UNIVERSIDAD EAFIT

Ph. D. Juan David Martínez Vargas
Ph. D. Raul Andrés Castañeda

Escuela de Ciencias Aplicadas e Ingeniería

Lecture 07
Deep Learning

Agenda

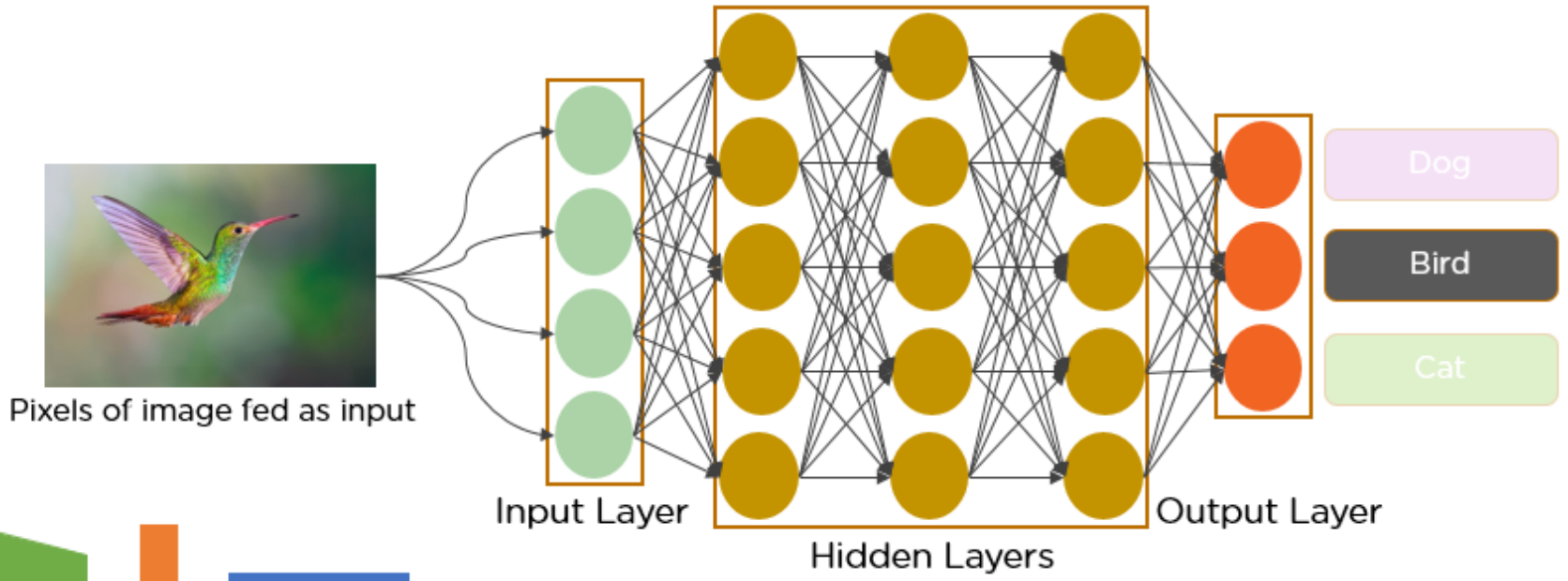
- **Redes Convolucionales**
- **Convolución**
- **ReLu**
- **Zero Padding**

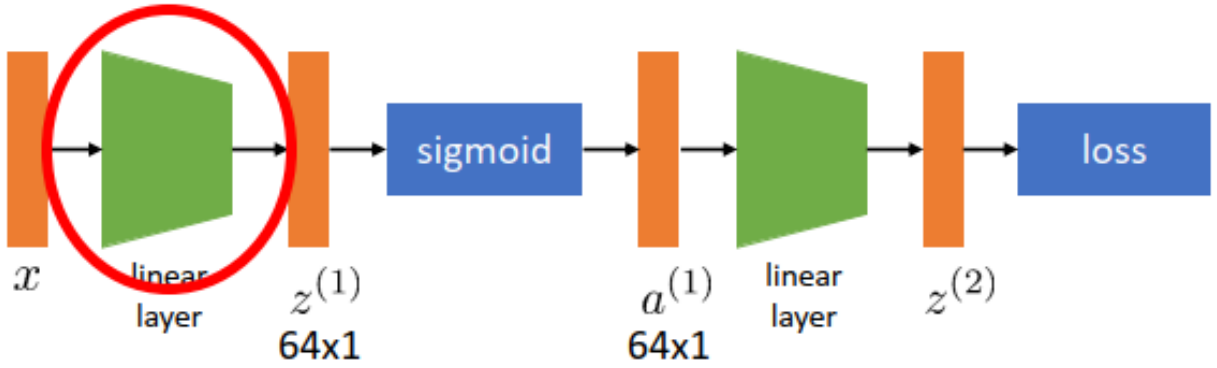
Redes Convolucionales

Las redes neuronales convolucionales (**CNN**) son una clase de arquitecturas diseñadas para datos con una topología tipo cuadrícula, donde las correlaciones espaciales locales se pueden explotar mediante operaciones de convolución.

Cuando se usa una red densa tradicional: La imagen debe convertirse en un vector.

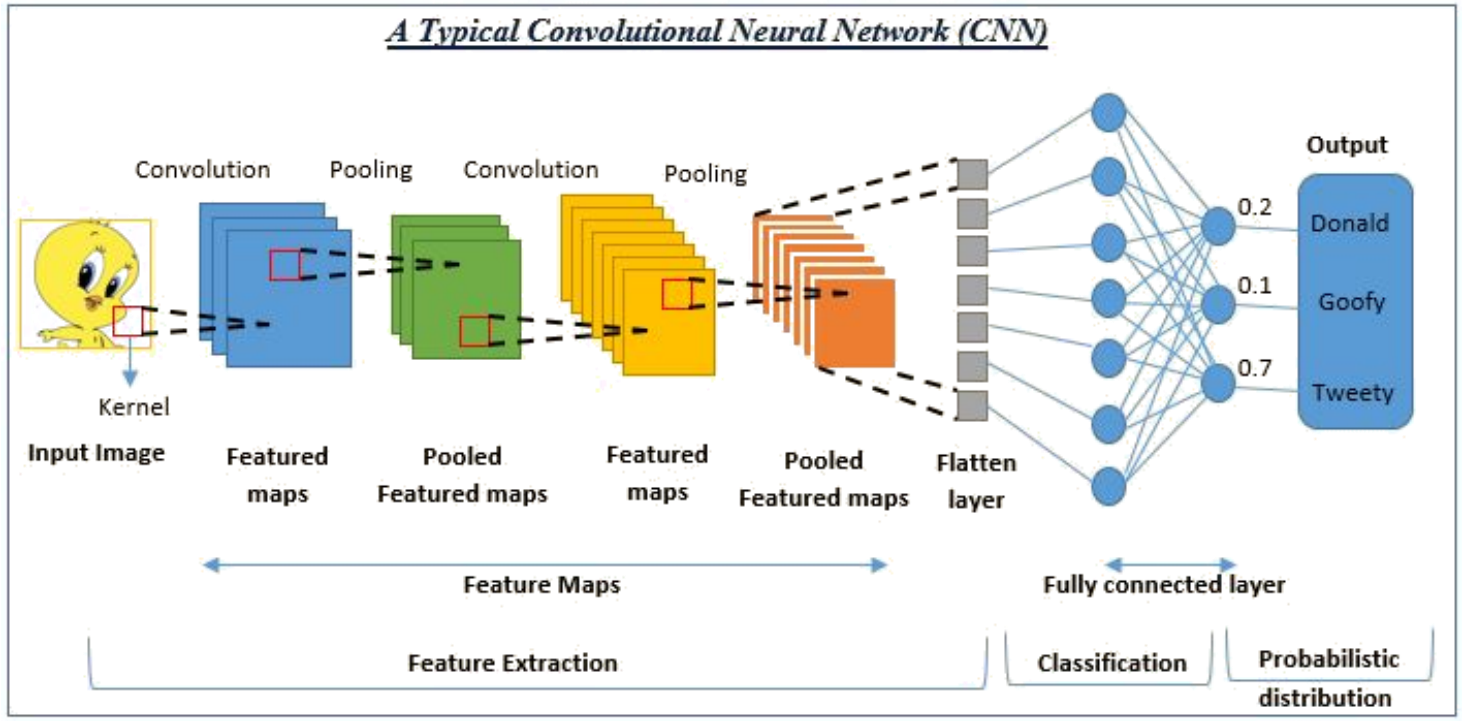
128 × 128 × 3 = 49.152
64 × 49152 ≈ 3'000.000

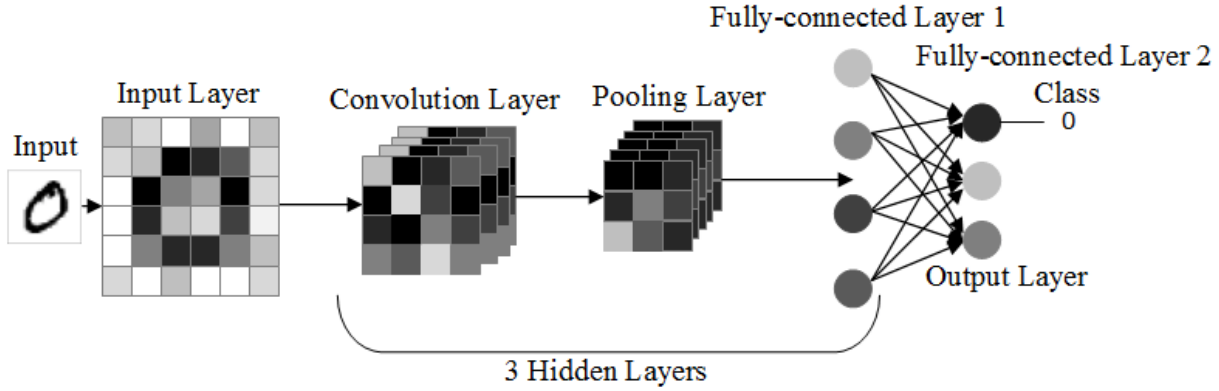




UNIVERSIDAD EAFIT

Redes Convolucionales





- En lugar de conectar todas las neuronas con todos los píxeles, la red aplica filtros pequeños llamados **kernels** que recorren la imagen y detectan patrones locales, como bordes, texturas o formas.
- Estas operaciones **generan mapas de características** (feature maps), que luego se reducen mediante capas de **pooling**, conservando la información más relevante.
- Finalmente, la información se aplana (**flatten**) y pasa a capas totalmente conectadas que realizan la clasificación.

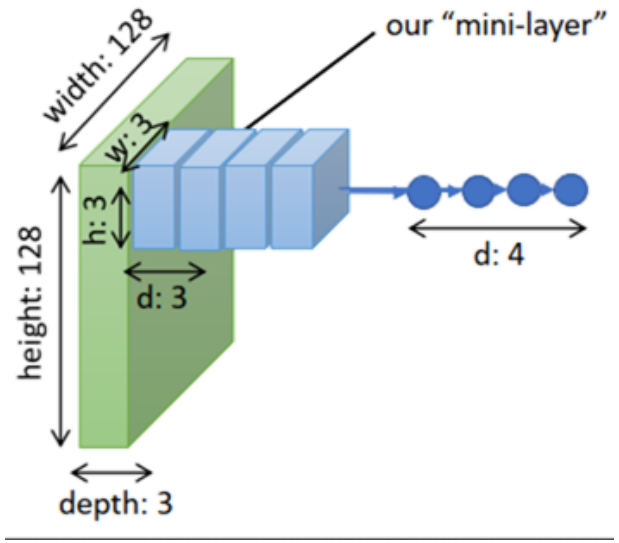
UNIVERSIDAD EAFIT®

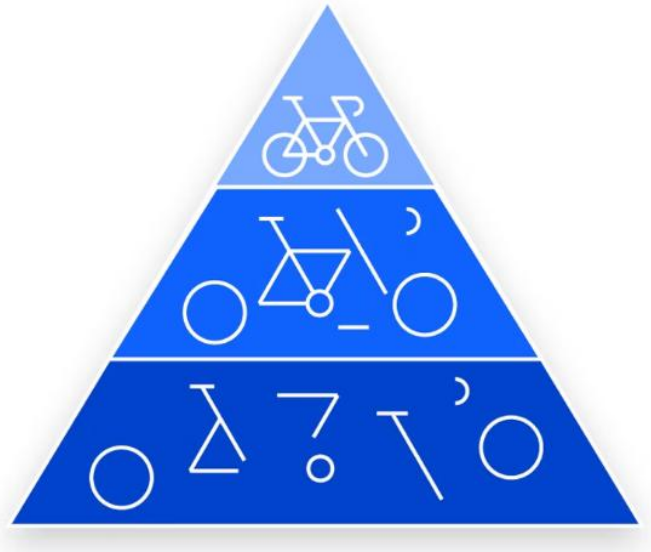
Redes Convolucionales

Conectar cada píxel con cada neurona, implica que el número de parámetros crece absurdamente rápido.

Las características importantes en imágenes son locales.

- ✓ Observar **regiones (patches)**
- ✓ Usar **filtros (kernels)**
- ✓ Reutilizamos los mismos pesos en toda la imagen





Los filtros son los componentes fundamentales de la capa convolucional, a partir de la cual se construyen los mapas de características.

Cada uno se centra en una característica específica que busca en la imagen a procesar.

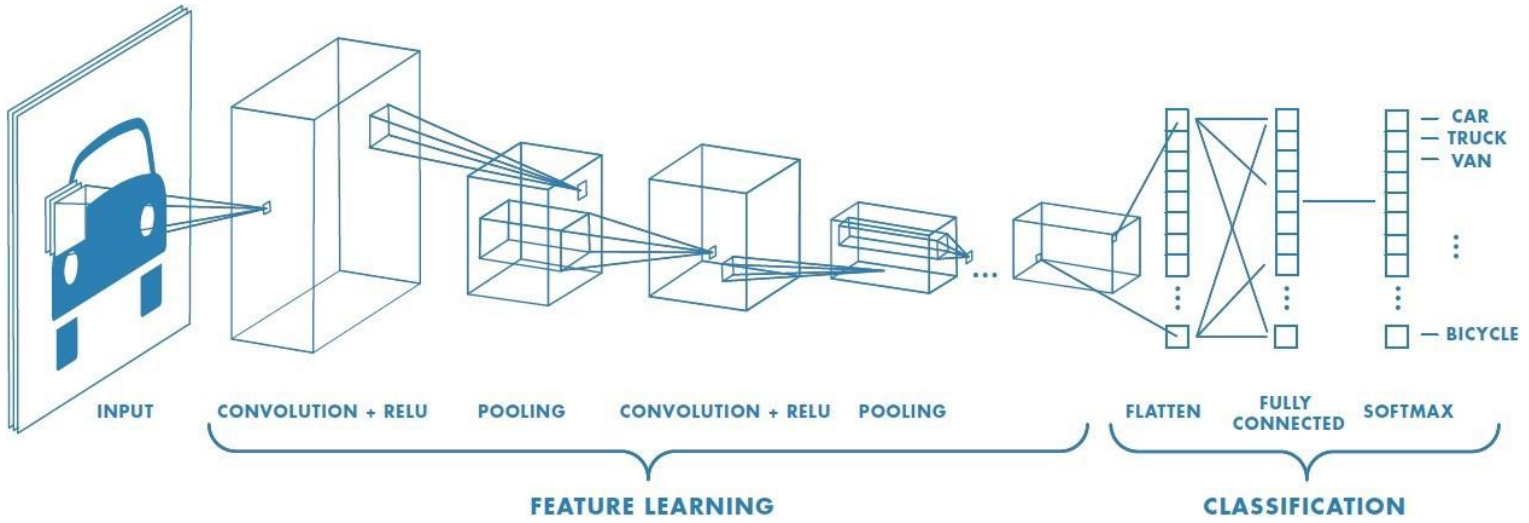
Redes Convolucionales

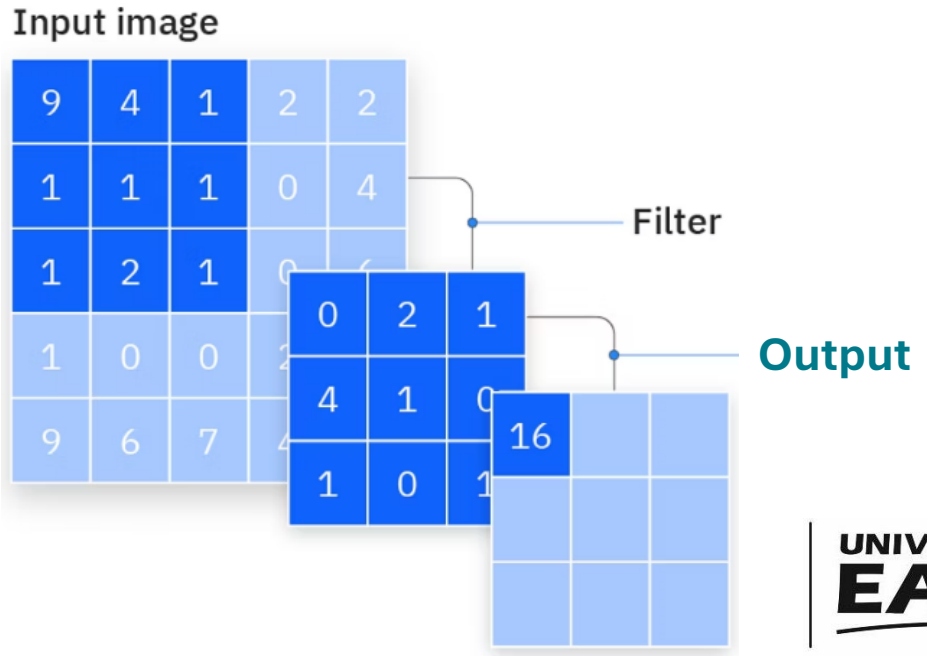
Breakdown of the key components

Capa Convolucional: El componente fundamental de las CNN. Implica un conjunto de filtros (Kernels). La convolución es una operación matemática que combina los datos de entrada con el filtro para **generar un mapa de características**.

Función de Activación: Comúnmente se aplica para introducir no linealidad. Esto ayuda a la red a aprender relaciones complejas en los datos.

Pooling Layer: Reduce las dimensiones espaciales de la representación y la cantidad de computación en la red (comúnmente Agrupamiento Máximo o Agrupamiento Promedio). El agrupamiento ayuda a **retener la información más importante** y descartar las características menos importantes.

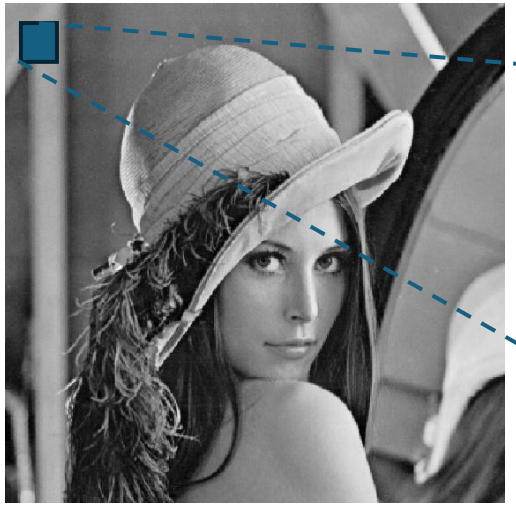




UNIVERSIDAD EAFIT

Kernels / Convolución

¿Qué es un kernel?



Image

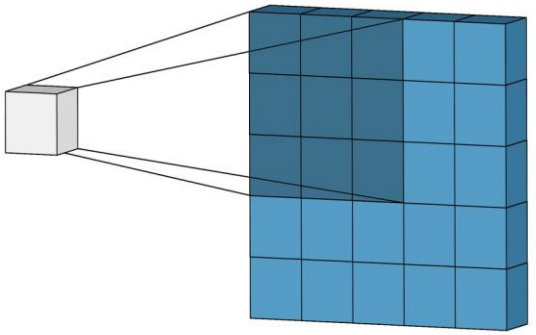
0	2	3
2	4	1
0	3	0

$\otimes$

Kernel

0	1	0
1	1	1
0	1	0

$$(0 \cdot 0) + (2 \cdot 1) + (3 \cdot 0) + (2 \cdot 1) + (4 \cdot 1) + (1 \cdot 1) + (0 \cdot 0) + (3 \cdot 1) + (0 \cdot 0) = 12$$

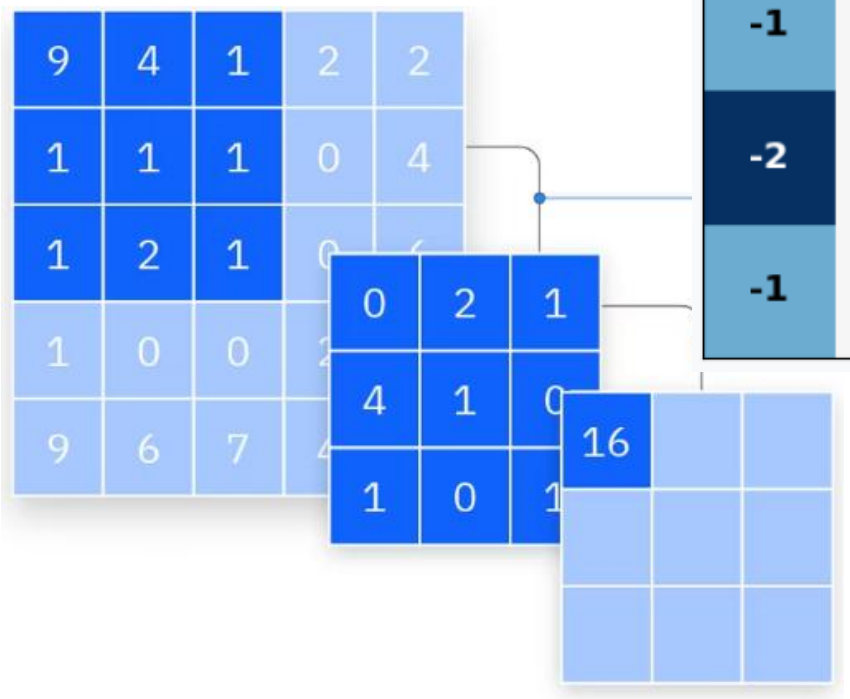


Un **kernel** es una pequeña matriz (ej: 3×3) que se desliza sobre la imagen de entrada realizando una operación llamada "**convolución**", donde se multiplica elemento a elemento con cada región de la imagen y se suman los resultados, produciendo un solo valor por posición.

$$W_{out} = W_{in} - K + 1 \qquad \sum_m \sum_n [\mathrm{Im}\,age \otimes Filter] = 12$$

$$H_{out} = H_{in} - K + 1 \qquad \sum_m \sum_n [i+m, j+n] \cdot K[m,n]$$

Kernels / Convolución



Sobel Vertical

-1	0	1
-2	0	2
-1	0	1

Sobel Horizontal

-1	-2	-1
0	0	0
1	2	1

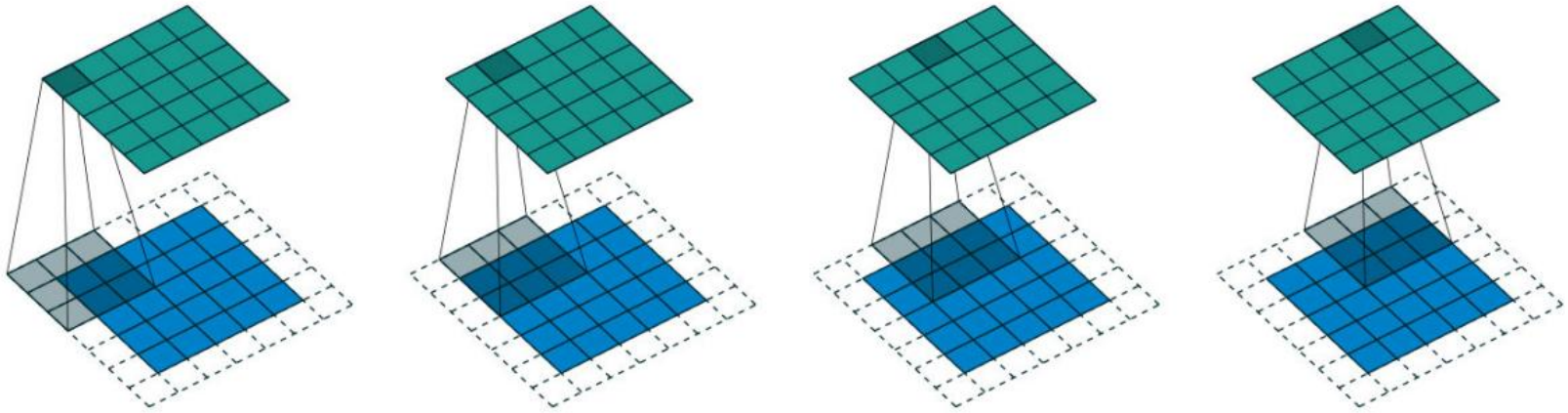
Laplaciano

0	-1	0
-1	4	-1
0	-1	0

Prewitt Diagonal

1	0	-1
0	0	0
-1	0	1

Los filtros NO se programan, se entrenan



Kernels / Convolución

Sobel Vertical

-1	0	1
-2	0	2
-1	0	1

Sobel Horizontal

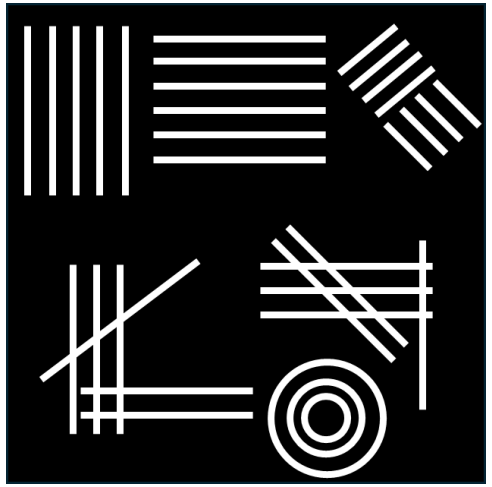
-1	-2	-1
0	0	0
1	2	1

Laplaciano

0	-1	0
-1	4	-1
0	-1	0

Prewitt Diagonal

1	0	-1
0	0	0
-1	0	1

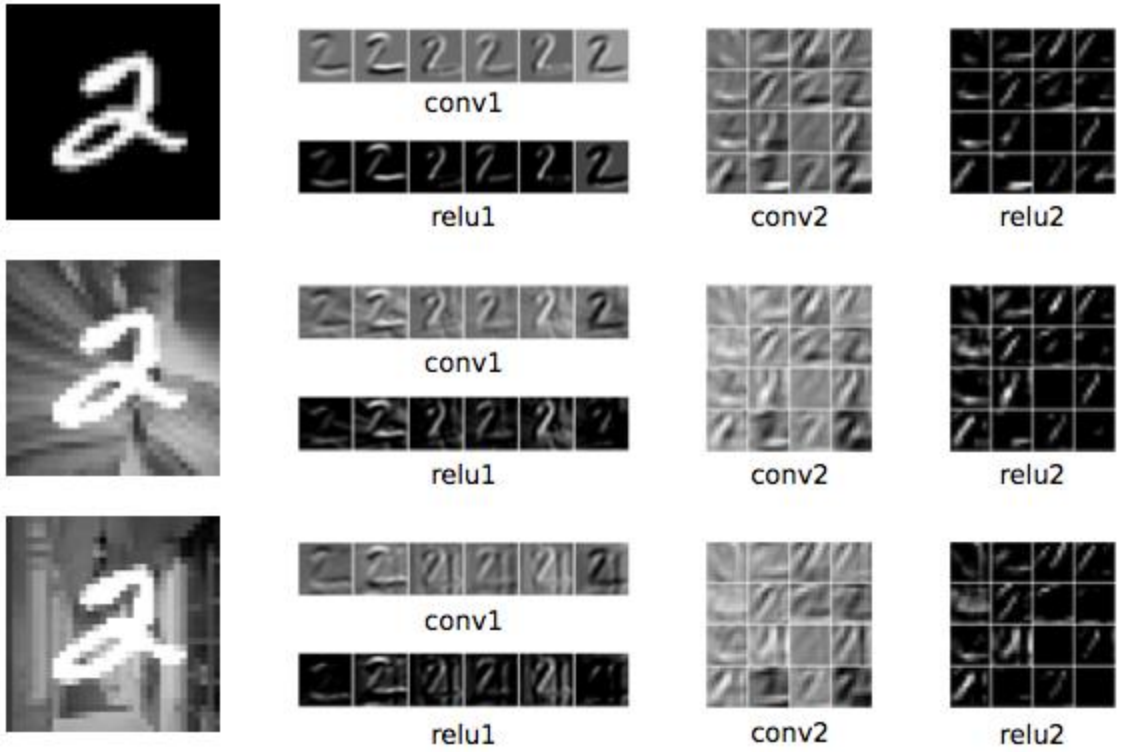


Al aplicar cada filtro sobre una imagen se genera el **mapa de características** resultante.

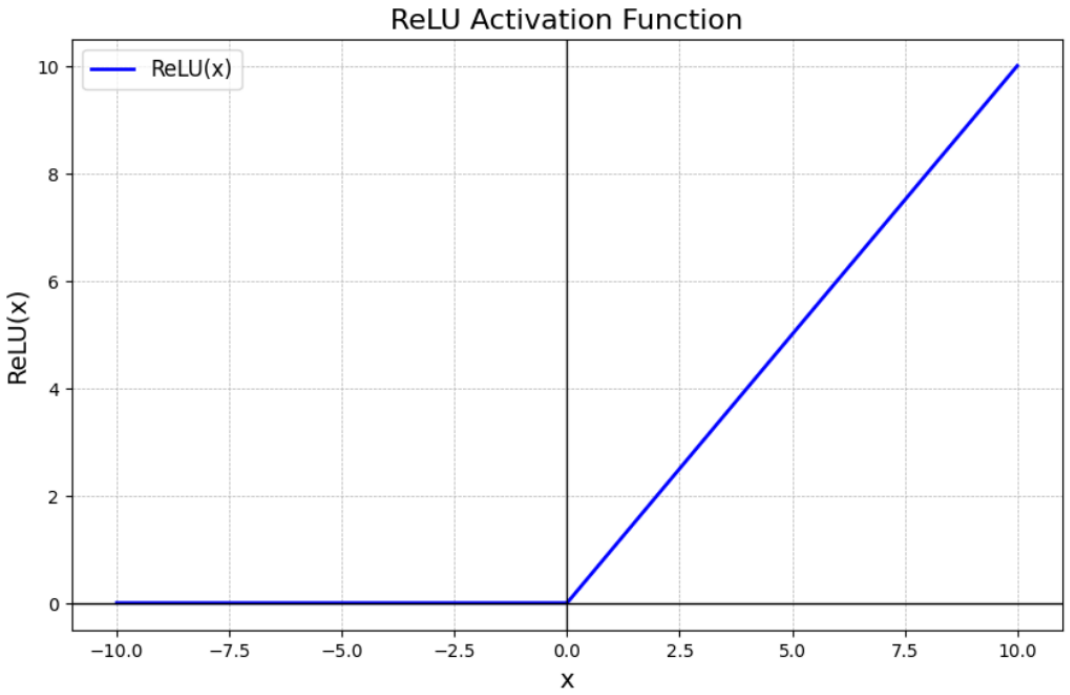
La idea clave: un filtro que detecta bordes verticales solo activará “fuertemente” en las regiones donde la imagen tenga ese patrón.

UNIVERSIDAD EAFIT

Kernels / Convolución



ReLU Activation Function $f(x) = \max(0, x)$

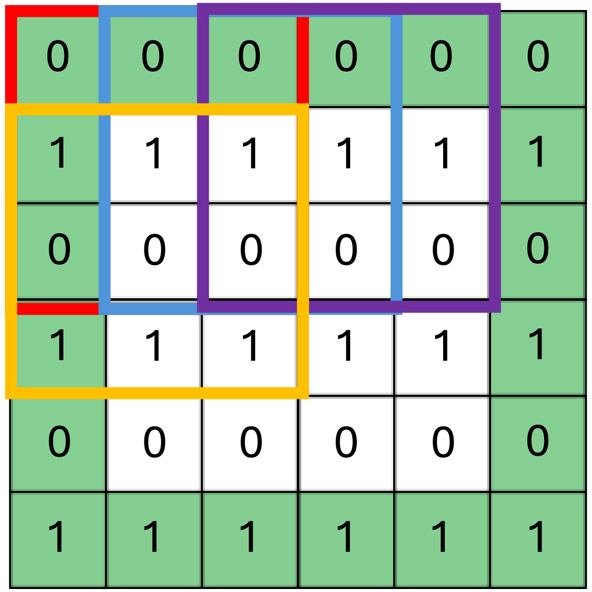


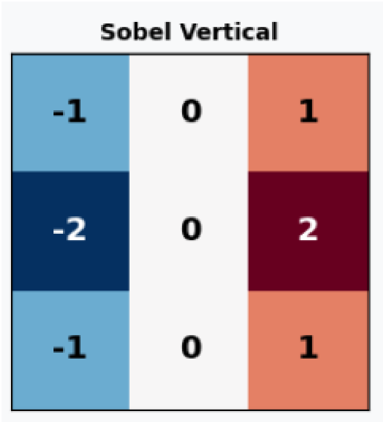
- ReLU is computationally efficient as it involves only a thresholding operation.
- ReLU is a non-linear function. This allows the model to learn more complex data patterns and model intricate relationships between features.

UNIVERSIDAD EAFIT

Kernels / Convolución

Image



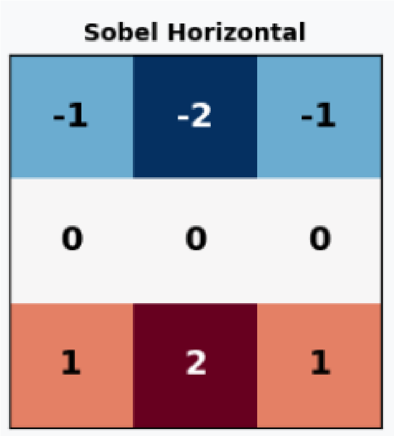


$$(0\cdot-1)+(0\cdot0)+(0\cdot1)+(1\cdot-2)+(1\cdot0)+(1\cdot2)+(0\cdot-1)+(0\cdot0)+(0\cdot1)=0$$

$$(0\cdot-1)+(0\cdot0)+(0\cdot1)+(1\cdot-2)+(1\cdot0)+(1\cdot2)+(0\cdot-1)+(0\cdot0)+(0\cdot1)=0$$

$$(0\cdot-1)+(0\cdot0)+(0\cdot1)+(1\cdot-2)+(1\cdot0)+(1\cdot2)+(0\cdot-1)+(0\cdot0)+(0\cdot1)=0$$

$$(1\cdot-1)+(1\cdot0)+(1\cdot1)+(0\cdot-2)+(0\cdot0)+(0\cdot2)+(1\cdot-1)+(1\cdot0)+(1\cdot1)=0$$



$$(0\cdot-1)+(0\cdot-2)+(0\cdot-1)+(1\cdot0)+(1\cdot0)+(1\cdot0)+(0\cdot1)+(0\cdot2)+(0\cdot1)=0$$

$$(0\cdot-1)+(0\cdot-2)+(0\cdot-1)+(1\cdot0)+(1\cdot0)+(1\cdot0)+(0\cdot1)+(0\cdot2)+(0\cdot1)=0$$

$$(0\cdot-1)+(0\cdot-2)+(0\cdot-1)+(1\cdot0)+(1\cdot0)+(1\cdot0)+(0\cdot1)+(0\cdot2)+(0\cdot1)=0$$

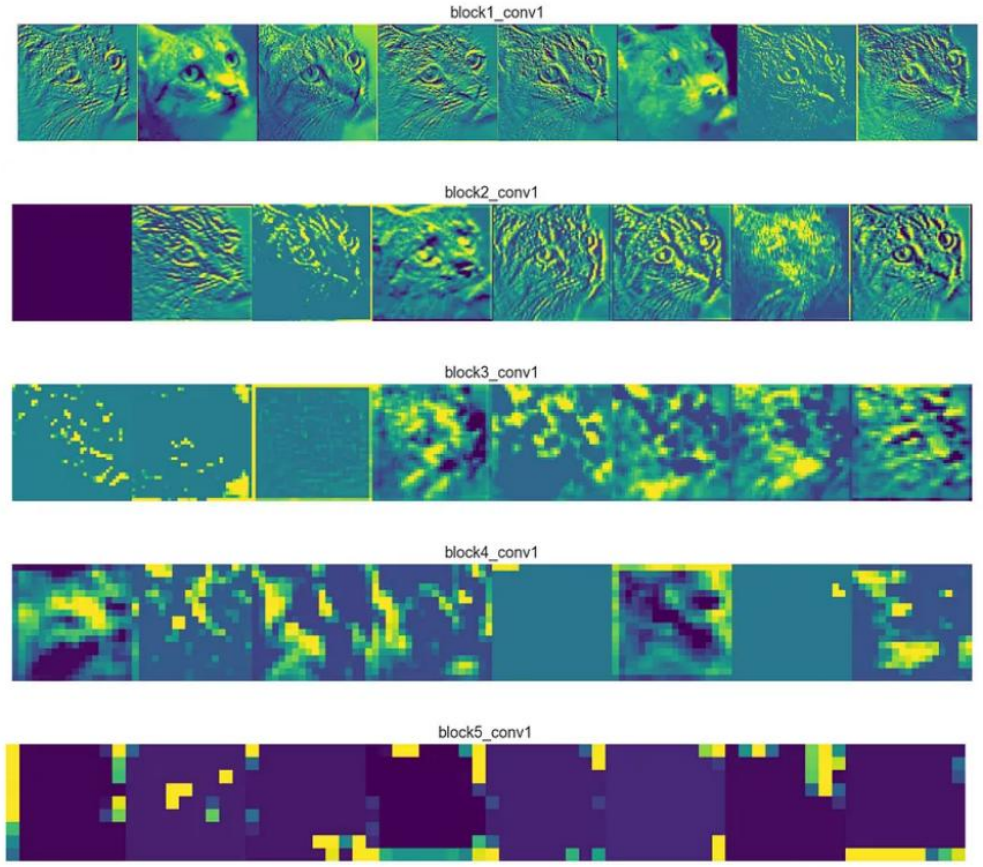
$$(1\cdot-1)+(1\cdot-2)+(1\cdot-1)+(0\cdot0)+(0\cdot0)+(0\cdot0)+(1\cdot1)+(1\cdot2)+(1\cdot1)=0$$

IMPORTANTE: Los kernels tipo Sobel estiman gradientes mediante diferencias ponderadas entre regiones vecinas; si no hay variación en la dirección analizada, la respuesta es cero.

Kernels / Convolución

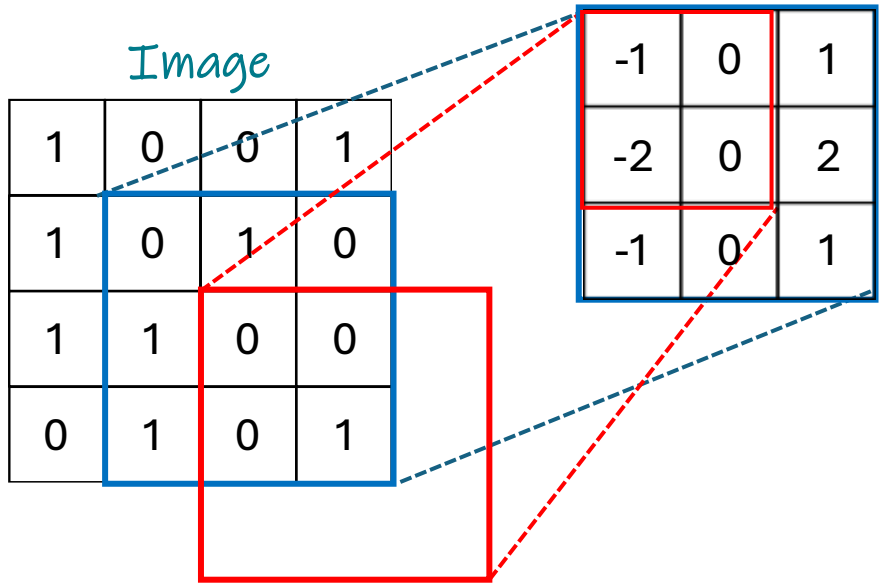
Los mapas de características más profundos codifican conceptos de alto nivel como "nariz de gato", mientras que los mapas de características de nivel inferior detectan bordes simples.

Por eso, los mapas de características más profundos contienen menos información sobre la imagen y más sobre su clase.



Kernels / Convolución

Zero Padding



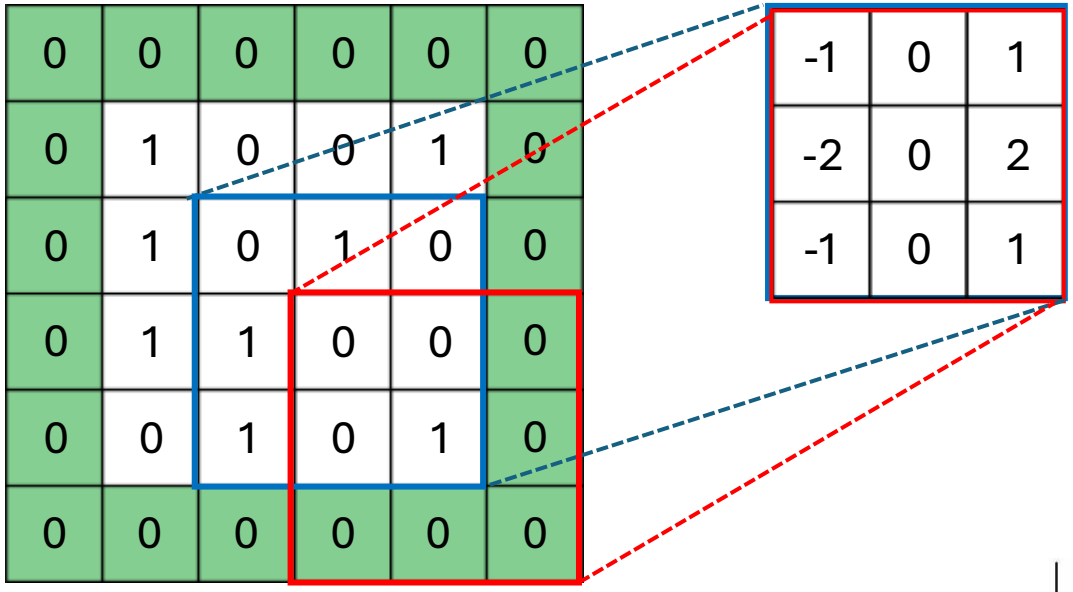
Zero padding es la técnica de agregar ceros alrededor de una imagen para permitir la convolución sin reducir su tamaño espacial.

- ✓ Pierdes resolución
- ✓ Pierdes información de bordes
- ✓ Red muy profunda colapsa dimensiones

$$O = \frac{I - k + 2p}{s} + 1$$

- I → Input dimension
- k → Kernel size
- p → Stride
- s → Padding
- o → Output size

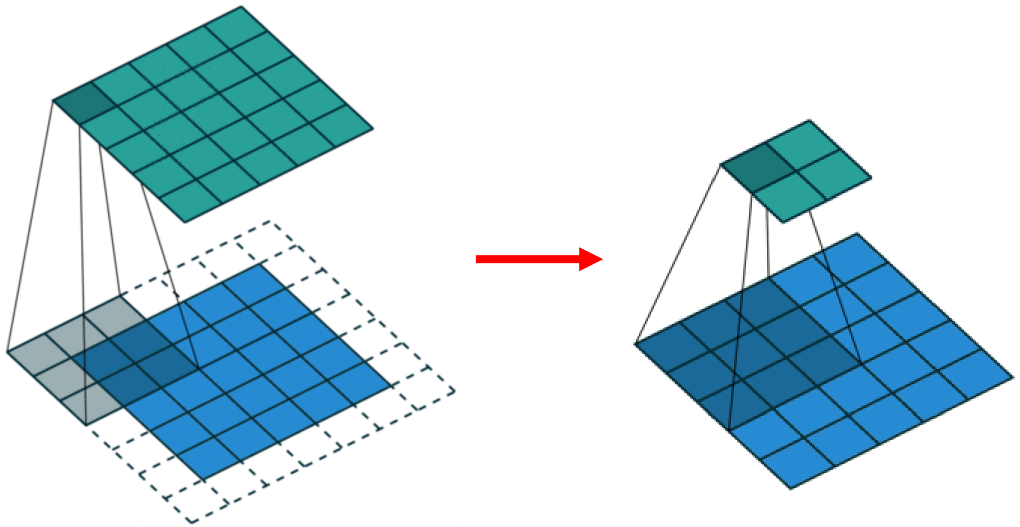
Zero padding



Kernels / Convolución

Striding: Se refiere al tamaño del paso con el que el filtro convolucional se desliza sobre los datos de entrada durante la operación de convolución.

$$H_{out} = \frac{H_{in}+2p-K}{s} + 1 \qquad W_{out} = \frac{W_{in}+2p-K}{s} + 1$$



Padding — Striding

Zero padding

0	0	0	0	0	0
0	1	0	0	1	0
0	1	0	1	0	0
0	1	1	0	0	0
0	0	1	0	1	0
0	0	0	0	0	0

Zero padding

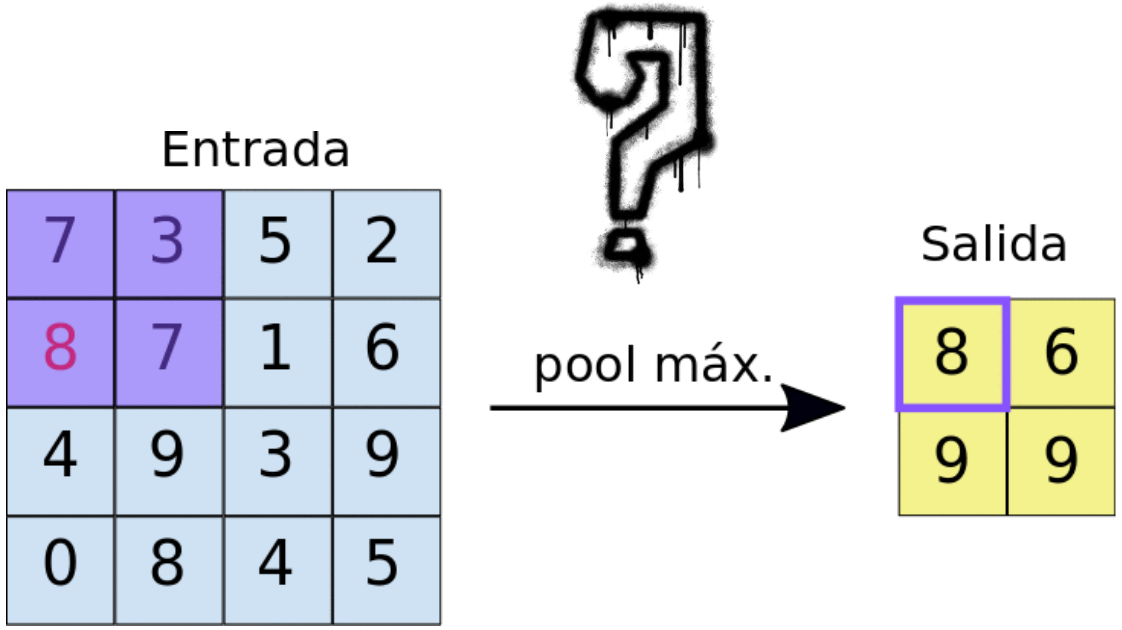
0	0	0	0	0	0
0	1	0	0	1	0
0	1	0	1	0	0
0	1	1	0	0	0
0	0	1	0	1	0
0	0	0	0	0	0

Zero padding

0	0	0	0	0	0
0	1	0	0	1	0
0	1	0	1	0	0
0	1	1	0	0	0
0	0	1	0	1	0
0	0	0	0	0	0

Kernels / Convolución

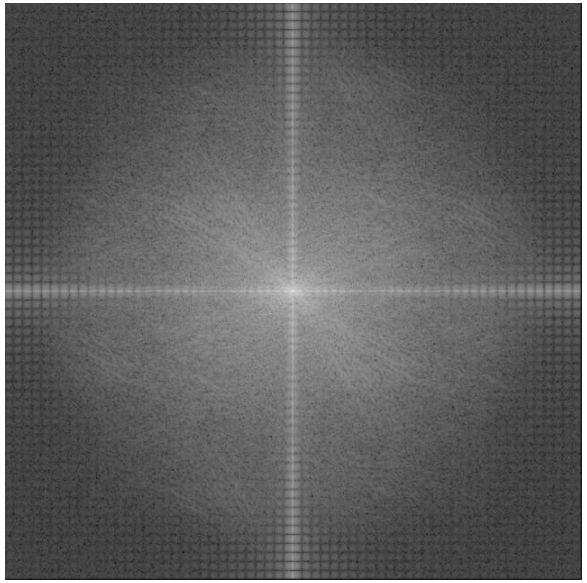
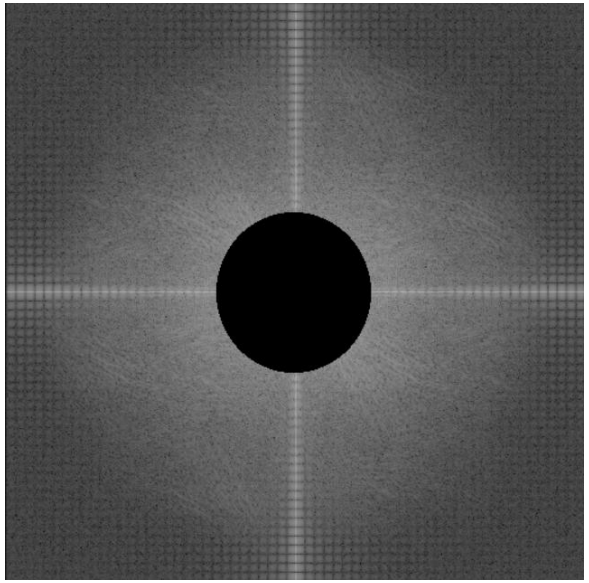
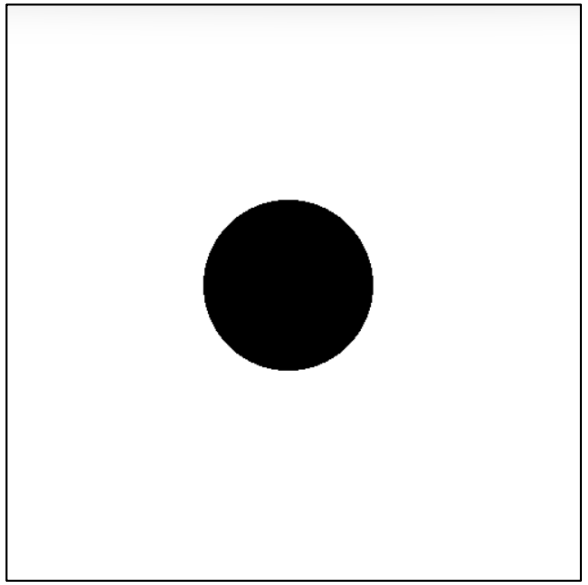
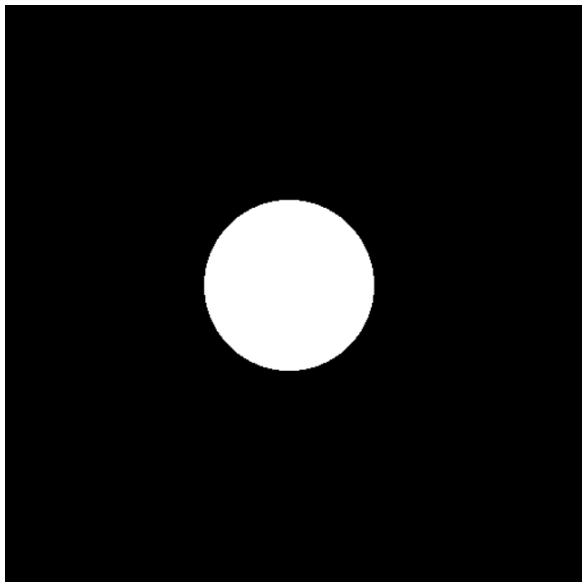
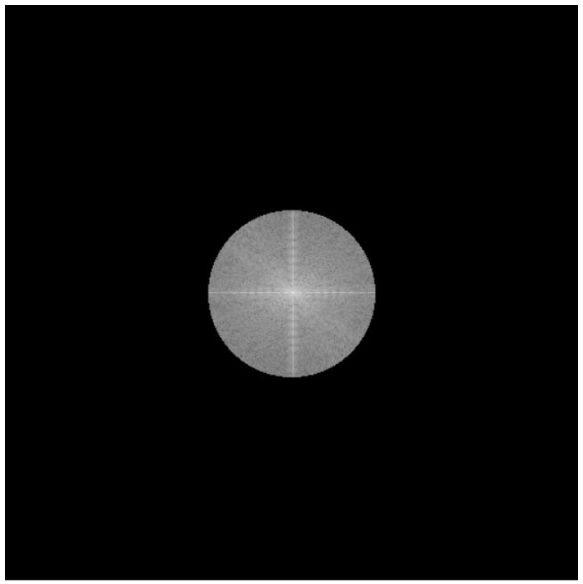
Pooling: Pooling en deep learning es una operación que **reduce el tamaño espacial** de los feature maps, resumiendo la información local..



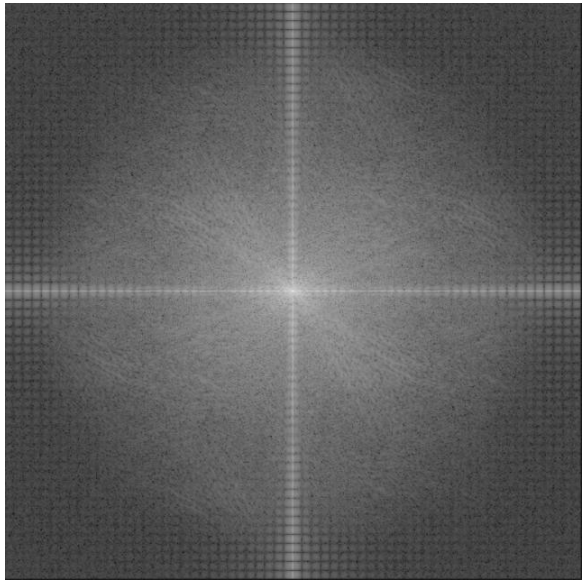
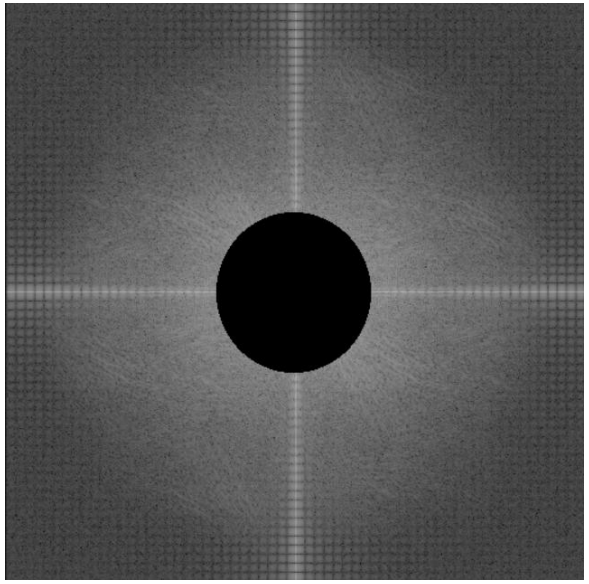
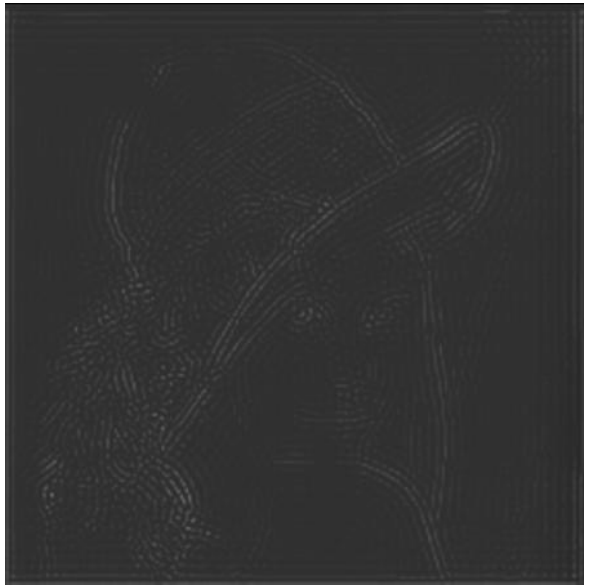
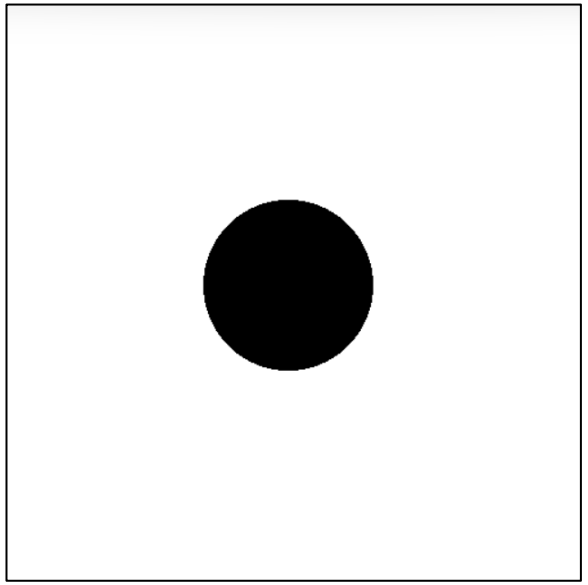
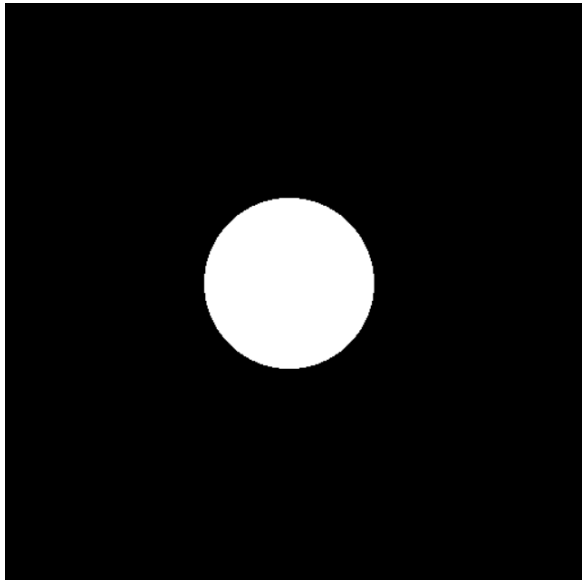
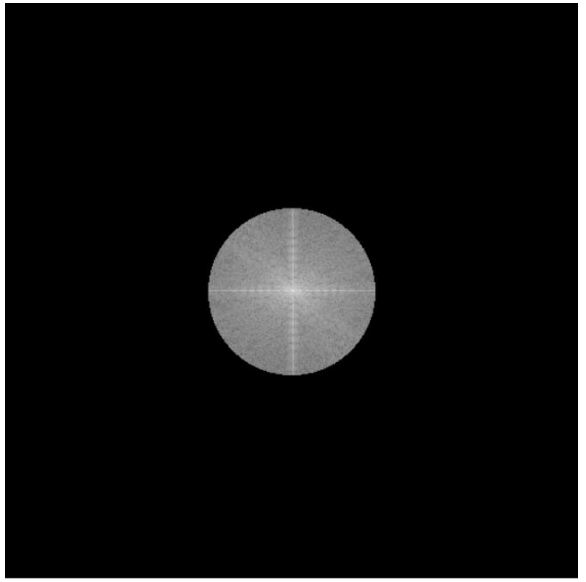
Max Pooling → Max pooling selects the maximum element from the region of the feature map covered by the filter. Thus, the output after max-pooling layer would be a feature map containing the most prominent features of the previous feature map.

Average Pooling → Average pooling computes the average of the elements present in the region of feature map covered by the filter. Average pooling gives the average of features present in a patch.

Kernels / Convolución

Kernels / Convolución

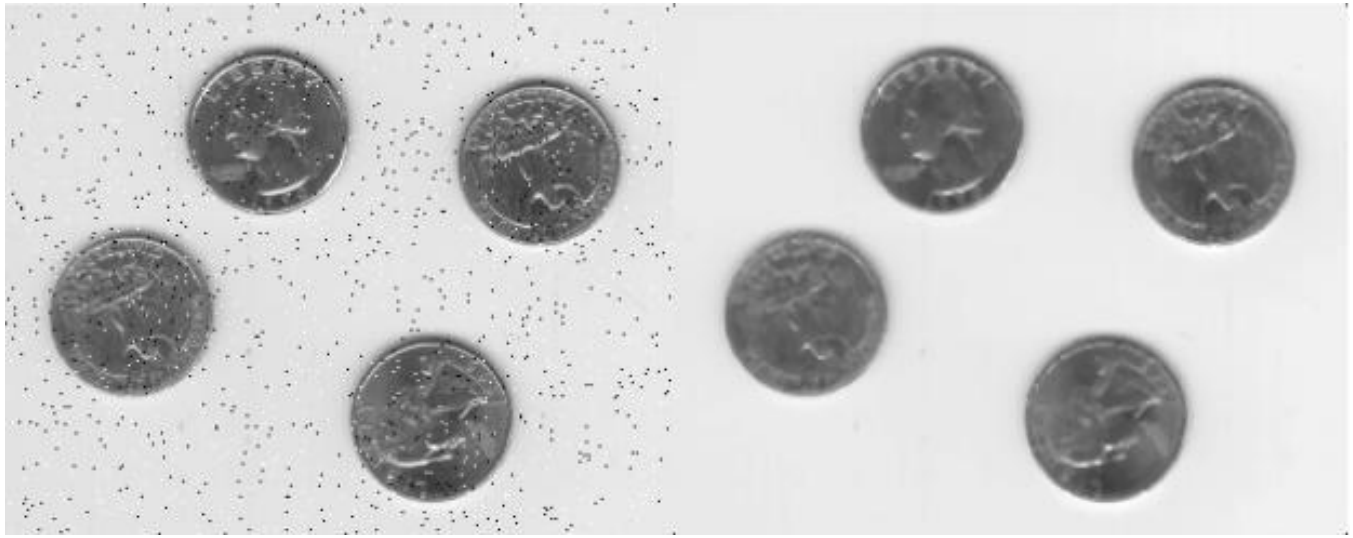

Kernels / Convolución

Las redes profundas pueden:

✓ Atenuar ruido

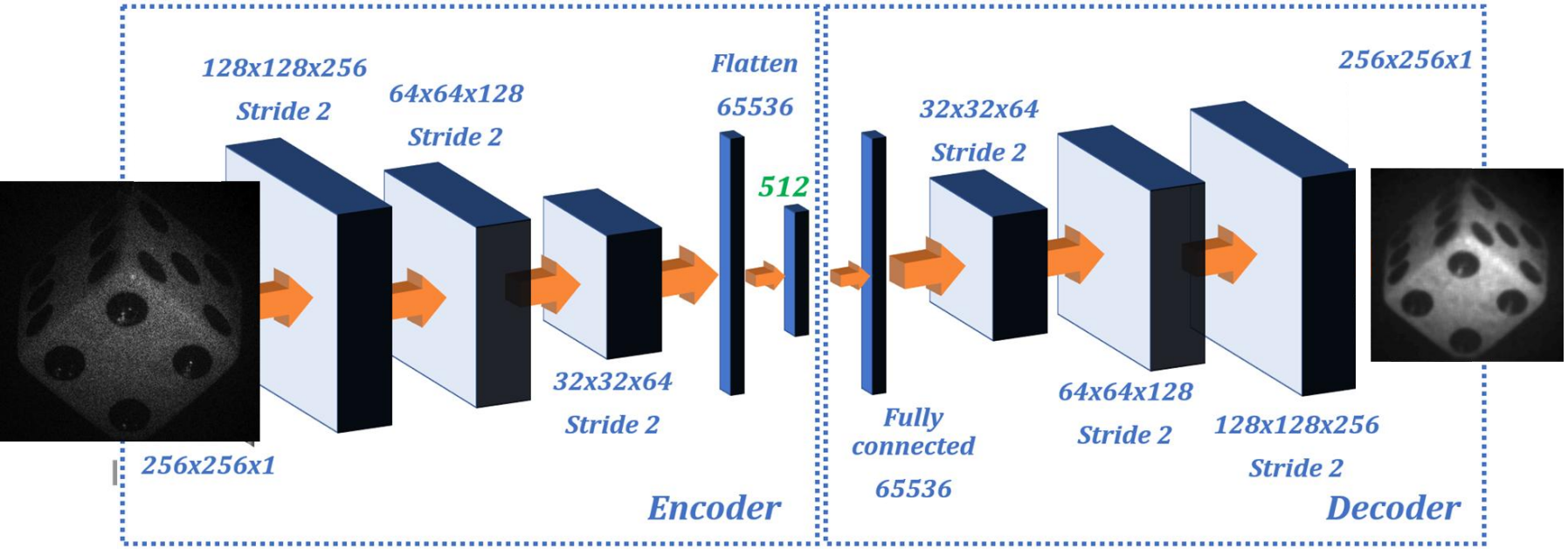
✓ Preservar bordes

✓ Reconstruir detalle



Noisy Image – Salt and pepper

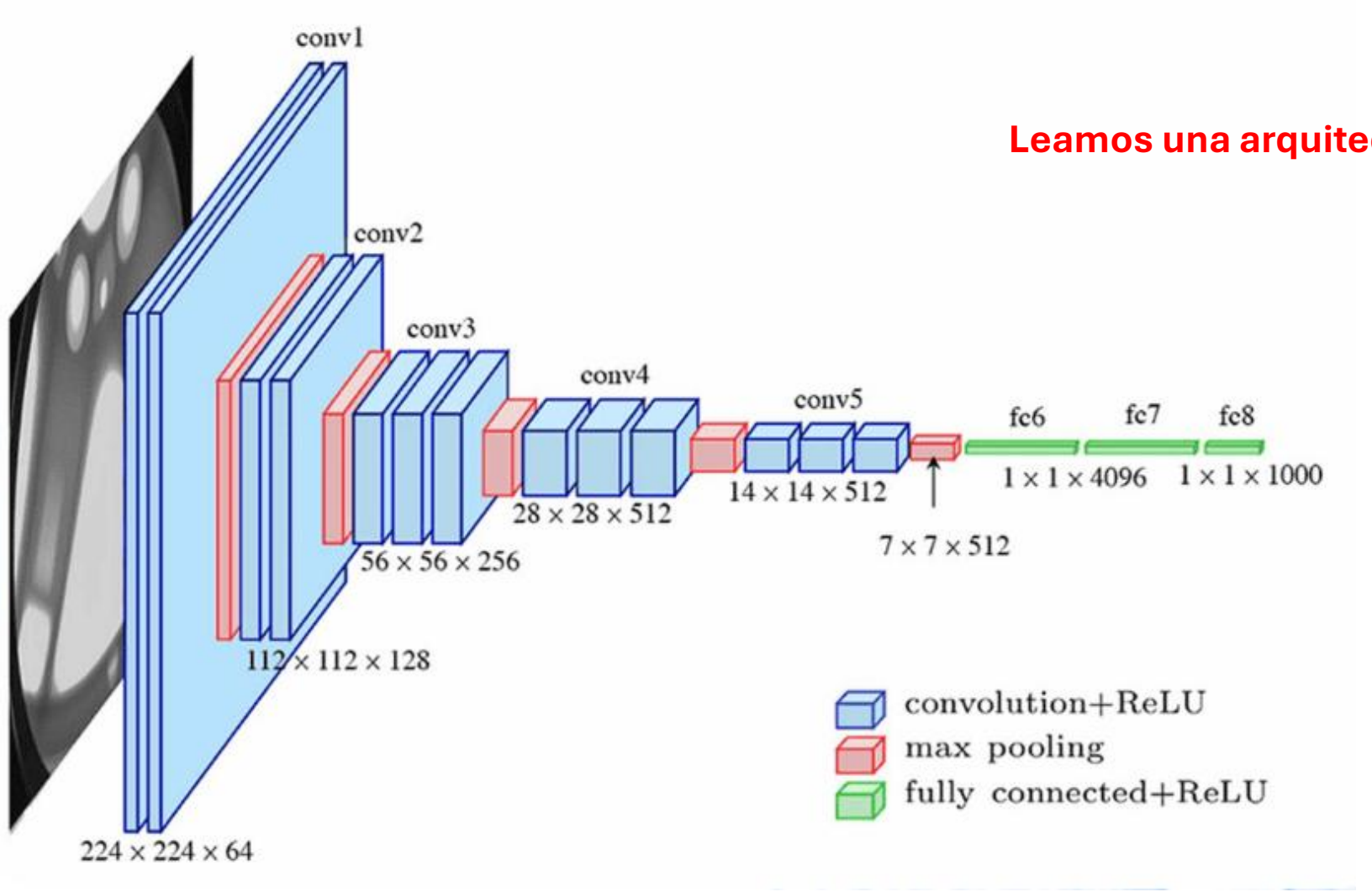
Reconstructed Image



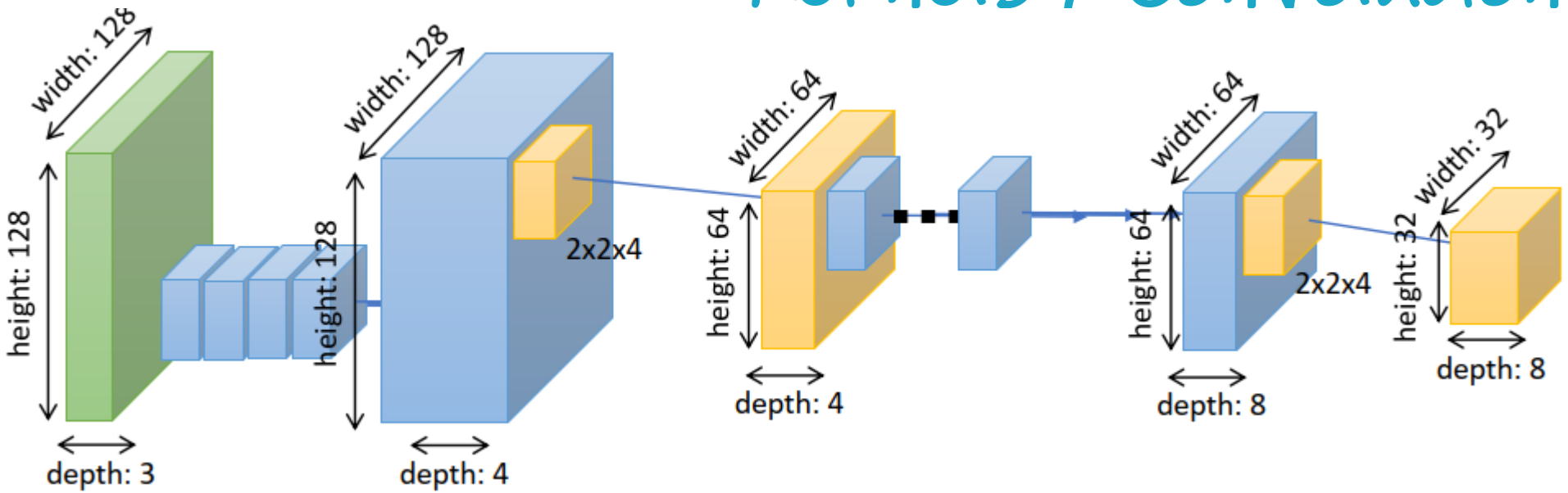
UNIVERSIDAD EAFIT

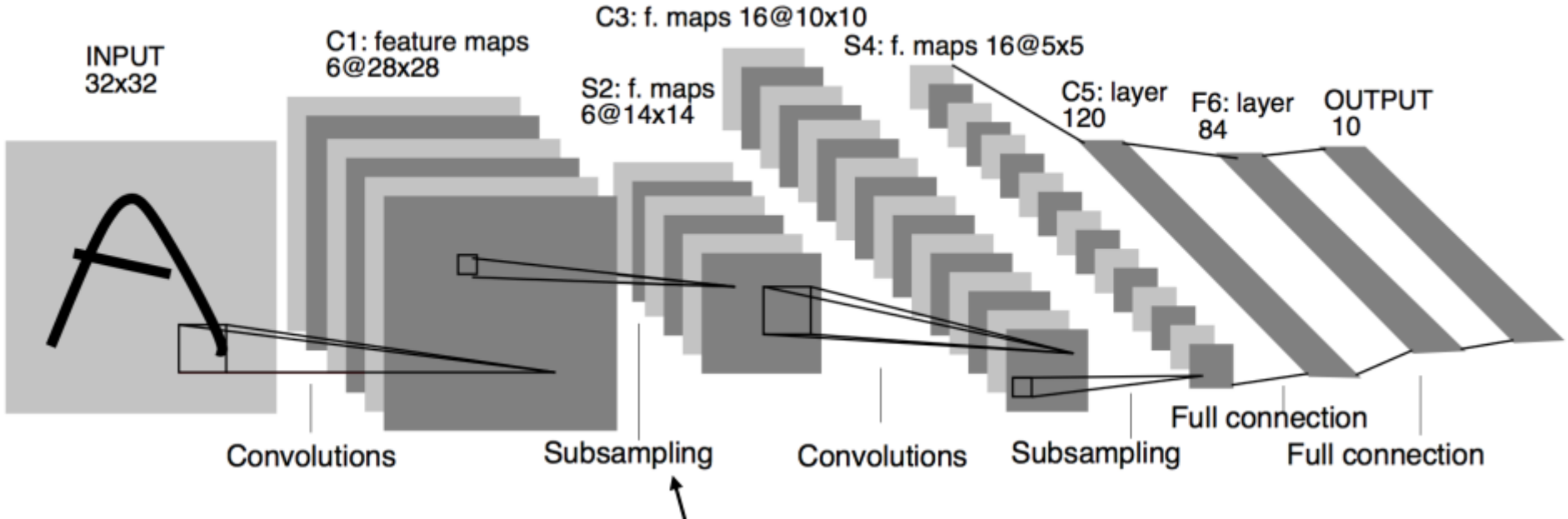
Kernels / Convolución

Leamos una arquitectura típica.



Kernels / Convolución

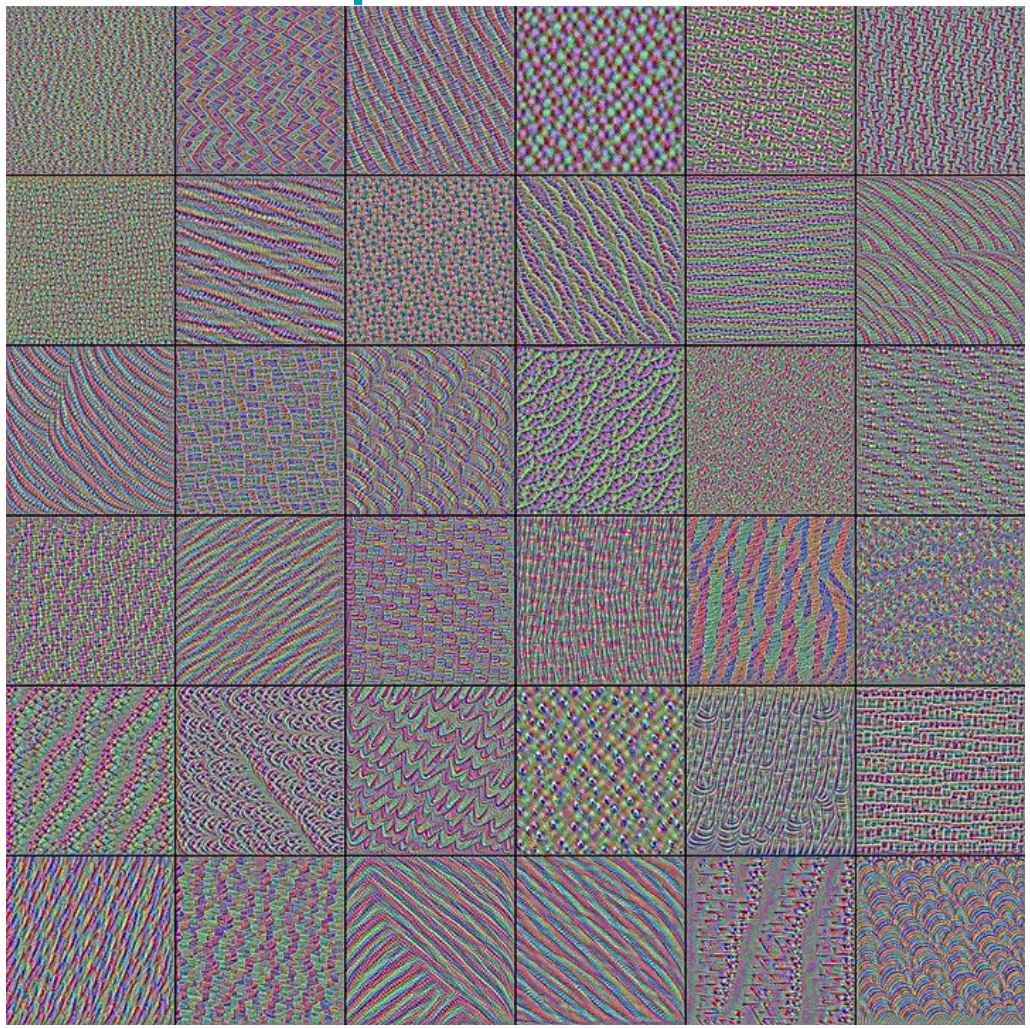




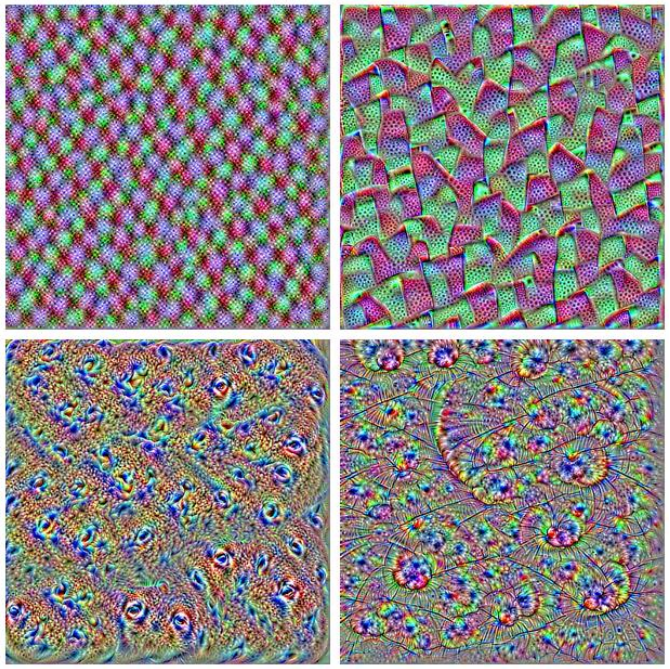
UNIVERSIDAD EAFIT

Redes Convolucionales

Feature Maps -> Extracted Characteristics



VGG16 convolution layers

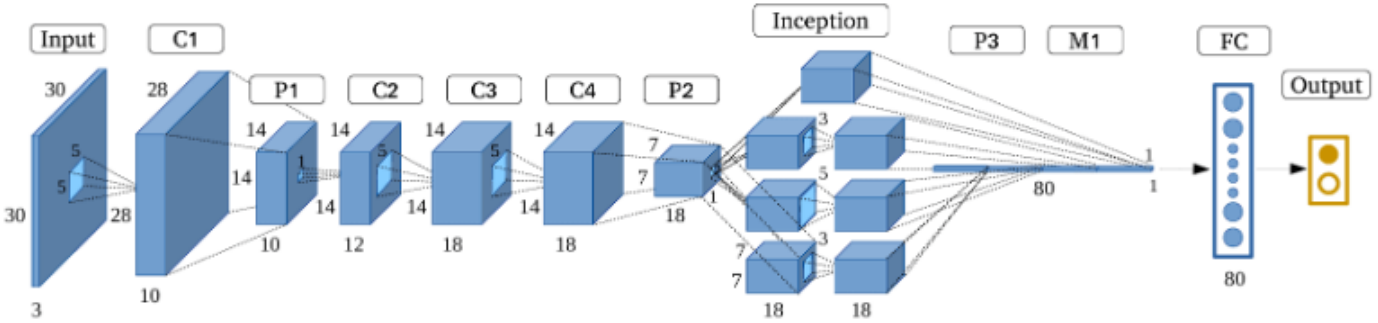


This is how the algorithm “sees”.

Redes Convolucionales

Link

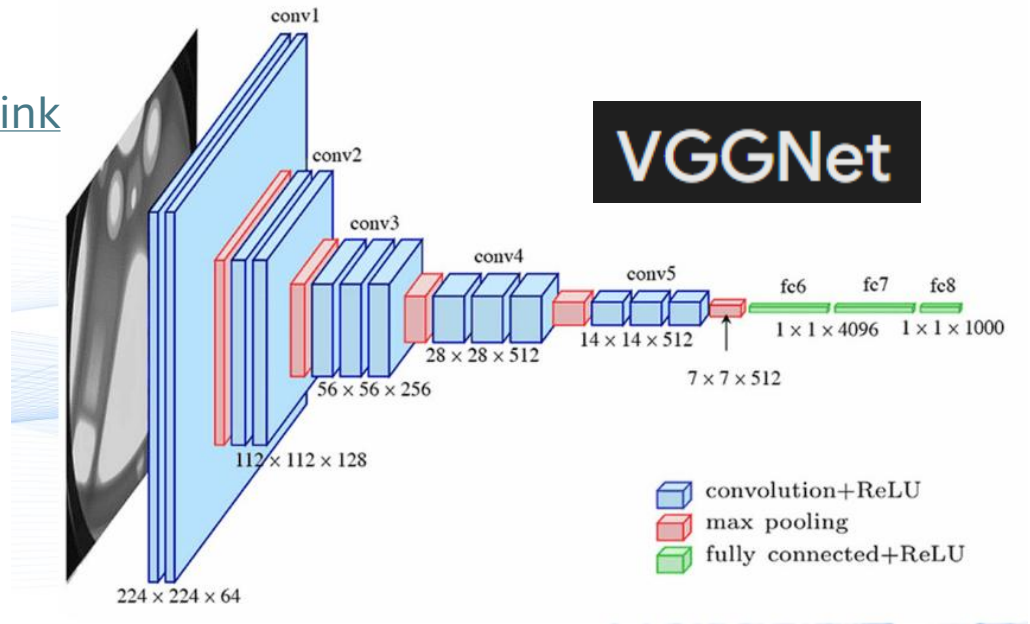
IMAGENET



Link

GoogLeNet

Link

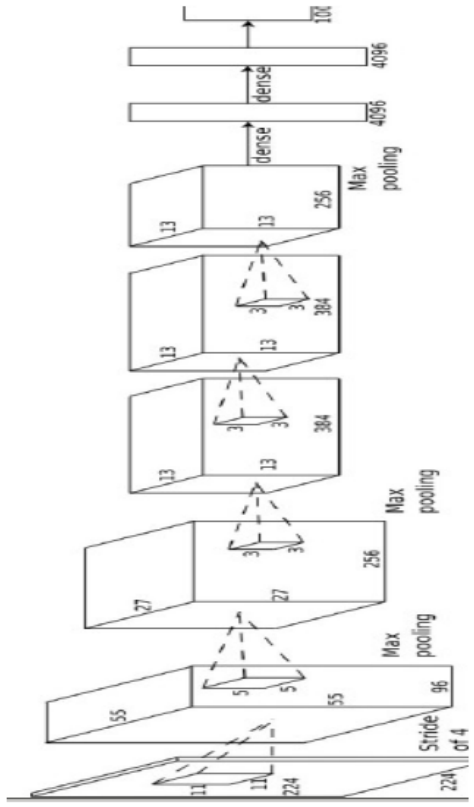


UNIVERSIDAD EAFIT

Redes Convolucionales

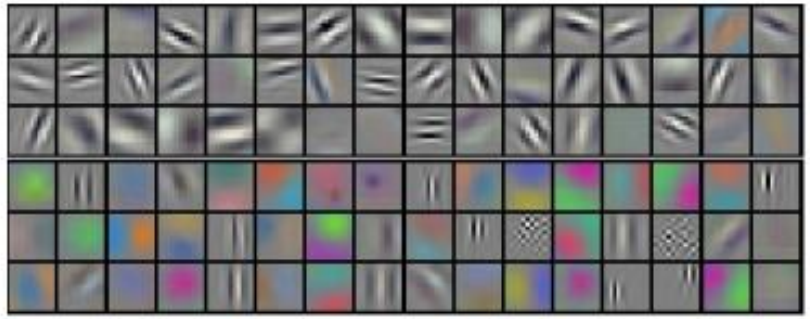
- 1000 categories, 1.5 Million labeled training samples
- Won the 2012 ImageNet LSVRC. 60 Million parameters, 832M MAC ops

4M	FULL CONNECT	4Mflop
16M	FULL 4096/ReLU	16M
37M	FULL 4096/ReLU	37M
	MAX POOLING	
442K	CONV 3x3/ReLU 256fm	74M
1.3M	CONV 3x3ReLU 384fm	224M
884K	CONV 3x3/ReLU 384fm	149M
	MAX POOLING 2x2sub	
	LOCAL CONTRAST NORM	
307K	CONV 11x11/ReLU 256fm	223M
	MAX POOL 2x2sub	
	LOCAL CONTRAST NORM	
35K	CONV 11x11/ReLU 96fm	105M



(Krizhevsky et al., 12')

AlexNet



96 convolutional filters on the first layer (filters are of size 11x11x3, applied across input images of size 224x224x3)

(Krizhevsky et al., 12')