CNN vs MLP

* MLPs use one perceptron for each input (eg. pixel in an image) and the amount of weights rapidly becomes unmanageable for large images. It includes to many parameters because it is fully connected. Hence, MLP can fall prey to overfitting quite easily.

* MLPs react differently to an input and its shifted version - they are not translation invariant.

Page No. _____
Date

* All the spatial information is lost when the image is flattened (matrix to vector) into an MLP.

* CNNs on the other hand benefit using their ability to develop an internal representation of a two-dimensional image. This helps the model to learn position and scale in variant structures in the data.

* The stride of the filter in the CNNs helps the filter to find and match patterns no matter where the pattern is located in a given image.

* The parning of filters in CNN essentially allows parameters sharing, weight sharing so that the filter looks for a specific pattern and is location invariant, that means can find patterns anywhere in the image.

Image Filtering

Filters are just systems that form a new and preferably enhanced image from a combination of the Original image's pixel value.
To better understand let's see image as a function.

Page No.
Date

* Images as functions

To better understand the inherent properties of images and the technical procedure used to manipulate and process them, we can think image which is comprised of individual pxs, as a function, f.
For grayscale image, each pixel would have an intensity b/w 0 to 255.

0 ⇒ black
255 ⇒ white

And $f(x,y)$ ⇒ gives the intensity of the image at pixel position (x,y).
assuming it is defined over a rectangle with a finite range.

→ In case of colored image, its just an extension of this.
$f(x,y)$ is a vector of 3 values.

In colored we have 3 colors, Red, Green, Blue (RGB) in a single px, in different proportions. Thus, it is represented by a 1x3 vector.
Since, 3 colors have values from 0 to 255, there are total of 256 x 256 x 256 = 16,777,216 combinations or color choices.

Page No.
Date

Hence,

$$f(x,y) = \begin{bmatrix} r(x,y) \\ g(x,y) \\ b(x,y) \end{bmatrix}$$

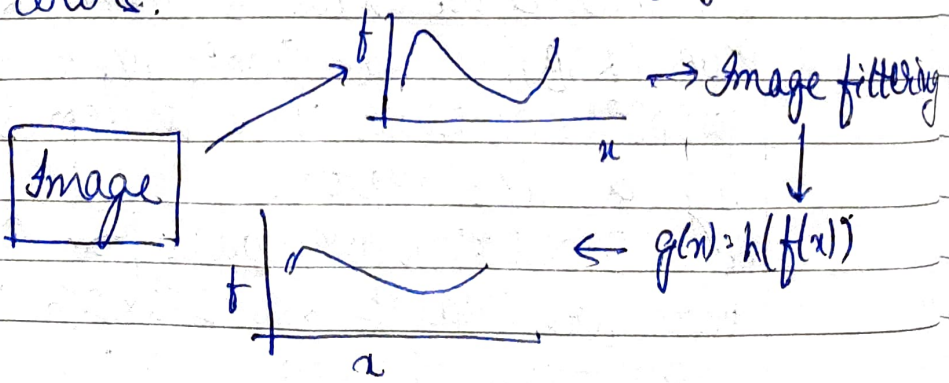
Thus with this, an image can be represented as a series of pixels or let's say matrix of pixels values.

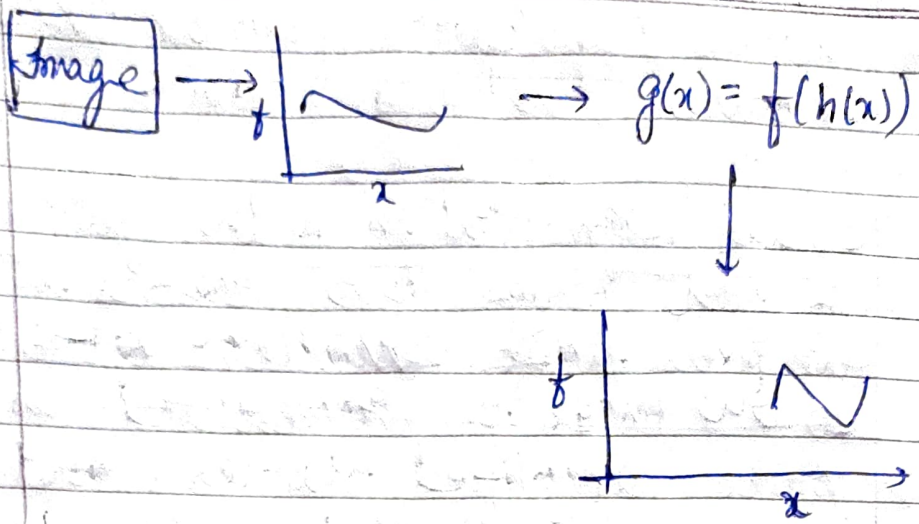
★ Image Processing

There are 2 main types of image processing:
- Image filtering
- Image ~~mapping~~ warping

→ Image filtering changes the range (i.e. the pixel values) of an image, so the colors of the image are altered without changing the px positions.

→ Image ~~mapping~~ warping changes the domain (i.e. the pixel positions) of an image, where points are mapped to other points without changing the colors.





The goal of filtering is to modify or enhance image properties and/or to extract valuable info. from the pictures such as edges, corners and blobs.

2 most common filters are:
- moving average filter
- image segmentation filter

The moving average filter replaces each pixel with the average pixel value of it and a neighbourhood window of adjacent pixels. The effect is a more smooth image with sharp features removed.

As seen with the moving average filters, blurring & sharpening filters, will produce unwanted artifact along the edges of the img. To get rid of the artifacts, zero padding, edge

value replication, mirror extension or other methods can be used.

Image segmentation is the partitioning of an image into regions where the pixels have smaller attributes, so the image is represented in a more simplified manner, and so we can then identify objects and boundaries more easily.

for eg: all pixels with an intensity greater than 100 are replaced with a white pixel (intensity 255) and all others are replaced with a black pixel. (intensity 0)

$$g[n,m] = \begin{cases} 255, & f[n,m] > 100 \\ 0, & \text{otherwise} \end{cases}$$

→ Convolution particularly 2D convolution are image filtering networks.

Page No.
Date

Convolution vs Correlation filtering

(Miscellaneous Topic)

* Convolutional operation widely used in CNN is a misnomer. The operation that is used is strictly speaking a correlation instead.
* There is a very slight difference b/w the two.

* Cross-correlation

→ Correlation is the process of moving a filter mask often referred to as kernel over the img and computing the sum of products at each location.

→ Correlation is the function of displacement of the filter. The first value of the correlation corresponds to zero displacement of the filter, the second value corresponds to one unit of displacement and so on.

→ Cross-correlation of img I using filter F

$$f \circ I(x) = \sum_{i=-N}^{N} F(i) I(x+i)$$

This is for crosscorrelation in 1D

→ In 2D the formula changes to, supposing our filter has odd number of elements, so it is represented by a $(2N+1, 2N+1)$ matrix

$$F \circ I(x,y) = \sum_{j=-N}^{N} \sum_{i=-N}^{N} F(i,j)\, I(x+i, y+j)$$

★ Convolution

→ Similar to correlation but with a slight difference.

→ Kernel is first flipped and then applied to the image. The kernel is flipped by an angle of 180 degrees.

So the formula changes to:

In 1D ⇒

$$F * I(x) = \sum_{i=-N}^{N} F(i)\, I(x-i)$$

In 2D ⇒

$$F * I(x,y) = \sum_{j=-N}^{N} \sum_{i=-N}^{N} F(i,j)\, I(x-i, y-j)$$