

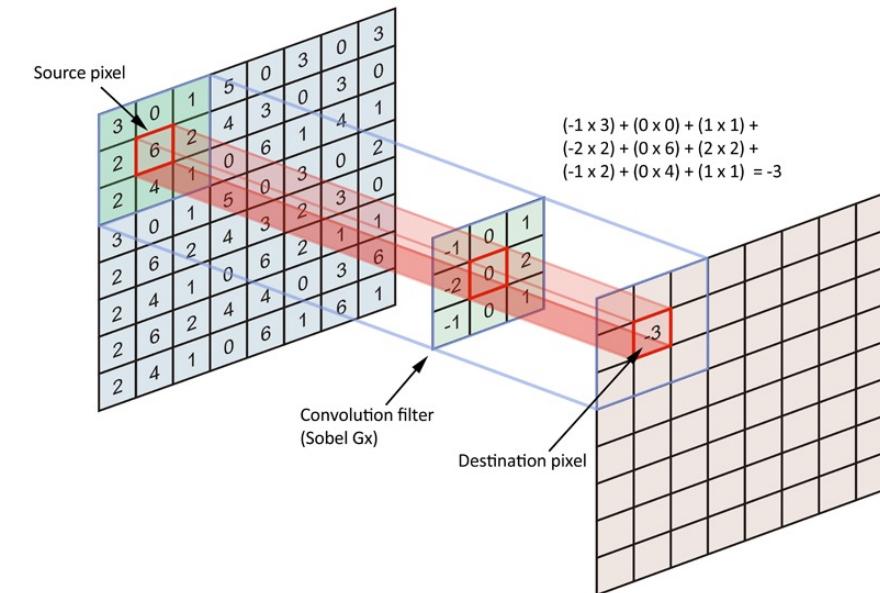
Clase 3

Redes Neuronales Convolucionales

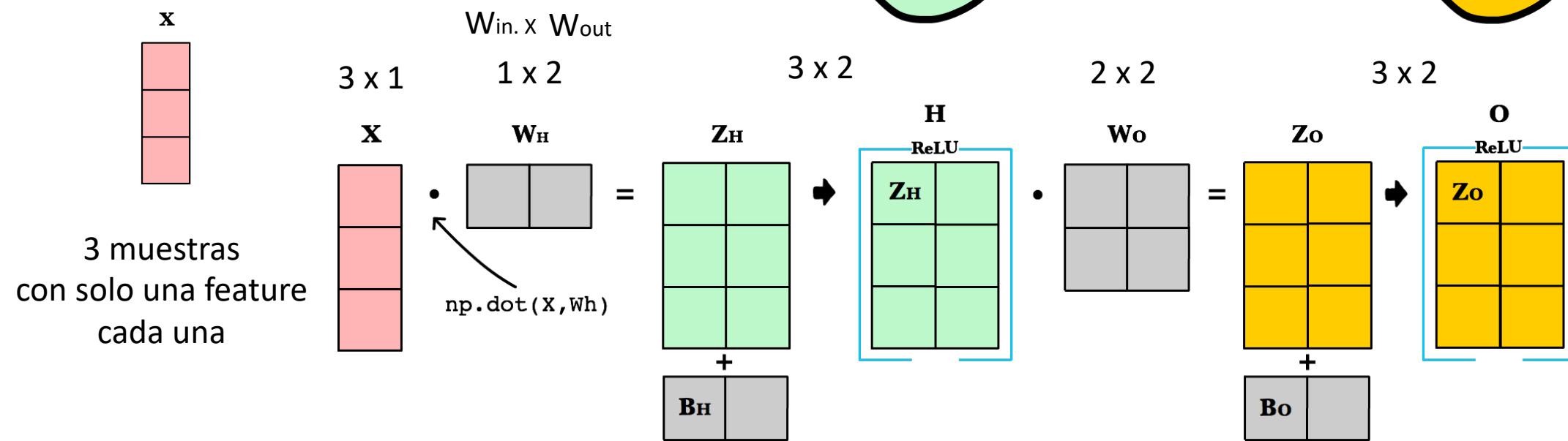
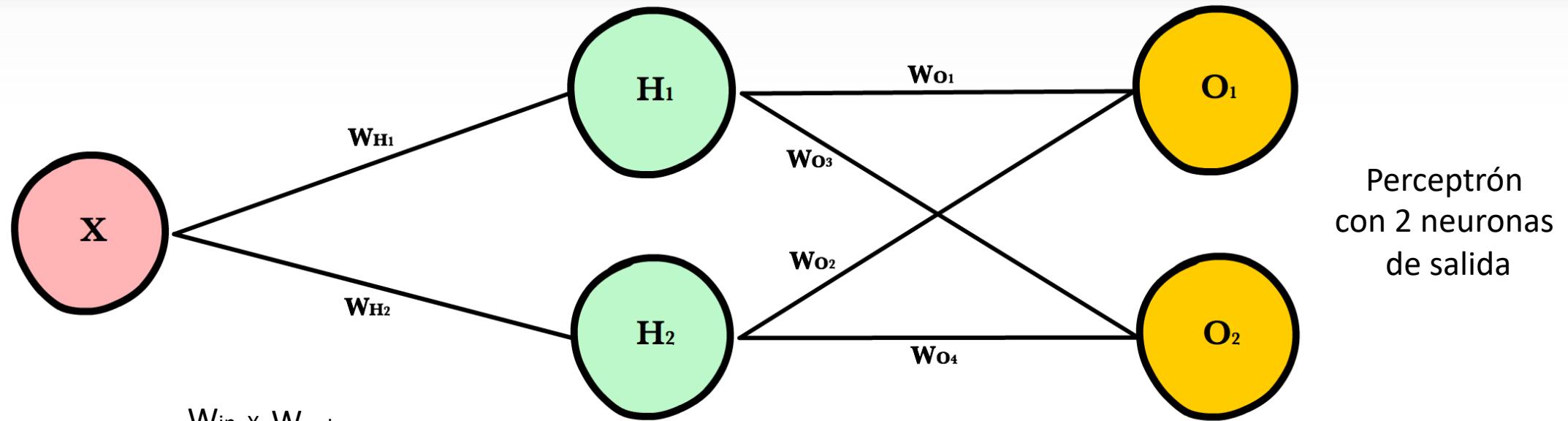
Enzo Ferrante

 eferrante@sinc.unl.edu.ar

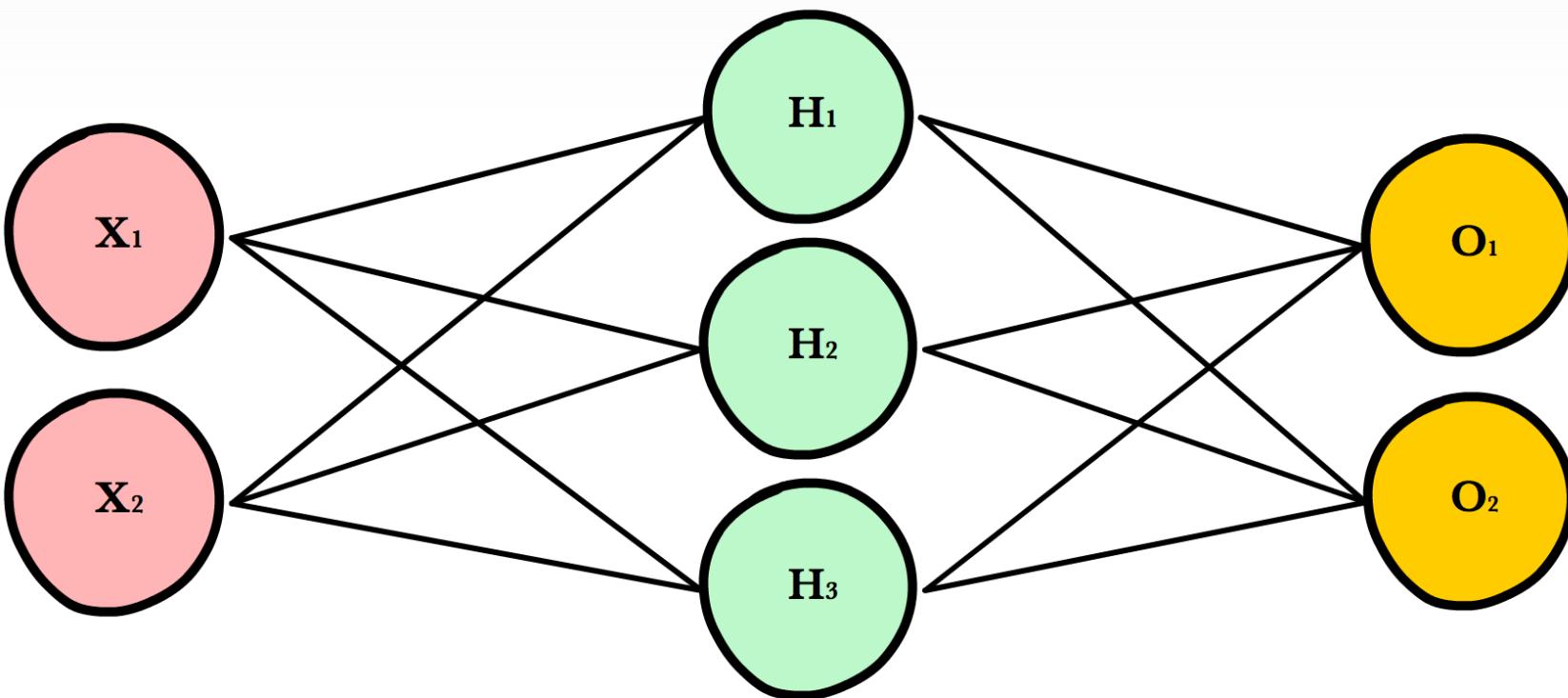
 @enzoferante



Repasso de percetrón



Repaso de percetrón



3 muestras
con 2 features
de entrada cada una

X_1	X_2
1	1
1	0
0	1
0	0

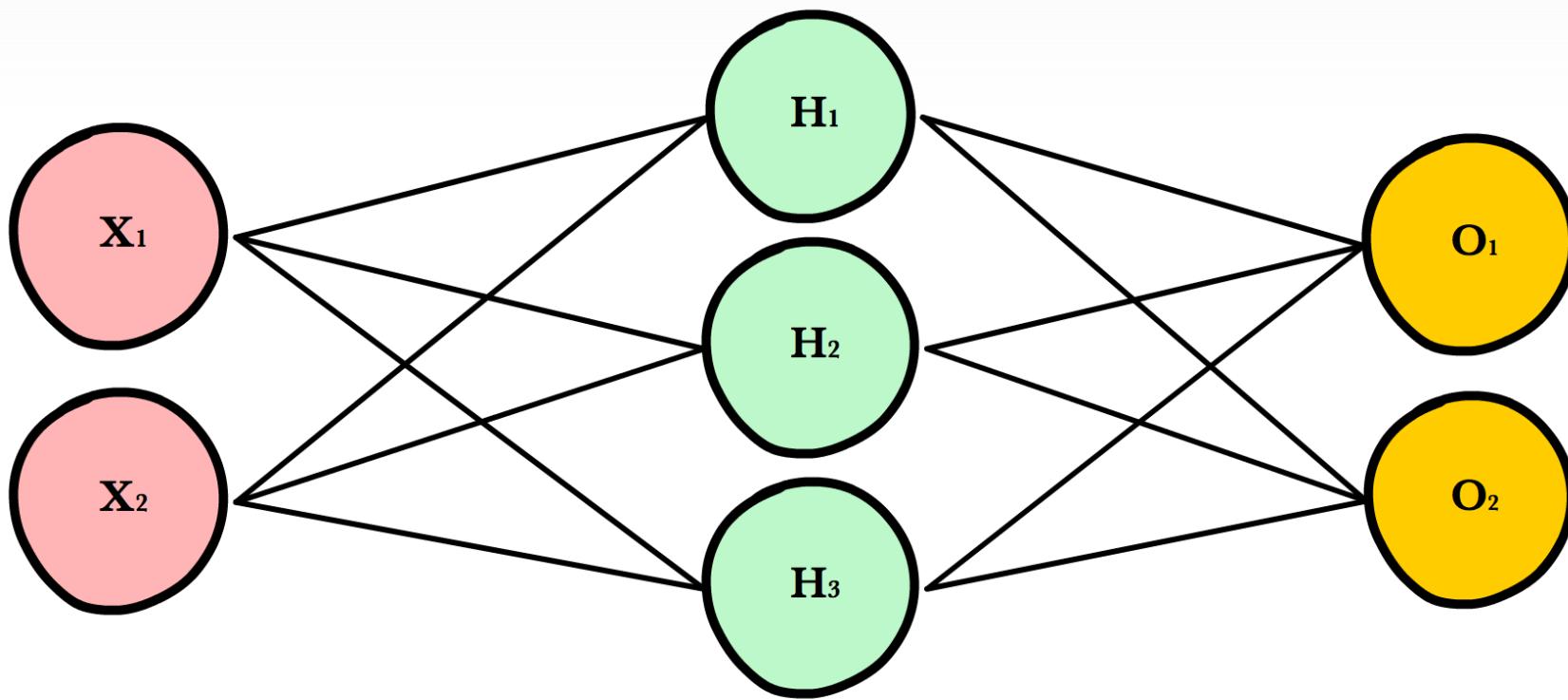
W_H
1 1 1
1 1 1

H_1	H_2	H_3
1 1 1	1 1 1	1 1 1
1 1 1	1 1 1	1 1 1
1 1 1	1 1 1	1 1 1
1 1 1	1 1 1	1 1 1

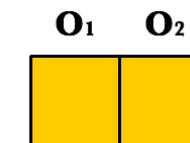
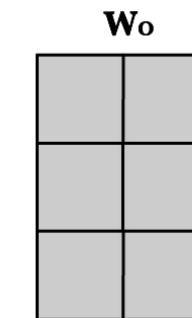
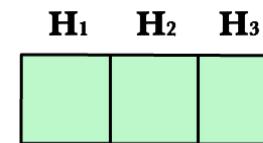
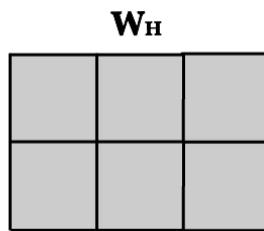
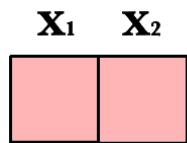
W_O
1 1 1
1 1 1

O_1	O_2
1 1 1	1 1 1
1 1 1	1 1 1
1 1 1	1 1 1
1 1 1	1 1 1

Repaso de percetrón

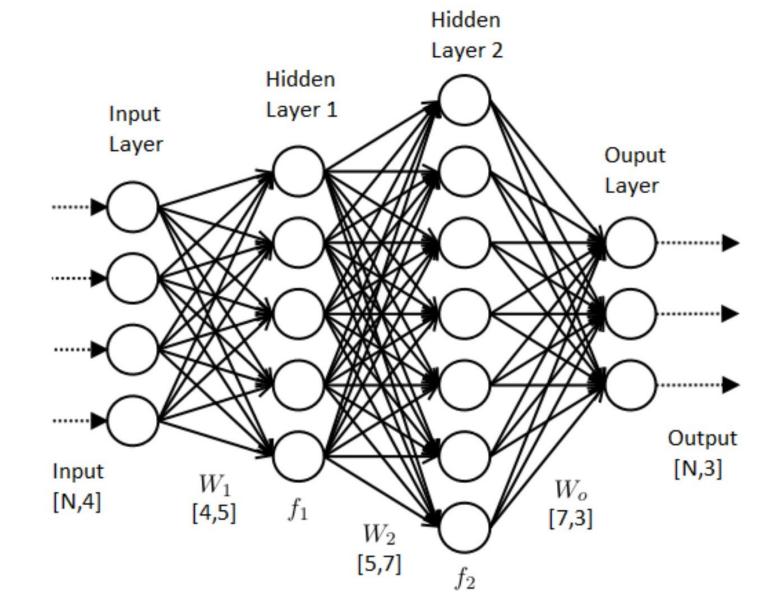
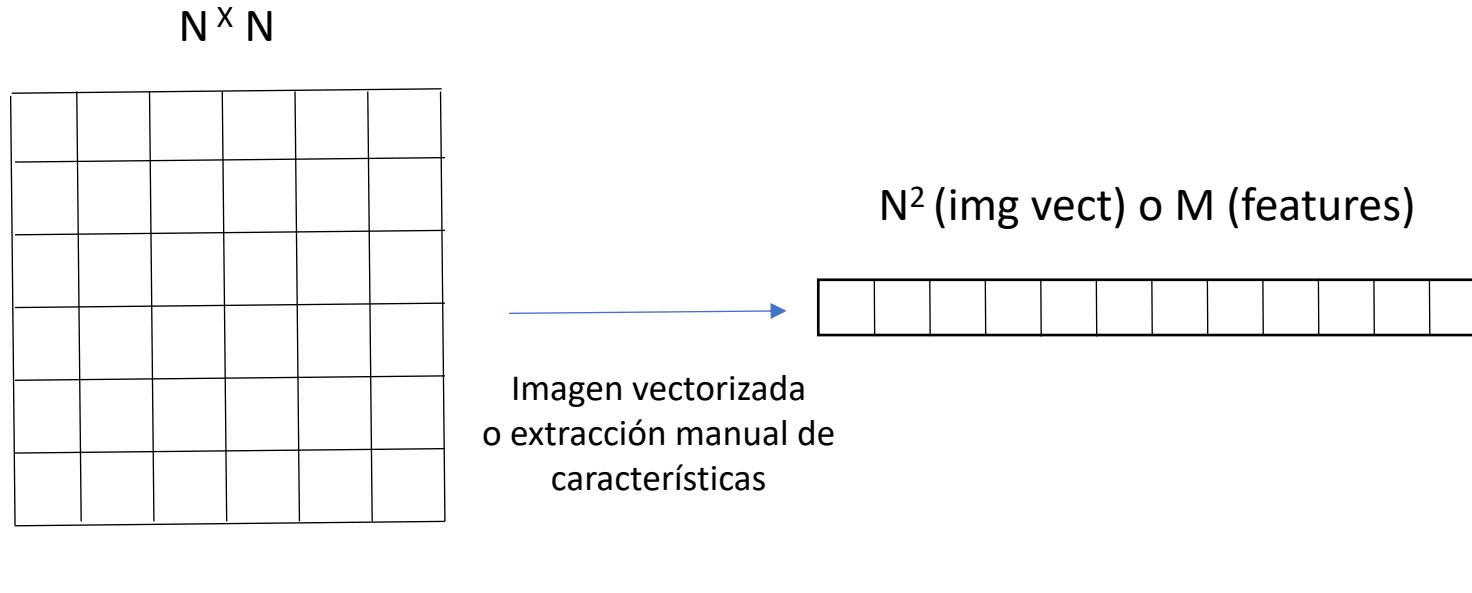


1 muestras
con 2 features
de entrada



El tamaño de las matrices
 W es independiente
de la cantidad de
elementos en el batch!

Cómo procesamos una imagen con un perceptrón?



Capas totalmente conectadas
(Fully connected layers)

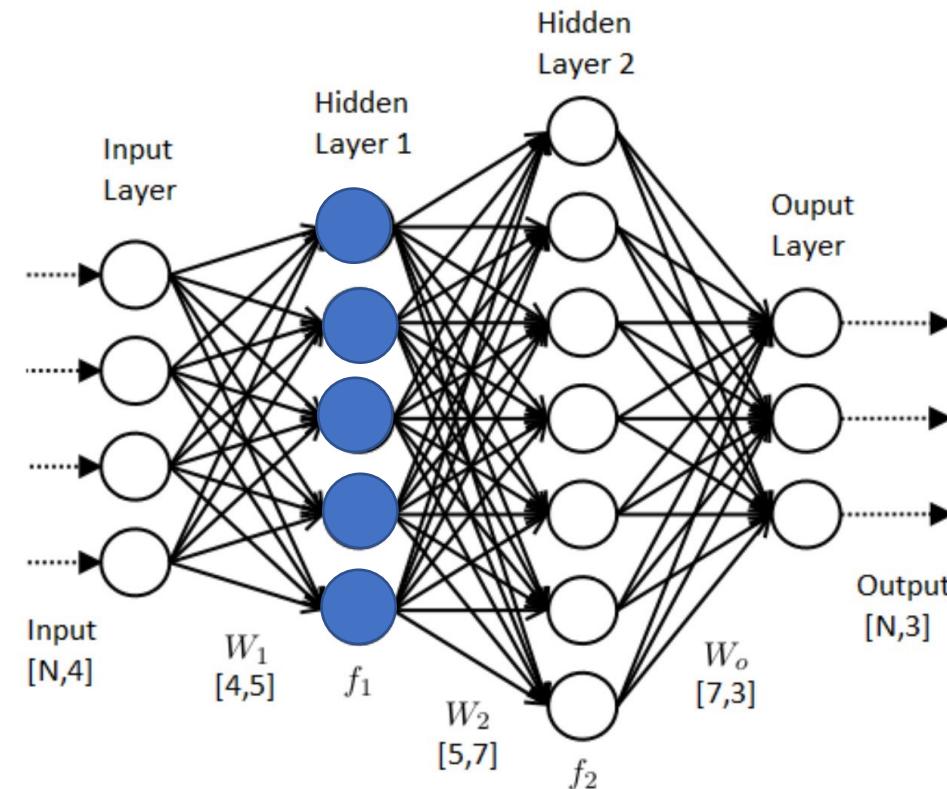
Cómo procesamos una imagen con un perceptrón?

~ 1 Millón de conexiones

x

1024 neuronas ocultas

~ 1024 Millones de parámetros



Cómo procesamos una imagen con un perceptrón?

Desventajas

- Se pierde la estructura original de los datos
- Para grandes imágenes, la cantidad de neuronas (y sus consecuentes conexiones) crece exponencialmente
- No tenemos una noción clara de análisis multi-escala/multi-resolución (algo que resulta útil en general en análisis de imágenes)

Redes neuronales inspiradas en el sistema visual

Arquitectura de la corteza visual del gato

(Hubel & Wiesel, 1962)

106

J. Physiol. (1962), 160, pp. 106-154
With 2 plates and 20 text-figures
Printed in Great Britain

RECEPTIVE FIELDS, BINOCULAR INTERACTION AND FUNCTIONAL ARCHITECTURE IN THE CAT'S VISUAL CORTEX

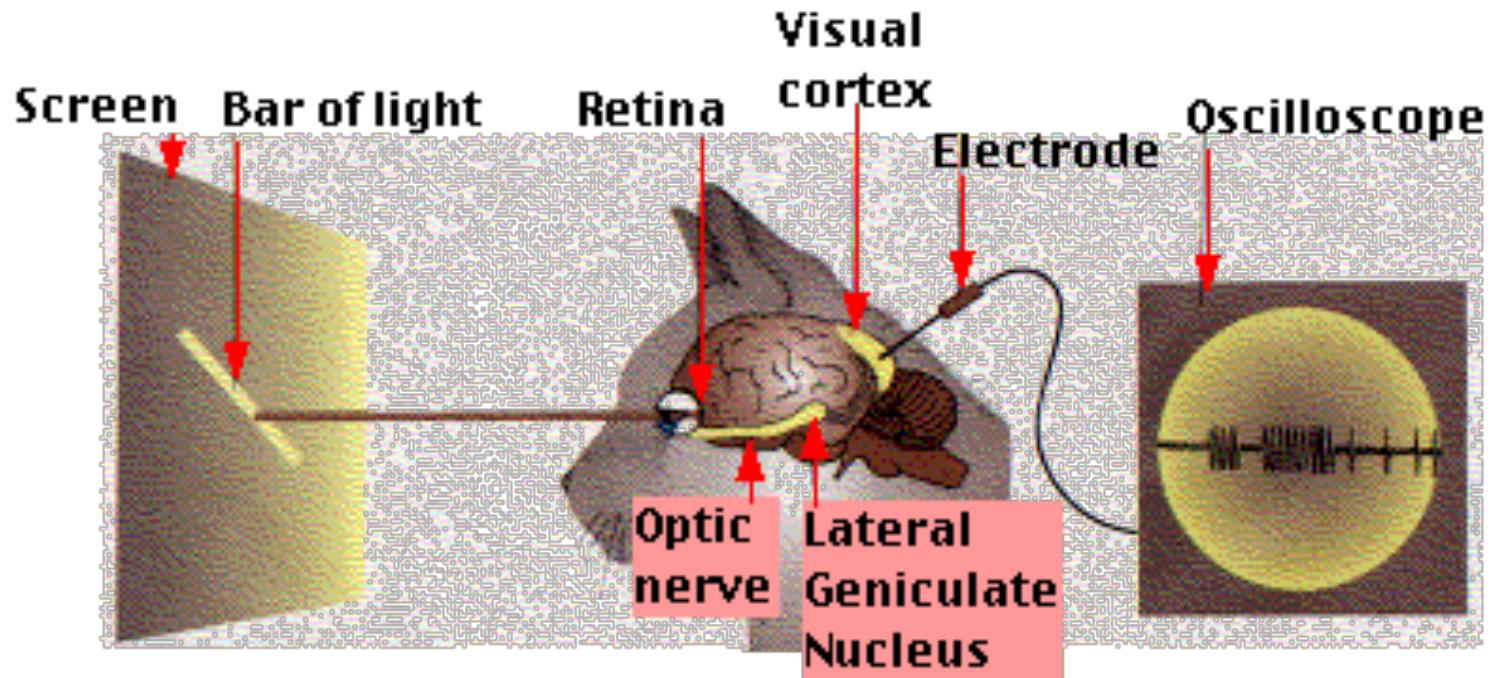
BY D. H. HUBEL AND T. N. WIESEL

*From the Neurophysiology Laboratory, Department of Pharmacology
Harvard Medical School, Boston, Massachusetts, U.S.A.*

(Received 31 July 1961)

What chiefly distinguishes cerebral cortex from other parts of the central nervous system is the great diversity of its cell types and interconnexions. It would be astonishing if such a structure did not profoundly modify the response patterns of fibres coming into it. In the cat's visual cortex, the receptive field arrangements of single cells suggest that there is a complexity far exceeding anything yet seen at lower

Redes neuronales inspiradas en el sistema visual



Redes neuronales inspiradas en el sistema visual

- Midieron la respuesta eléctrica en el cerebro de un gato estimulándolo con patrones simples en una pantalla.
- Encontraron que las **neuronas de la corteza visual** temprana se organizan de **forma jerárquica**, donde las **primeras** reaccionan a **patrones simples como líneas**, y las **posteriores** capas responden a **patrones más complejos** combinando las activaciones que reciben.
- En el modelo propuesto, las neuronas en las capas superiores tienen un mayor **campo receptivo** y son **menos sensibles a la posición desde la cual proviene dicho estímulo**.

Neocognitrón: red neuronal artificial inspirada en (Hubel & Wiesel, 1962)

(Fukushima 1980)

Biol. Cybernetics 36, 193–202 (1980)

Biological
Cybernetics
© by Springer-Verlag 1980

Neocognitron: A Self-organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position

Kunihiko Fukushima

NHK Broadcasting Science Research Laboratories, Kinuta, Setagaya, Tokyo, Japan

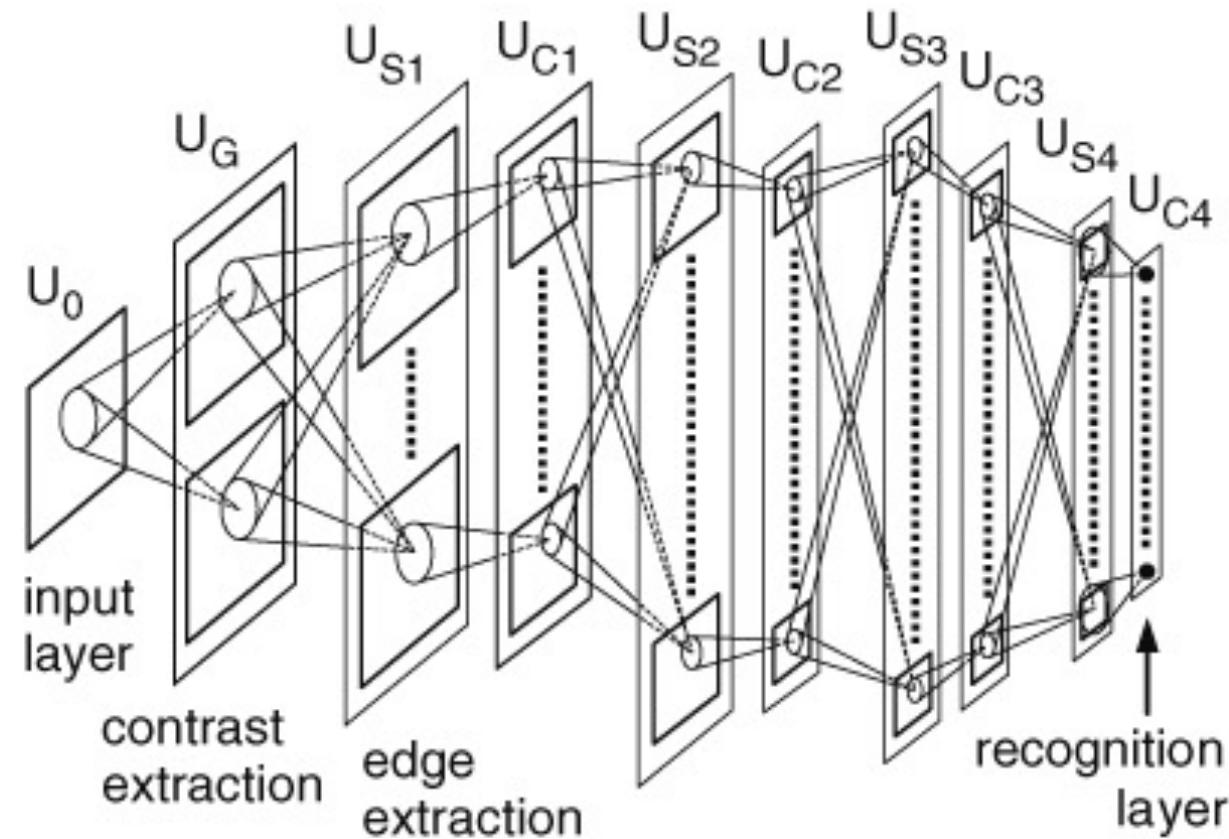
Abstract. A neural network model for a mechanism of visual pattern recognition is proposed in this paper. The network is self-organized by “learning without a teacher”, and acquires an ability to recognize stimulus patterns based on the geometrical similarity (Gestalt) of their shapes without affected by their positions. This network is given a nickname “neocognitron”. After completion of self-organization, the network has a structure similar to the hierarchy model of the visual nervous system proposed by Hubel and Wiesel. The network consists of an input layer (photoreceptor array) followed by a cascade connection of a number of modular structures, each of which is composed of two

reveal it only by conventional physiological experiments. So, we take a slightly different approach to this problem. If we could make a neural network model which has the same capability for pattern recognition as a human being, it would give us a powerful clue to the understanding of the neural mechanism in the brain. In this paper, we discuss how to synthesize a neural network model in order to endow it an ability of pattern recognition like a human being.

Several models were proposed with this intention (Rosenblatt, 1962; Kabrisky, 1966; Giebel, 1971; Fukushima, 1975). The response of most of these models, however, was severely affected by the shift in

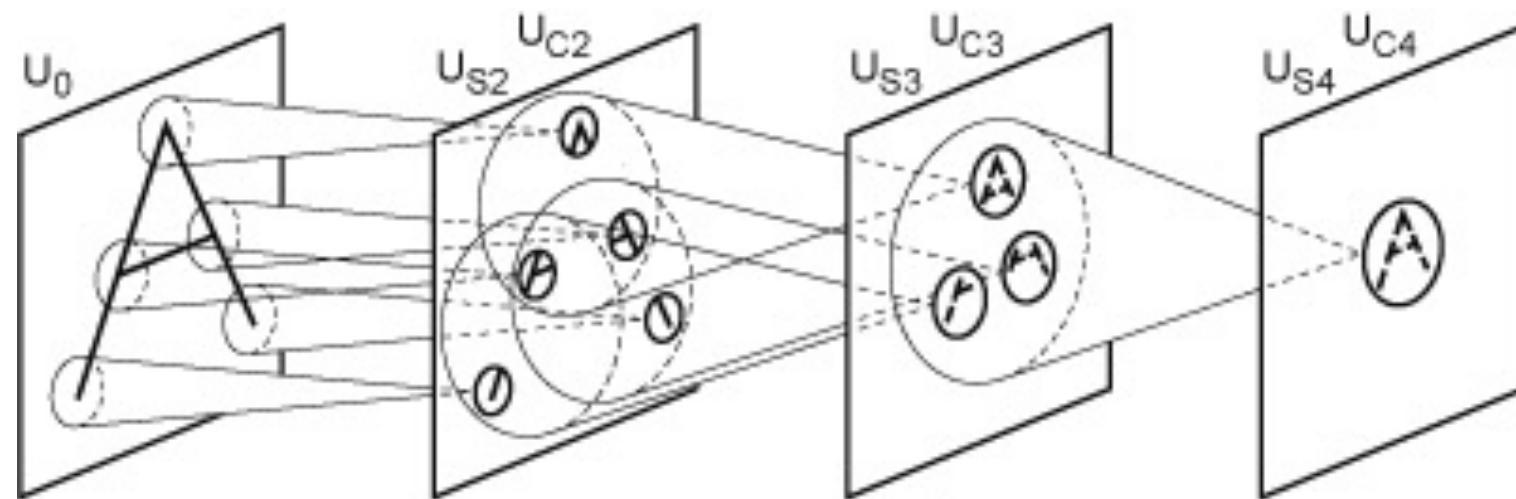
Neocognitrón: red neuronal artificial inspirada en (Hubel & Wiesel, 1962)

(Fukushima 1980)



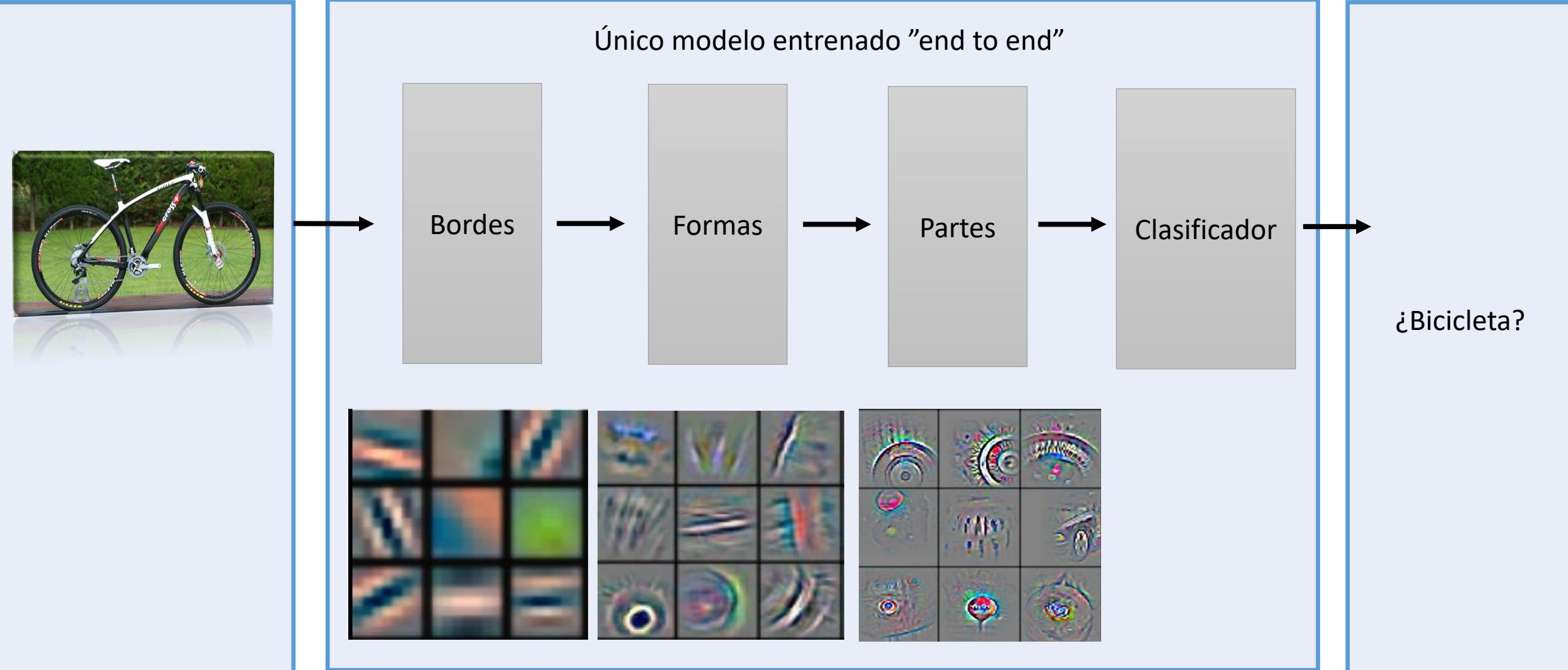
Neocognitrón: red neuronal artificial inspirada en (Hubel & Wiesel, 1962)

(Fukushima 1980)



Deep Learning

Multiples niveles de abstracción



Redes neuronales convolucionales

(Lecun 1998)

Gradient-Based Learning Applied to Document Recognition

YANN LECUN, MEMBER, IEEE, LÉON BOTTOU, YOSHUA BENGIO, AND PATRICK HAFFNER

Invited Paper

Multilayer neural networks trained with the back-propagation algorithm constitute the best example of a successful gradient-based learning technique. Given an appropriate network architecture, gradient-based learning algorithms can be used to synthesize a complex decision surface that can classify high-dimensional patterns, such as handwritten characters, with minimal preprocessing. This paper reviews various methods applied to handwritten character recognition and compares them on a standard handwritten digit recognition task. Convolutional neural networks, which are specifically designed to deal with the variability of two dimensional (2-D) shapes, are shown to outperform all other techniques.

Real-life document recognition systems are composed of multiple modules including field extraction, segmentation, recognition, and language modeling. A new learning paradigm, called graph transformer networks (GTN's), allows such multimodule systems to learn using gradient-based methods so as to

NN	Neural network.
OCR	Optical character recognition.
PCA	Principal component analysis.
RBF	Radial basis function.
RS-SVM	Reduced-set support vector method.
SDNN	Space displacement neural network.
SVM	Support vector method.
TDNN	Time delay neural network.
V-SVM	Virtual support vector method.

I. INTRODUCTION

Over the last several years, machine learning techniques, particularly when applied to NN's, have played an increasingly important role in the design of pattern recognition systems. In fact, it could be argued that the availability of cheap computing power has been a crucial factor in the success of learning applications such as

Redes Neuronales Convolucionales

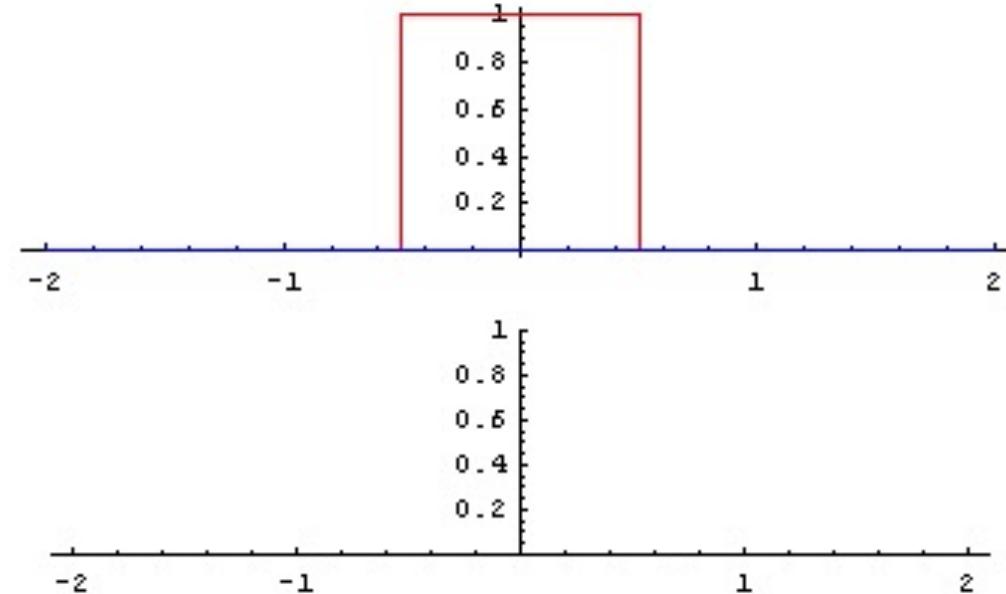
"Convolutional networks are simply neural networks that use convolution in place of general matrix multiplication in at least one of their layers"

Deep Learning, MIT Press, Goodfellow et al, 2016

Convolución

Convolución 1D

Representa la integral del producto de una función x con una versión trasladada y reflejada de otra w



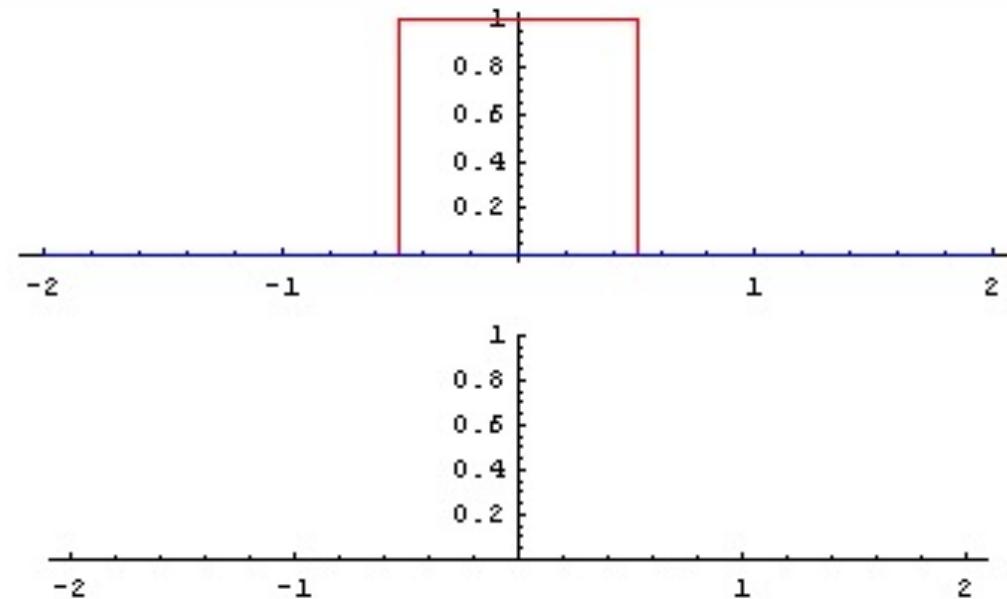
$$s(t) = \int x(a)w(t-a)da$$

Señal convolucionada

Señal de
entrada

Filtro

Convolución 1D



$$s(t) = (x * w)(t)$$

Convolución 1D

Discretización de la señal

$$s(t) = \int x(a)w(t - a)da$$

Señal Continua

$$s(t) = (x * w)(t) = \sum_{a=-\infty}^{\infty} x(a)w(t - a)$$

Señal Discreta

Convolución 2D

$$S(i, j) = (I * K)(i, j) = \sum_m \sum_n I(m, n)K(i - m, j - n)$$



La convolución es comutativa

$$S(i, j) = (K * I)(i, j) = \sum_m \sum_n I(i - m, j - n)K(m, n)$$

Propiedad útil ya que el rango de valores válidos de m, n de una imagen es mayor en general que el del kernel.

Correlación cruzada

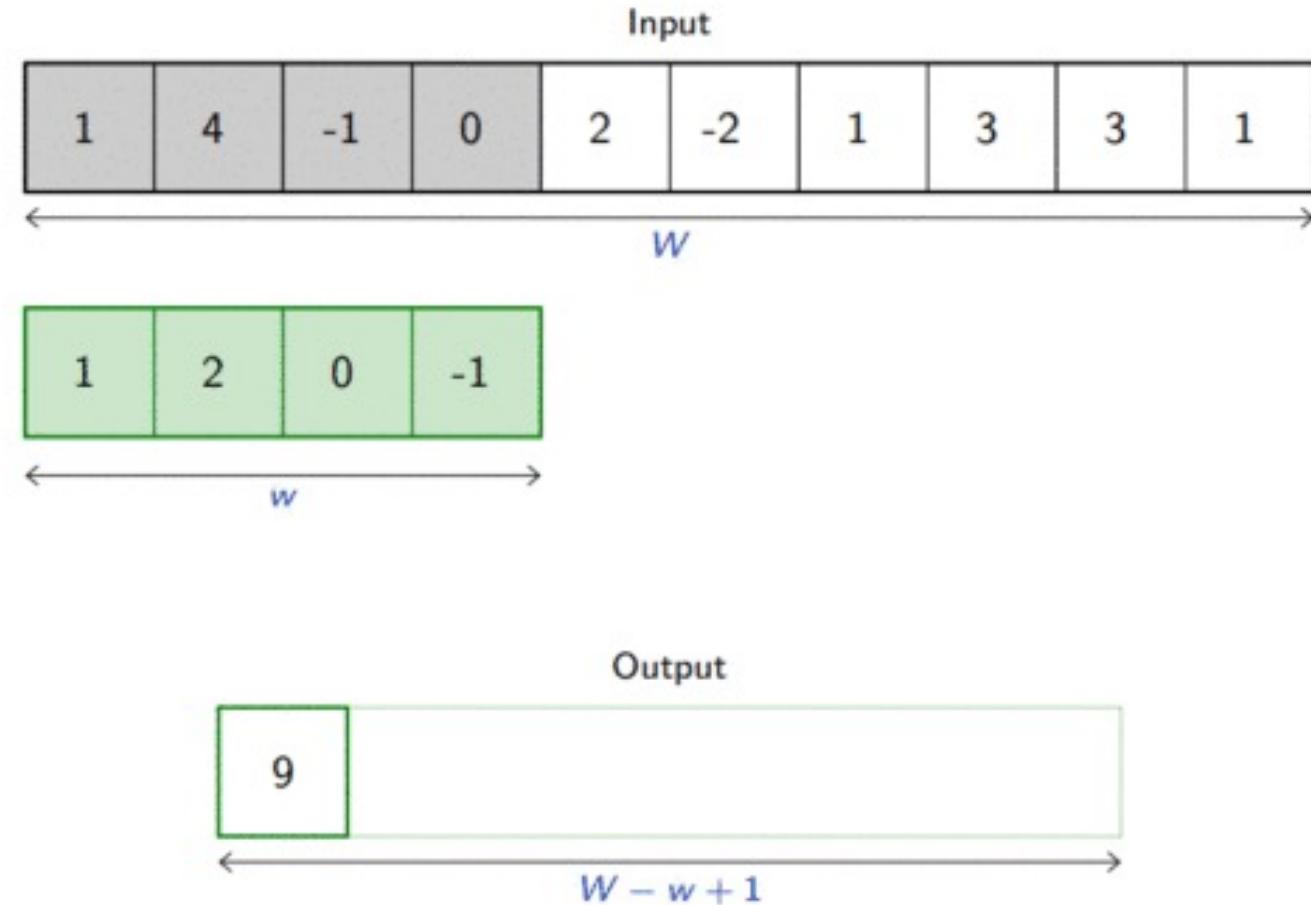
$$S(i, j) = (K * I)(i, j) = \sum_m \sum_n I(i - m, j - n) K(m, n)$$



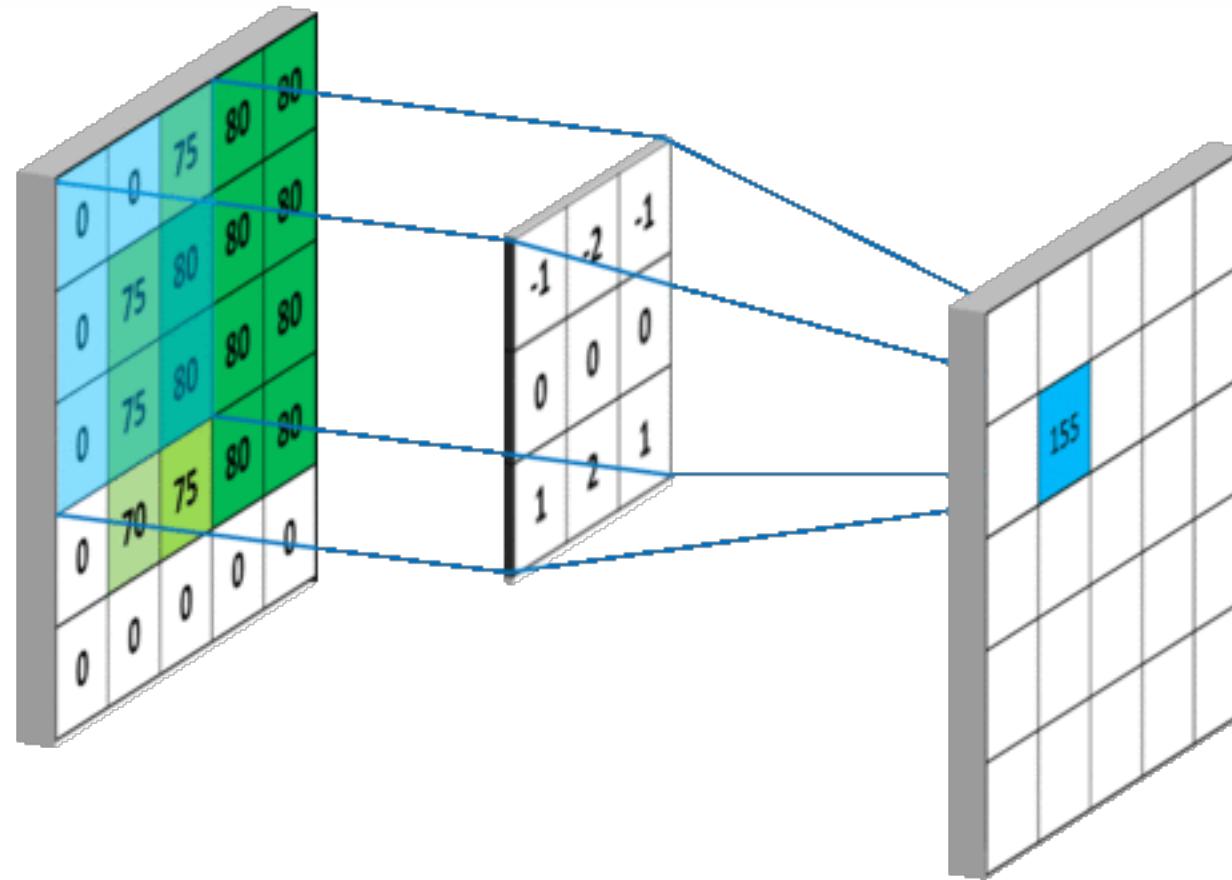
En el contexto de redes neuronales convolucionales se suele utilizar **la correlación cruzada** en lugar de la convolución para evitar el flipping del kernel.

$$S(i, j) = (I * K)(i, j) = \sum_m \sum_n I(i + m, j + n) K(m, n)$$

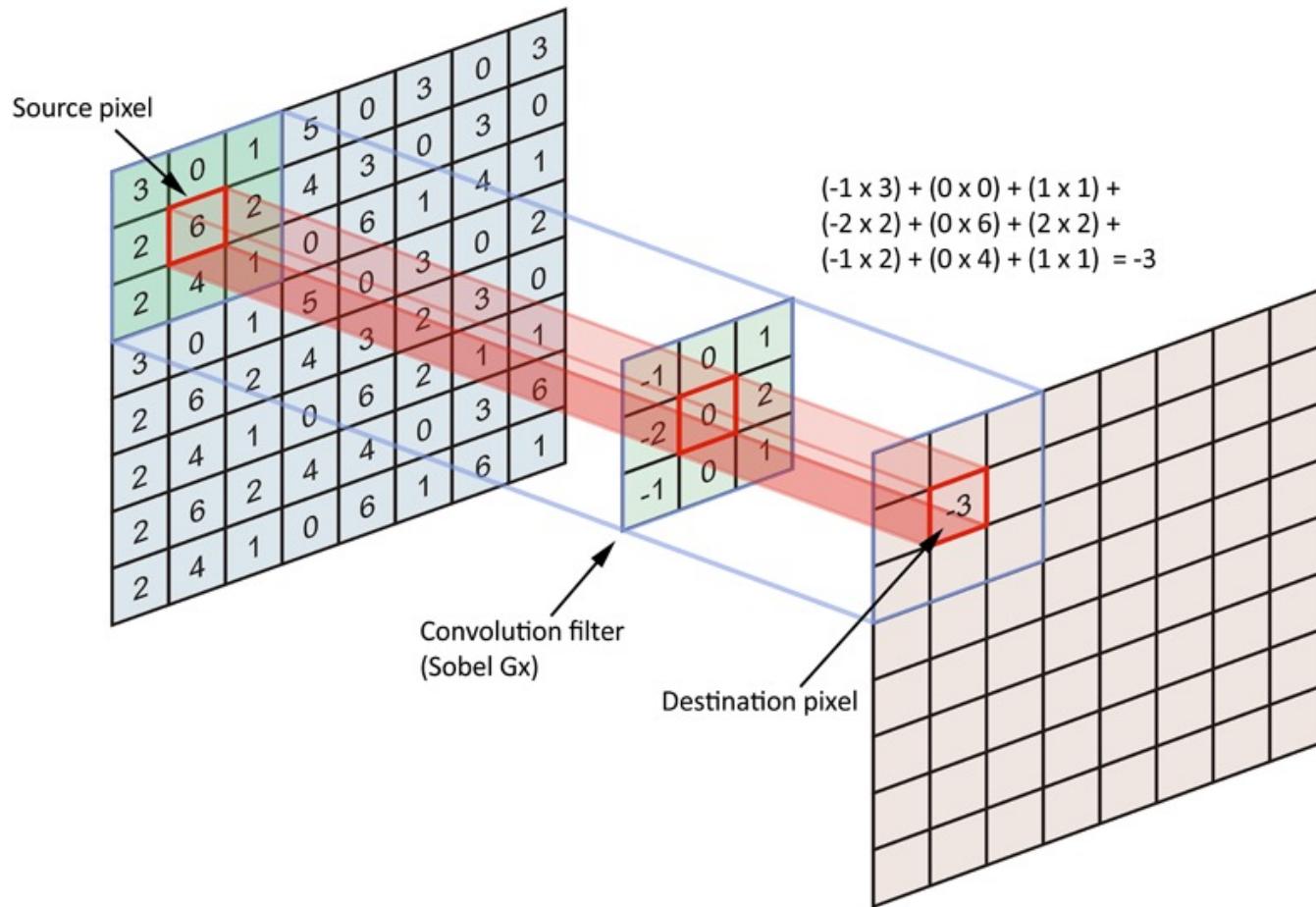
Convolución 1D



2D Convolution



Convolución 2D



Ejemplos de filtros



$$* \begin{bmatrix} 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \end{bmatrix} =$$



Ejemplos de filtros



$$* \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} =$$



Ejemplos de filtros



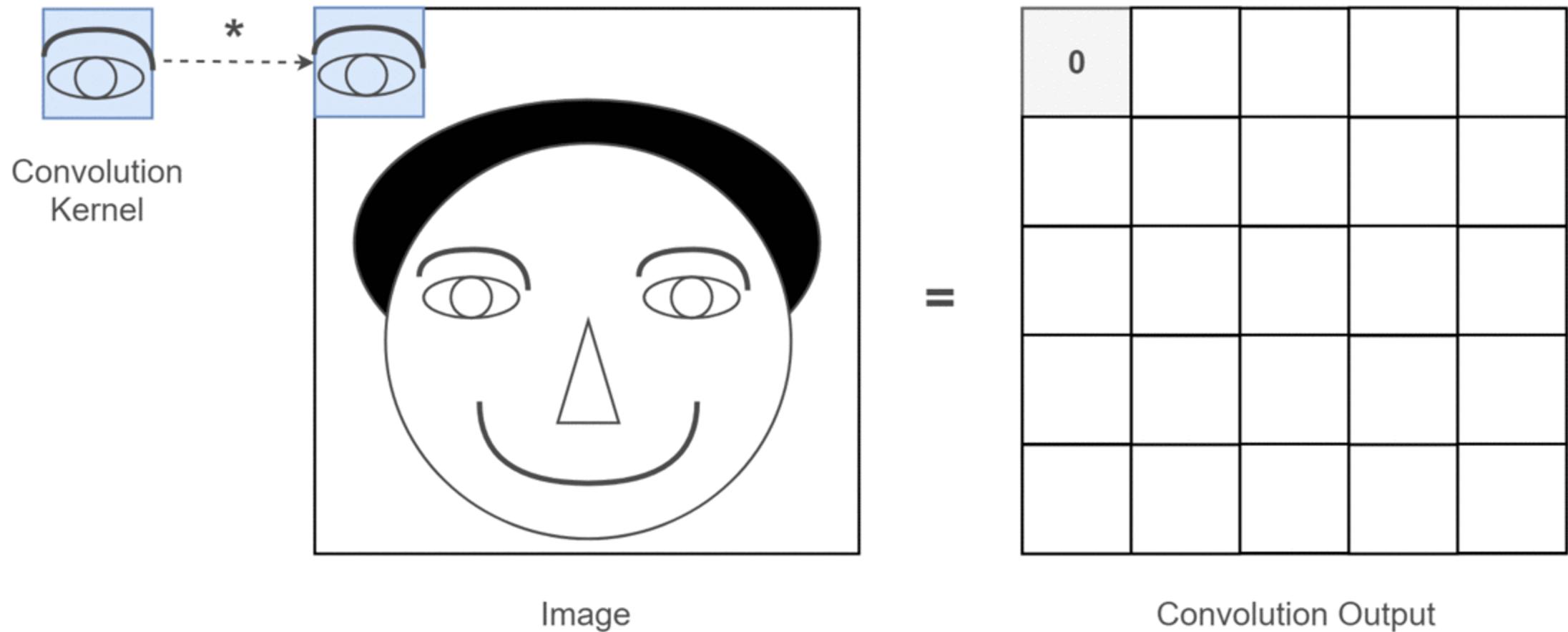
$$* \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} =$$



Ejemplos de filtros

<https://setosa.io/ev/image-kernels/>

Convolución 2D

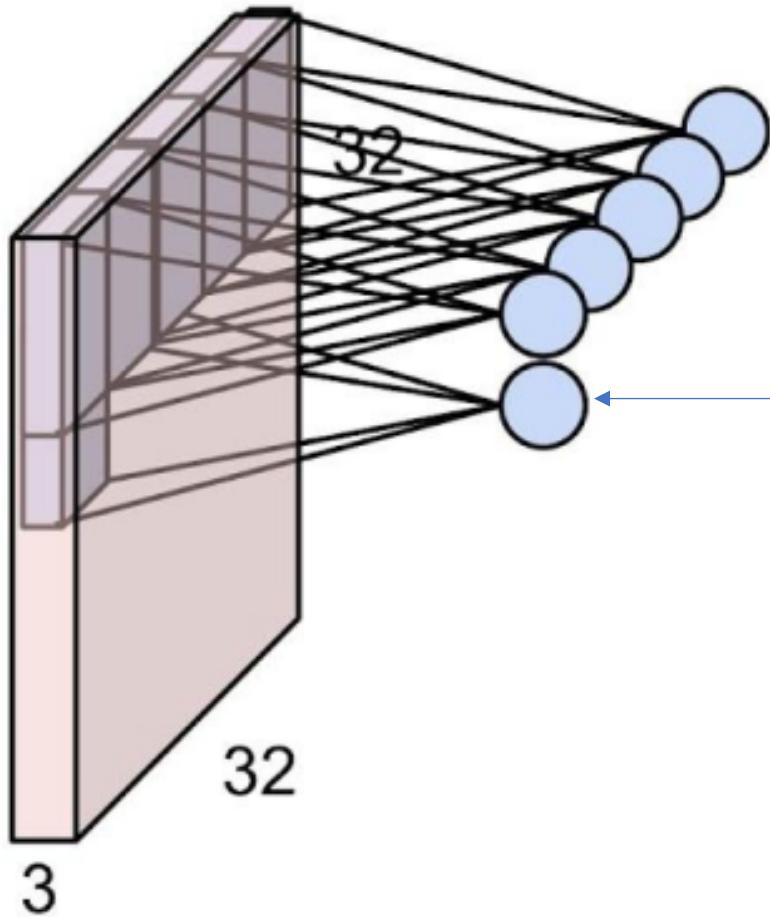


Redes Neuronales Convolucionales



Capas convolucionales

2D Convolution

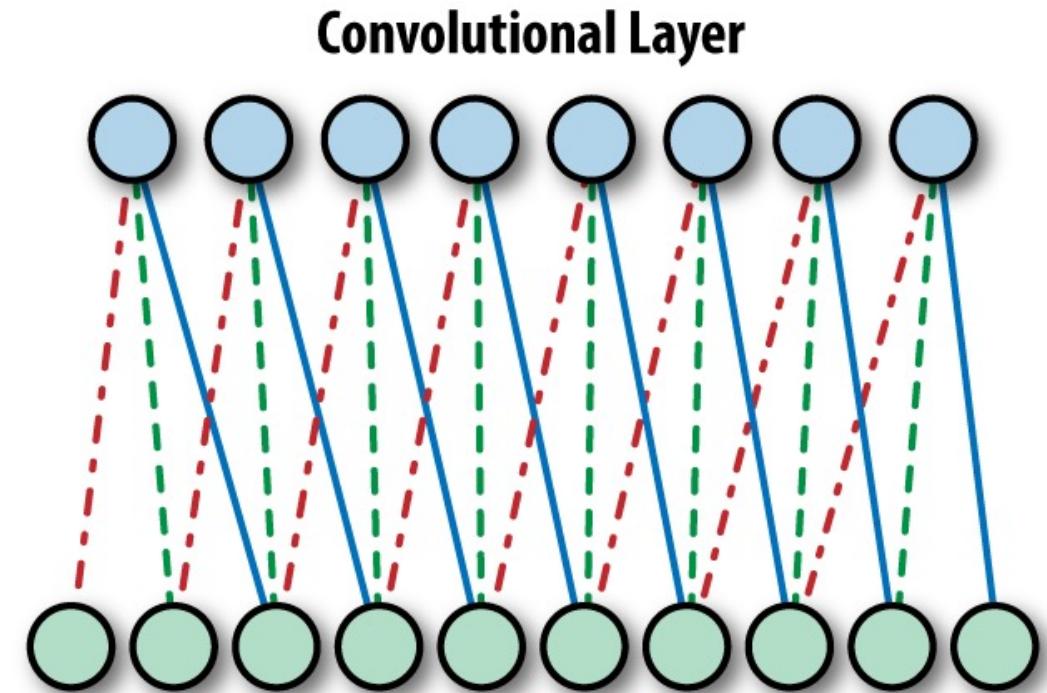
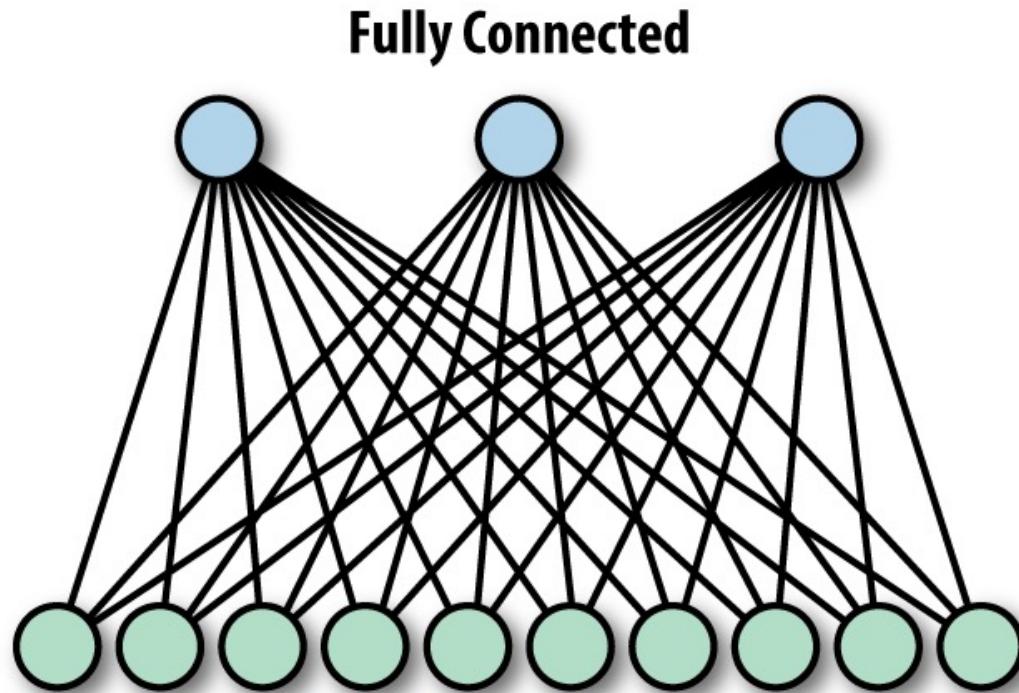


Las neuronas convolucionales tienen soporte Local y pesos compartidos.

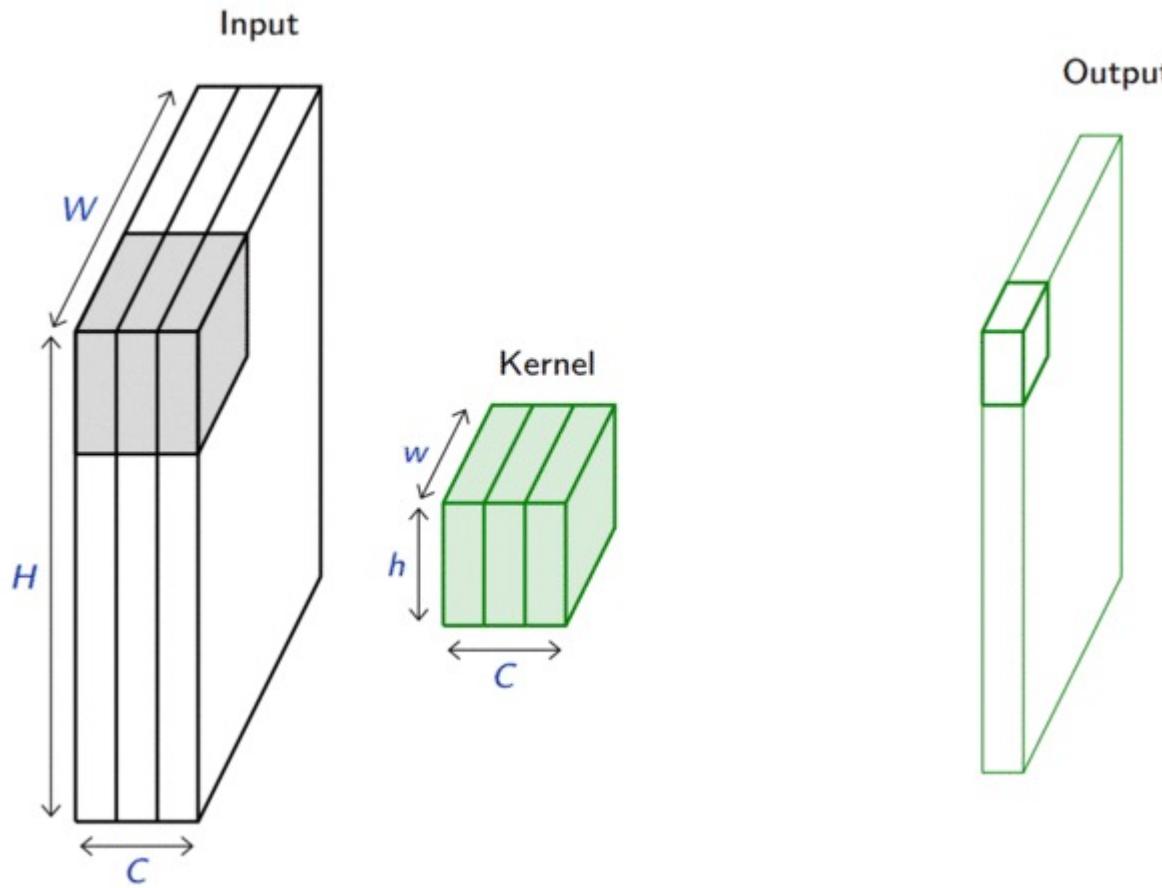
Las funciones de activación no lineales (ReLU, etc) se aplican elemento a elemento en las neuronas de salida

Notar que las neuronas convolucionales se organizan como un feature map (tienen 'estructura').

Capas totalmente conectadas vs capas convolucionales



Convolución 2D



La conectividad es local en el espacio y total en profundidad

Weight-sharing el número de parámetros e introduce invarianza a la traslación

Las funciones de activación no lineal (ej. ReLU) son aplicadas a cada elemento individualmente

Convolución 2D

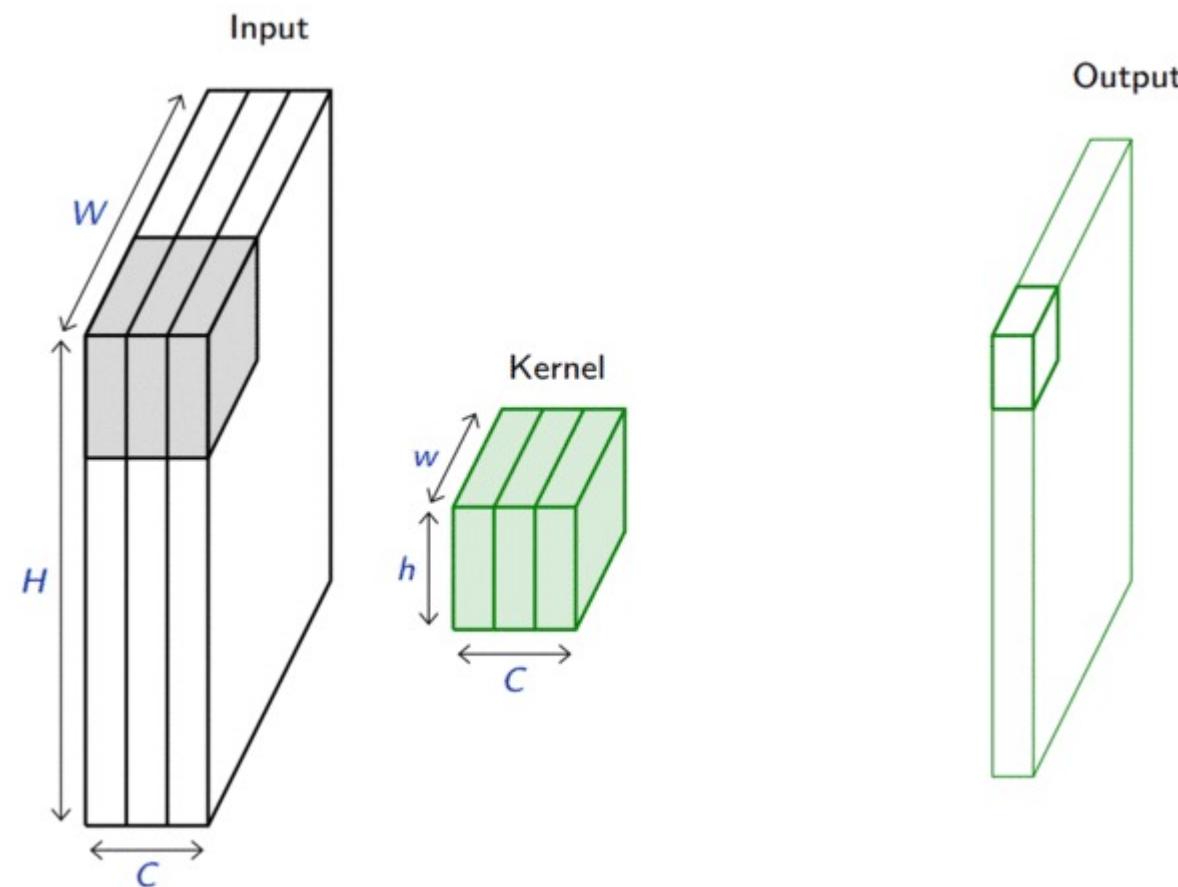
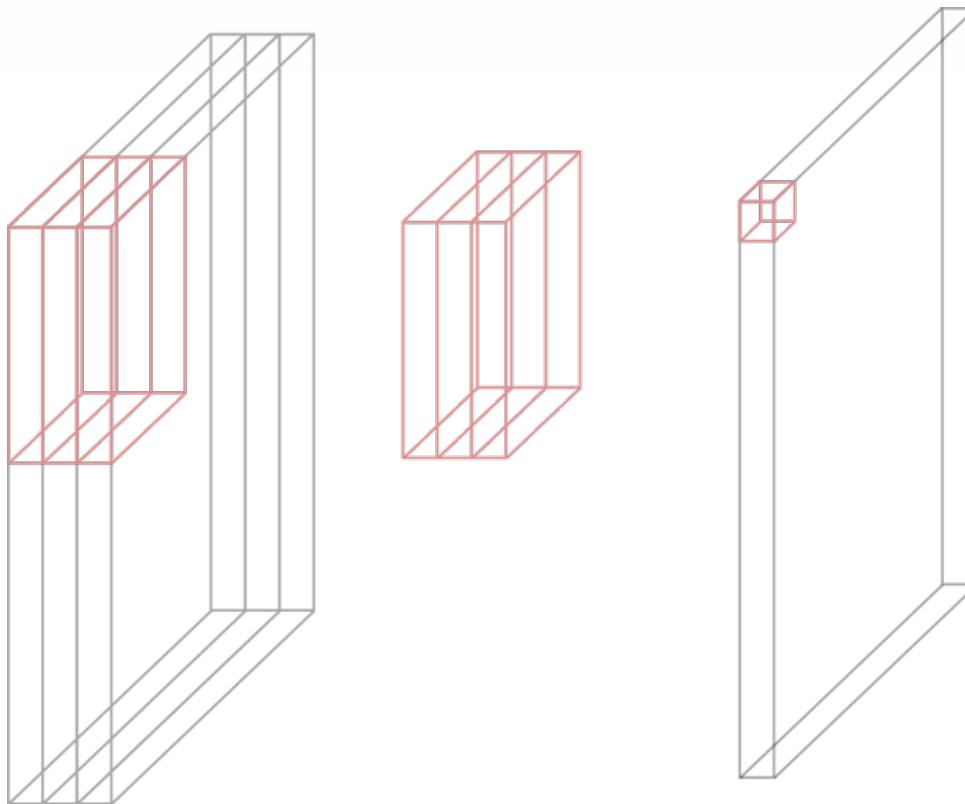


Imagen extraída de: <https://documents.epfl.ch/users/f/fl/fleuret/www/dlc/dlc-slides-4a-dag-autograd-conv.pdf>

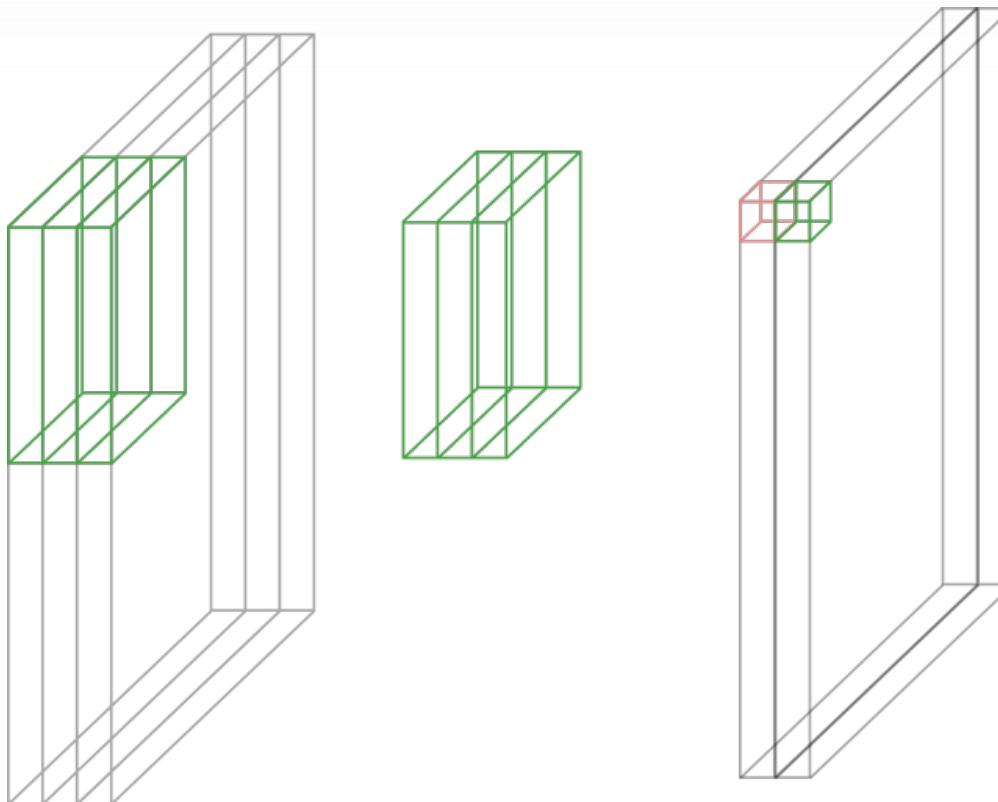
Salida de una capa convolucional



Parámetros a aprender en la capa



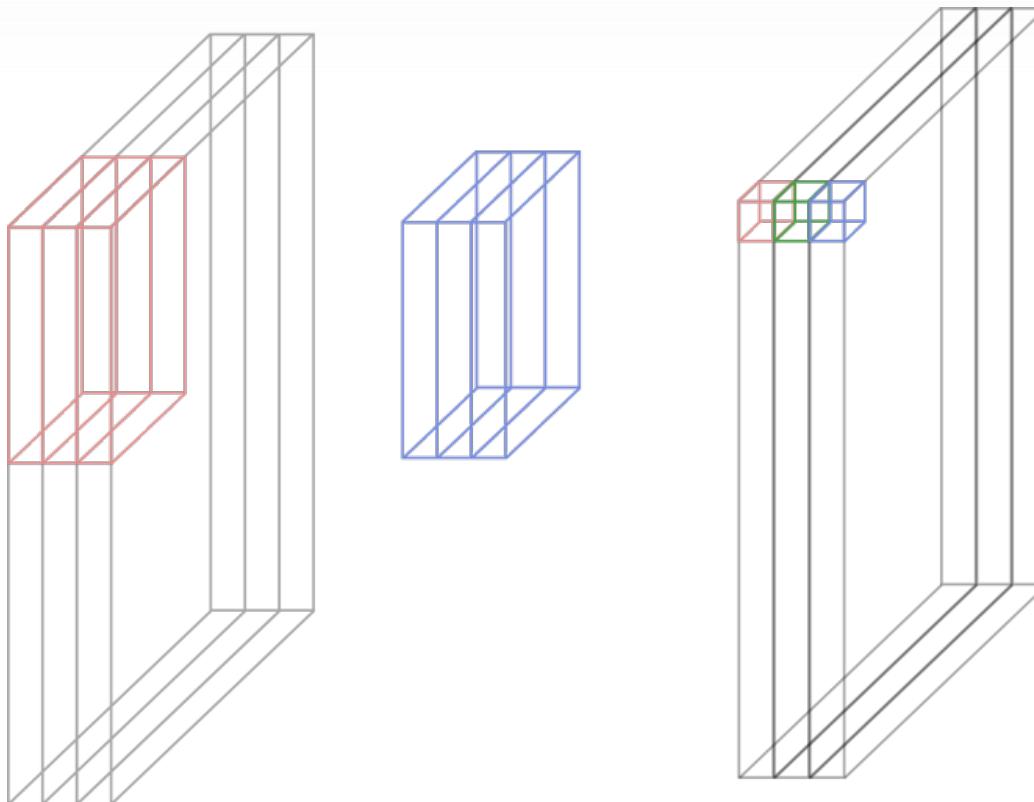
Salida de una capa convolucional



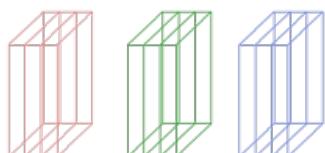
Parámetros a aprender en la capa



Salida de una capa convolucional



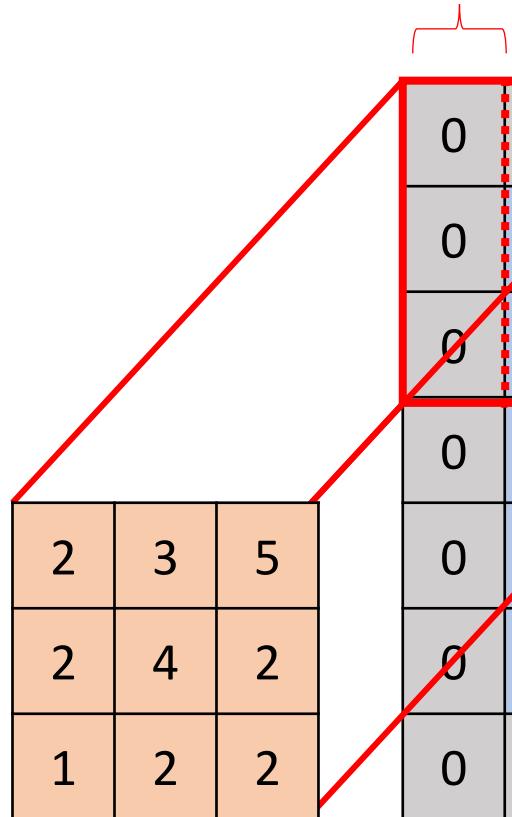
Parámetros a aprender en la capa



Hiperparámetros de una capa convolucional

Relación entrada/salida

Stride: $S = 1$



Entrada: $W = 5 \times 5$

0	0	0	0	0	0	0
0	0	2	2	1	2	0
0	4	3	0	1	2	0
0	2	1	3	2	1	0
0	3	1	2	0	1	0
0	4	2	0	1	1	0
0	0	0	0	0	0	0

Padding $P = 1$

Salida: $O = 5 \times 5$

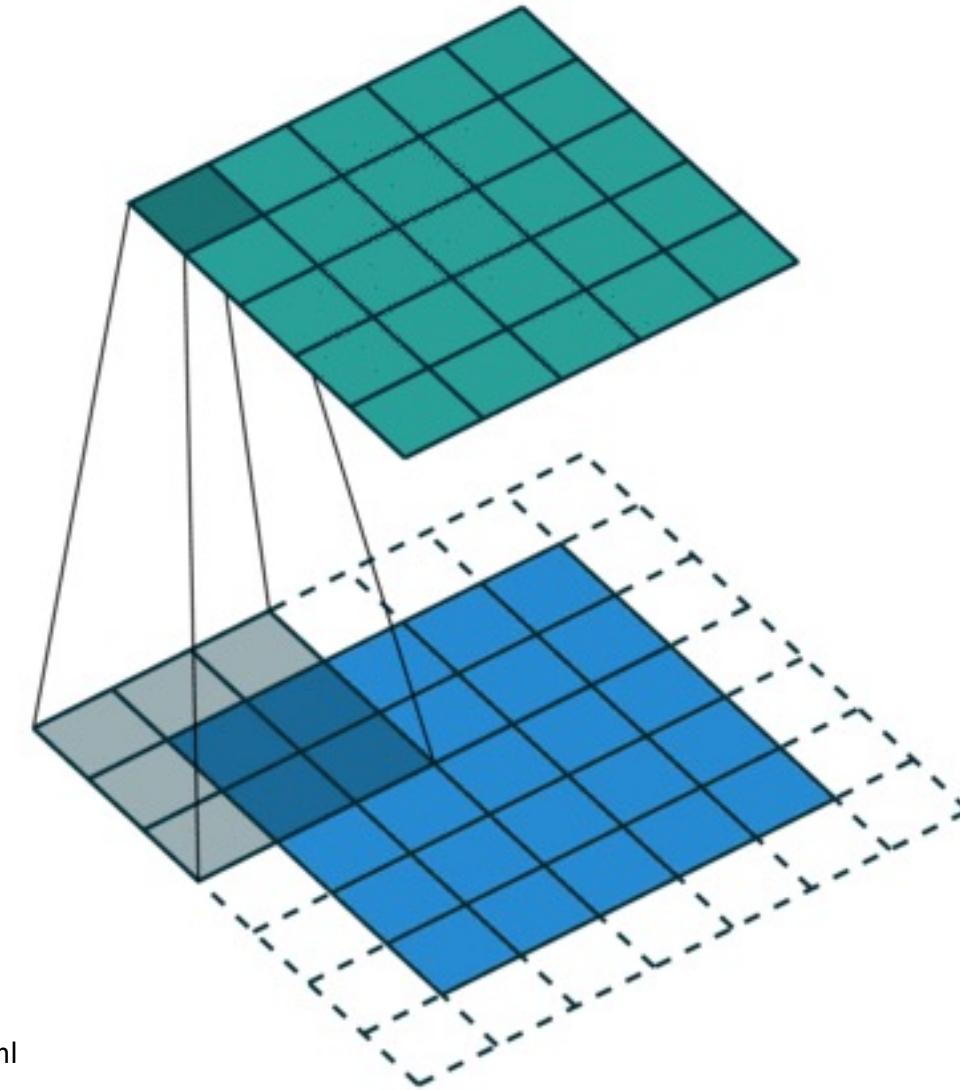
Conv

x	x	x	x	x
x	x	x	x	x
x	x	x	x	x
x	x	x	x	x
x	x	x	x	x

$$O = \frac{(W - F + 2P)}{S} + 1$$

Convolución con P=1, S=1 y F=3

Mantiene las dimensiones



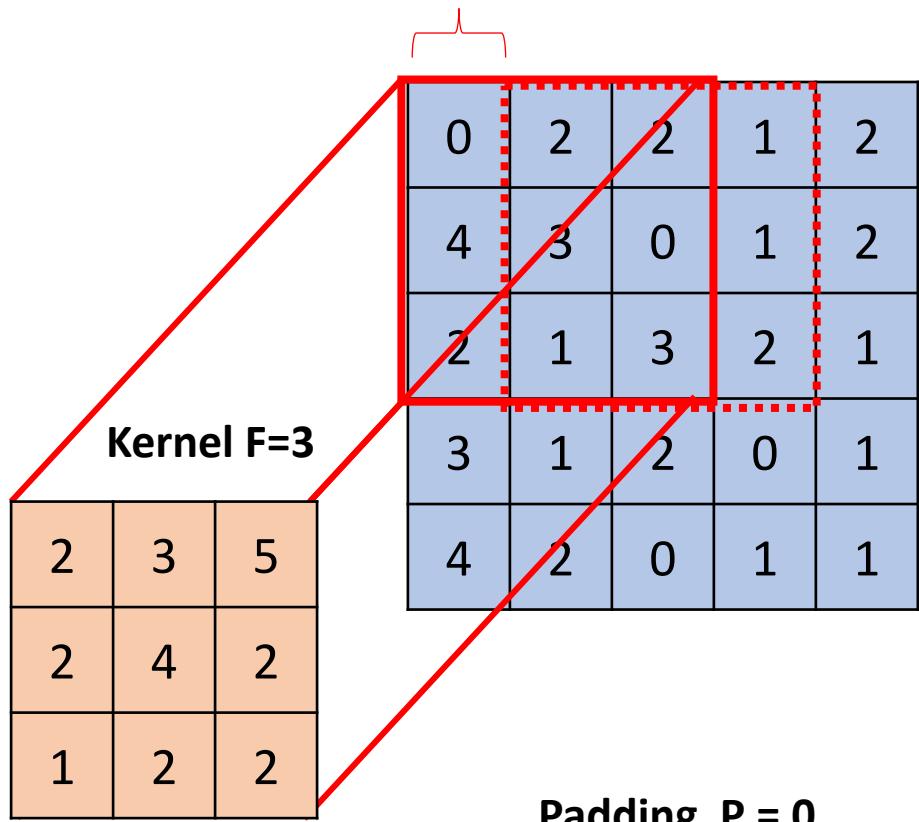
Animación extraída de
http://deeplearning.net/software/theano/tutorial/conv_arithmetic.html

Hiperparámetros de una capa convolucional

Relación entrada/salida

Stride: S = 1

Entrada: W = 5x5



Salida: O = 3x3

Conv

$$O = \frac{(W - F + 2P)}{S} + 1$$

Convolución con P=0, S=1, F=3, w=5

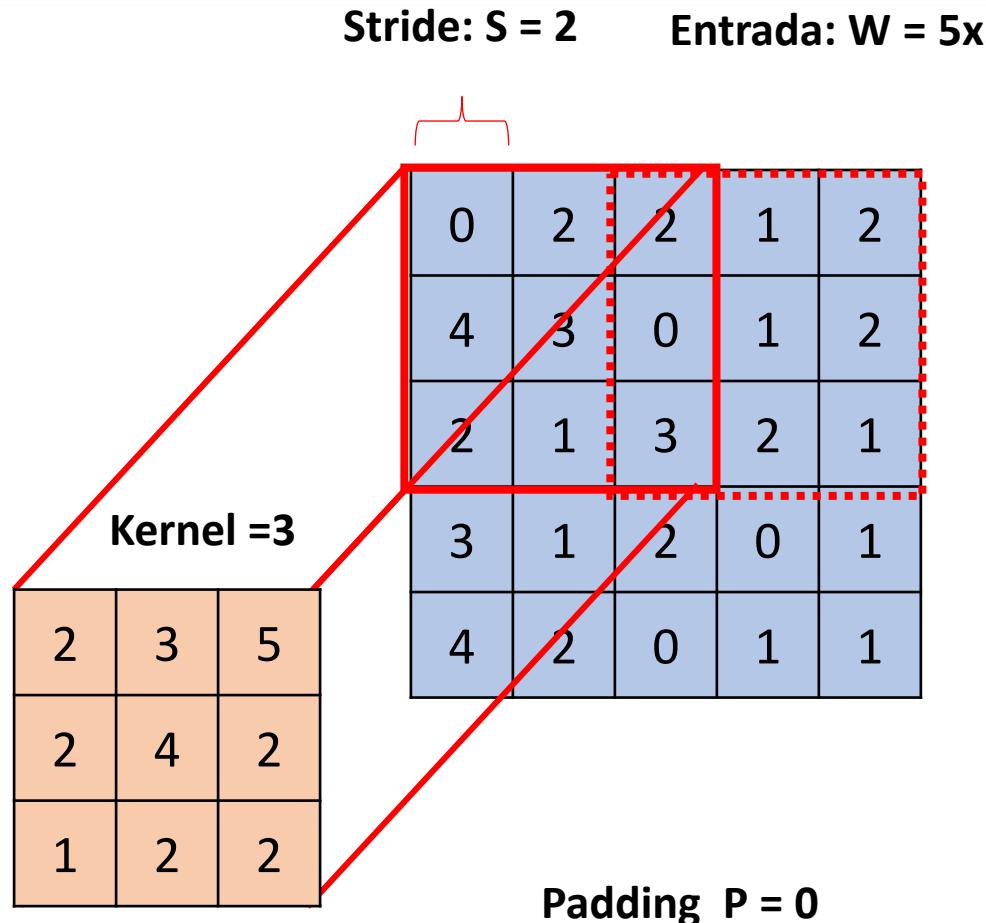
El tamaño de salida O se reduce a 3

3 ₀	3 ₁	2 ₂	1	0
0 ₂	0 ₂	1 ₀	3	1
3 ₀	1 ₁	2 ₂	2	3
2	0	0	2	2
2	0	0	0	1

12	12	17
10	17	19
9	6	14

Hiperparámetros de una capa convolucional

Relación entrada/salida



Salida: O = 3x3

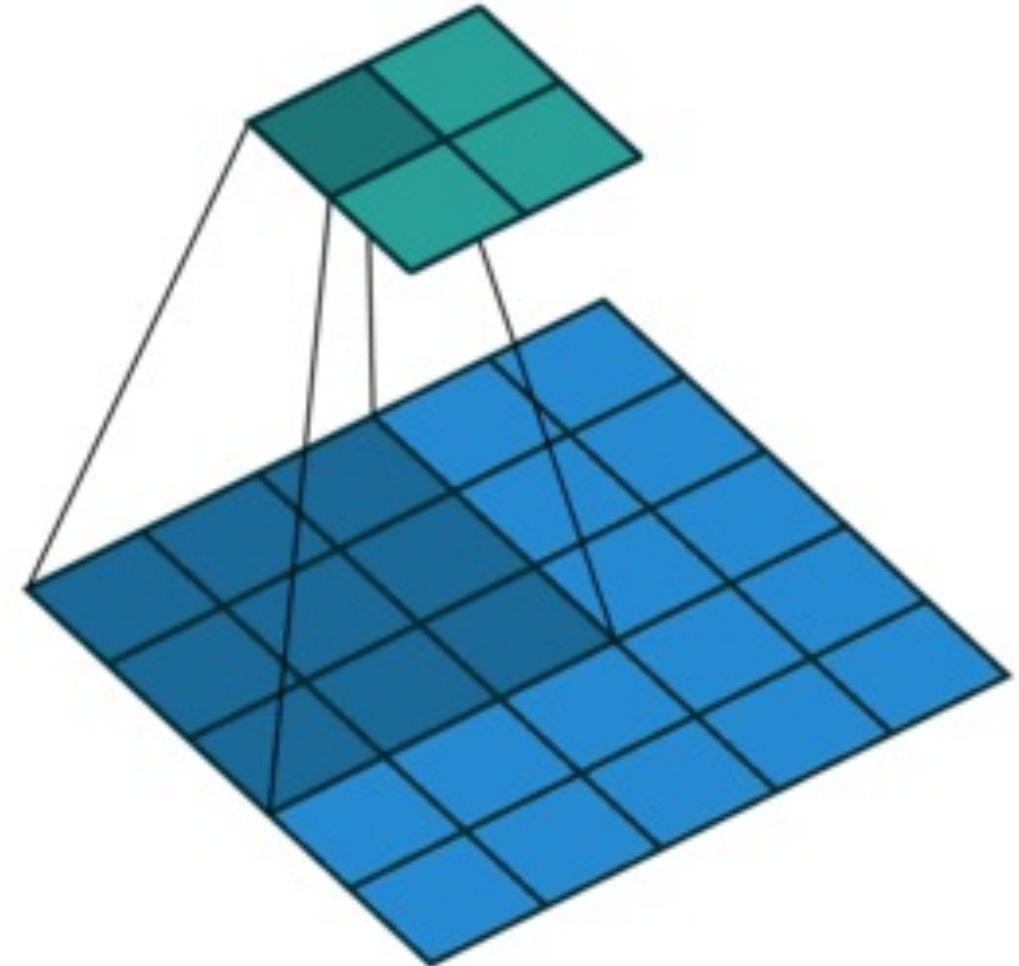
Conv

x	x
x	x

$$O = \frac{(W - F + 2P)}{S} + 1$$

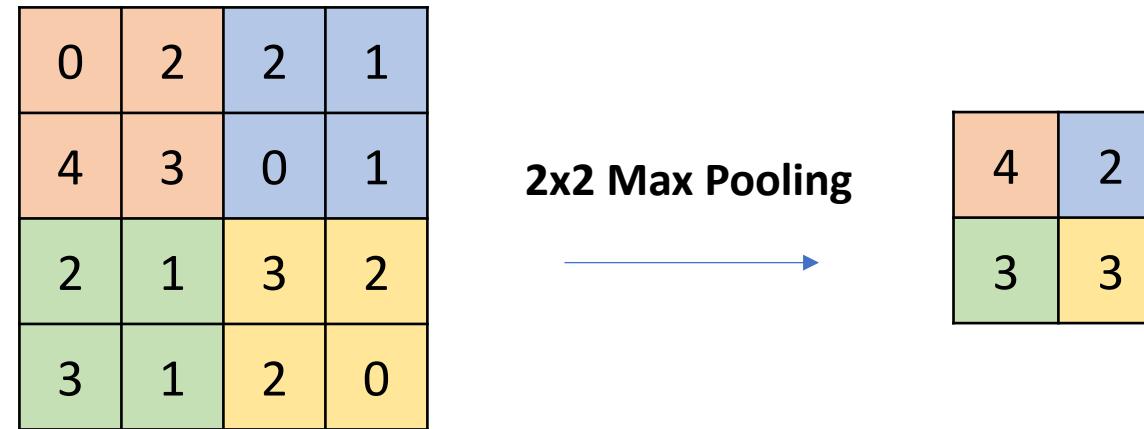
Convolución con P=0, S=2, F=3, W=5

El tamaño de salida O
se reduce a 2



Animación extraída de
http://deeplearning.net/software/theano/tutorial/conv_arithmetic.html

Capa de Max-Pooling



- Reduce la dimensión de los feature maps
- Contribuye a la invarianza respecto a pequeñas traslaciones en las imágenes de entrada

Capa de Avg-Pooling

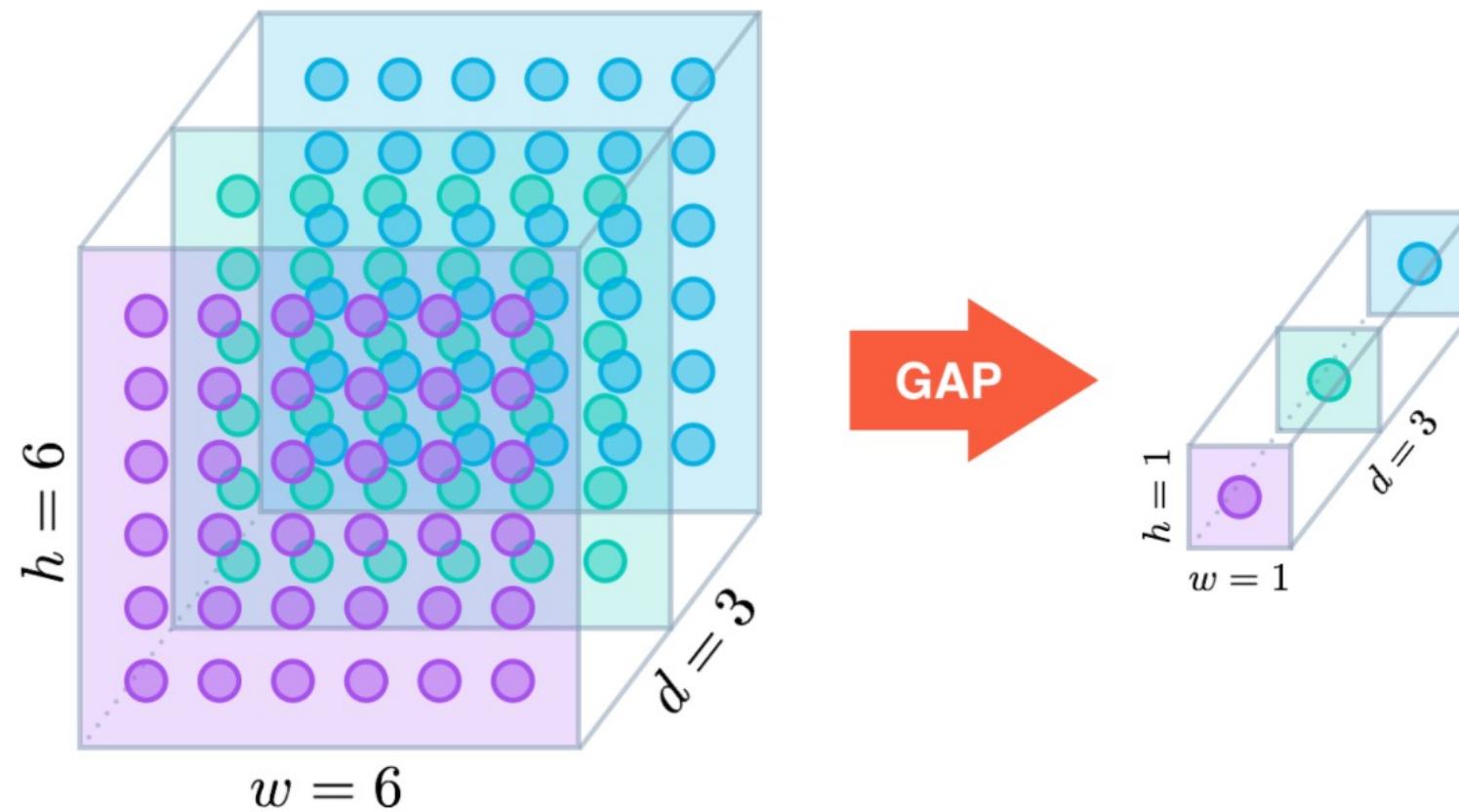
0	2	2	1
4	3	0	1
2	1	3	2
3	1	2	0

2x2 Avg Pooling

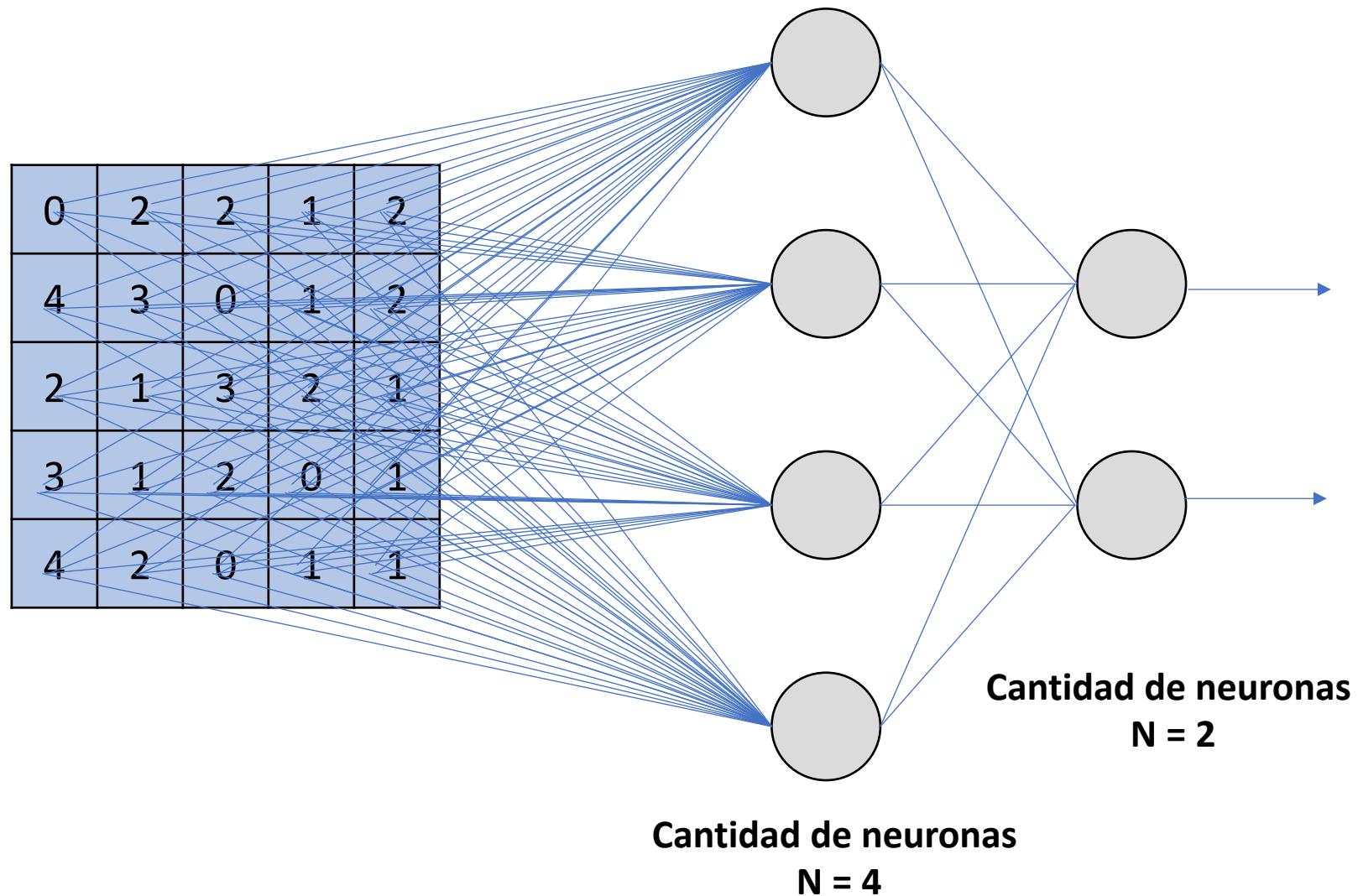


9/4	4/4
7/4	7/4

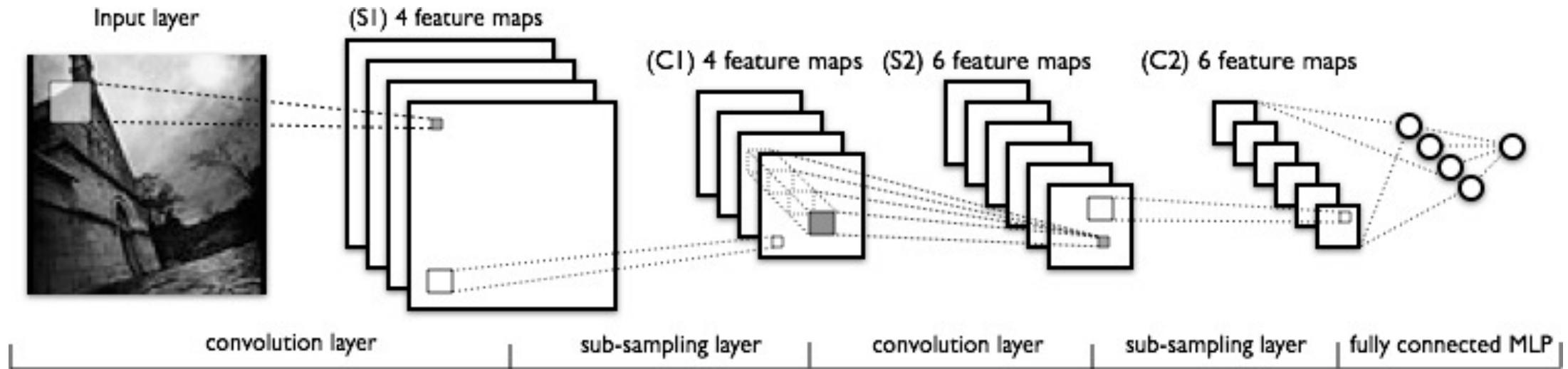
Global Average Pooling



Capa Fully Connected (Totalmente Conectada)



Redes Neuronales Convolucionales



- Las convoluciones tienen soporte local
- Implementan 'weight sharing' reduciendo el número de parámetros e introduciendo invarianza espacial

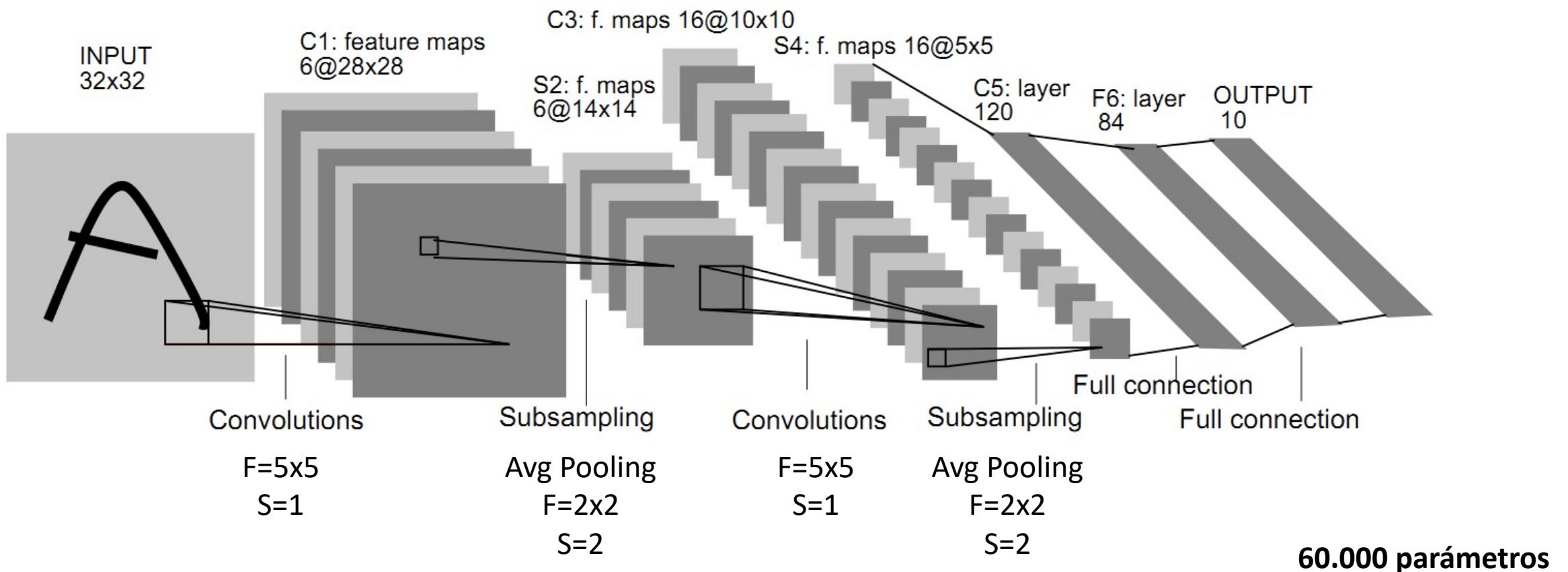
ConvNets para el análisis de imágenes: Ventajas

- Naturalmente adaptadas a la **estructura regular** de las imágenes (por medio de la operación de convolución)
- Invariantes respecto a translaciones
- Aprendizaje **end-to-end**
- Bajos requerimientos de memoria: **weight sharing**
- **Eficientes** en test-time
- Buen grado de **generalización** si se entranan con suficientes datos

Algunas arquitecturas clásicas para clasificación

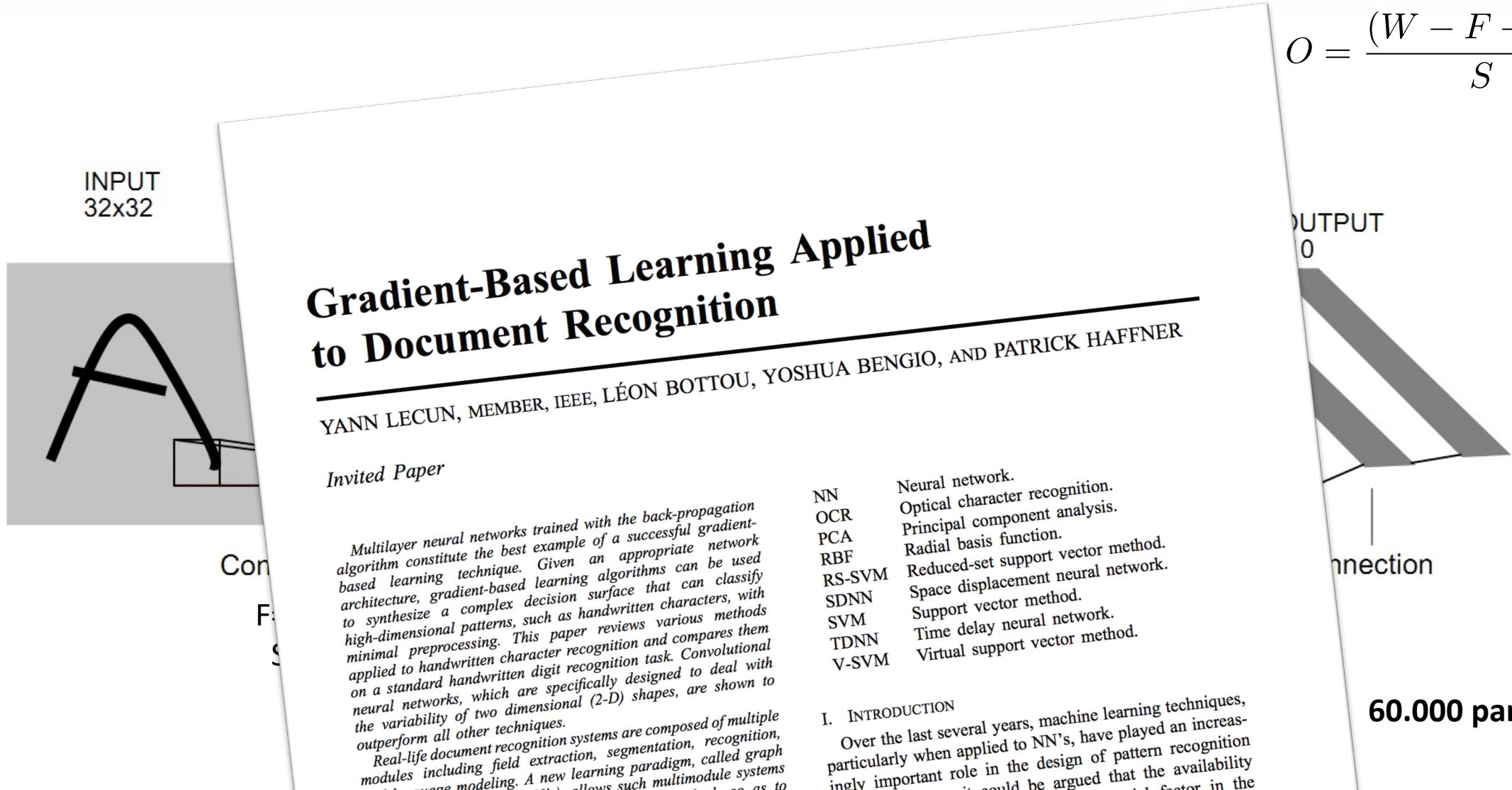
LeNet-5 (1998, Clasificación de dígitos MNIST)

$$O = \frac{(W - F + 2P)}{S} + 1$$



Algunas arquitecturas clásicas para clasificación

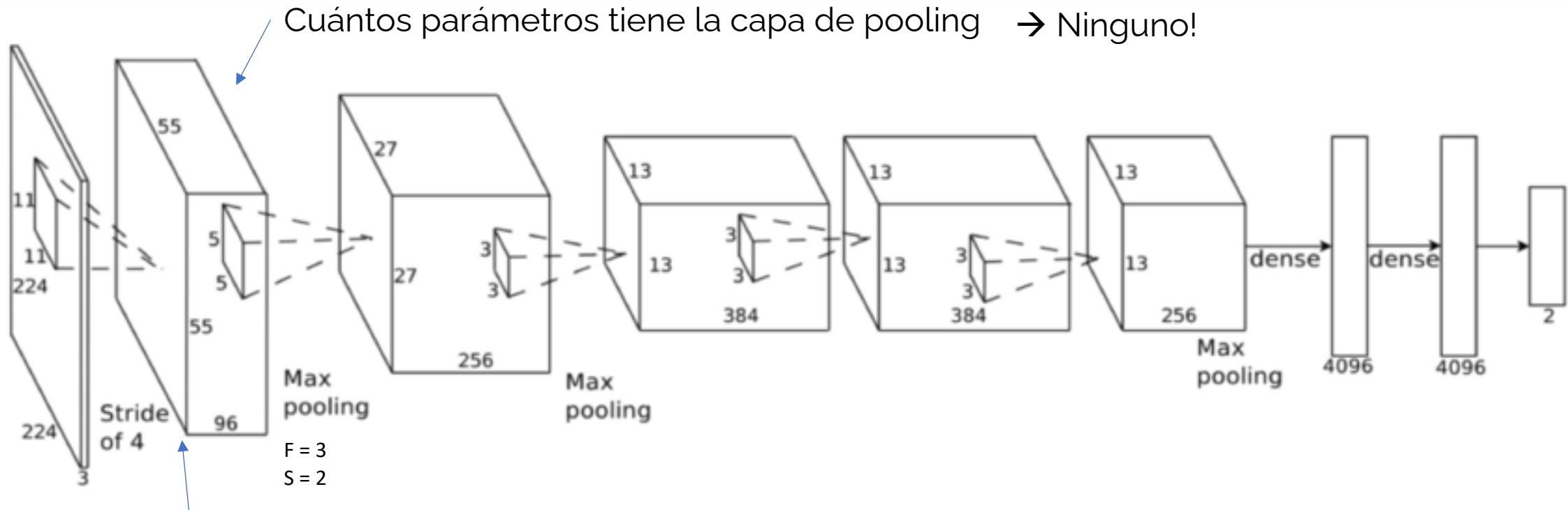
LeNet-5 (1998, Clasificación de dígitos MNIST)



$$O = \frac{(W - F + 2P)}{S} + 1$$

Algunas arquitecturas clásicas para clasificación

AlexNet (2012)



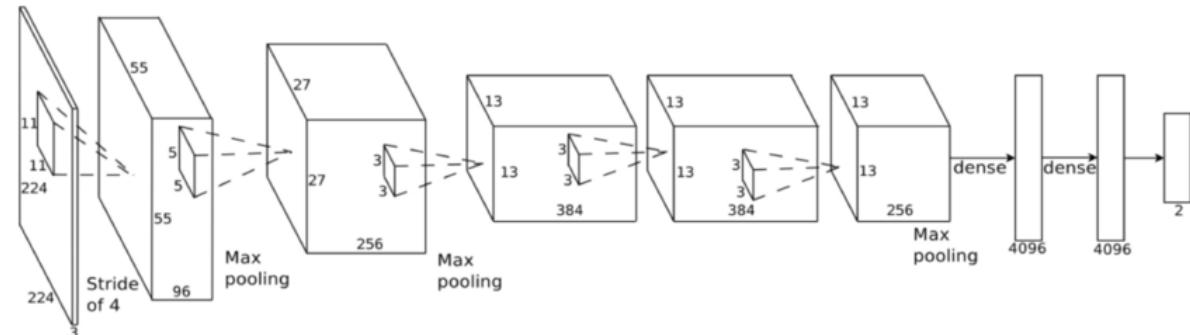
- Cada capa tiene muchos más feature maps que LeNet. Total de parámetros: 60 millones.

$$O = \frac{(W - F + 2P)}{S} + 1$$

Algunas arquitecturas clásicas para clasificación

AlexNet (2012)

Layer	Units	Weights
L_1 (Conv)	290,400	34,848
L_2 (Conv)	186,624	307,200
L_3 (Conv)	64,896	884,736
L_4 (Conv)	64,869	663,552
L_5 (Conv)	43,264	442,368
L_6 (Dense)	4096	37,748,736
L_7 (Dense)	4096	16,777,216
L_8 (Dense)	1000	4,096,000
Conv Subtotal	650,080	2,332,704
Dense Subtotal	9192	58,621,952
Total	659,272	60,954,656

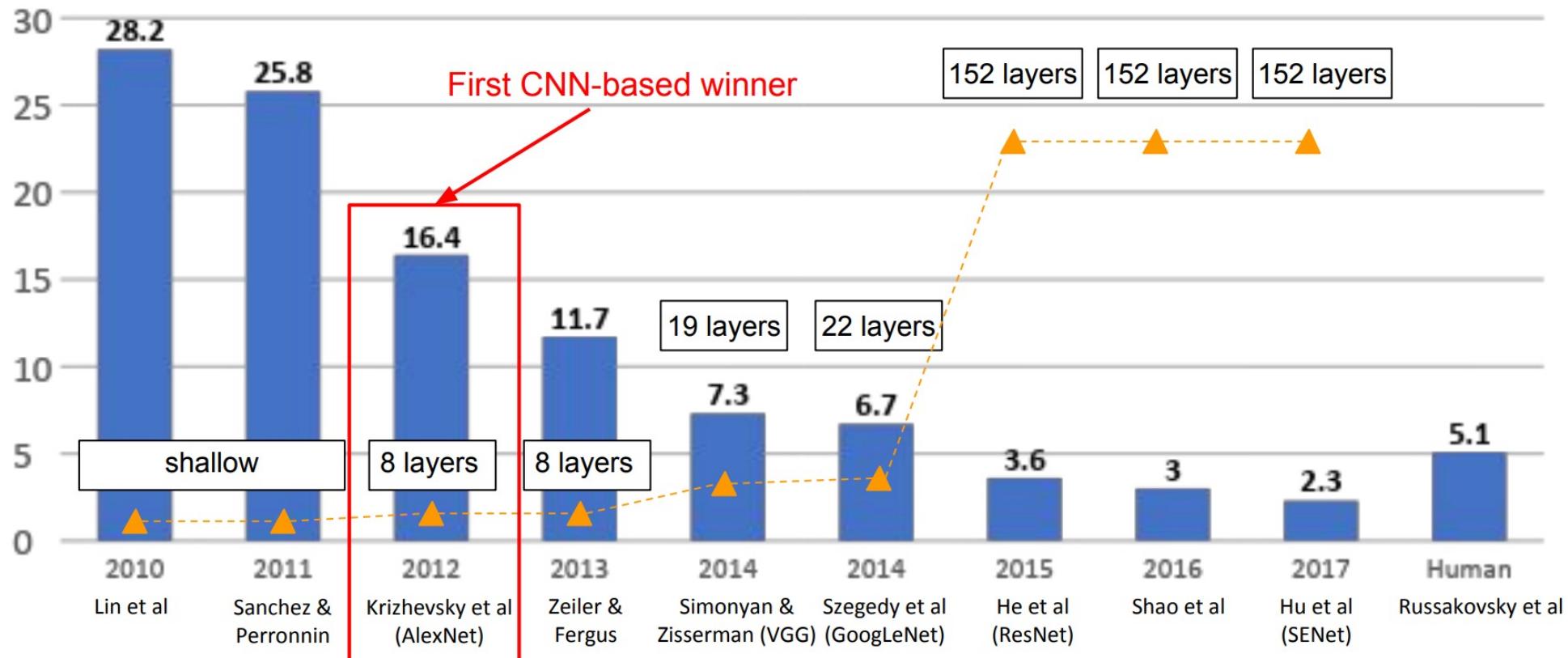


De los 60 Millones de parámetros,
58 Millones corresponden a las últimas capas densas!

Algunas arquitecturas clásicas para clasificación

AlexNet (2012)

ImageNet Large Scale Visual Recognition Challenge (ILSVRC) winners



Algunas arquitecturas clásicas para clasificación

AlexNet (2012)

ImageNet Classification with Deep Convolutional Neural Networks

Alex Krizhevsky
University of Toronto
kriz@cs.utoronto.ca

Ilya Sutskever
University of Toronto
ilya@cs.utoronto.ca

Geoffrey E. Hinton
University of Toronto
hinton@cs.utoronto.ca

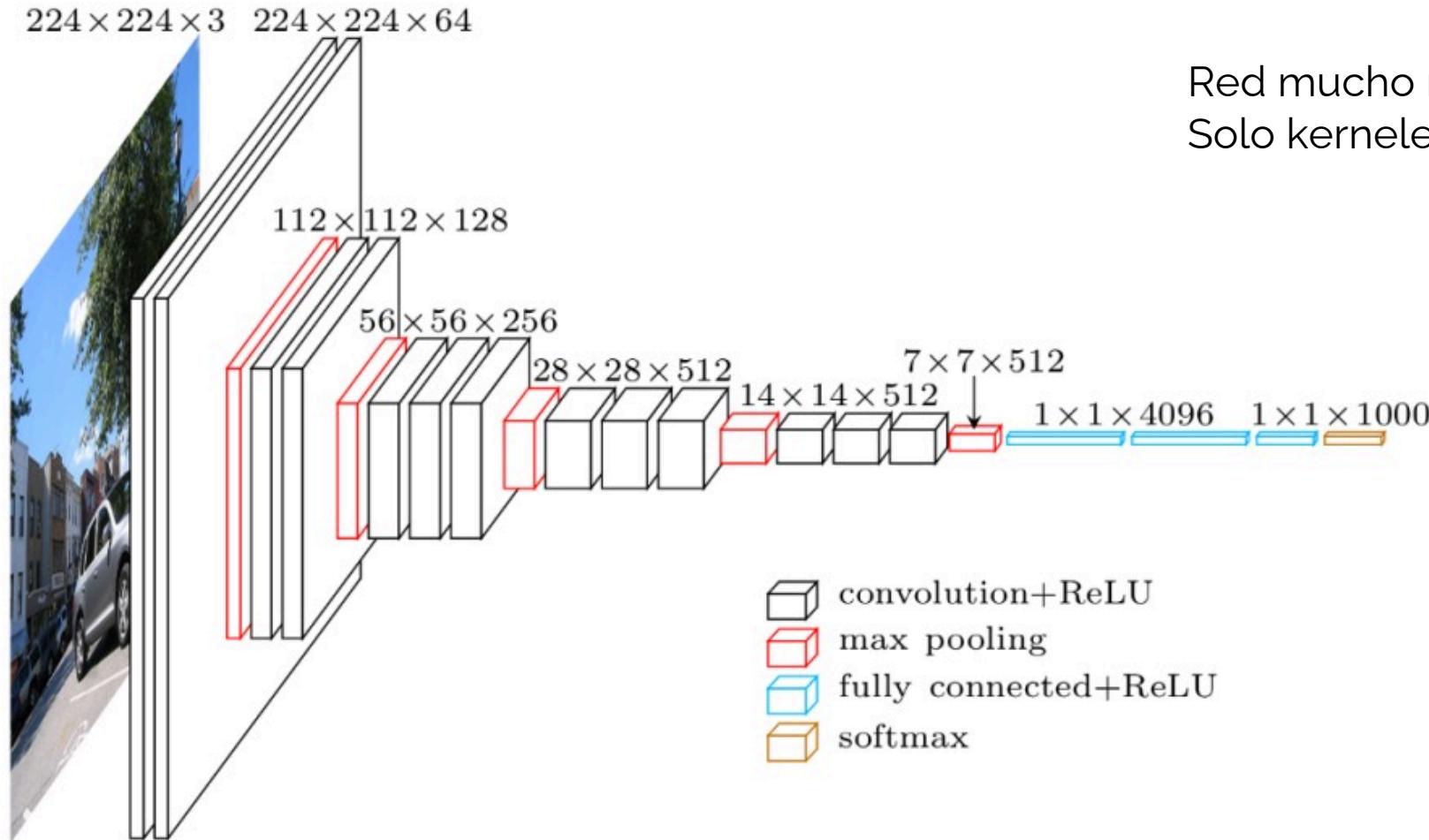
Abstract

We trained a large, deep convolutional neural network to classify the 1.2 million high-resolution images in the ImageNet LSVRC-2010 contest into the 1000 different classes. On the test data, we achieved top-1 and top-5 error rates of 37.5% and 17.0% which is considerably better than the previous state-of-the-art. The neural network, which has 60 million parameters and 650,000 neurons, consists of five convolutional layers, some of which are followed by max-pooling layers, and three fully-connected layers with a final 1000-way softmax. To make training faster, we used non-saturating neurons and a very efficient GPU implementation of the convolution operation. To reduce overfitting in the fully-connected layers we employed a recently-developed regularization method called “dropout” that proved to be very effective. We also entered a variant of this model in the ILSVRC-2012 competition and achieved a winning top-5 test error rate of 15.3%, tied by the second-best entry.



Algunas arquitecturas clásicas para clasificación

VGGNet (2014)

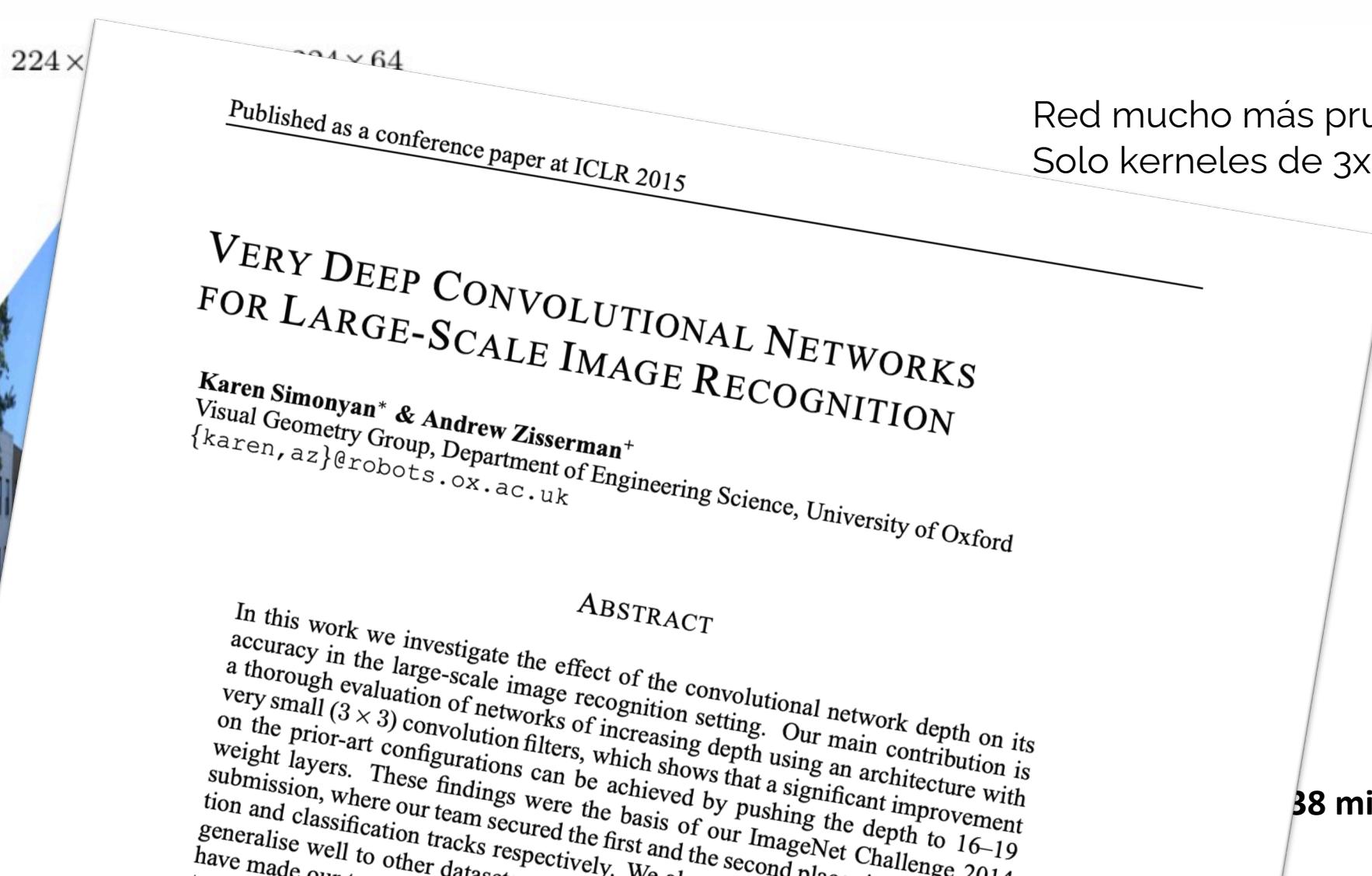


Red mucho más profunda, que utiliza
Solo kernel de 3x3 y pooling de 2x2

138 millones de parámetros

Algunas arquitecturas clásicas para clasificación

VGGNet (2014)



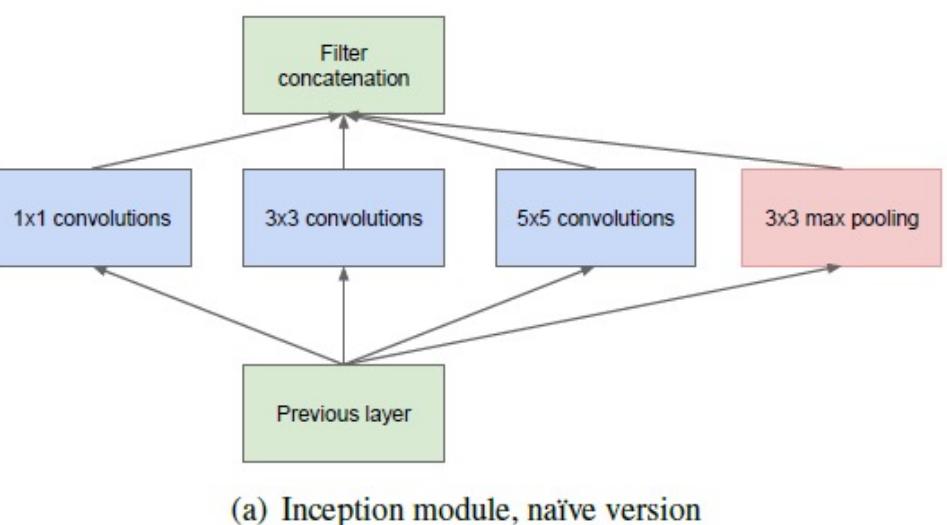
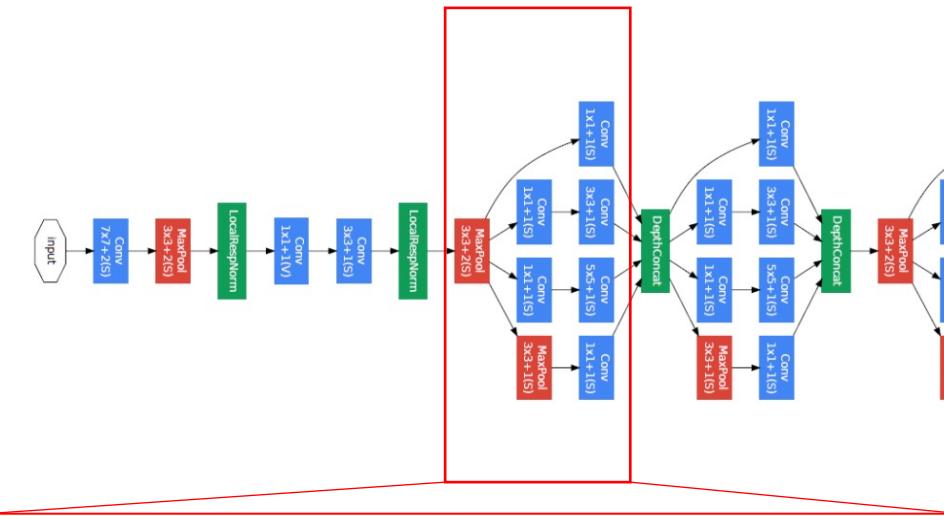
Red mucho más prunda, que utiliza
Solo kerneles de 3×3 y pooling de 2×2

38 millones de parámetros

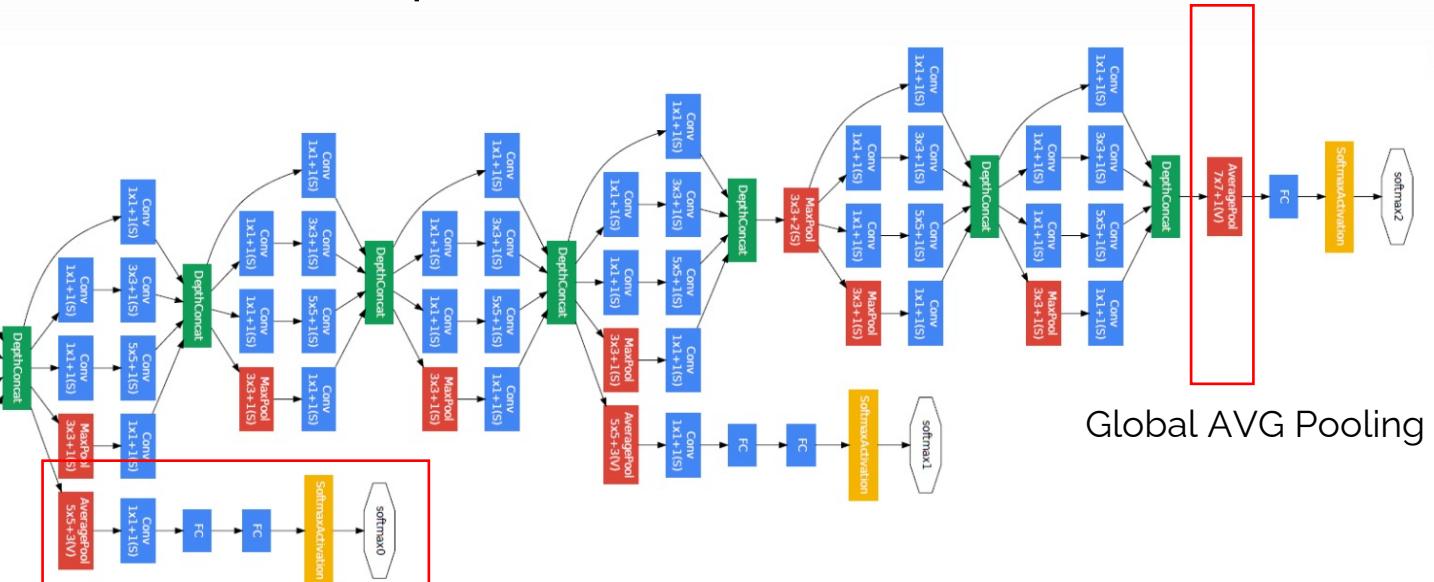
Algunas arquitecturas clásicas para clasificación

GoogLeNet (2014, InceptionV1)

Inception Modules



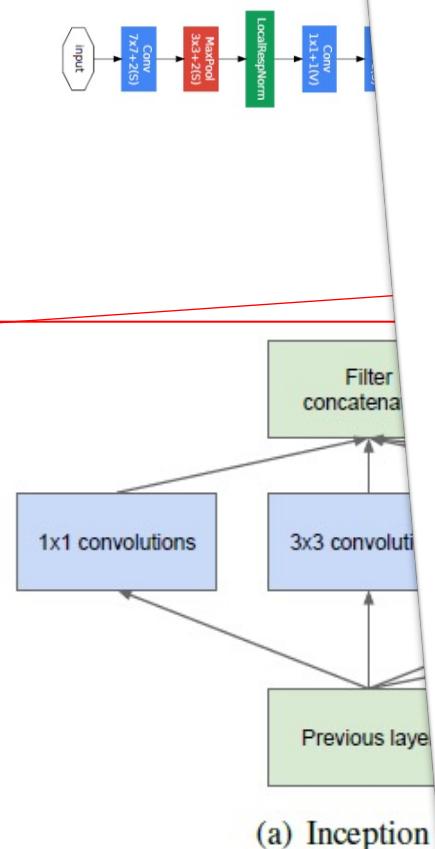
Clasificadores auxiliares



4 Millones de parámetros

Algunas arquitecturas clásicas para clasificación

Going deeper with convolutions



Christian Szegedy
Google Inc.

Pierre Sermanet
Google Inc.

Vincent Vanhoucke
Google Inc.

Wei Liu
University of North Carolina, Chapel Hill

Scott Reed
University of Michigan

Dragomir Anguelov
Google Inc.

Andrew Rabinovich
Google Inc.

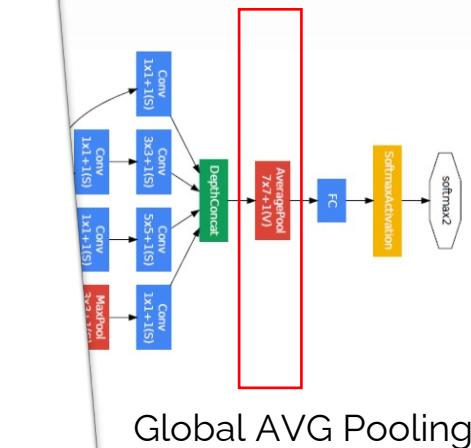
Yangqing Jia
Google Inc.

Dumitru Erhan
Google Inc.

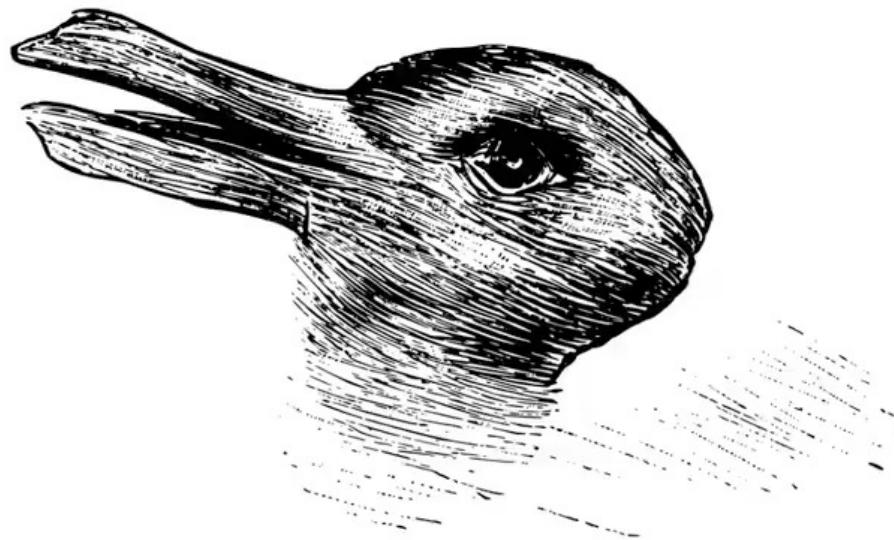
Abstract

We propose a deep convolutional neural network architecture codenamed Inception, which was responsible for setting the new state of the art for classification and detection in the ImageNet Large-Scale Visual Recognition Challenge 2014 and 2015. The main hallmark of this architecture is the improved utilization

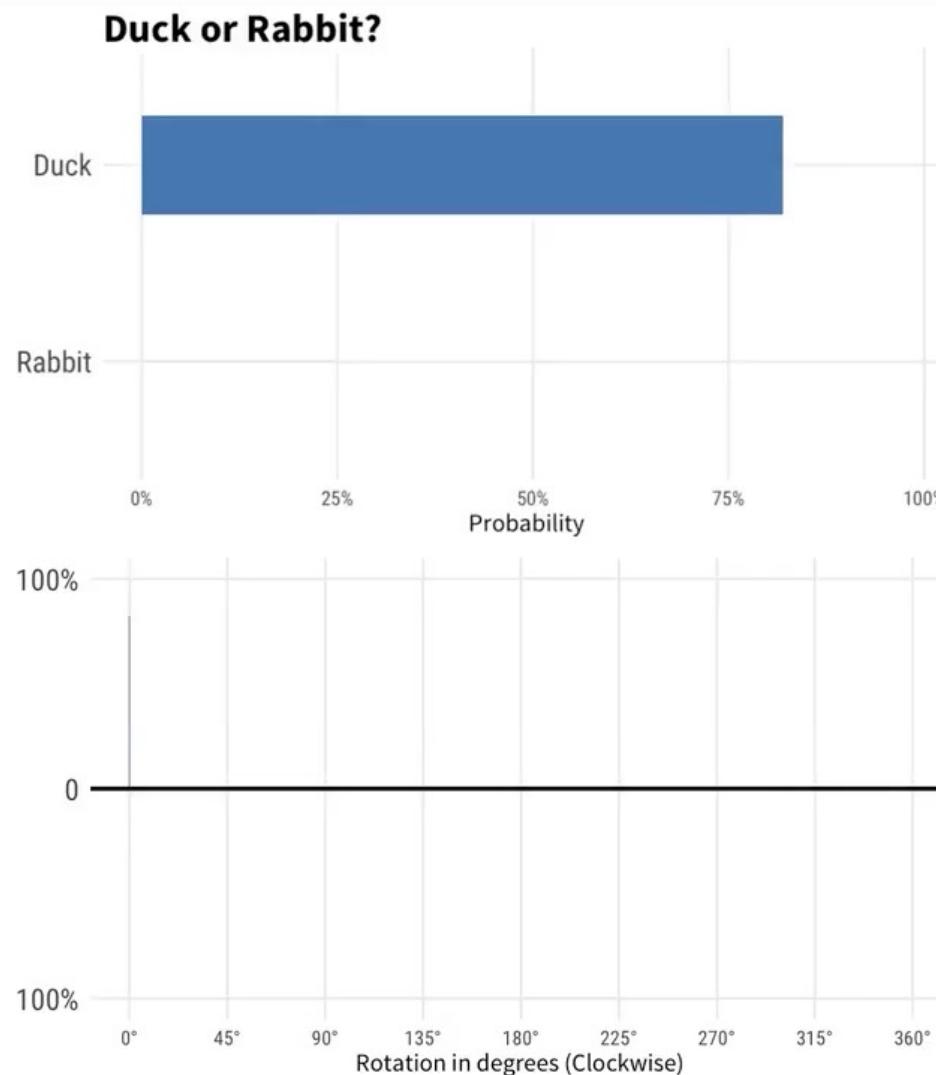
de parámetros



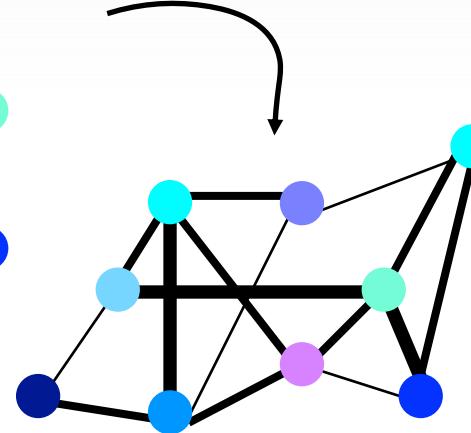
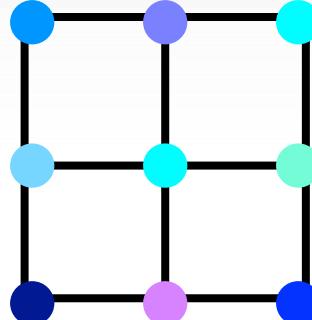
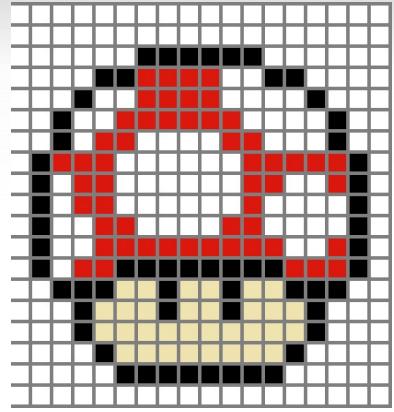
Salidas de una CNN para clasificación de imágenes



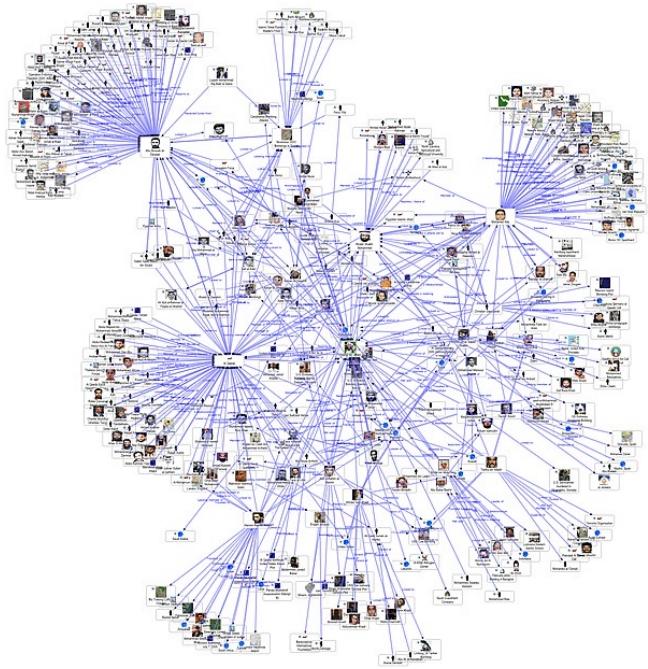
Predictions provided by the Google Cloud Vision API
Animation by Max Woolf (@minimaxir)



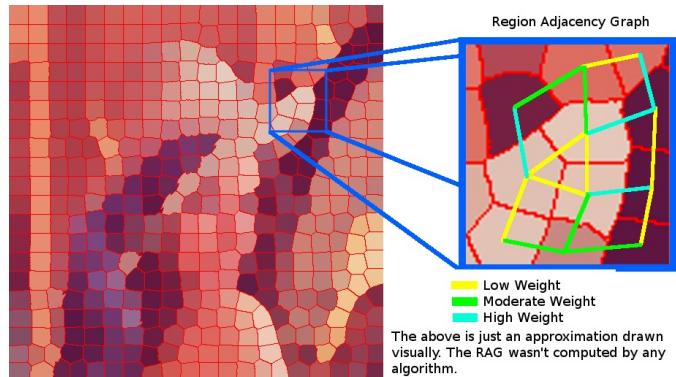
Redes Neuronales Convolucionales en grafos



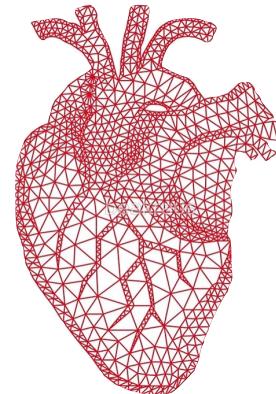
Redes sociales



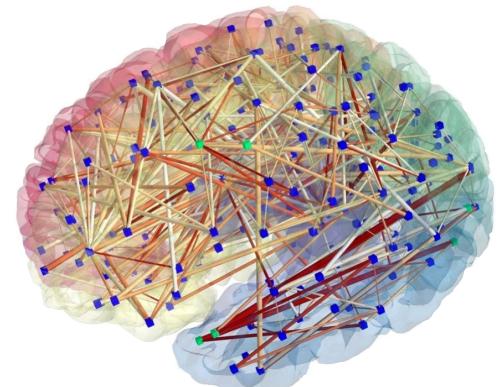
Superpixels



Meshes

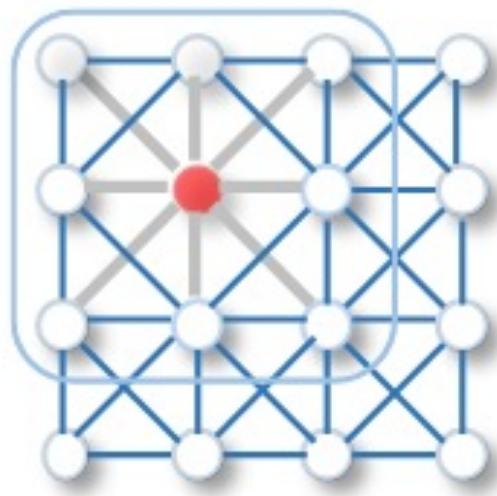


Brain Networks

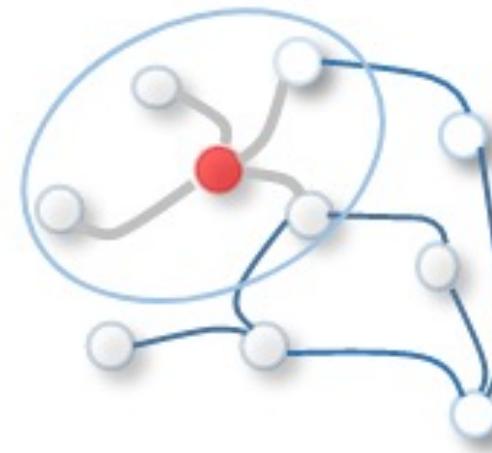


ConvNets en grafos y dominios irregulares

Convolución 2D clásica

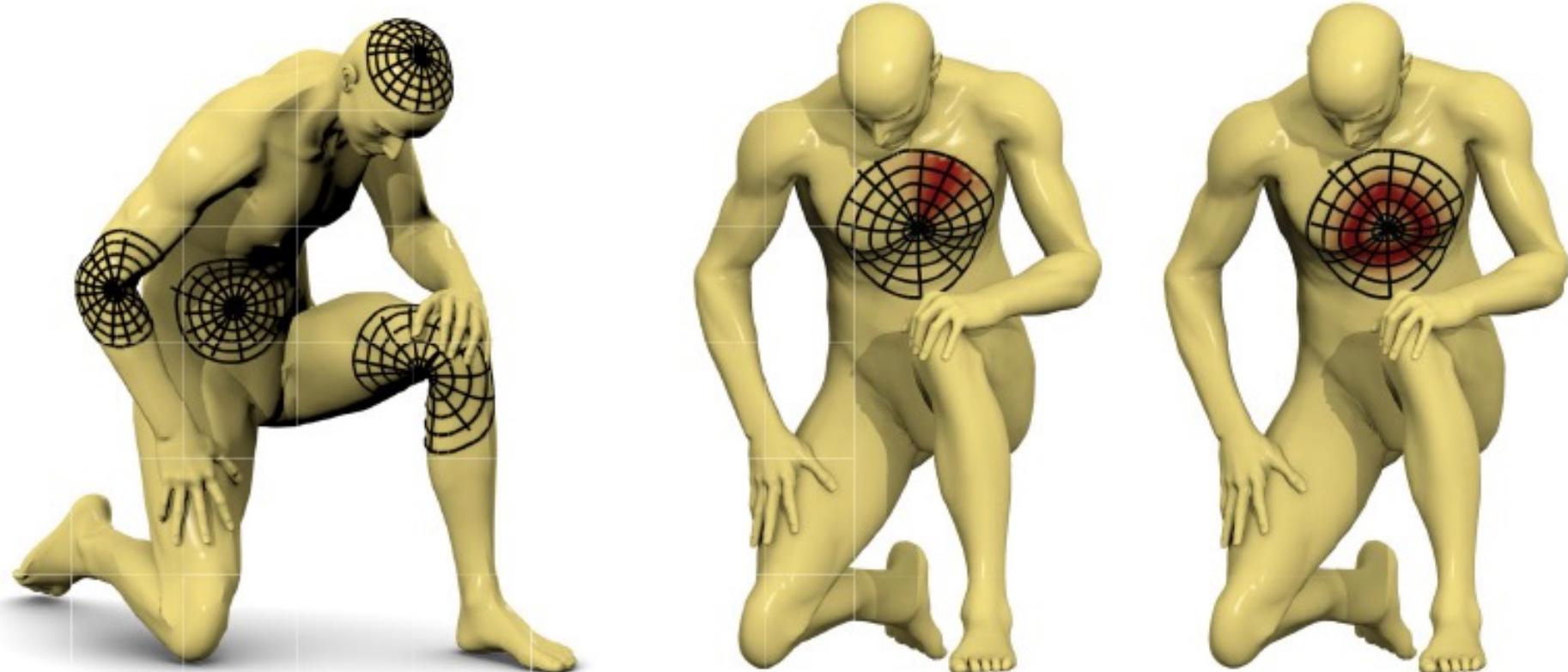


Convolución en grafos



"A Comprehensive Survey on Graph Neural Networks" Wu et al, 2019

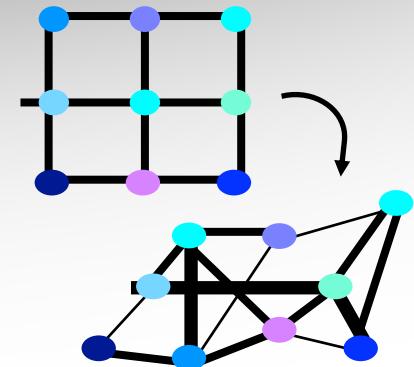
ConvNets en grafos y dominios irregulares



ConvNets en grafos

Geometric deep learning:
going beyond Euclidean data

Michael M. Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, Pierre Vandergheynst



- **Filtrado espacial**: La convolución se define directamente en el dominio original del grafo
 - ✓ Interpretación intuitiva y analogía simple con convoluciones en dominios estructurados
 - Requiere definir un sistema de vecindad y un orden para los nodos
- **Filtrado espectral**: Convoluciones en el dominio espacial corresponden a multiplicaciones en el dominio de Fourier
 - ✓ No se requiere la noción de orden entre vecinos
 - ✓ Definición precisa del concepto de convolución por medio del análisis espectral de grafos
 - ✓ Se pueden modelar filtros estrictamente localizados en el grafo
 - Los filtros que se aprenden en una estructura de grafo dada no pueden ser transferidos a otras

Clase 3

Redes Neuronales Convolucionales

Enzo Ferrante

 eferrante@sinc.unl.edu.ar

 @enzoferante

