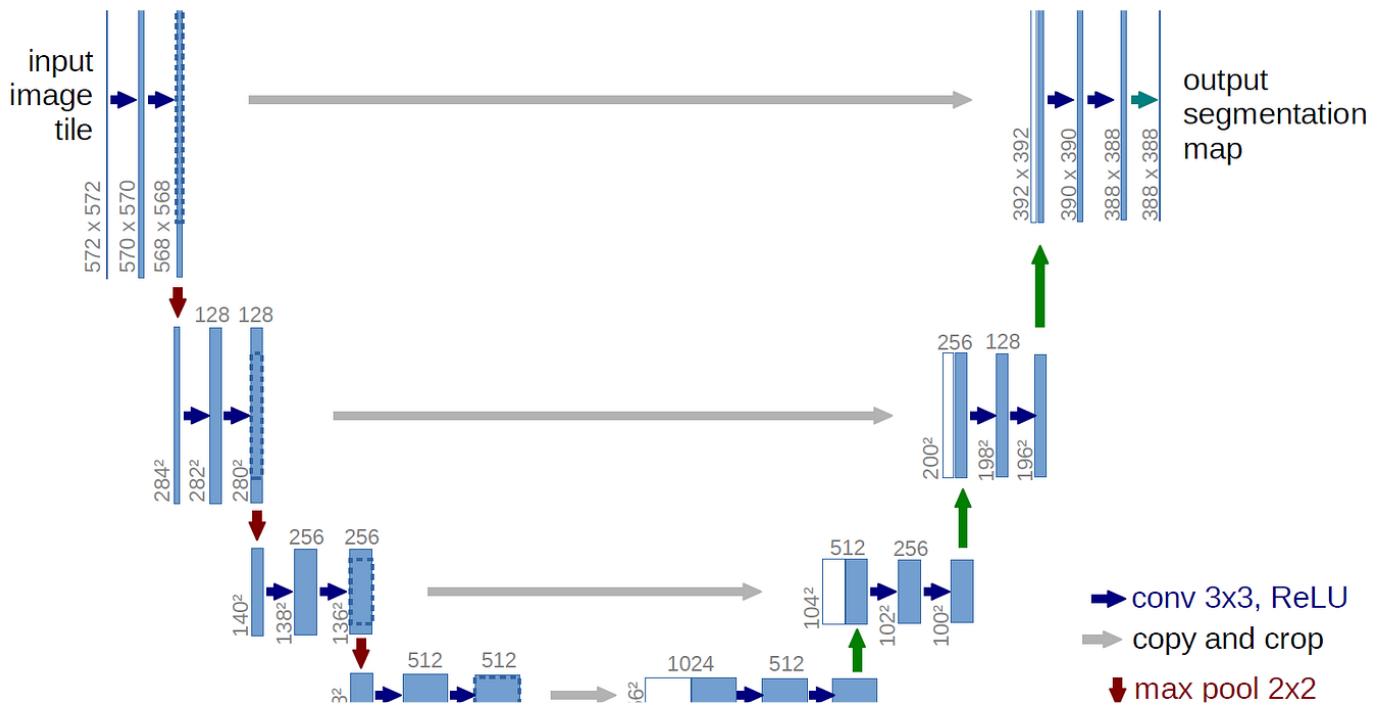
Hi Everyone! This post starts with the basics of graphs and moves forward until covering the General Framework of Graph neural networks...

14 min read · May 17

30



...



Jeremy

Notable CNN architectures and how they work

Including AlexNet, VGGNet, UNet

6 min read · Apr 24

12

1

...





Martin Thissen in MLearning.ai

Understanding and Coding the Attention Mechanism—The Magic Behind Transformers

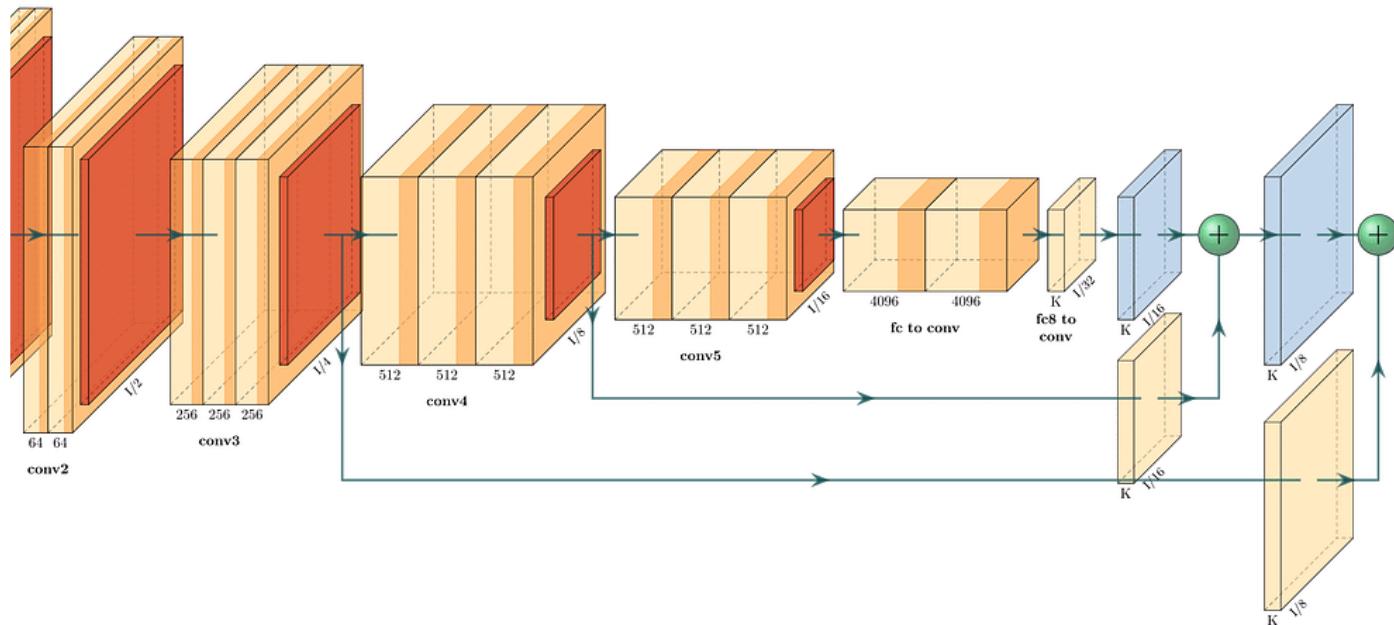
In this article, I'll give you an introduction to the attention mechanism and show you how to code the attention mechanism yourself.

★ · 12 min read · Dec 6, 2022

271 2



...



Clément Delteil in Towards AI

Creating Stunning Neural Network Visualizations with ChatGPT and PlotNeuralNet

Presenting PlotNeuralNet, a LaTeX / Python package to visualize Neural Networks

★ · 8 min read · Mar 8

98



...

See more recommendations