

# Redes Convolucionales

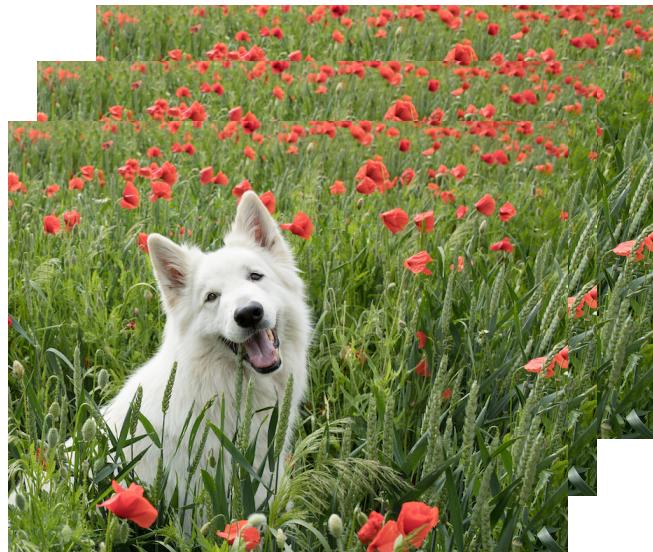
Pooling



Prof. Ricardo Ñanculef - Departamento de Informática UTFSM 2022

# Principios Básicos

- Las capas de una red convolucional serán capaces de **patrones locales en grandes arreglos multi-dimensionales de datos.** Esto se logrará mediante 3 principios básicos de diseño:

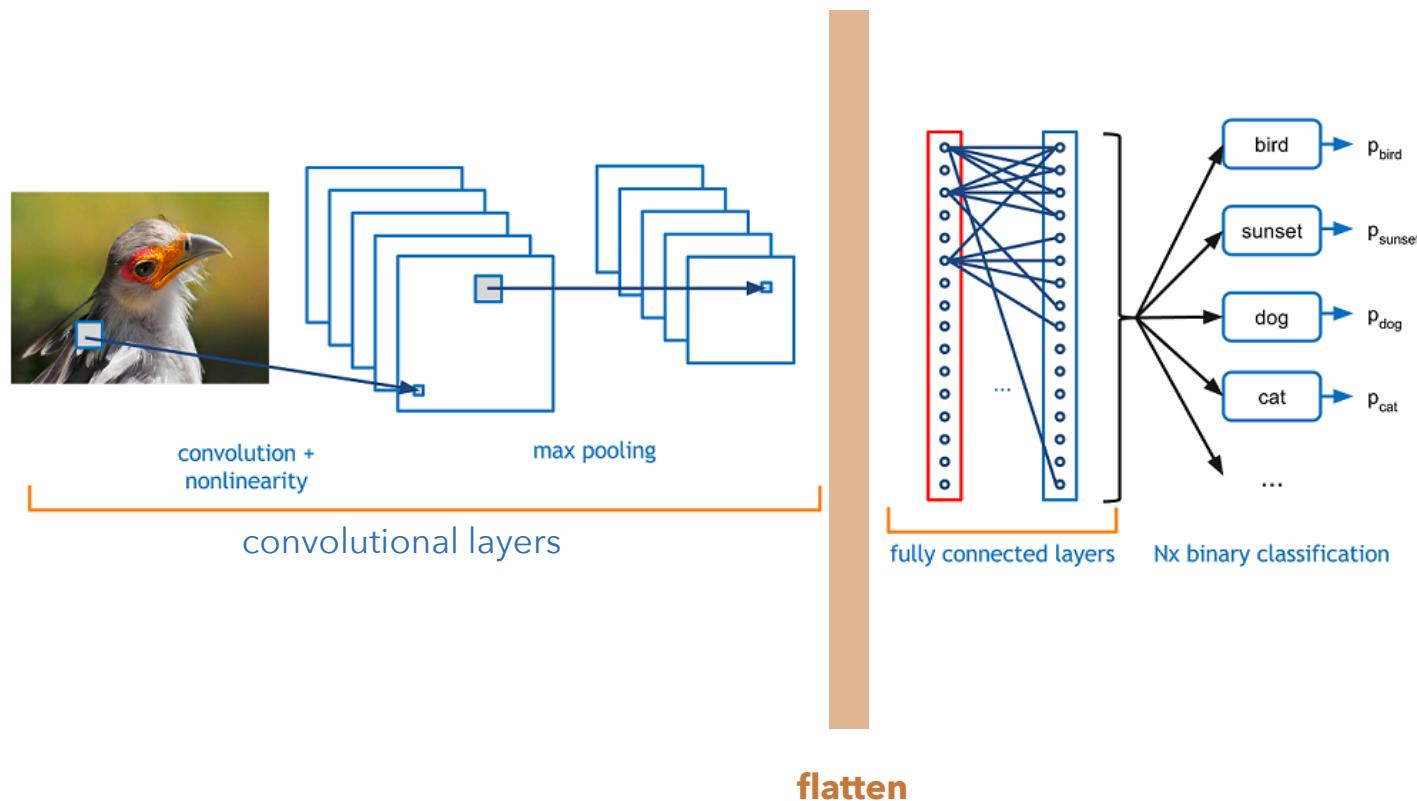


- **Conectividad local.**
- Compartición de pesos.
- Pooling.

- A esto sumaremos un principio transversal de diseño: mantener la topología de los datos a medida que se propaga por la red.

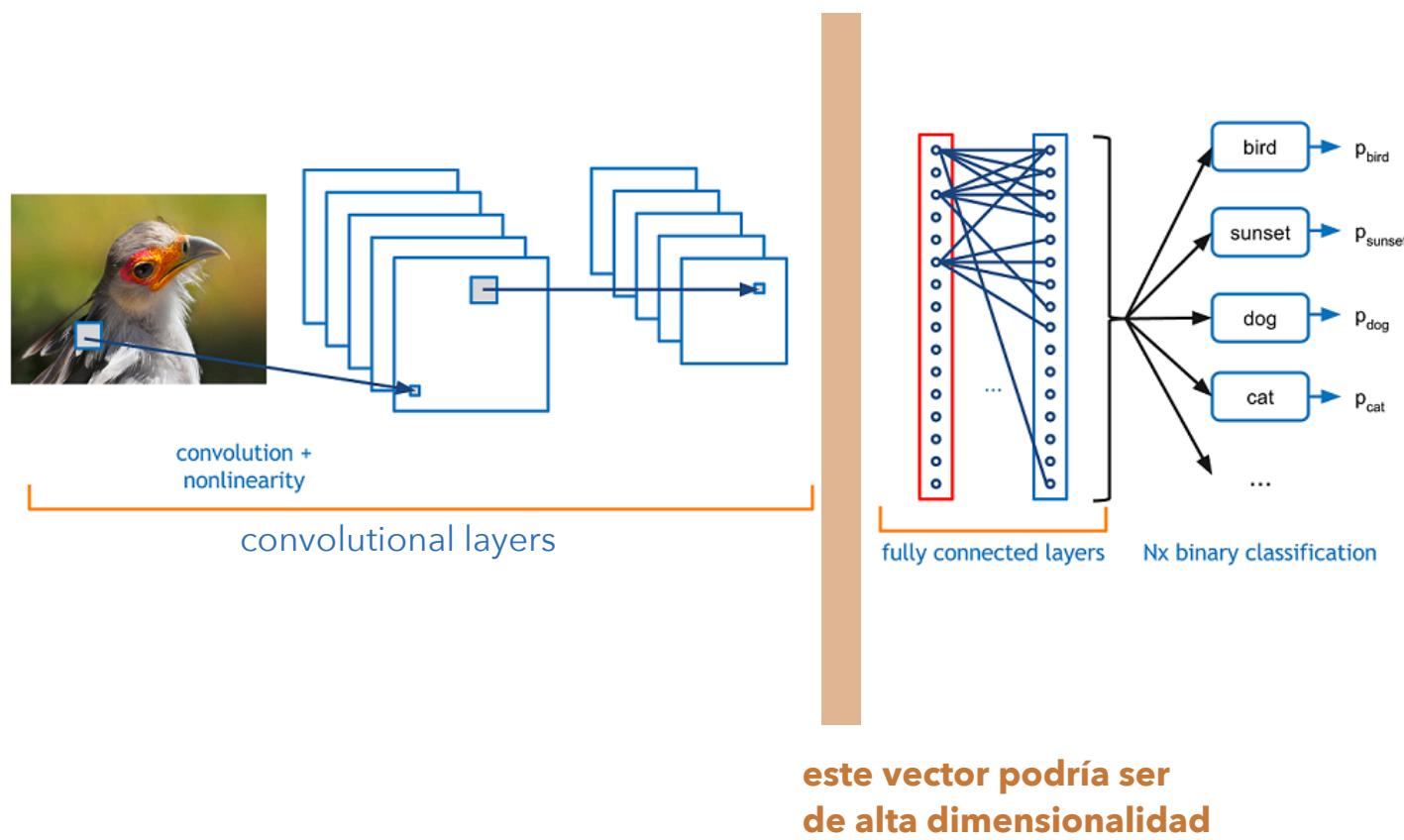
# Capas Convolucionales

- Después de apilar un determinado número de capas convolucionales, tendremos características suficientemente potentes como para alimentar una red feed-forward densa.



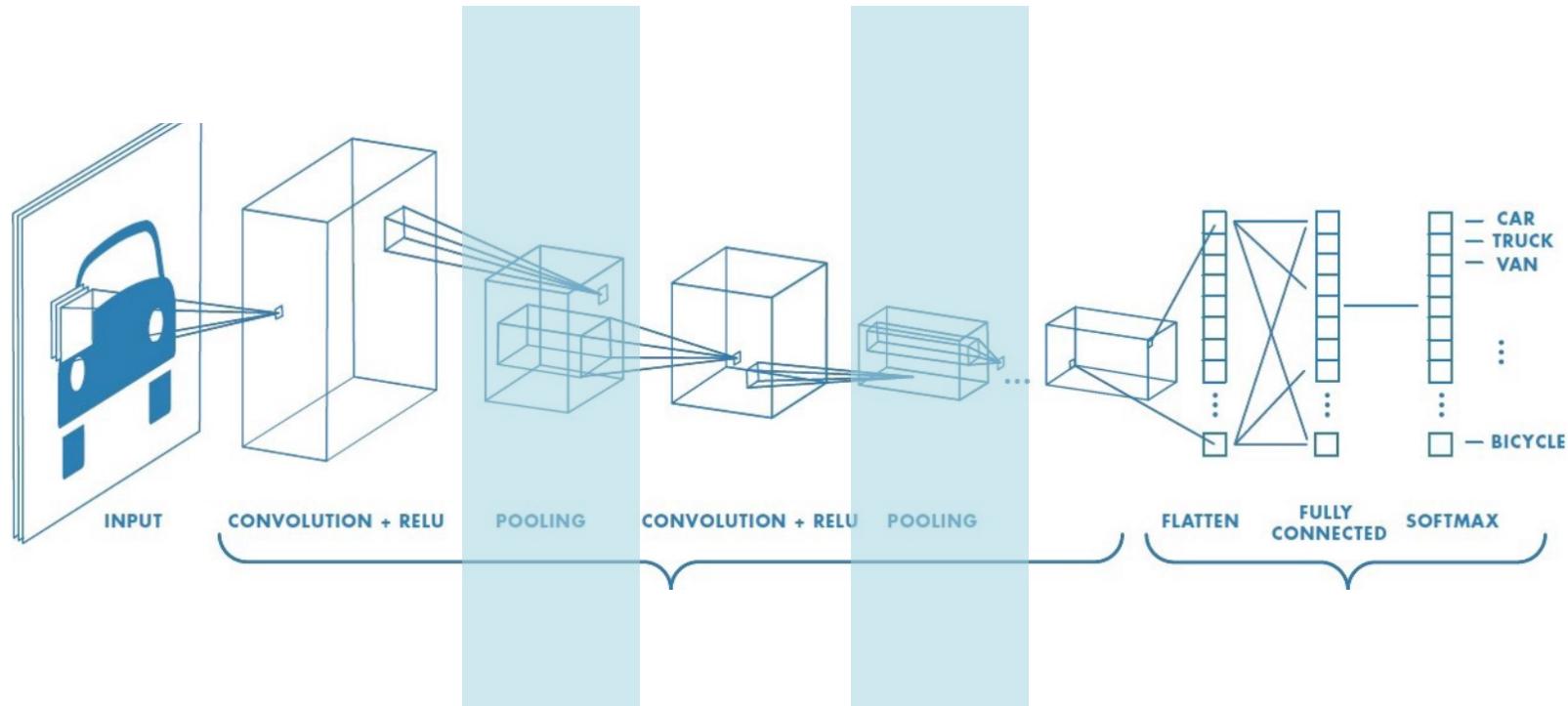
# Capas Convolucionales

- Lamentablemente, aunque la conectividad local reduce el número de parámetros enmendables de las capas, no reduce significativamente la dimensionalidad del volumen de datos. Esto puede causar problemas al momento de conectar una capa densa como una softmax.



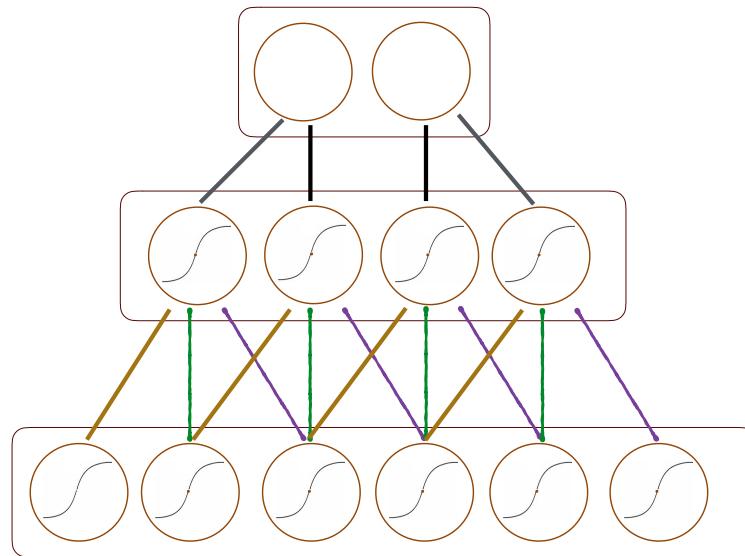
# Pooling

- Aparece así la última idea clave en el diseño de una red convolucional: la capa de pooling. El objetivo de esta capa será reducir la dimensionalidad del volumen entrante y típicamente se utilizará después de 1 o más capas convolucionales.



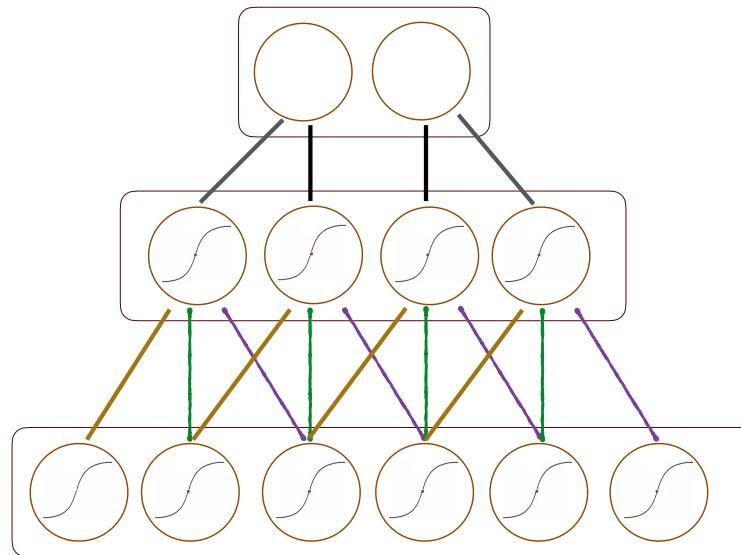
# Pooling

- La conectividad de la capa quedará definida especificando la forma del campo receptivo y del stride, exactamente como en el caso convolucional.



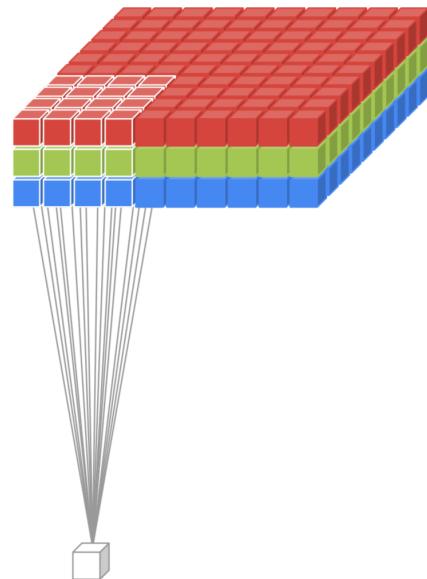
# Pooling

- Sin embargo, el valor del stride será determinado para reducir la dimensionalidad del volumen entrante (típicamente 2 en cada dimensión).



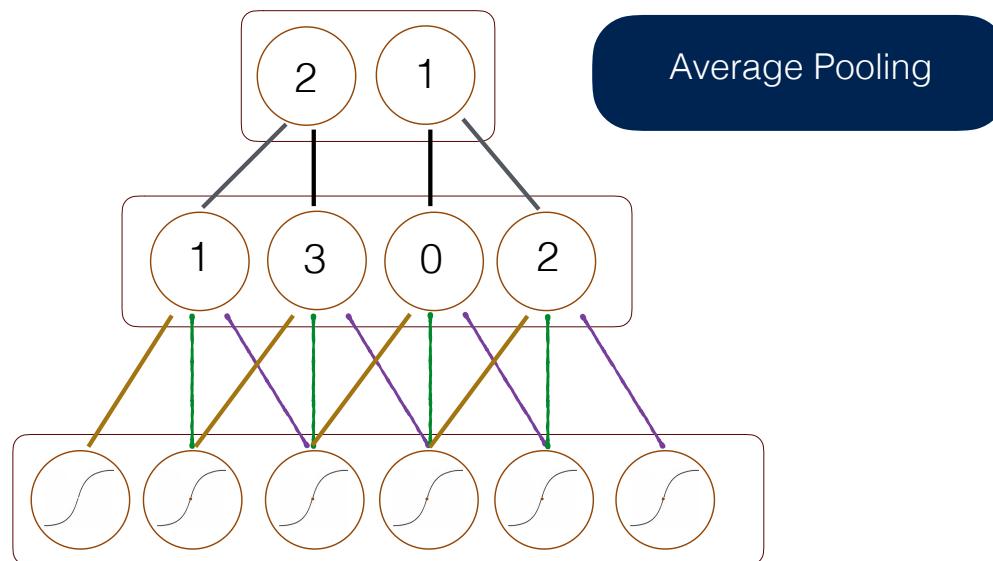
# Pooling 2D

- Usualmente, las capas de pooling serán definidas de modo consistente con la forma de las capas convolucionales que la precedan: si se trata de capas 2D usaremos un campo receptivo 2D, si se trata de capas 1D usaremos un campo receptivo 1D, etc.



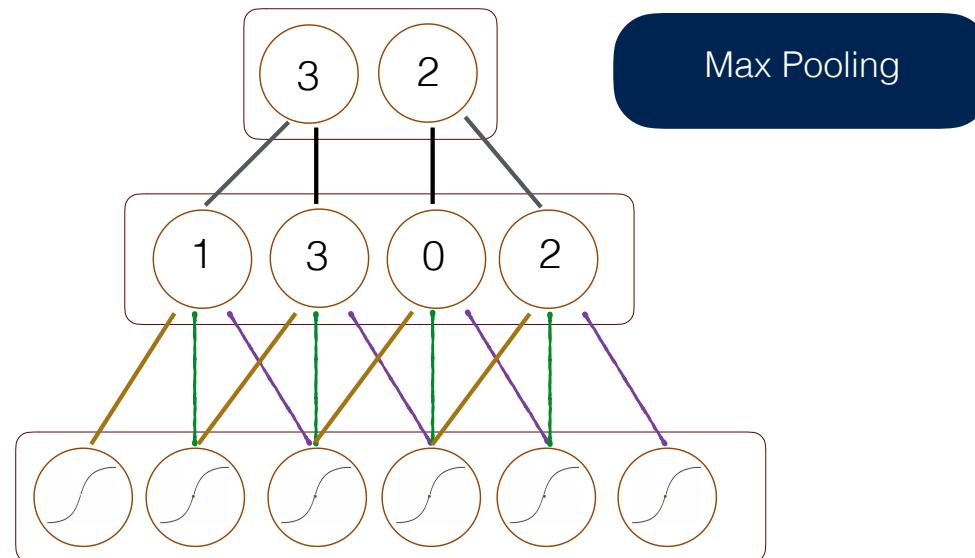
# Average Pooling

- A diferencia de las neuronas convolucionales, la operación que las neuronas de pooling aplicarán sobre su campo receptivo es reemplazar los valores allí observados por un único valor, utilizando una estadística sencilla como la **media** ...



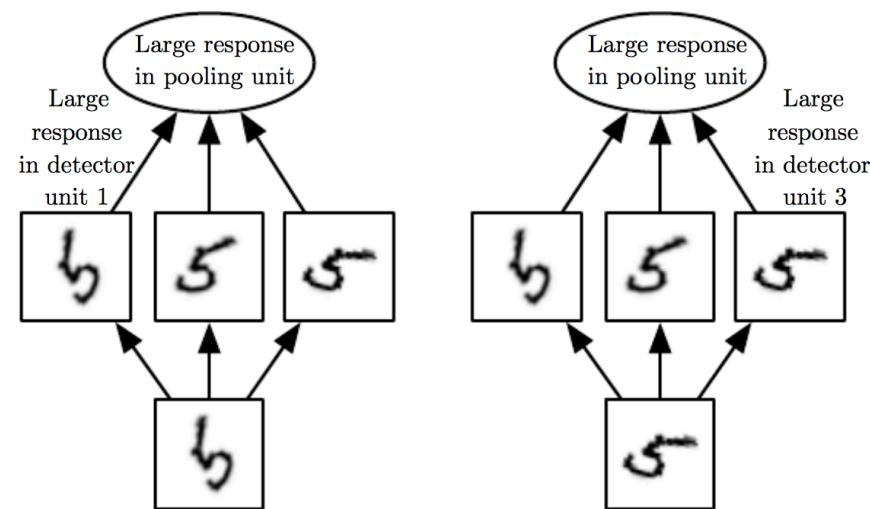
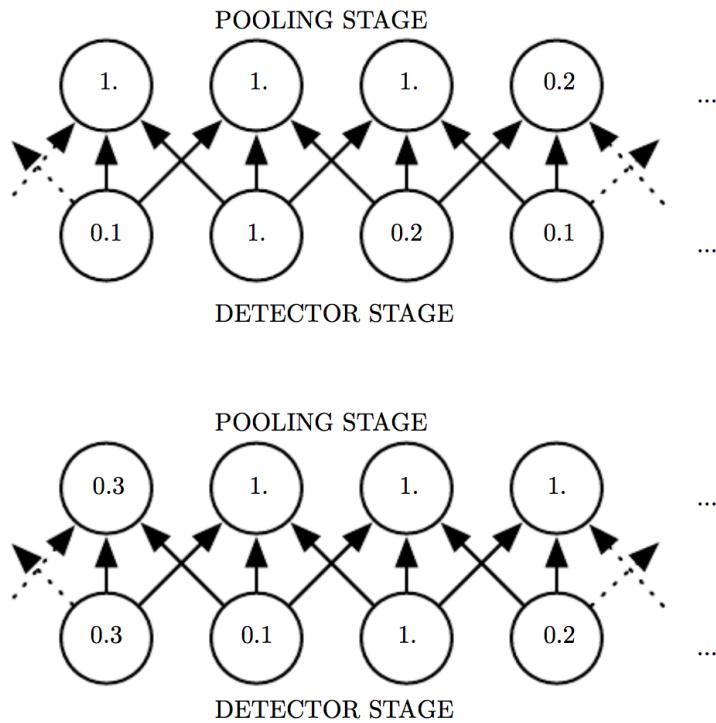
# Max Pooling

- ... o más frecuentemente el **máximo**.



# Max Pooling

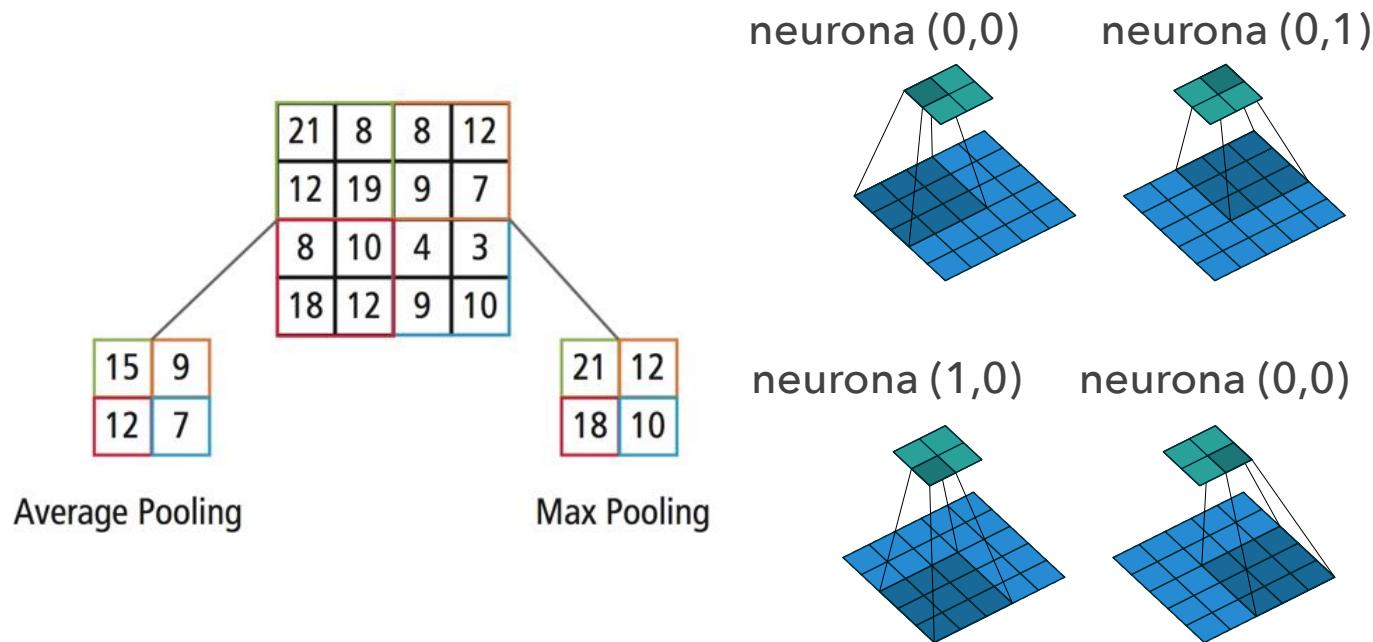
- En la práctica, la operación de max pooling otorga a la red la capacidad de ser **robusta frente a variaciones de la posición de un patrón (invarianza traslacional)**.



Deep Learning (2006). Ian Goodfellow, Yoshua Bengio and Aaron Courville. MIT Press.

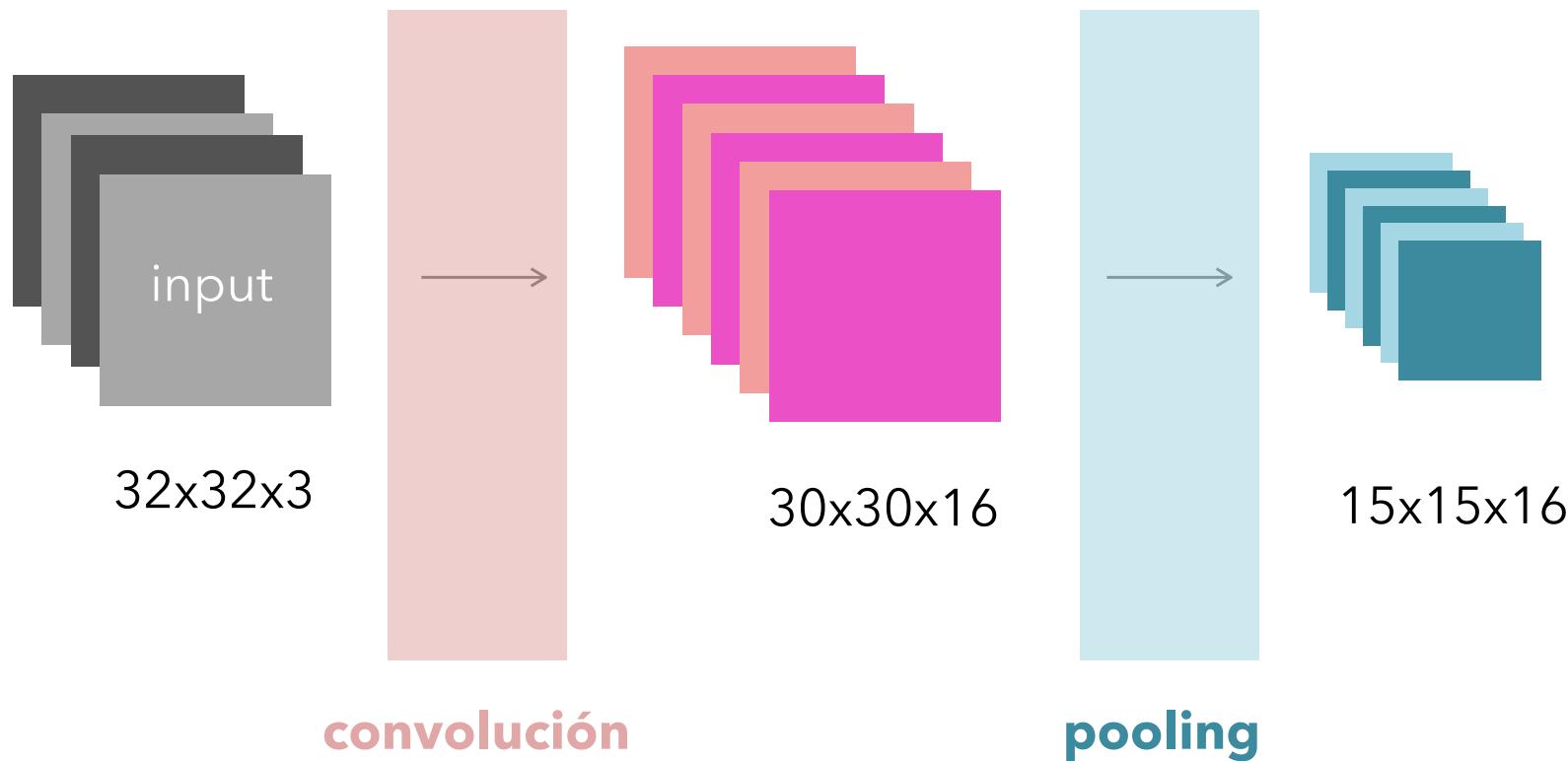
# Pooling 2D

- Max y Average Pooling 2D:



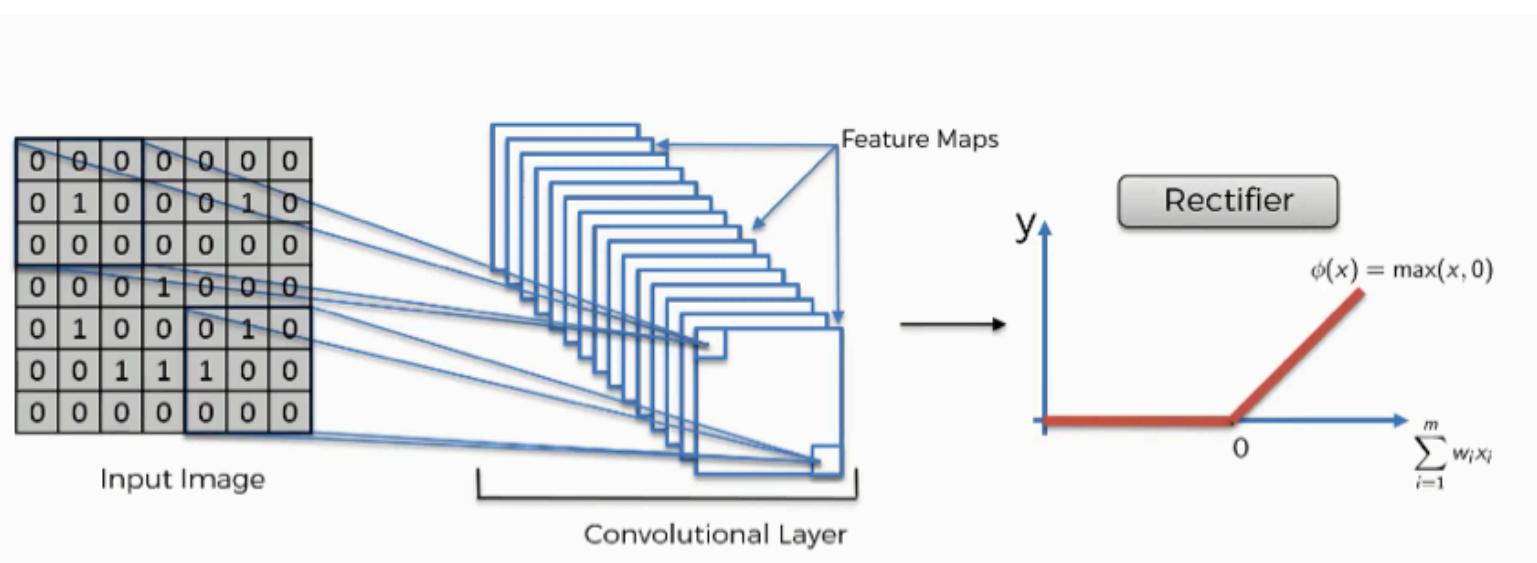
# Pooling 2D

- En una capa de pooling estándar, la operación de muestreo se aplica **a cada canal de entrada de manera independiente**, de modo que se preserve el número de canales de la capa convolucional anterior.



# Pooling 2D

- Normalmente la operación de pooling (max, promedio, etc) **no está parametrizada** (no es aprendible) y **no se acompaña de una función de activación no-lineal** como sí ocurre en las capas convolucionales ordinarias.



- Como consecuencia, una capa de pooling tiene sólo hyper-parámetros (no aprendibles): tamaño, stride y función a aplicar.

# Pooling vía Capas Convolucionales

- Una forma (durante un tiempo popular) de parametrizar una capa de pooling con campo receptivo  $S_i \times S_j$  y stride unitario es implementarla usando capas convolucionales con campo receptivo  $S_i \times S_j$  y stride  $S_i \times S_j$ .

## STRIVING FOR SIMPLICITY: THE ALL CONVOLUTIONAL NET

**Jost Tobias Springenberg\***, **Alexey Dosovitskiy\***, **Thomas Brox**, **Martin Riedmiller**

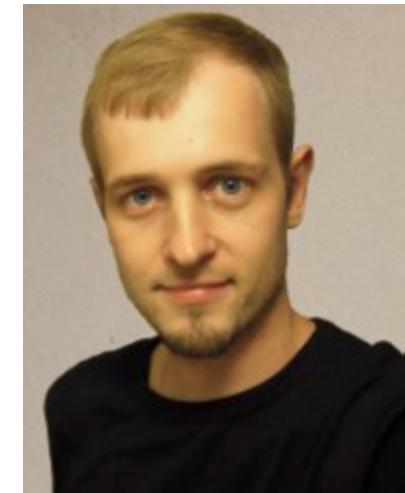
Department of Computer Science

University of Freiburg

Freiburg, 79110, Germany

{springj, dosovits, brox, riedmiller}@cs.uni-freiburg.de

Model		
Strided-CNN-C	ConvPool-CNN-C	All-CNN-C
Input $32 \times 32$ RGB image		
$3 \times 3$ conv. 96 ReLU	$3 \times 3$ conv. 96 ReLU	$3 \times 3$ conv. 96 ReLU
$3 \times 3$ conv. 96 ReLU with stride $r = 2$	$3 \times 3$ conv. 96 ReLU	$3 \times 3$ conv. 96 ReLU
	$3 \times 3$ max-pooling stride 2	$3 \times 3$ conv. 96 ReLU with stride $r = 2$
$3 \times 3$ conv. 192 ReLU	$3 \times 3$ conv. 192 ReLU	$3 \times 3$ conv. 192 ReLU
$3 \times 3$ conv. 192 ReLU with stride $r = 2$	$3 \times 3$ conv. 192 ReLU	$3 \times 3$ conv. 192 ReLU
	$3 \times 3$ max-pooling stride 2	$3 \times 3$ conv. 192 ReLU with stride $r = 2$



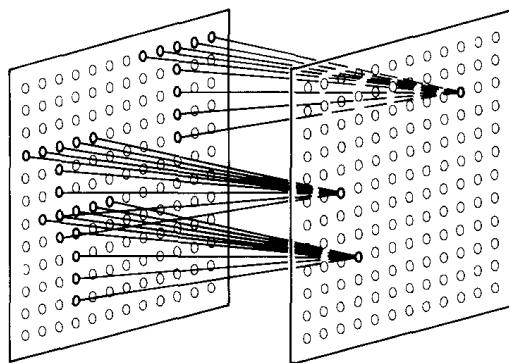
# Un Poco de Historia

- La idea de una arquitectura de red basada en conectividad local, que imitara la habilidad del sistema visual para procesar estímulos, había sido ya propuesta por K. Fukushima en 1980 como parte de un modelo denominado **Neo-Cognitrón**.

## **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

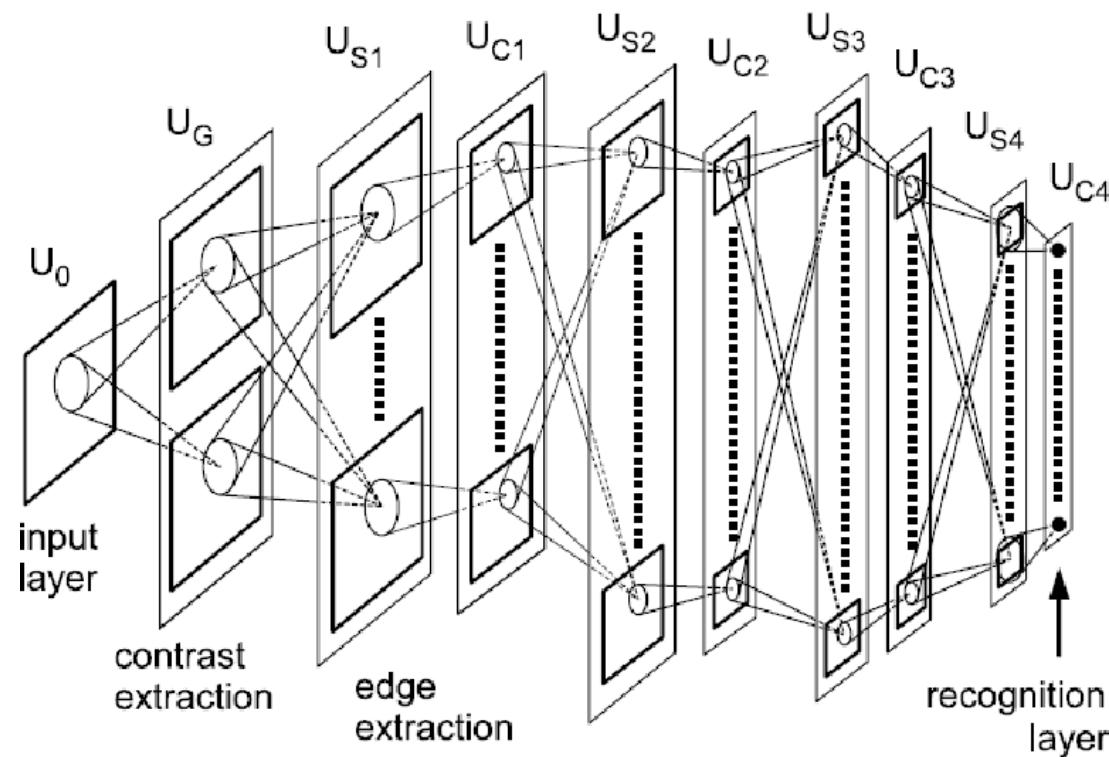


**Fig. 3.** Illustration showing the input interconnections to the cells within a single cell-plane



# Un Poco de Historia

- **Neo-Cognitrón (1980)**



# Un Poco de Historia

- La combinación de esta idea con un método (back-propagation) que hiciese las capas *aprendibles desde ejemplos* de manera simple y eficiente se produjo sólo hacia fines de los 90' con la introducción de **LeNet5** y su demostración experimental en el **dataset MNIST**.

PROC. OF THE IEEE, NOVEMBER 1998

1

## Gradient-Based Learning Applied to Document Recognition

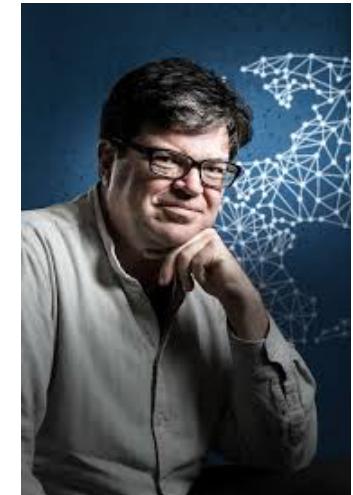
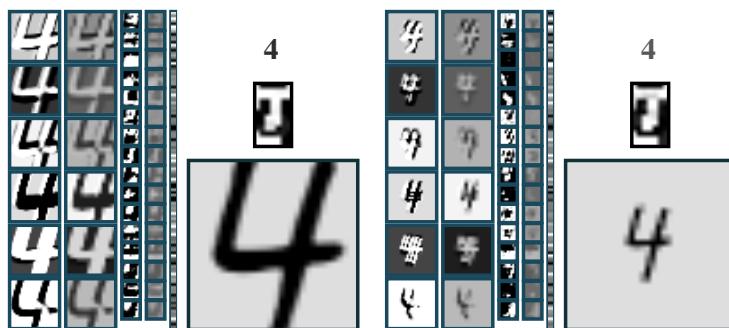
Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner

### Abstract—

Multilayer Neural Networks trained with the backpropagation 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, that are specifically designed to deal with the variability of 2D shapes, are

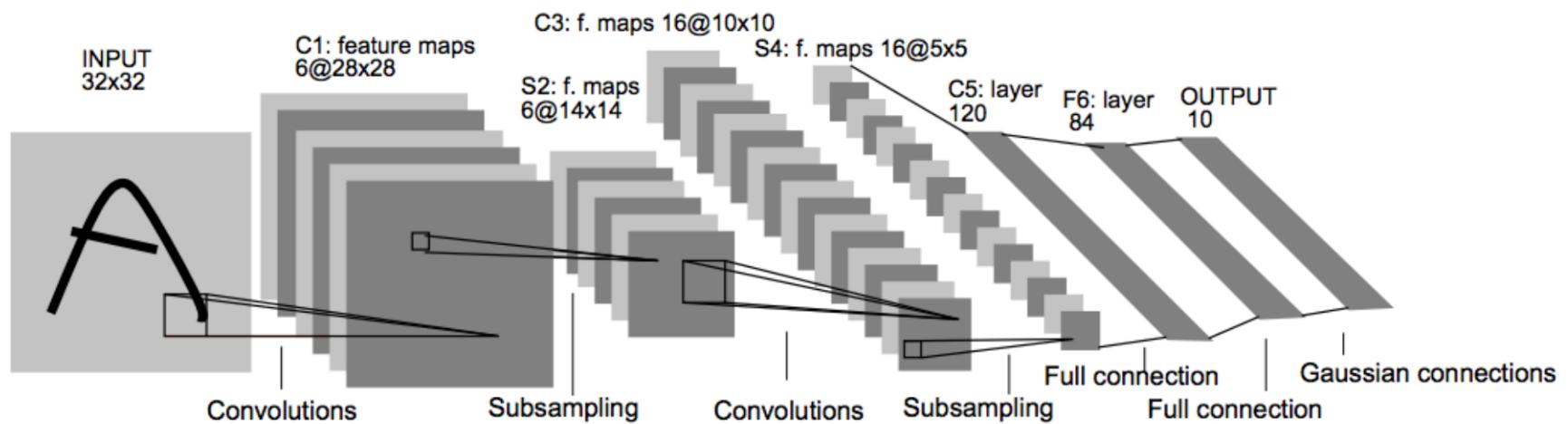
### I. INTRODUCTION

Over the last several years, machine learning techniques, particularly when applied to neural networks, have played an increasingly important role in the design of pattern recognition systems. In fact, it could be argued that the availability of learning techniques has been a crucial factor in the recent success of pattern recognition applications such as continuous speech recognition and handwriting recognition.



# Un Poco de Historia

- **LeNet5 (1998):**

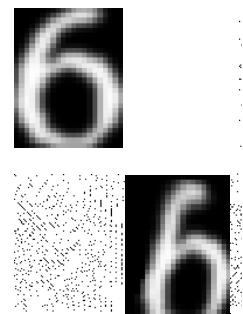
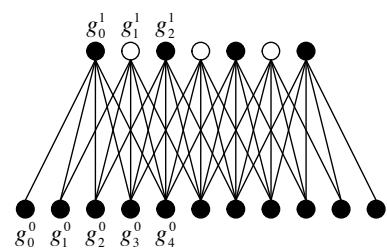


# Un Poco de Historia

- La arquitectura siguió siendo investigada por algunos grupos sin ser adoptada masivamente ni en la literatura ni en la práctica, muy probablemente por el costo computacional necesario para entrenarlas en la práctica. Por ejemplo, hacia 2003 Simard propone modificaciones al método de entrenamiento de LeNet5 que la hacen más liviana computacionalmente.

## Best Practices for Convolutional Neural Networks Applied to Visual Document Analysis

Patrice Y. Simard, Dave Steinkraus, John C. Platt  
*Microsoft Research, One Microsoft Way, Redmond WA 98052*  
*{patrice,v-davste,jplatt}@microsoft.com*



# Un Poco de Historia

- El uso de la arquitectura sólo se masificó hacia 2012, cuando Alex Krizhevsky ganó la versión de ese año del ImageNet challenge con un modelo, hoy famoso, denominado **AlexNet**.

---

## 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

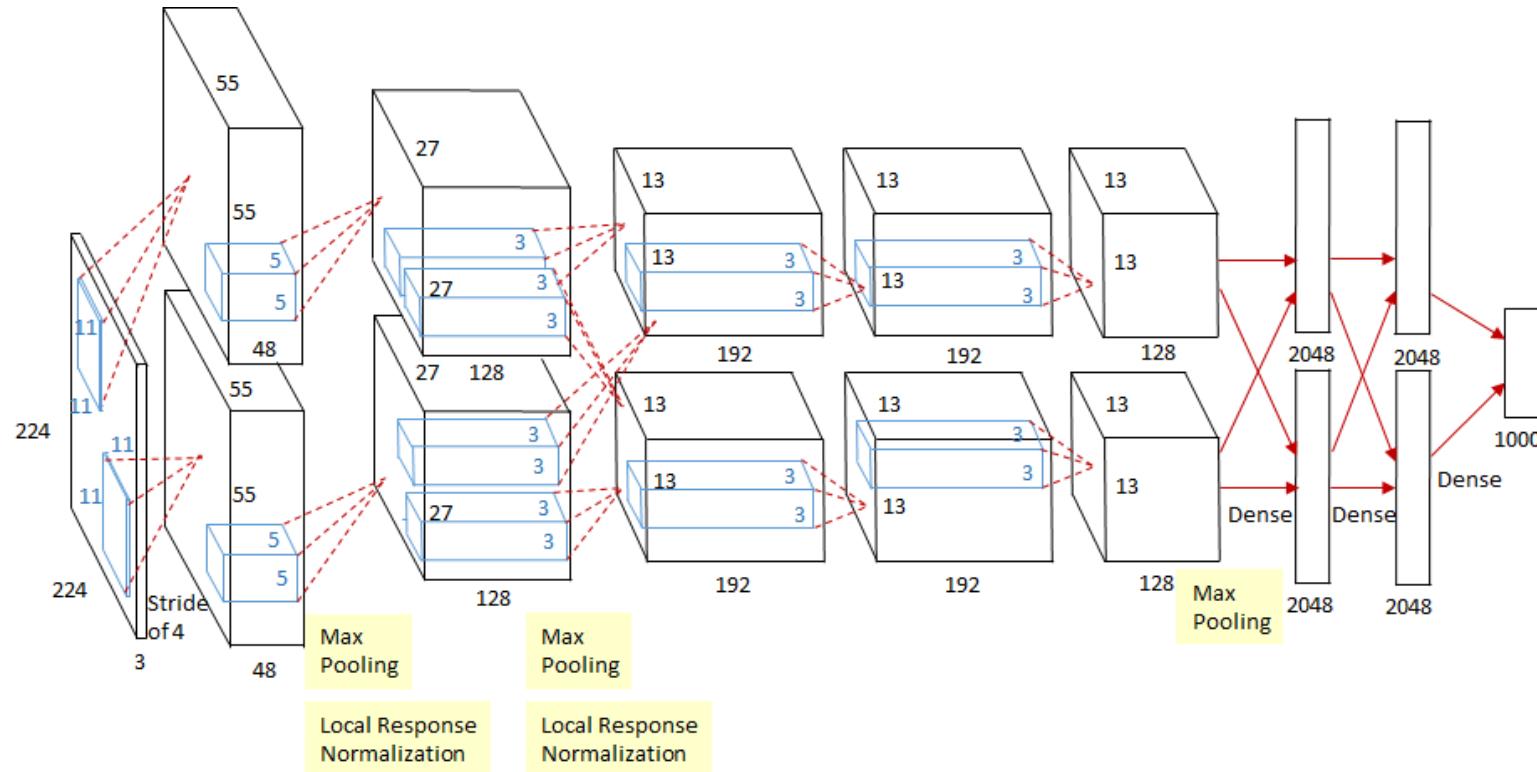
### Abstract

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%, compared to 26.2% achieved by the second-best entry.



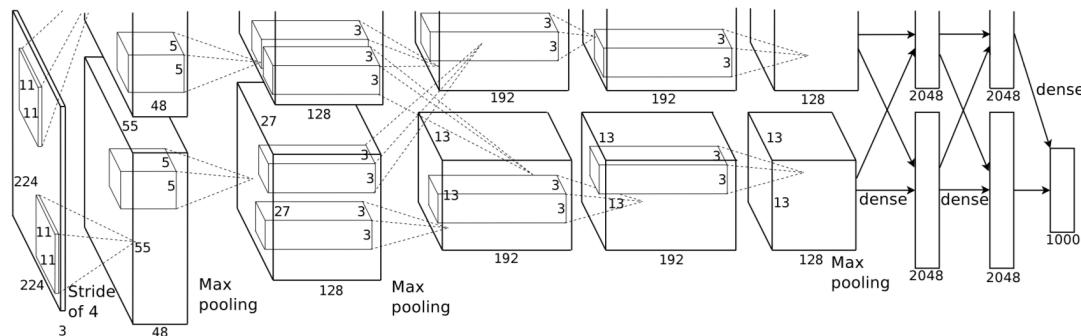
# Un Poco de Historia

- **AlexNet (2012):**



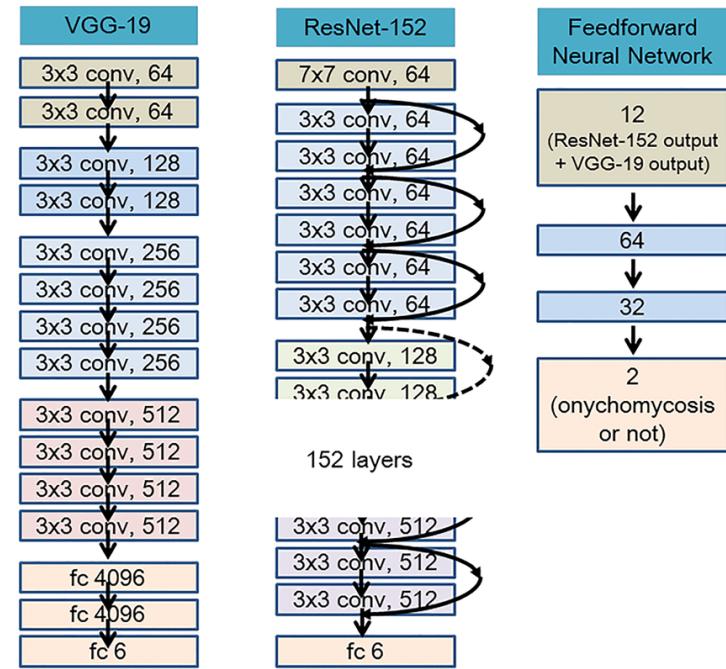
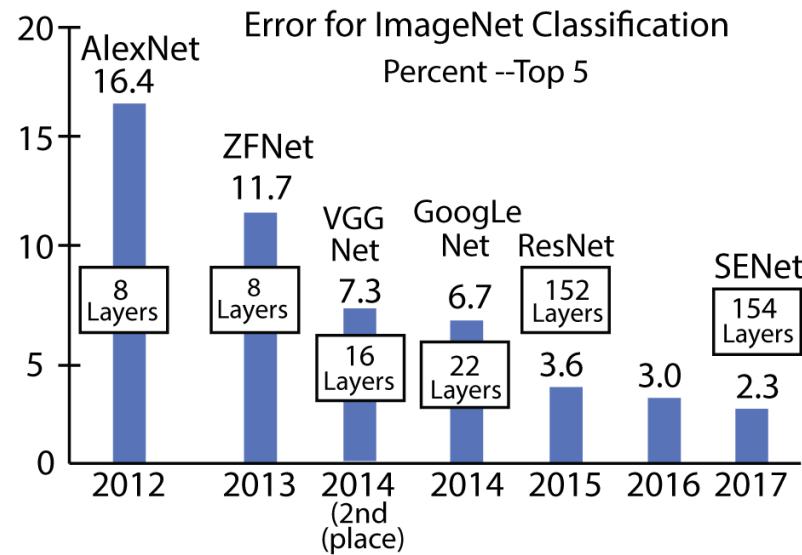
# Un Poco de Historia

- El modelo tenía 60 millones de parámetros, 650.000 neuronas y fue entrenado sobre 15 millones de imágenes de 256x256 pixeles (65536 dimensiones de entrada), pertenecientes a más de 22.000 categorías. Eso sólo fue posible gracias al uso de GPUs (GTX 580), algo relativamente novedoso para la época .



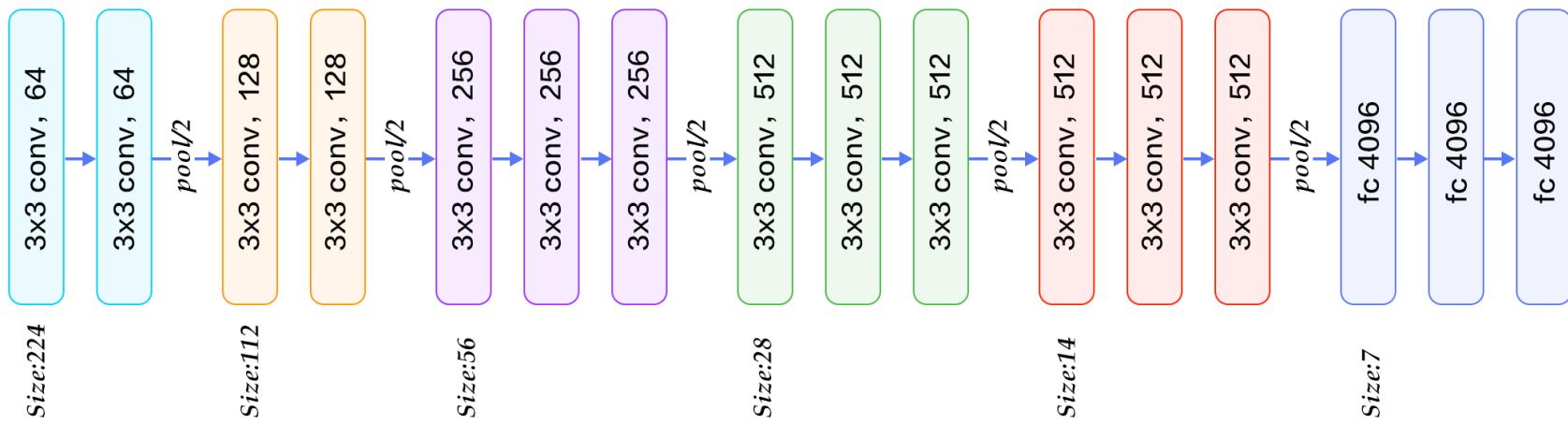
# Un Poco de Historia

- Desde ese momento, el uso de redes convencionales en problemas de visión artificial es sumamente masivo.



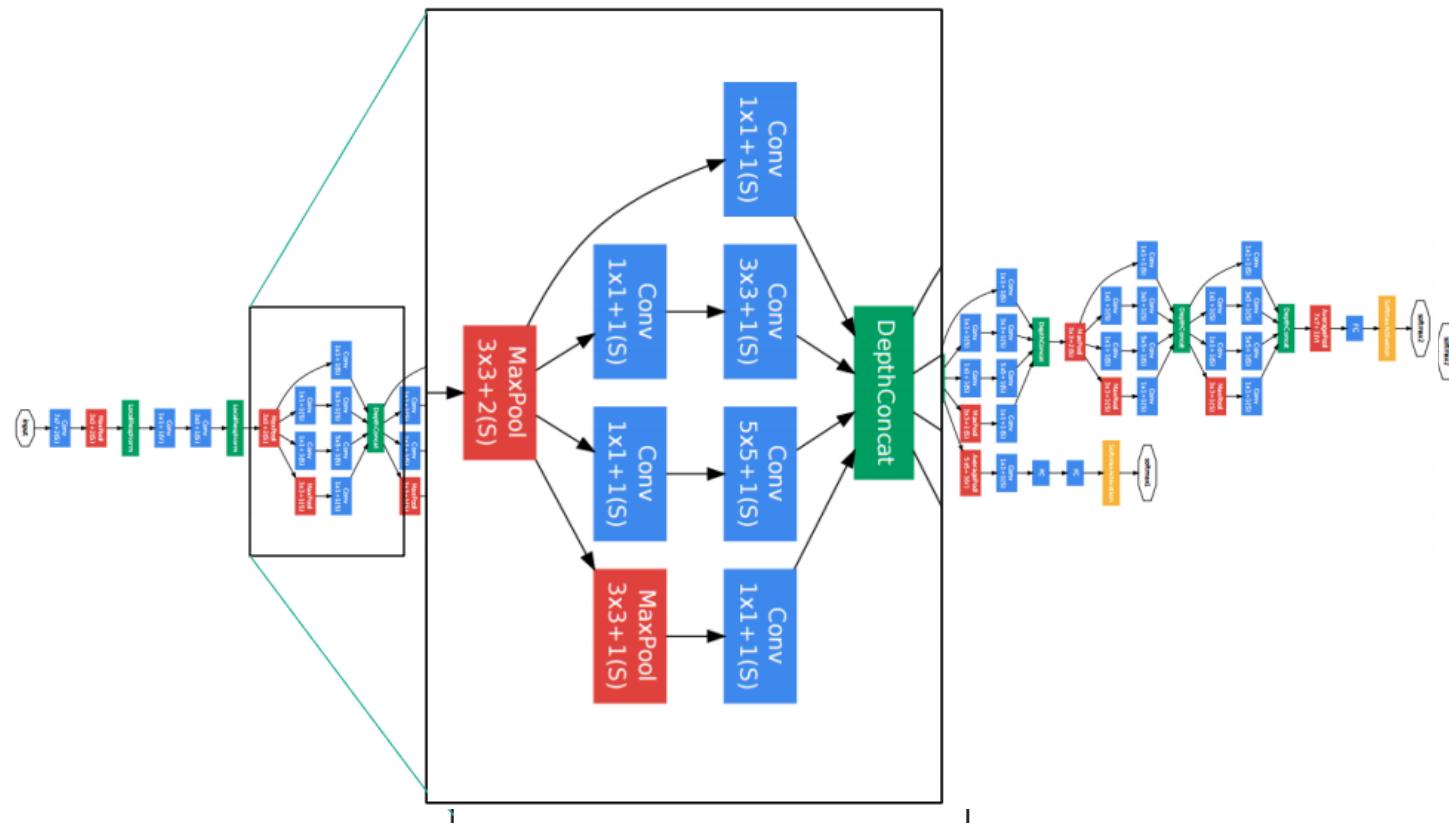
# Un Poco de Historia

- **VGGNet (2014):**



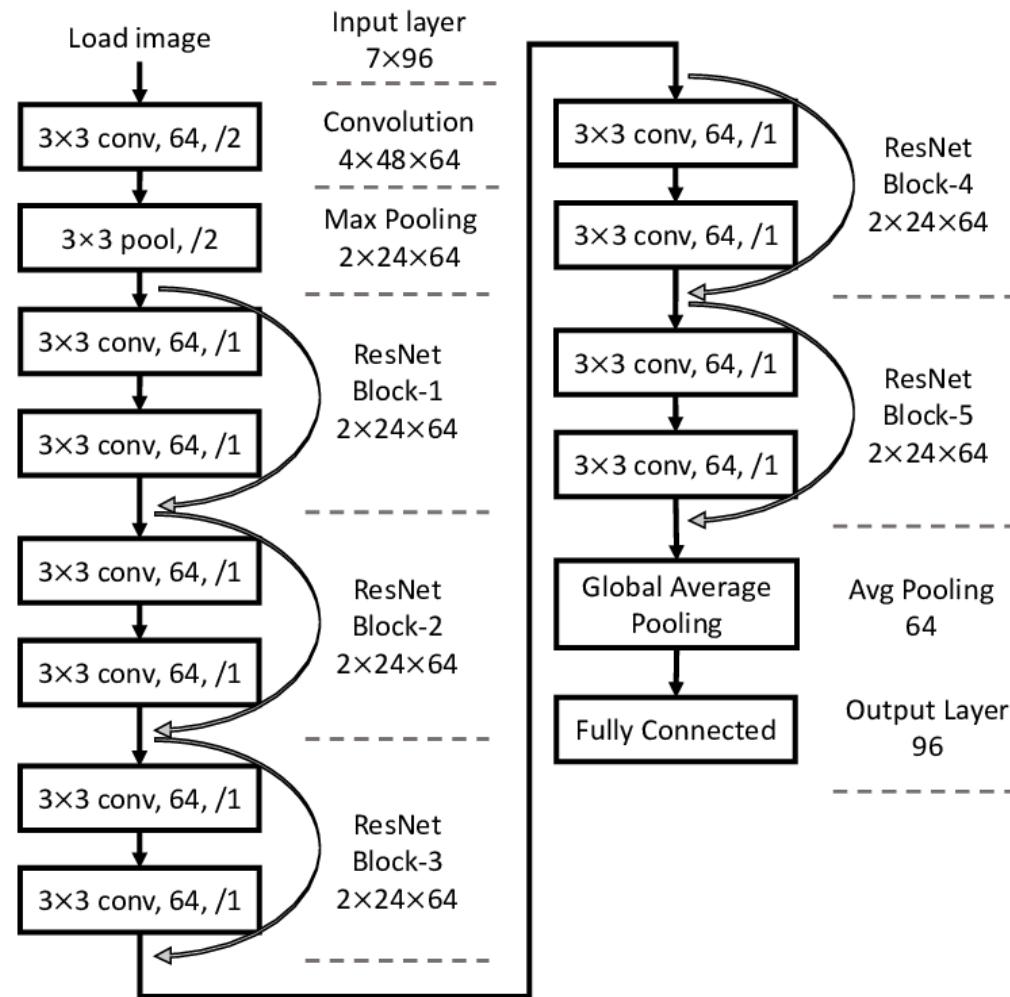
# Un Poco de Historia

- **GoogleNet (2014):**



# Un Poco de Historia

- **ResNet (2015):**



# Entonces ...

- Una red convolucional (CNN) se construye sobre 3 ideas fundamentales: conectividad limitada, compartición de pesos y agresiva reducción de dimensionalidad vía pooling.
- La capa de pooling tiene el objetivo de reducir la dimensionalidad del volumen entrante y de favorecer la invarianza de la red ante traslaciones de patrones locales en los datos de entrada.
- Típica la capa no es entrenable, siendo sus únicos (hyper) parámetros: el tamaño de los campos receptivos, el stride y la función a aplicar.
- El patrón típico de composición con capas convolucionales es CPCPCP .. o bien CCPCCP ... pero naturalmente no se trata de una regla rígida.

