Convolutional Neural Network (1)

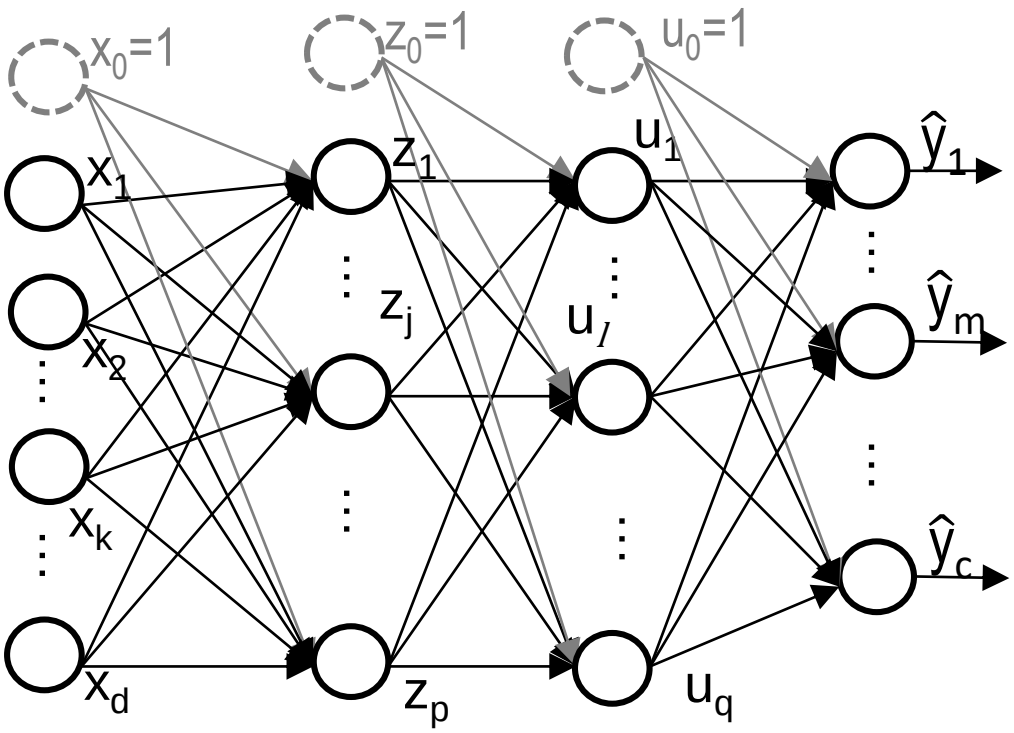
Hanwool Jeong

hwjeong@kw.ac.kr

Deep MLP Training

X

Design Matrix (N×d)



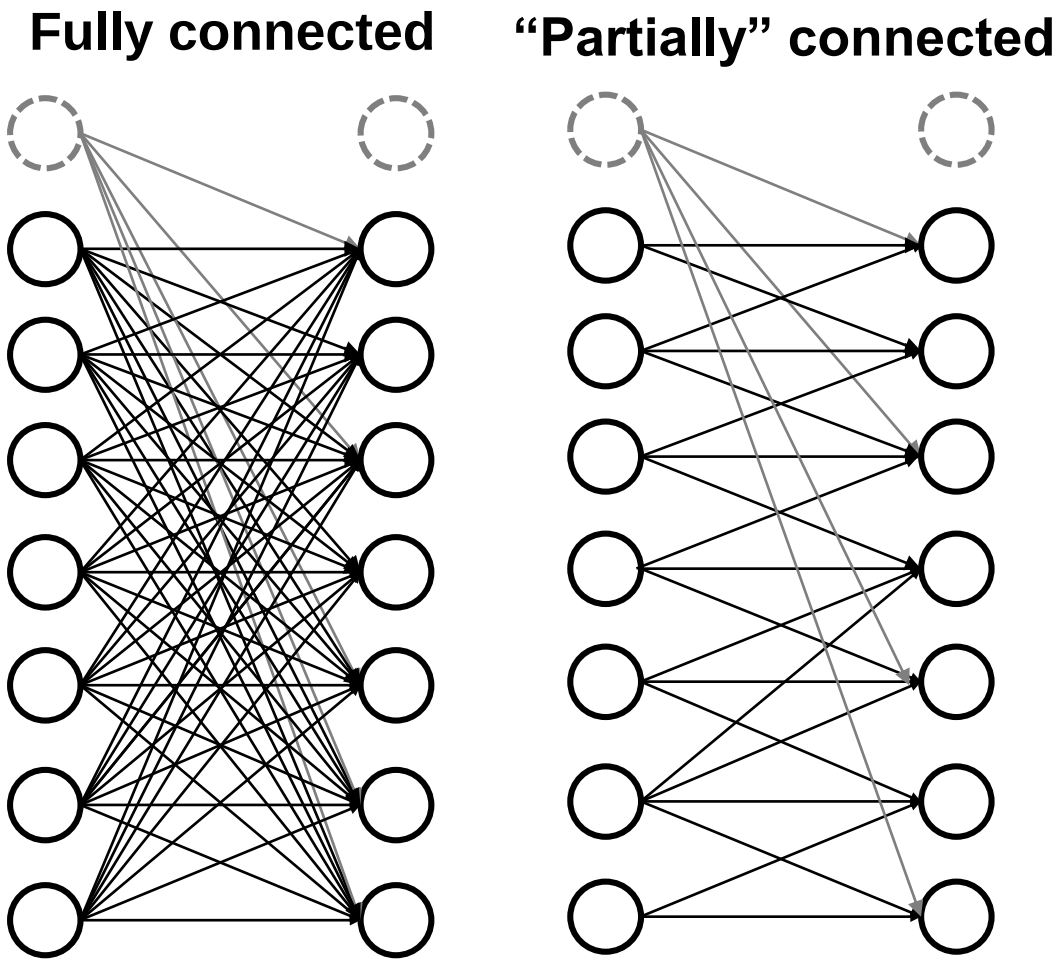
y

Response Vector (Nx1)

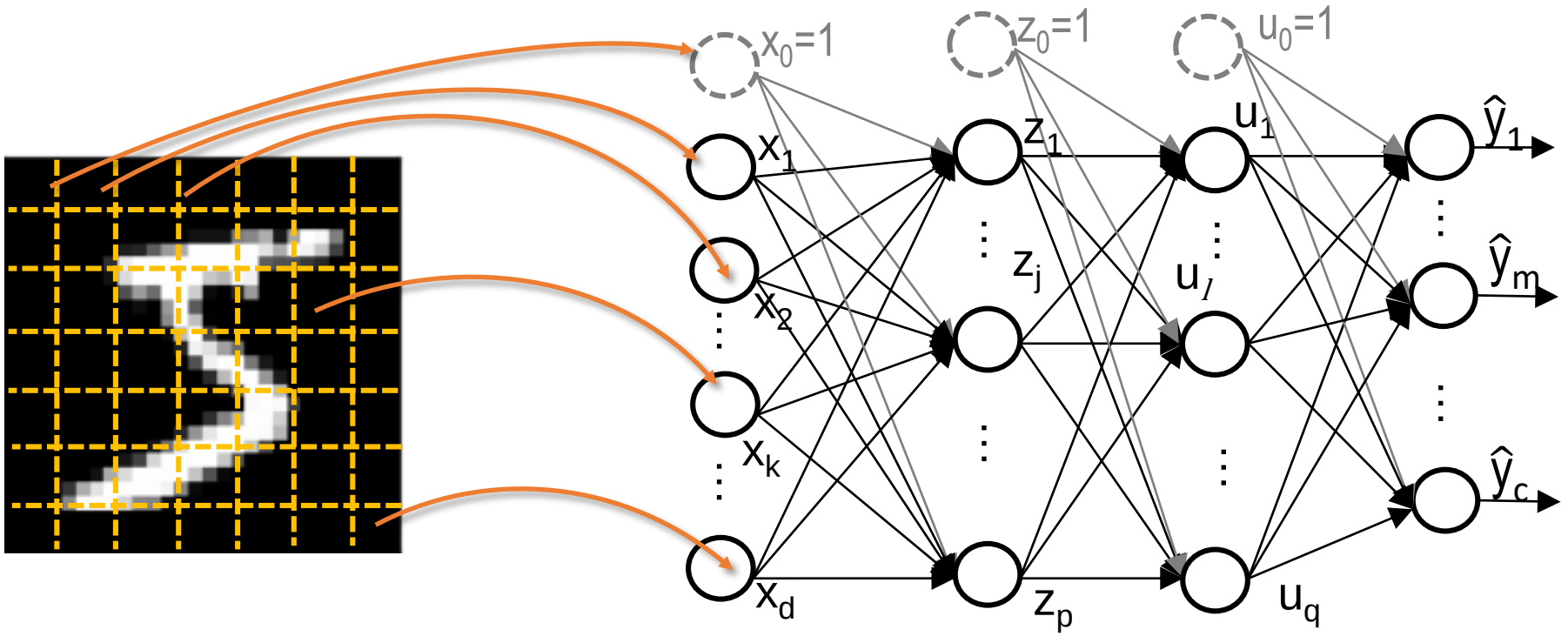
Large calculation from fully connected (imagine large N, d)

Revisit Deep MLP Limitations

- Large computation
- Overfitting risks



Loss of Information from Flattening



Flattening for Applying NN

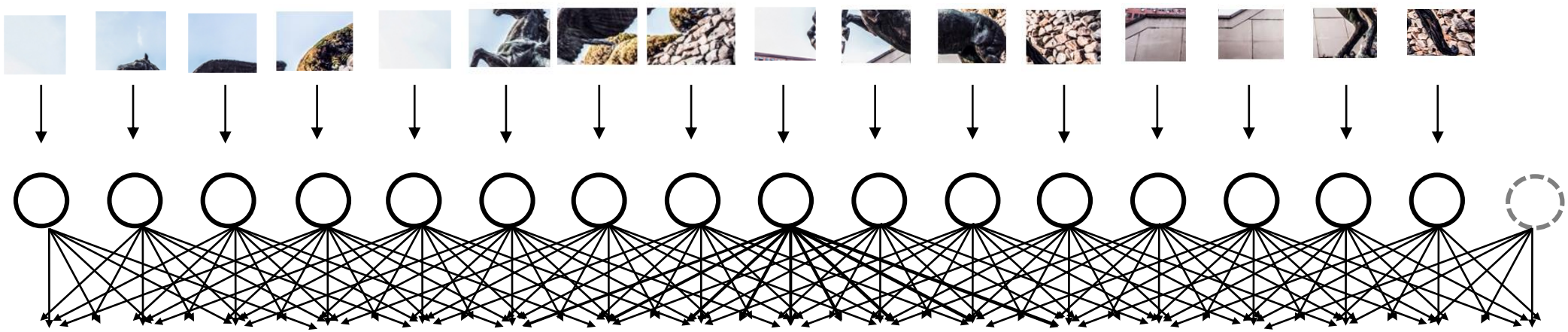
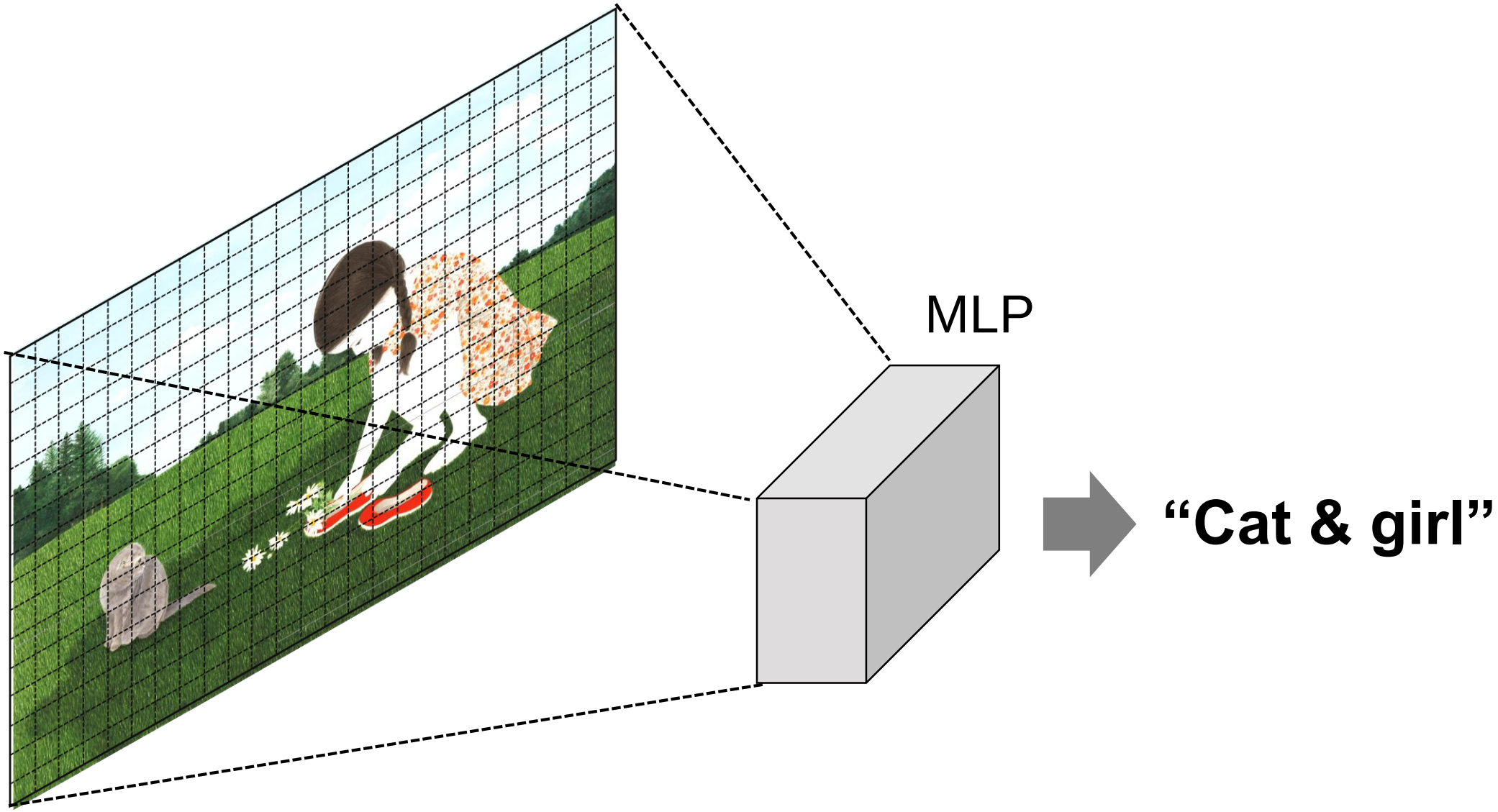


Image Recognition Example



Grid Structured Data

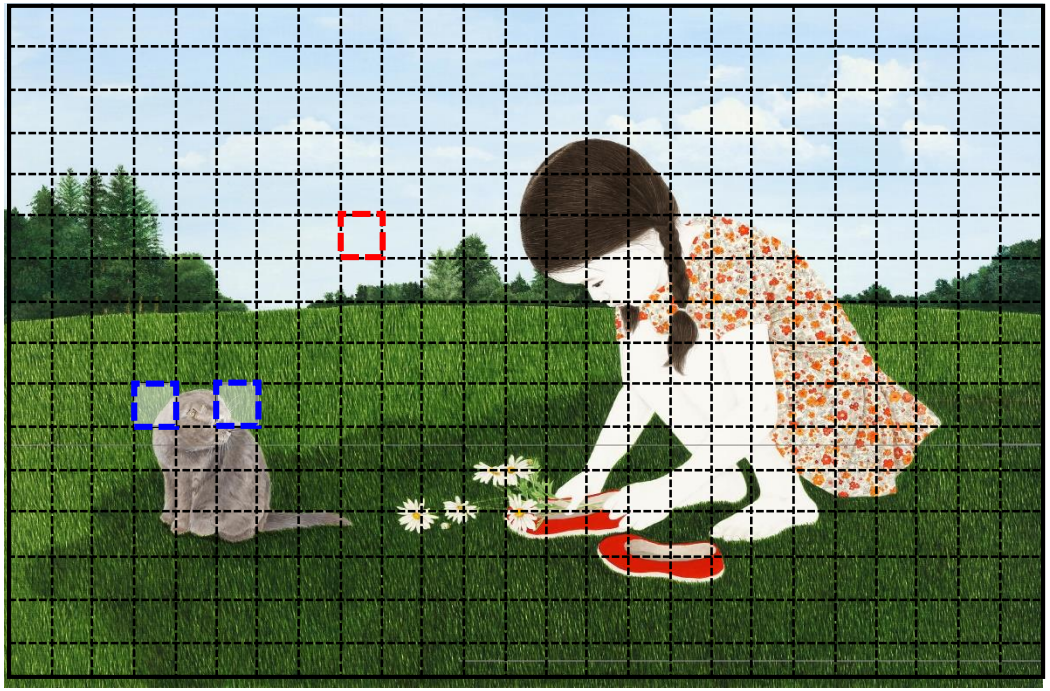
- May have millions of pixels
- Pixel location itself is highly valuable information.

➔ **Locality** feature**!**

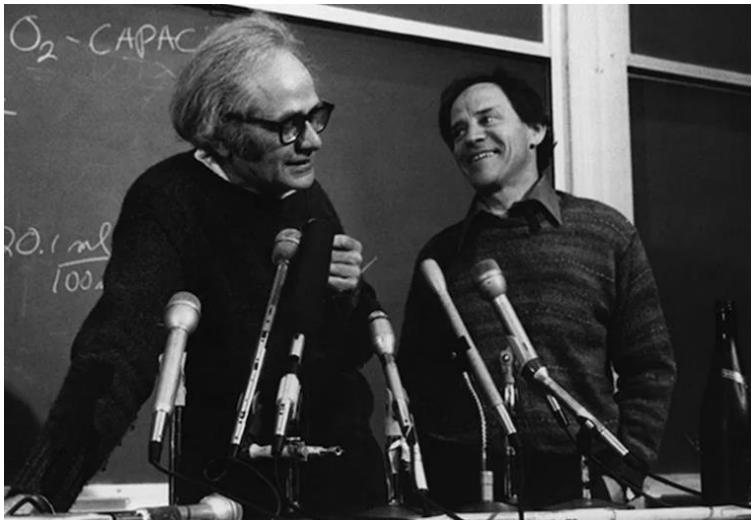
➔**Neighboring components have high correlation**

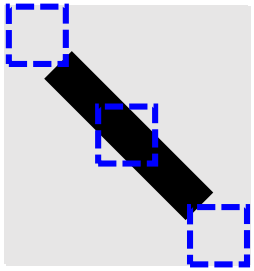
➔**Components far apart have little correlation**

How can we Consider this respect into MLP classification?

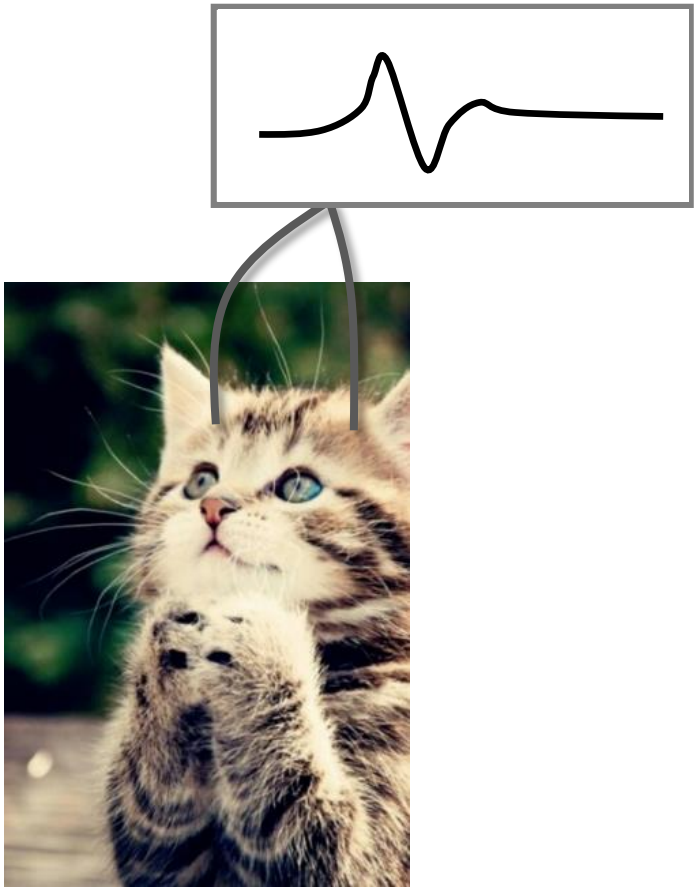


Research on Visual System and Brain of Animal by Hubel & Wiesel, 1959



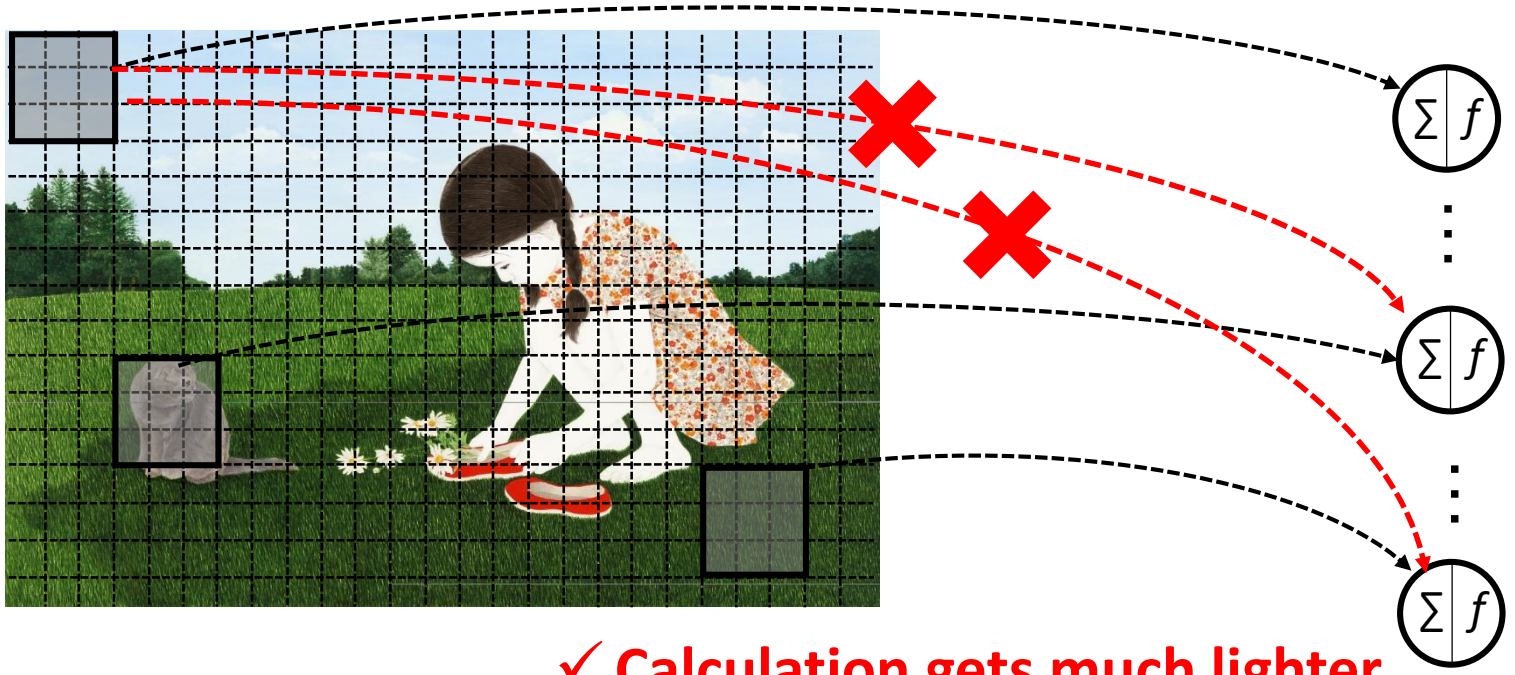


Processed by different **field**



Conclusion so Far;
Motivation for Partial Connection

- Calculation is heavy due to fully connection
- Locality should be considered for efficient information processing



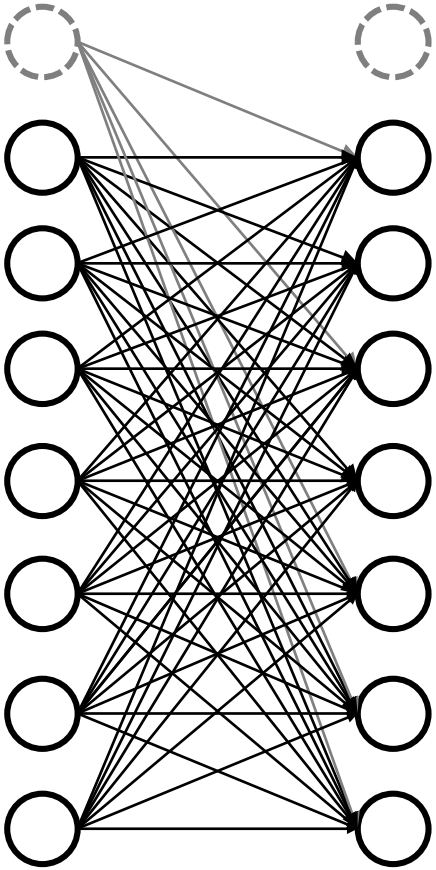
✓ **Calculation gets much lighter**

✓ **Considers locality**

Image, Voice, Video, etc..

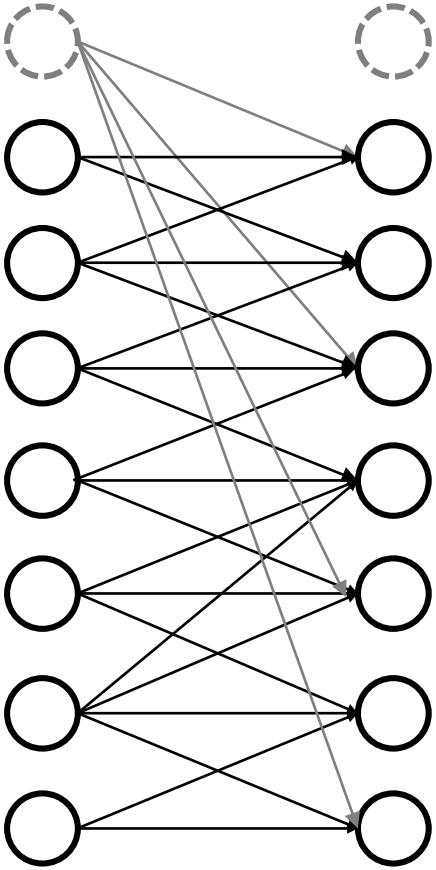
Revisit Fully Connection vs. Partially Connection

Fully connected

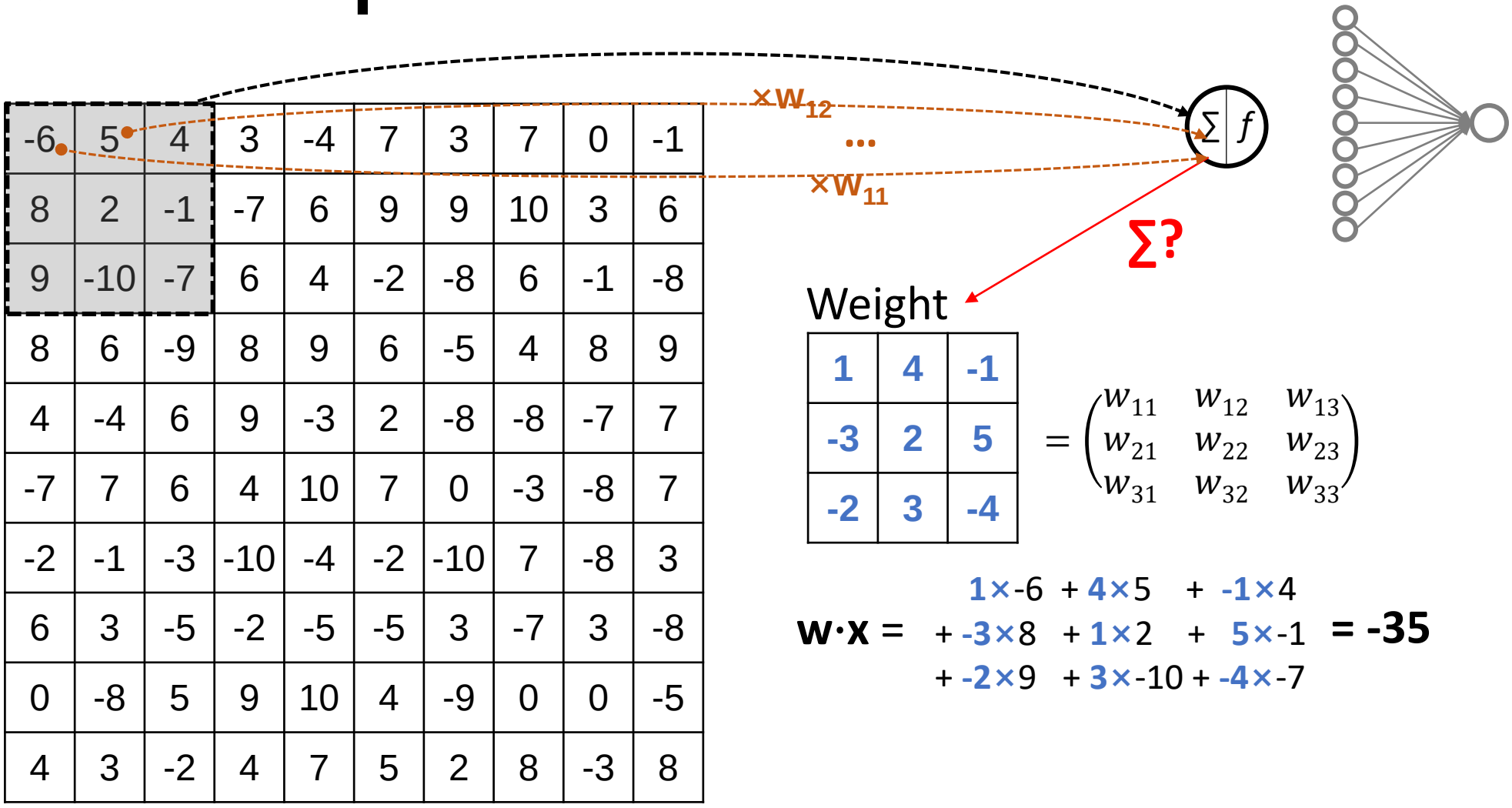


vs.

“Partially” connected



Inner Product Calculation; 3x3 example



-6	5	4	3	-4	7	3	7	0	-1
8	2	-1	-7	6	9	9	10	3	6
9	-10	-7	6	4	-2	-8	6	-1	-8
8	6	-9	8	9	6	-5	4	8	9
4	-4	6	9	-3	2	-8	-8	-7	7
-7	7	6	4	10	7	0	-3	-8	7
-2	-1	-3	-10	-4	-2	-10	7	-8	3
6	3	-5	-2	-5	-5	3	-7	3	-8
0	-8	5	9	10	4	-9	0	0	-5
4	3	-2	4	7	5	2	8	-3	8

Weight

1	4	-1
-3	2	5
-2	3	-4

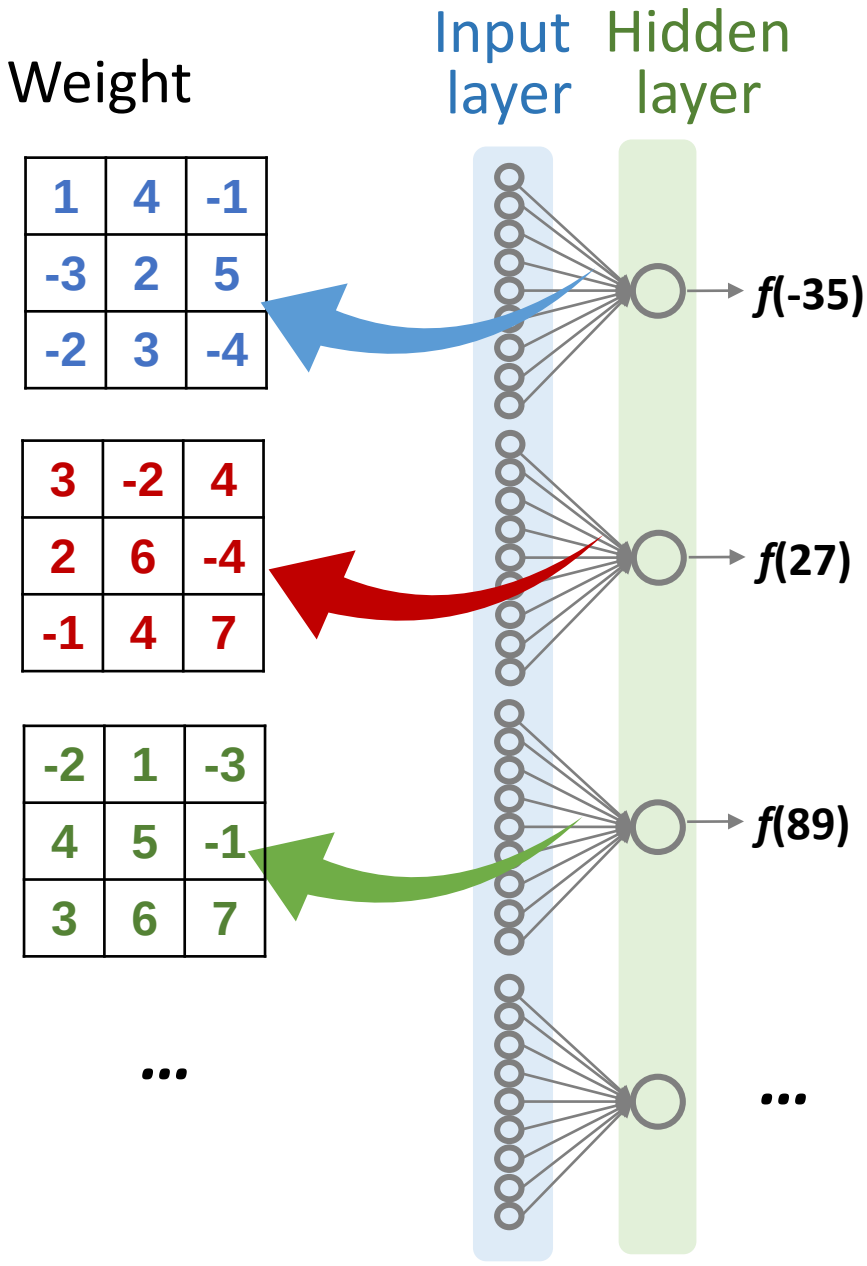
$$= \begin{pmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{pmatrix}$$

$$\mathbf{w \cdot x} = \begin{matrix} 1\times-6 + 4\times5 + -1\times4 \\ +\,-3\times8 + 1\times2 + 5\times-1 \\ +\,-2\times9 + 3\times-10 + -4\times-7 \end{matrix} = -35$$

What's Next Step?

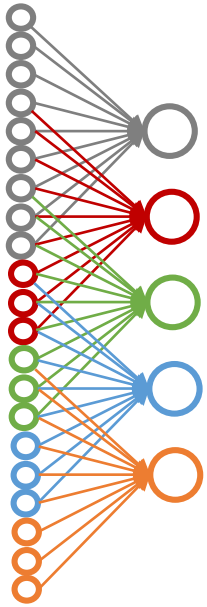
Correlated Correlated :Why are these ignored?

-6	5	4	3	-4	7	3	7	0	-1
8	2	-1	-7	6	9	9	10	3	6
9	-10	-7	6	4	-2	-8	6	-1	-8
8	6	-9	8	9	6	-5	4	8	9
4	-4	6	9	-3	2	-8	-8	-7	7
-7	7	6	4	10	7	0	-3	-8	7
-2	-1	-3	-10	-4	-2	-10	7	-8	3
6	3	-5	-2	-5	-5	3	-7	3	-8
0	-8	5	9	10	4	-9	0	0	-5
4	3	-2	4	7	5	2	8	-3	8



Overlap is Desirable

-6	5	4	3	-4	7	3	7	0	-1
8	2	-1	-7	6	9	9	10	3	6
9	-10	-7	6	4	-2	-8	6	-1	-8
8	6	-9	8	9	6	-5	4	8	9
4	-4	6	9	-3	2	-8	-8	-7	7
-7	7	6	4	10	7	0	-3	-8	7
-2	-1	-3	-10	-4	-2	-10	7	-8	3
6	3	-5	-2	-5	-5	3	-7	3	-8
0	-8	5	9	10	4	-9	0	0	-5
4	3	-2	4	7	5	2	8	-3	8



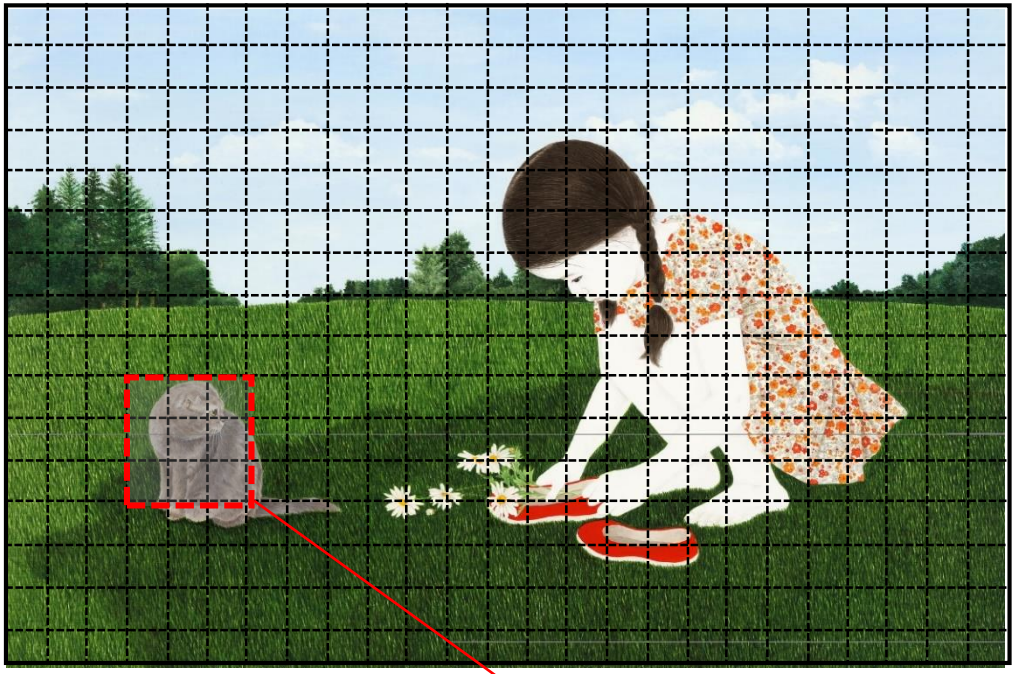
You see
partial connection?

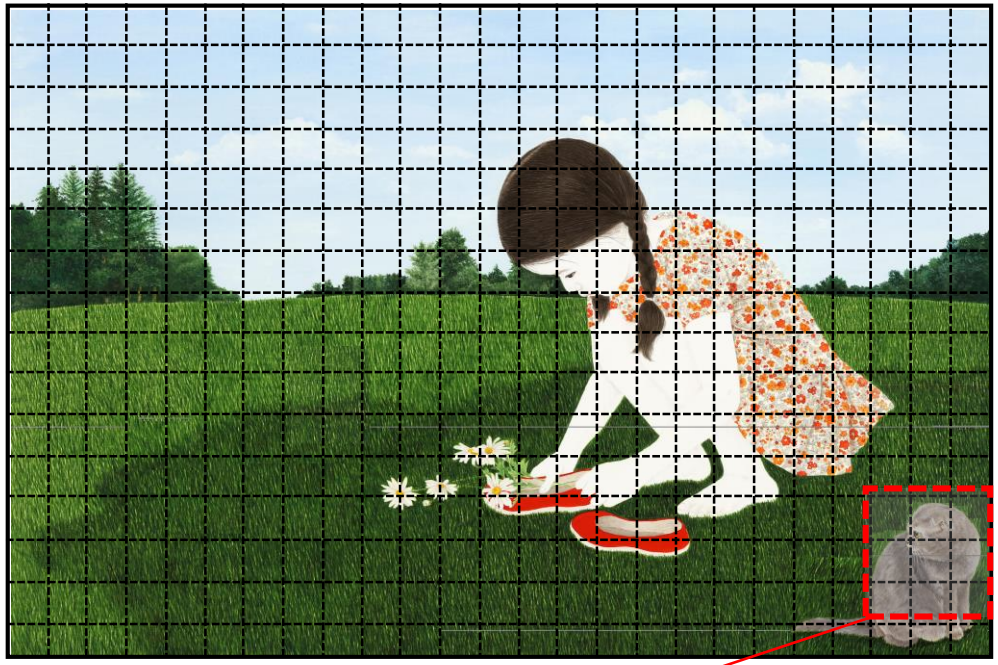
Locality
is considered

+ lighter calculation

Stride can be different

What Else Should We Consider in Image?





Should have the same calculation results, like

I see hair

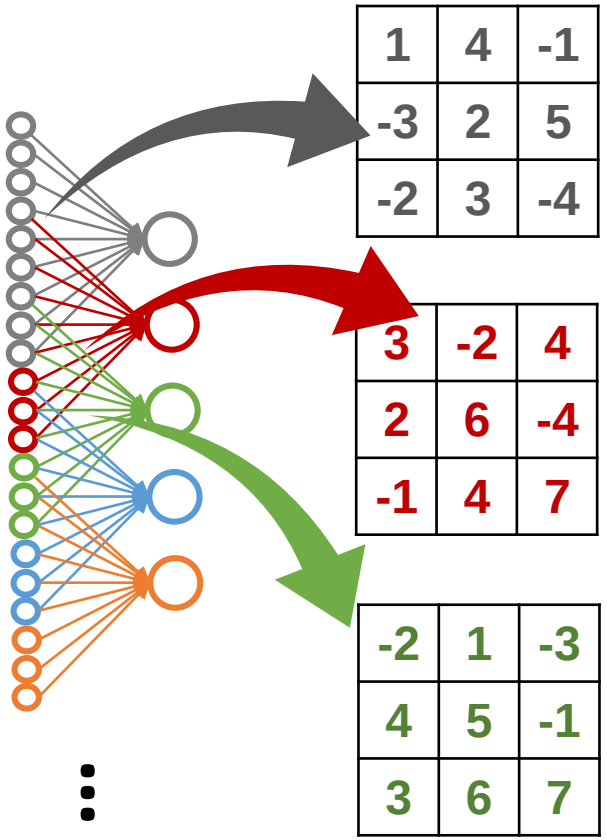
I see an eye

I see something…. cute…

How can we Consider this respect into MLP classification?

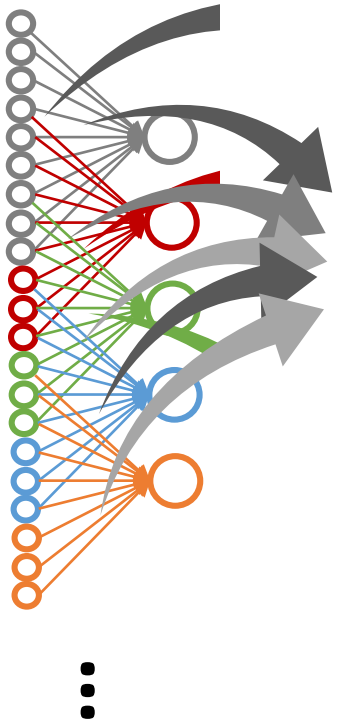
Revisit “Weight” in MLP

-6	5	4	3	-4	7	3	7	0	-1
8	2	-1	-7	6	9	9	10	3	6
9	-10	-7	6	4	-2	-8	6	-1	-8
8	6	-9	8	9	6	-5	4	8	9
4	-4	6	9	-3	2	-8	-8	-7	7
-7	7	6	4	10	7	0	-3	-8	7
-2	-1	-3	-10	-4	-2	-10	7	-8	3
6	3	-5	-2	-5	-5	3	-7	3	-8
0	-8	5	9	10	4	-9	0	0	-5
4	3	-2	4	7	5	2	8	-3	8



Using the Same Weight for Translation Invariance to Location

-6	5	4	3	-4	7	3	7	0	-1
8	2	-1	-7	6	9	9	10	3	6
9	-10	-7	6	4	-2	-8	6	-1	-8
8	6	-9	8	9	6	-5	4	8	9
4	-4	6	9	-3	2	-8	-8	-7	7
-7	7	6	4	10	7	0	-3	-8	7
-2	-1	-3	-10	-4	-2	-10	7	-8	3
6	3	-5	-2	-5	-5	3	-7	3	-8
0	-8	5	9	10	4	-9	0	0	-5
4	3	-2	4	7	5	2	8	-3	8



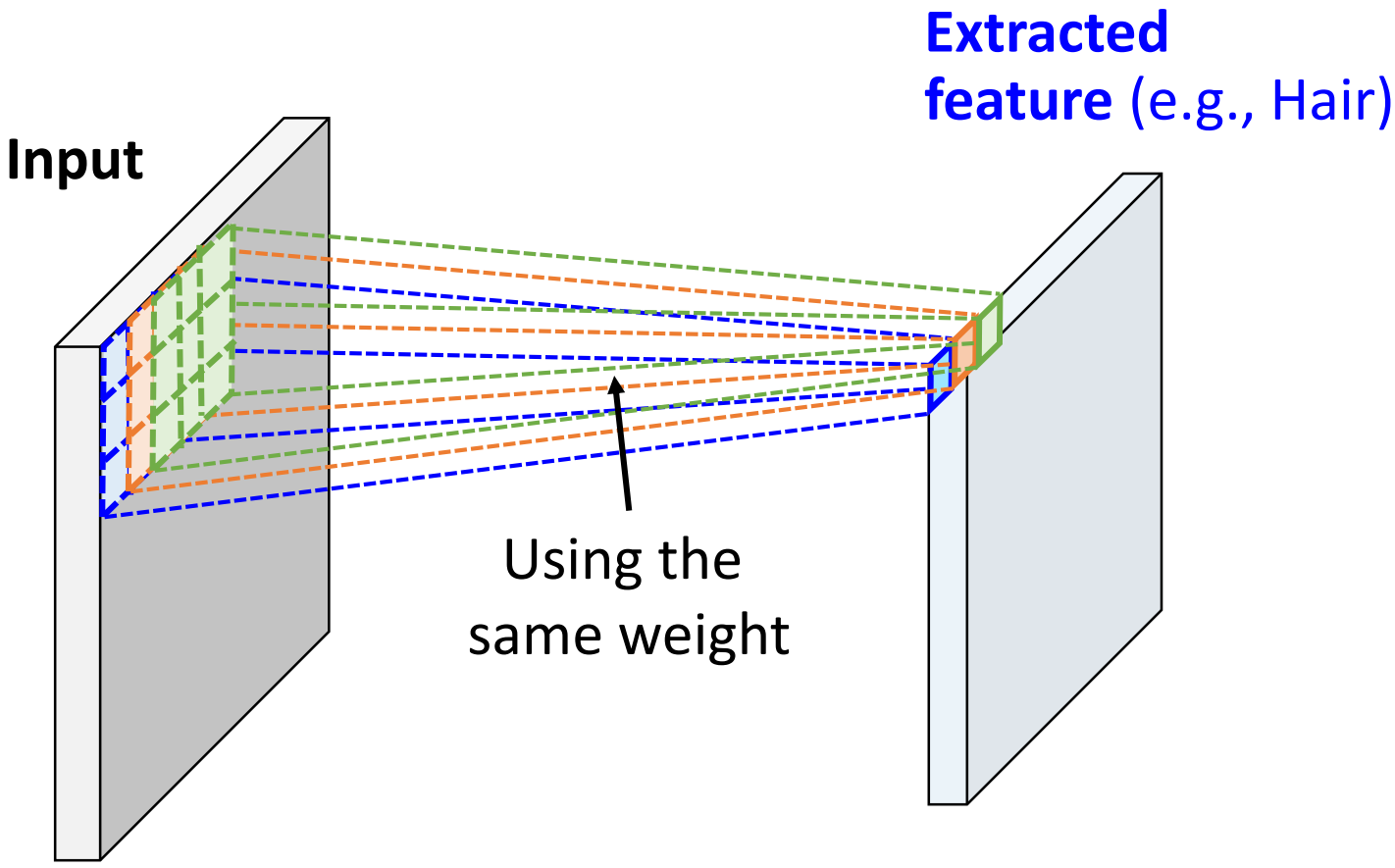
1	**4**	**-1**
-3	**2**	**5**
-2	**3**	**-4**

Feature

"Extracting hair"

➔ **Parameter Sharing or Weight Sharing**

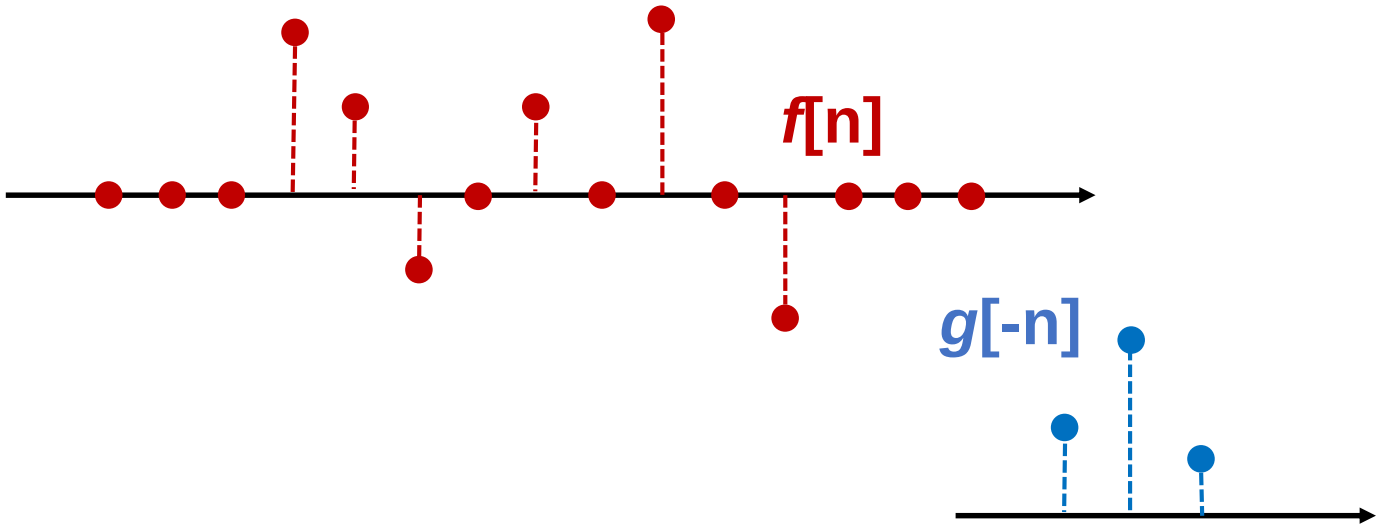
Typical Representation of NN for Image Tensor



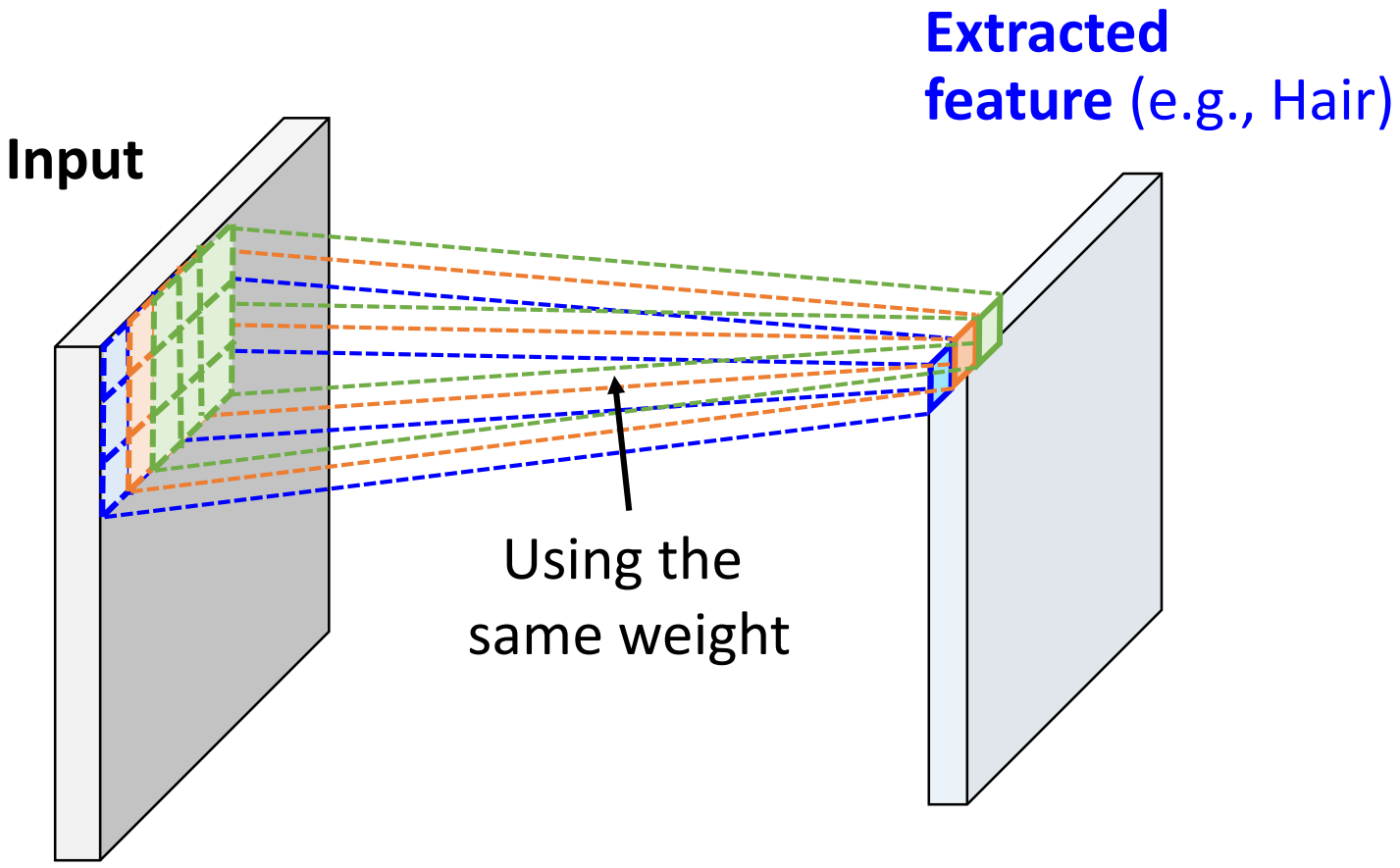
You Feel Something Familiar

- Revisit the convolution operation:

$$(f * g)[n] = \sum_{m=-\infty}^{\infty} f[m]g[n-m]$$



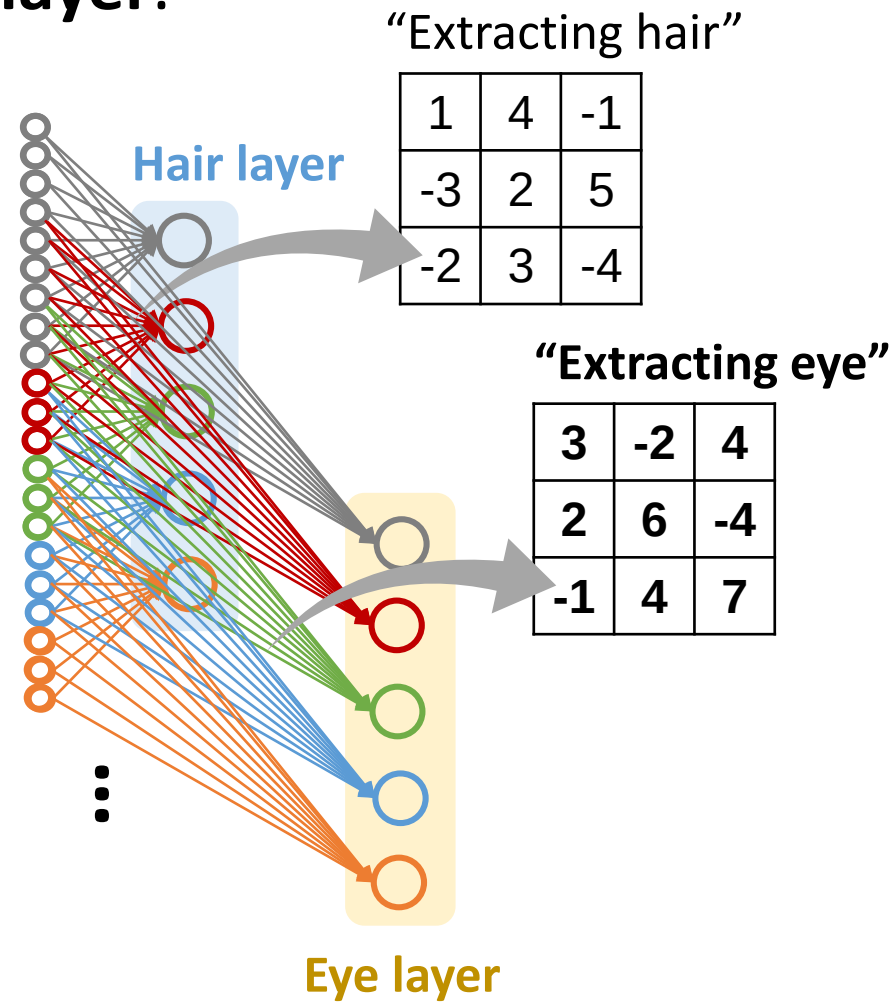
Called
“Convolution Neural Network” (CNN)



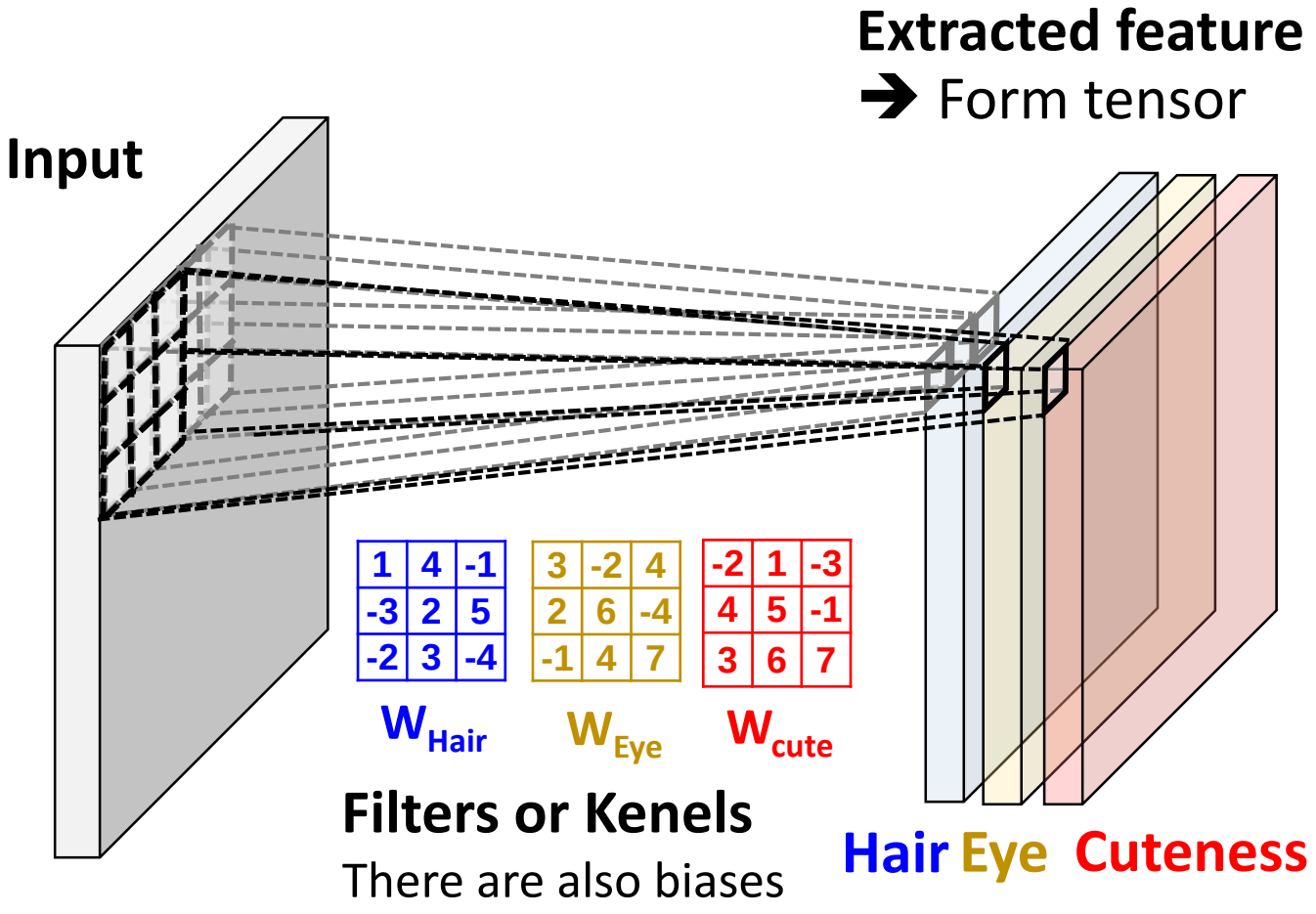
How About “Eye” or “Cuteness”?

- We can repeat the weighted sum operation for the input **to form another feature extracted layer**.

-6	5	4	3	-4	7	3	7	0	-1
8	2	-1	-7	6	9	9	10	3	6
9	-10	-7	6	4	-2	-8	6	-1	-8
8	6	-9	8	9	6	-5	4	8	9
4	-4	6	9	-3	2	-8	-8	-7	7
-7	7	6	4	10	7	0	-3	-8	7
-2	-1	-3	-10	-4	-2	-10	7	-8	3
6	3	-5	-2	-5	-5	3	-7	3	-8
0	-8	5	9	10	4	-9	0	0	-5
4	3	-2	4	7	5	2	8	-3	8

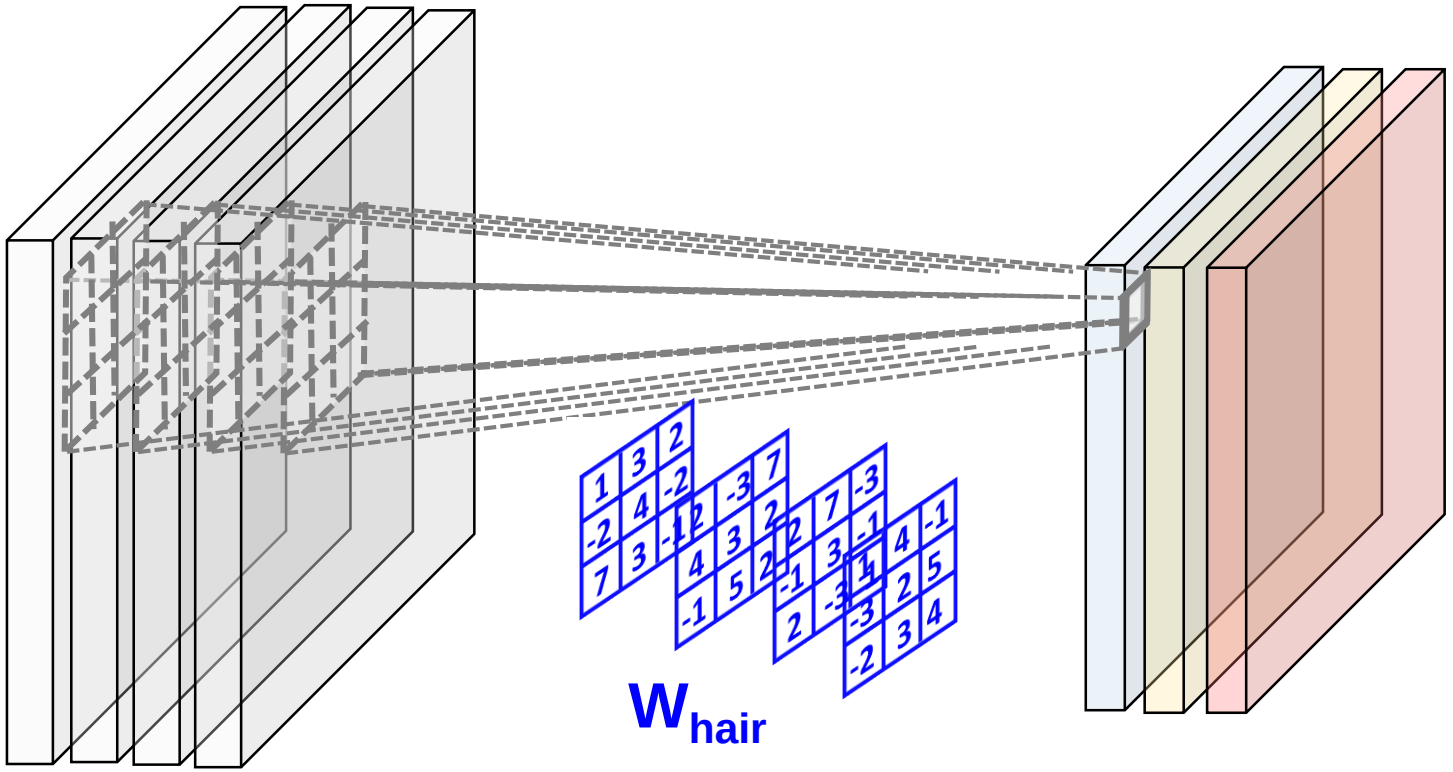


Multiple Feature Maps



Muti-Channel Input

Multi-channel Input
(e.g., RGB)



➔ **Feature Map extraction for 2D multi-channel input in CNN**

Checkpoints

- ✓Development of CNN from Deep MLP
- ✓Locality
- ✓Parameter sharing
- ✓Multiple feature map
- ✓Multiple channel input
- ✓Coming up next: multi-layer CNN and various CNN models