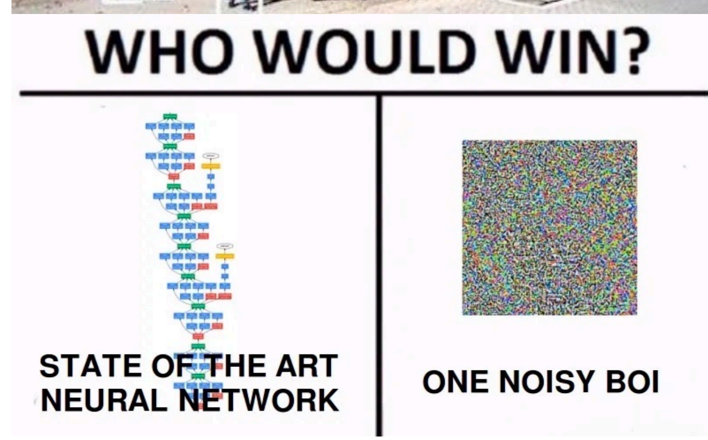
March 20, 2023

DSML: Computer Vision.

CNNs: Dealing with Overfitting.

Class starts @ 9:05 pm.



Recap:

* Common problems in computer vision?

  Classification, matching, detection, segmentation, generation.

* Challenges?

  Occlusion; deformations, zoom/ distance, illumination, loss of spatial info.

* What is the current state-of-the-art for natural image classification?

  CNN - Convolutional Neural Networks.

* What is the inspiration for the SOTA?

  Visual Cortex.

* What are different types of convolutions? Are they linear?

  valid, full, same. (All are linear operations).

Agenda for today:

* Types of convolutions:
  - → Strided convolutions.
  - → Dilated Convolutions.

* Pooling operations:
  - → Max pooling
  - → Average pooling.

* Putting it all together: Apparel classification case study.
  - → ANN approach.
  - → CNN approach.

Strided Convolution.

$m \times n \xrightarrow[\text{conv}]{\text{strided}} \dfrac{m}{2} \times \dfrac{n}{2}$

Image. $(m \times n)$ → Conv-2D. → Image

Kernel. $(a \times b)$ → Conv-2D.

Conv-2D. ↓ Elementwise multiplication + addition.

Kernel ↓ Filter. ↓ Some 2-D matrix (usually a square).

: (i) Full : $\dfrac{m+a-1}{k}, \dfrac{n+b-1}{l}$.

(ii) Valid : $\dfrac{m-a+1}{k}, \dfrac{n-b+1}{l}$

(iii) Same : $\dfrac{m}{k}, \dfrac{n}{l}$.

Objective of strided conv:

Reduce the size of the output image by a factor $k \times l$.

1	0	-1
1	0	-1
1	0	-1

3x3

10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0

6x6

-20	20	0
-30	30	0
-30	30	0

-20	0	20	20	0	0
-30	0	30	30	0	0
-30	0	30	30	0	0
-20	0	30	30	0	0
-30	0	30	30	0	0
-20	0	20			0

whi l

Dilated Convolutions.

Image $(m \times n)$ → Conv-2D. → Image : (i) Full : $m+a-1,\ n+b-1$.
Kernel $(a \times b)$ → Conv-2D.

(ii) Valid : $m-a+1,\ n-b+1$

(iii) Same : m , n

Conv-2D. ↓ Elementwise multiplication + addition.

Kernel ↓ Filter. ↓ Some 2-D matrix (usually a square).

$(a \times b)$ → $(2a \times 2b)$

Objective of Dilated conv:

(i) Efficient way to do conv with large filters.

(ii) Control of receptive field size.

Dilation ↕ (same meaning) Upsampling.

0	0	0	0	0					
0	0	0							
0	0	10	10	10	0	0	0		
0	0	10	10	10	0	0	0		
0	0	10	10	10	0	0	0		
0	0	10	10	10	0	0	0		
0	0	10	10	10	0	0	0		
0	0	10	10	10	0	0	0		
0	0	0	0						

6×6

1	0	0	0	-1	0
0	0	0	0	0	0
1	0	0	0	-1	0
0	0	0	0	0	0
1	0	0	0	-1	0
0	0	0	0	0	0

6 × 6

1	0	-1
1	0	-1
1	0	-1

3×3

Downsampling.

Max. pooling.

$$\frac{m}{a} \times \frac{n}{b}.$$

Image ($m \times n$) → Maxpool. → "Same"

Kernel ($a \times b$) → Maxpool.

→ Similar to Conv 2D without overlap.

→ Instead of elementwise mul & addition, we take the max value and store that.

3×3

10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0

10	0
10	0

Average Pooling

$m \times n$

Image $\rightarrow$ Avgpool. $\rightarrow$ "Same"

Kernel $\rightarrow$

$a \times b$

$$\frac{m}{a} \times \frac{n}{b}.$$

→ Similar to Conv 2D without overlap.

→ Instead of elementwise mul & addition, we take the avg. value and store that.

3×3

10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0

6×6

10	0
10	0

$$\frac{6}{3} \times \frac{6}{3} = 2 \times 2$$

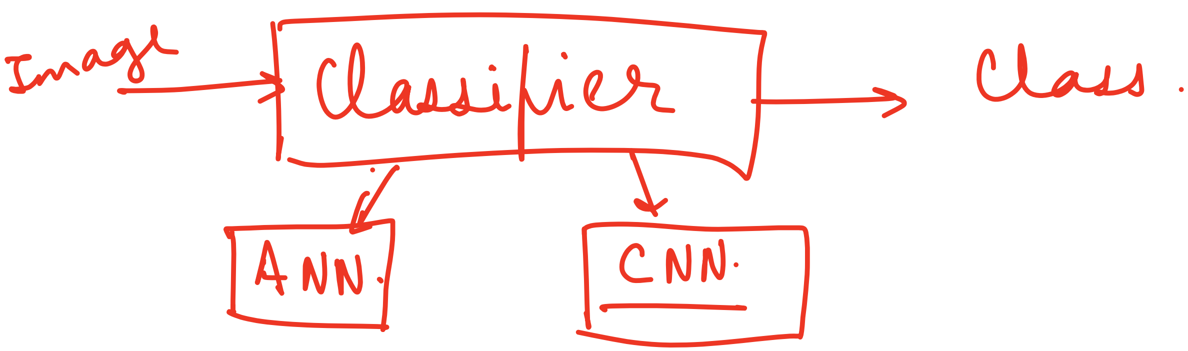
3×3

Case study : Apparel Classification.

Operation	Input sizes: Kernel	Input sizes: Image	Output sizes
(1) Conv2D (Valid)	$a \times b$	$m \times n$	$m-a+1 \times n-b+1$
(2) Conv2D (Same)	$a \times b$	$m \times n$	$m \times n$
(3) Conv2D (Full)	$a \times b$	$m \times n$	$m+a-1 \times n+b-1$
(4) Conv2D (Strided)	$a \times b$	$m \times n$, $k \times k$	$\frac{m}{k} \times \frac{n}{k}$
(5) Conv2D (Dilated)	$a \times b$	$m \times n$, $k \times k$	$m \times n$
(6) MaxPool2D	$a \times b$	$m \times n$	$\frac{m}{a} \times \frac{n}{b}$
(7) AvgPool2D	$a \times b$	$m \times n$	$\frac{m}{a} \times \frac{n}{b}$

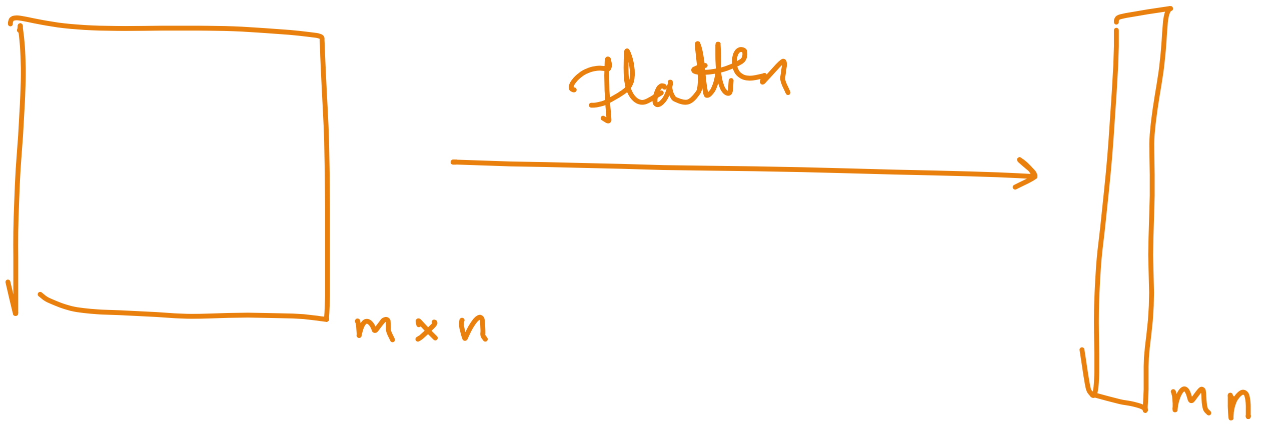
```
clothing-dataset-small
├── test
│   ├── dress
│   ├── hat
│   ├── longsleeve
│   ├── outwear
│   ├── pants
│   ├── shirt
│   ├── shoes
│   ├── shorts
│   ├── skirt
│   └── t-shirt
├── train
│   ├── dress
│   ├── hat
│   ├── longsleeve
│   ├── outwear
│   ├── pants
│   ├── shirt
│   ├── shoes
│   ├── shorts
│   ├── skirt
│   └── t-shirt
└── validation
    ├── dress
    ├── hat
    ├── longsleeve
    ├── outwear
    ├── pants
    ├── shirt
    ├── shoes
    ├── shorts
    ├── skirt
    └── t-shirt
```

Structure of the labelled dataset.



ANN Solution

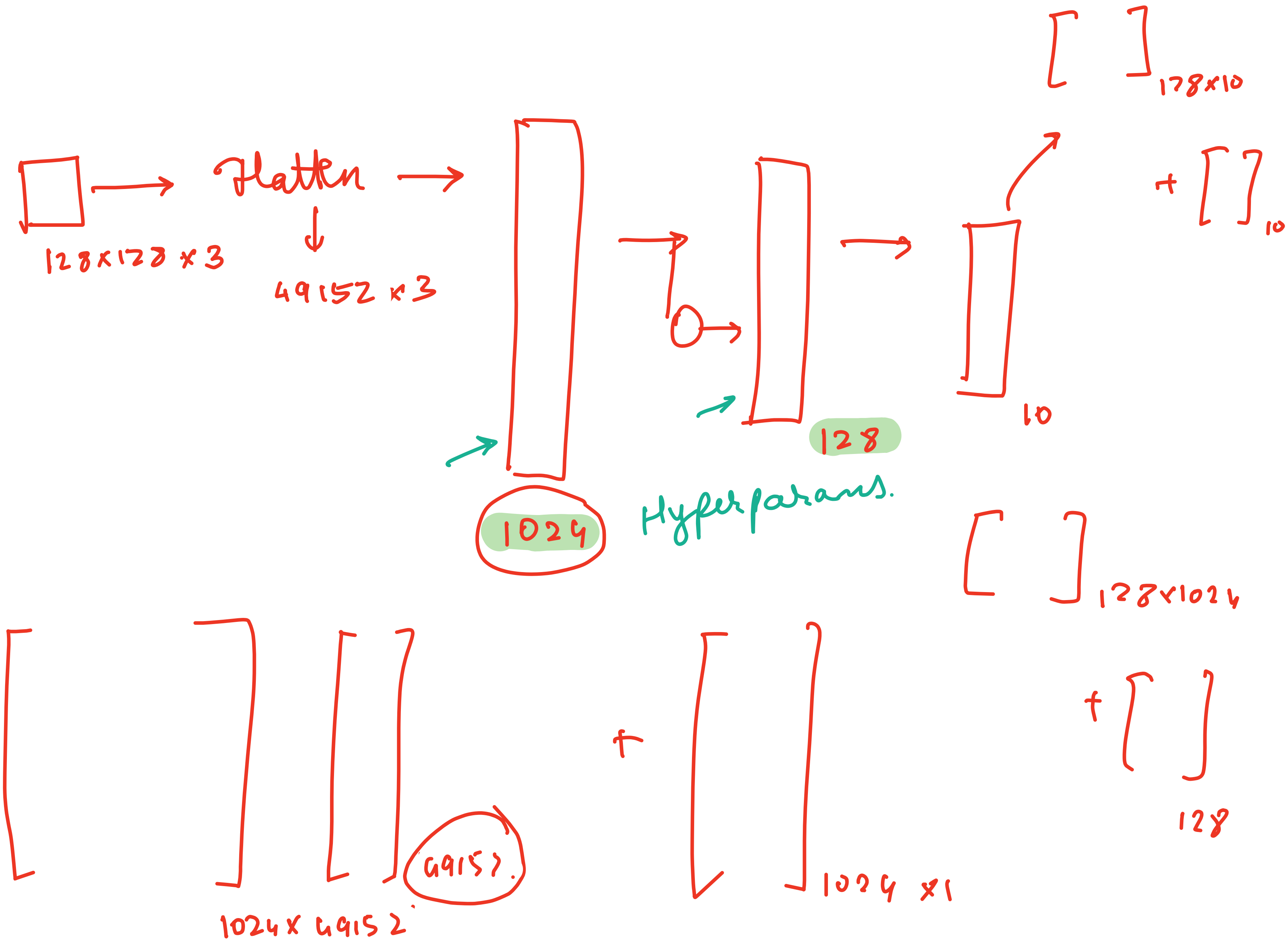
1. Resize

2. 

3. Value preprocessing : Scaling

So that [0 - 255] → [0 - 1].

[min-max scaling.


128x128 x 3
Flatten
49152 x 3
1024
128
10
Hyperparams.
128x10
+
10
128x1024
+
128
1024 x 49152
49152
+
1024 x1

Main Takeaway:

(Dense)

* CNNs >> ANNs

for Image Classification!!

49% > 35%

Next Class!!

* Why did it do better?
* Can we fix the overfitting problem?
* How are the different types of convs. useful here?

How to do better?

(a) More data

(b) Better preprocessing.

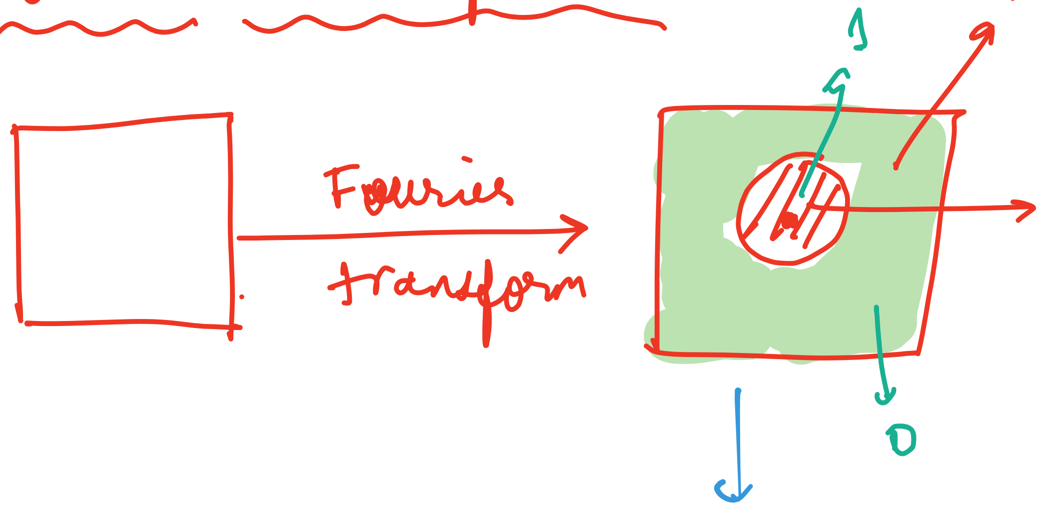
(c) Better learning rate.

(d) Different architecture.

(e) Regularization.

Dilated convolutions.

1] Fourier transform.

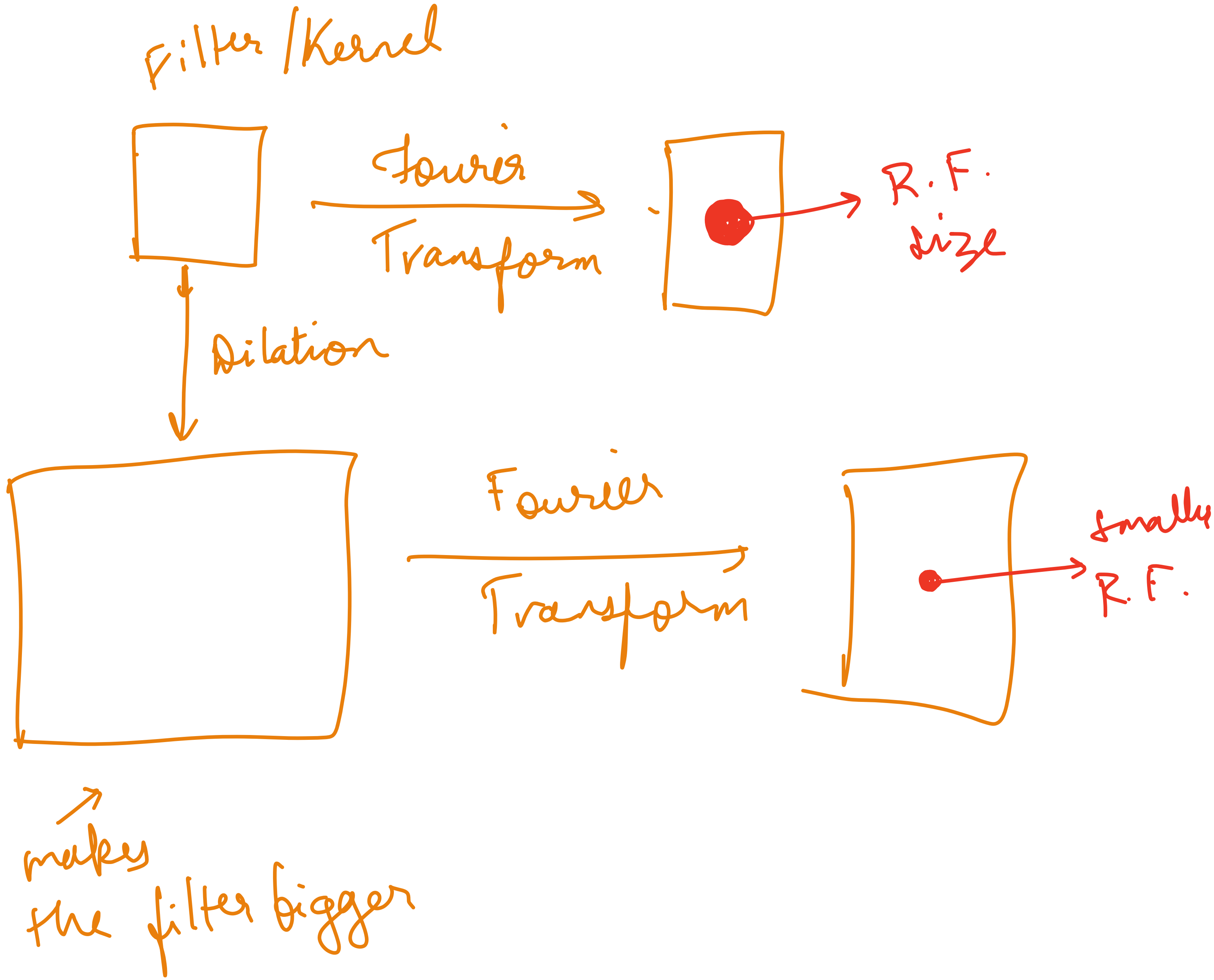


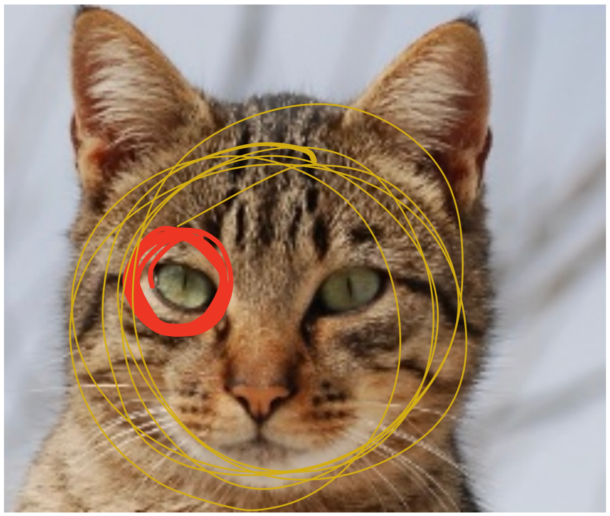
High frequency region.

low frequency region

Blurred version of the image.

Filter / Kernel

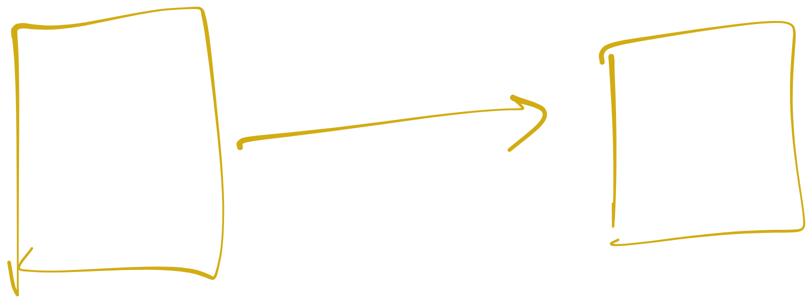

Filter/Kernel
Fourier
Transform
R.F.
size
Dilation
Fourier
Transform
smaller
R.F.
makes
the filter bigger



R.F → small

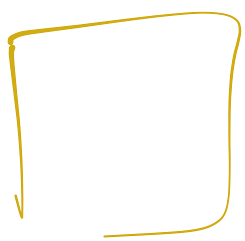
R.F → large

Step 1 : Take FFT → Fast Fourier transform.



FFT

Step 2 : Multiply with a low-pass mask.

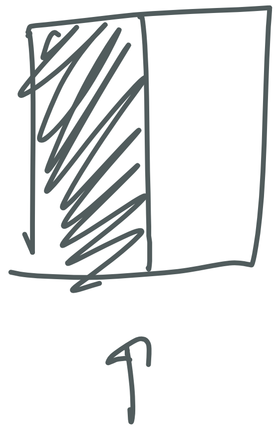


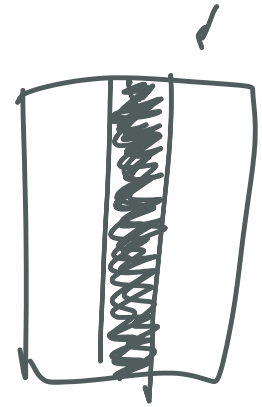
Im1 Im2

(1) $Im1 * Im2$

Are both equivalent!!

(2) $IFFT(FFT(IM1) \times FFT(IM2))$





0 30 30 0

10 10 10 0 0 0

10 10 10 0 0 0

Test:

im3, y_3

Train

im1, y_1

im2, y_2