Assignment 01 Solutions

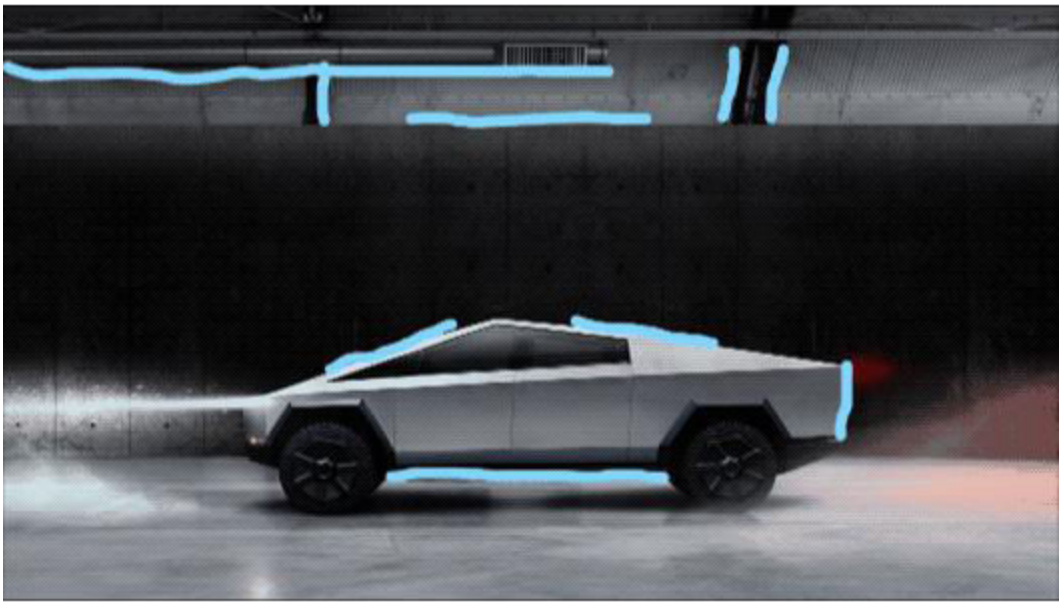
1. What exactly is a feature?

Ans: A feature is a measurable property of the object you're trying to analyze. In datasets, features appear as columns:

The image above contains a snippet of data from a public dataset with information about passengers on the ill-fated Titanic maiden voyage. Each feature, or column, represents a measurable piece of data that can be used for analysis: Name, Age, Sex, Fare, and so on. Features are also sometimes referred to as "variables" or "attributes." Depending on what you're trying to analyze, the features you include in your dataset can vary widely.

2. For a top edge detector, write out the convolutional kernel matrix.

Ans. ## 1. Vertical Edge Detection



Notice how the light intensity differs at the edges of the cyber-truck and the environment. In particular, at the top of the truck, the truck has brighter pixels and the environment has darker pixels. Whereas at the bottom, the floor has brighter pixels than the rocker panel of the truck. So where there are drastic changes in brightness, there are edges of the object (vertical edges, horizontal edges, 45 degree, 78 degree, 62 degree angled edges etc). The purpose of detecting sharp changes in image brightness is to capture important events and changes in properties of the world. For now lets focus on detecting vertical edges.

Suppose that you want to detect vertical edges on the cyber-truck image. How would you detect vertical edges? To simplify this problem, it would be best to consider a simpler problem.

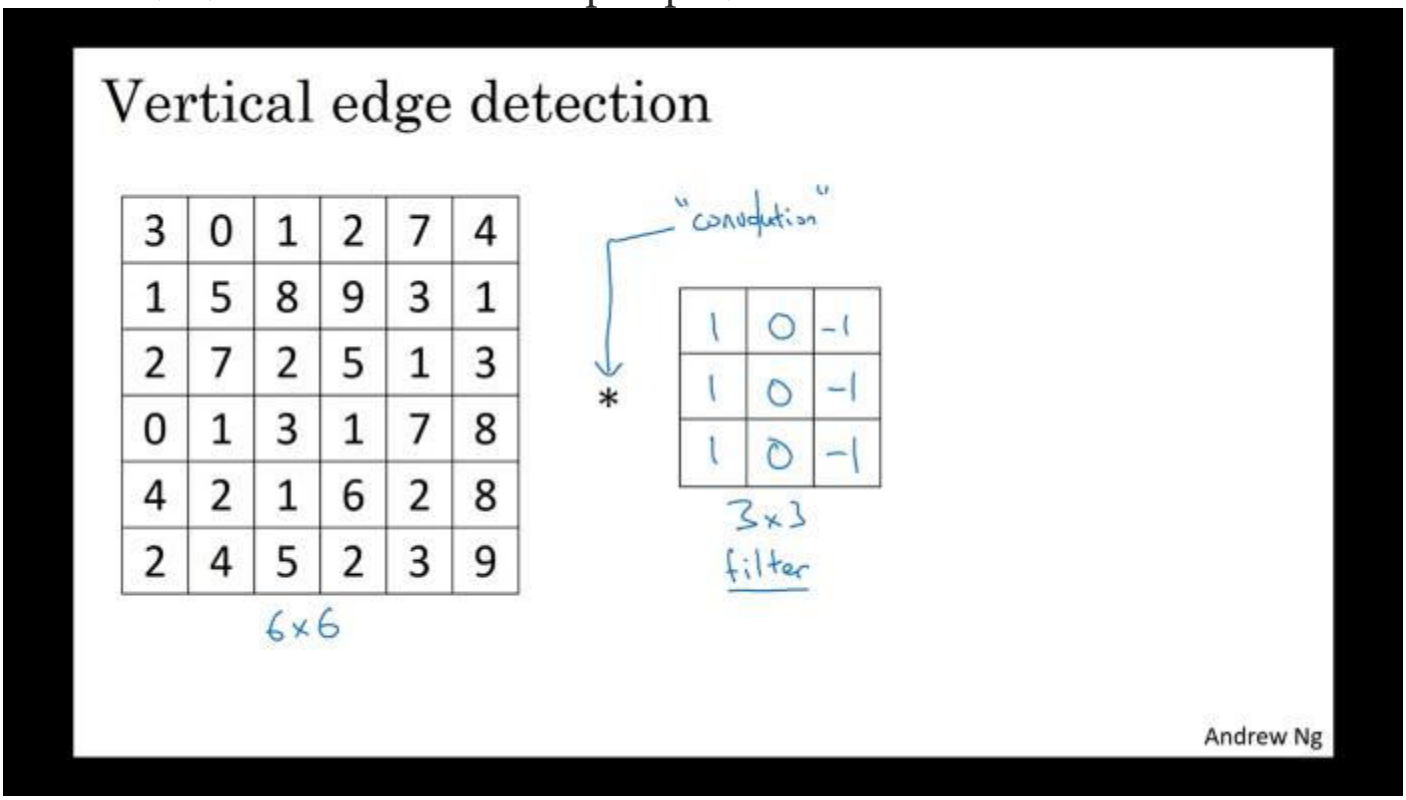


FIGURE 1: Vertical edge detection. From the deep learning specialization CNN course on Coursera by Andrew Ng and deeplearning.ai .

The grid on the *left* in figure 1 above represents a gray scale image with a 6 by 6 resolution. The numbers are the intensity values. Yes, intensity is a measurable quantity in physics and, in this context of light, intensity is just a measure of brightness. There is an actual mathematical formula to calculate light intensity, basically, the the brighter the image the higher the intensity values on the left 6 by 6 grid/matrix. Since this is a grayscale image, this is just a 6 by 6 by 1 matrix rather than 6 by 6 by 3 when it's a color image with 3 separate channels (the red, green and blue channels).

In order to detect edges or lets say vertical edges in his image, what you can do is construct a 3 by 3 matrix and in the terminology of convolutional neural networks, this is going to be called a **filter** (sometimes research papers will call this a kernel instead of a filter but I am going to use the filter terminology in this blog post).

And what you are going to do is take the 6 by 6 image and convolve it (*the convolution operation is denoted by this asterisk*) and convolve it with the 3 by 3 filter.

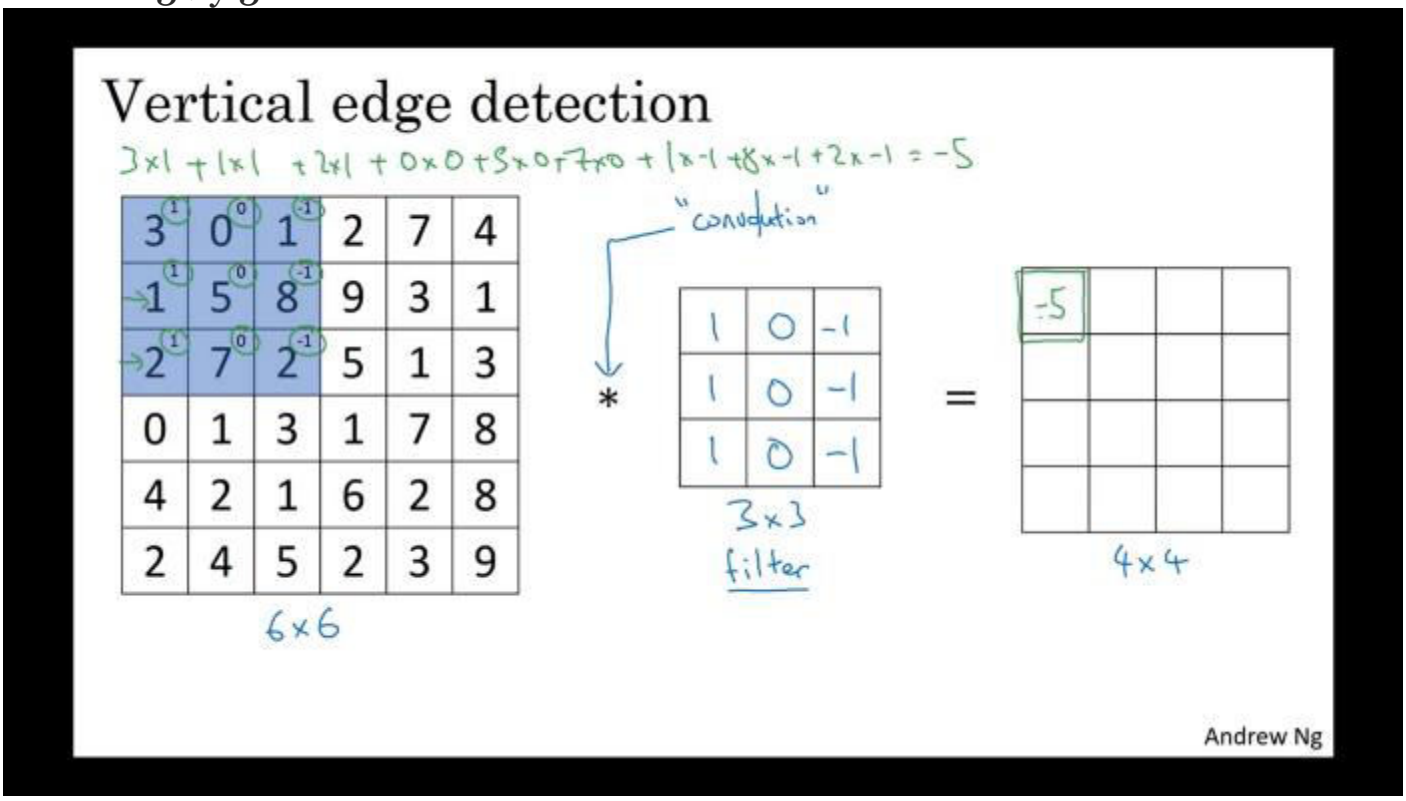


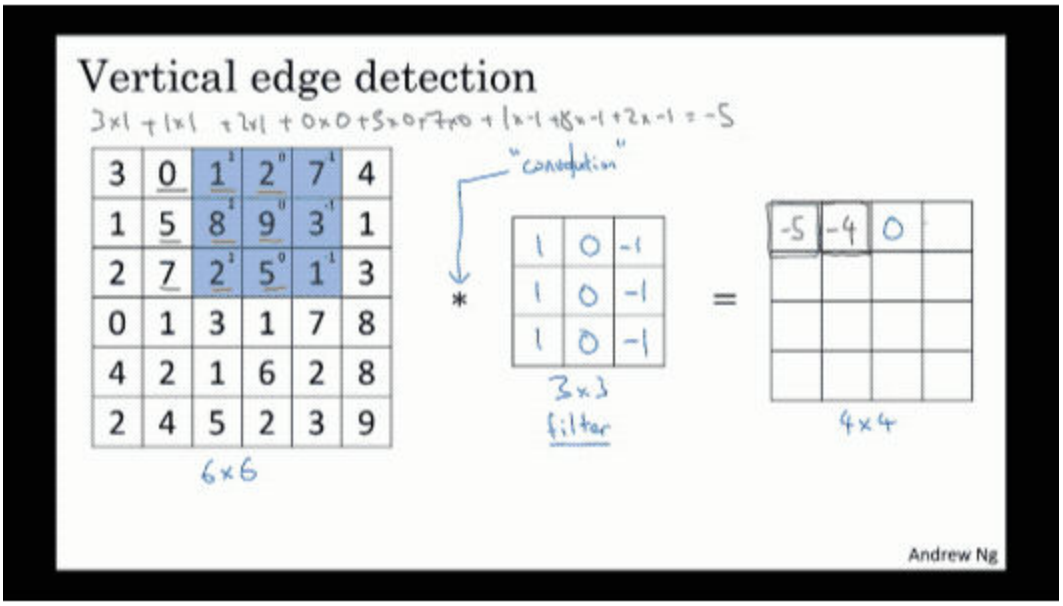
FIGURE 2: Computing the first entry of the 4 by 4 output.

The output of convoluting the 6 by 6 matrix with a 3 by 3 matrix will be a 4 by 4 matrix. The way you compute this 4 by 4 output is as follows, to compute the first elements, the upper left element of this 4 by 4

matrix, what you are going to do is take the 3 by 3 filter and paste it on top of the 3 by 3 region of your original input image. Notice the convolution matrix entries (1, 1, 1, 0, 0, 0, -1, -1, -1) are written in the top right corners of the blue region and circled in green.

And what you should do is take the element wise product of the entries in the blue 3 by 3 region and corresponding the filter matrix entries which are circles in green. Then add them all up and you should get -5. This -5 value will be the first entry of the 4 by 4 output as shown in figure 2 on the right.

Next, to figure out what is this second entry is, you are going to take the blue square and shift it one step to the right like so and you are going to do the same element wise product and then addition. You do the same for the third , fourth entries and so on as illustrated by GIF 1 below.



GIF 1: Convolution of a 6 by 6 image by a 3 by 3 filter to get the output entries.

So why is this doing vertical edge detection? Lets look at another example.

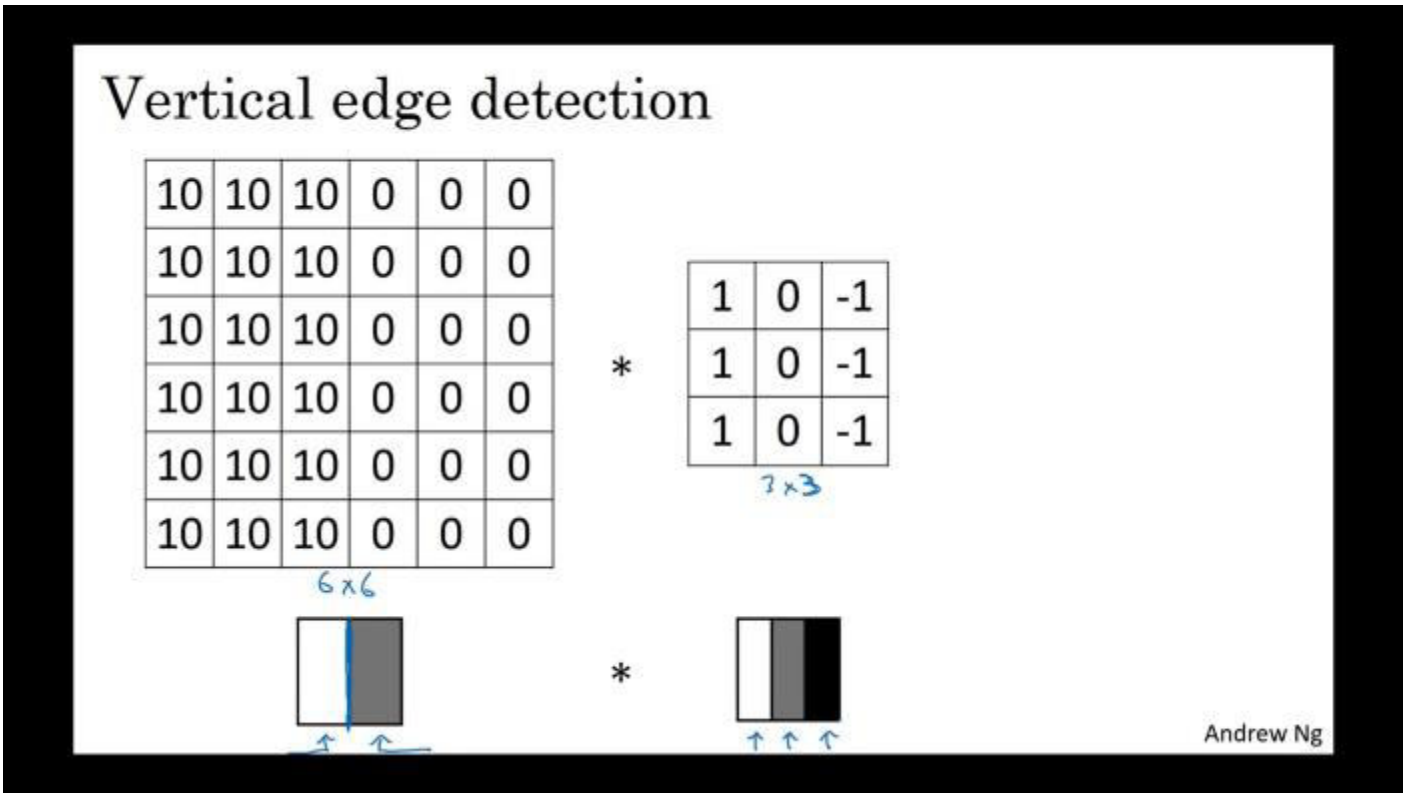


FIGURE 3: Simplified example

To illustrate vertical edge detection, we are going to use a simplified image in figure 3 on the left. The 10s, give you brighter pixel intensive values and the right half gives you darker pixel intensive values (Andrew Ng used a shade of gray to denote zeros, although maybe it could also be drawn as black). When you convolve the 6 by 6 input matrix with the 3 by 3 filter and so this 3 by 3 filter can be visualized as follows, where it is lighter, brighter pixels on the left and then this mid tone zeroes in the middle and then darker on the right (the small image with 3 shades below the filter in figure 4 below) and what you get is this matrix on the right, as shown in figure 4 below.

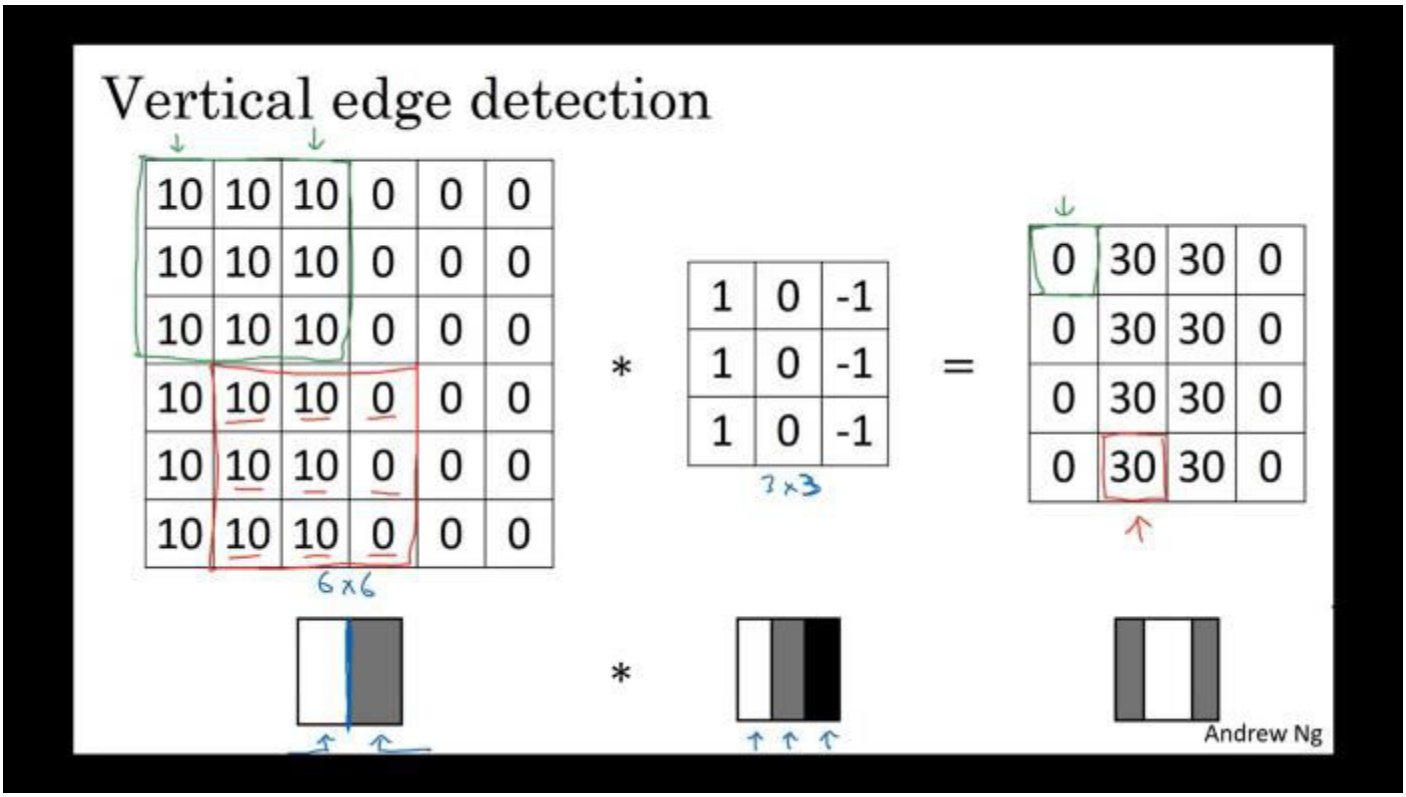
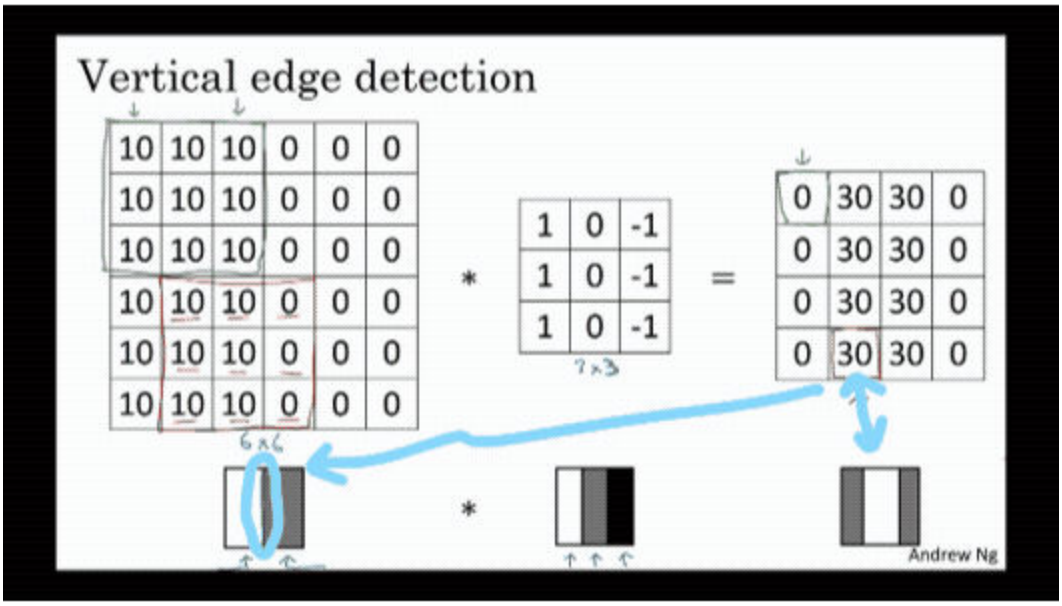


FIGURE 4: convolution of a 6 by 6 image with a 3 by 3.

Now, if you plot this right most matrix's image it will look like that where there is this lighter region right in the middle and that corresponds to having detected this vertical edge down the middle of your 6 by 6 image, as illustrated by GIF 2 below.



GIF 2: The 30s give the lighter region right in the middle and that corresponds to having detected the vertical edge (the white and gray image on the left).

One intuition to take away from vertical edge detection is that a vertical edge is *the middle in the 6 by 6 image is really where there could be bright pixels on the left and darker pixels on the right and that is why the vertical edge detector thinks its a vertical edge over there.* The convolution operation gives you a convenient way to specify how to find these vertical edges in an image.

3. Describe the mathematical operation that a 3x3 kernel performs on a single pixel in an image.

Ans. In image processing, a kernel, convolution matrix, or mask is a small matrix. It is used for blurring, sharpening, embossing, edge detection, and more. This is accomplished by doing a convolution between a kernel and an image.

4. What is the significance of a convolutional kernel added to a 3x3 matrix of zeroes?

Ans. The Convolution Matrix filter uses a first matrix which is the Image to be treated. The image is a bi-dimensional collection of pixels in rectangular coordinates. The used kernel depends on the effect you want.

GIMP uses 5x5 or 3x3 matrices. We will consider only 3x3 matrices, they are the most used and they are enough for all effects you want. If all border values of a kernel are set to zero, then system will consider it as a 3x3 matrix.

The filter studies successively every pixel of the image. For each of them, which we will call the "initial pixel", it multiplies the value of this pixel and values of the 8 surrounding pixels by the kernel corresponding value. Then it adds the results, and the initial pixel is set to this final result value.

5. What exactly is padding?

Ans: Padding is a term relevant to convolutional neural networks as it refers to the amount of pixels added to an image when it is being processed by the kernel of a CNN. For example, if the padding in a CNN is set to zero, then every pixel value that is added will be of value zero. If, however, the zero padding is set to one, there will be a one pixel border added to the image with a pixel value of zero.

Padding works by extending the area of which a convolutional neural network processes an image. The kernel is the neural networks filter which moves across the image, scanning each pixel and converting the data into a smaller, or sometimes larger, format. In order to assist the kernel with processing the image, padding is added to the frame of the image to allow for more space for the kernel to cover the image. Adding padding to an image processed by a CNN allows for more accurate analysis of images.

6. What is the concept of stride?

Ans: Stride is a component of convolutional neural networks, or neural networks tuned for the compression of images and video data. Stride is a parameter of the neural network's filter that modifies the amount of movement over the image or video. For example, if a neural network's stride is set to 1, the filter will move one pixel, or unit, at a time. The size of the filter affects the encoded output volume, so stride is often set to a whole integer, rather than a fraction or decimal.

Imagine a convolutional neural network is taking an image and analyzing the content. If the filter size is 3x3 pixels, the contained nine pixels will be converted down to 1 pixel in the output layer. Naturally, as the stride, or movement, is increased, the resulting output will be smaller. Stride is a parameter that works in conjunction with padding, the feature that adds blank, or empty pixels to the frame of the image to allow for a minimized reduction of size in the output layer. Roughly, it is a way of increasing the size of an image, to counteract the fact that stride reduces the size. Padding and stride are the foundational parameters of any convolutional neural network.

7. What are the shapes of PyTorch's 2D convolution's input and weight parameters?

Ans. Two-dimensional convolution is applied over an input given by the user where the specific shape of the input is given in the form of size, length, width, channels, and hence the output must be in a convoluted manner is called PyTorch Conv2d.

8. What exactly is a channel?

Ans: Just like any other layer, a convolutional layer receives input, transforms the input in some way, and then outputs the transformed input to the next layer. The inputs to convolutional layers are called input channels, and the outputs are called output channels.

9.Explain relationship between matrix multiplication and a convolution?

Ans. convolution is same as matrix multiplication(where matrix X and Y matrix of signal) but ONLY IN FREQUENCY DOMAIN.