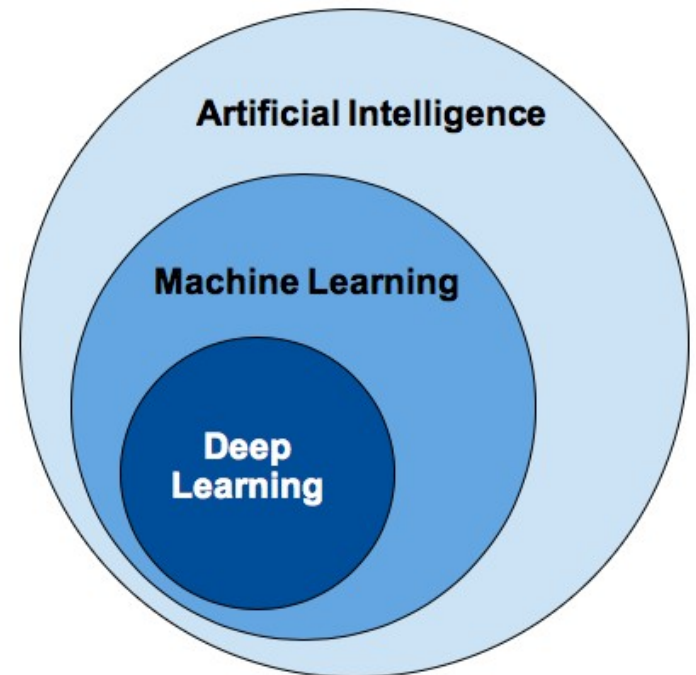


Redes profundas y redes convolucionales

Basado en
charlas de
Lucas C. Uzal y
otros



“Revolution of Depth”

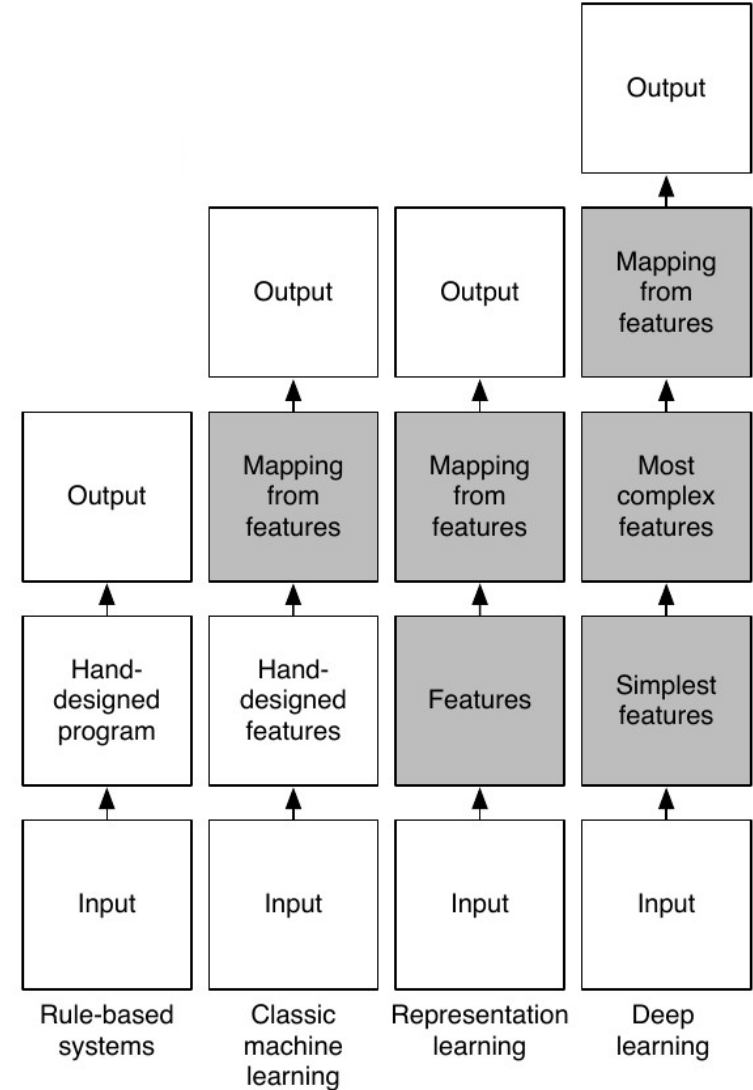
Situación a mitad de los “2000”

- Machine Learning está establecido como campo de investigación con alto potencial tecnológico
- Las SVM dominan el campo
- Tipo de solución basado en el conocimiento de expertos:
 - Medir cosas (features) sobre los datos, sugeridas por conocimiento del problema
 - Aplicar un método de ML a esos features para decidir
- Las redes neuronales están en un “invierno”:
 - Se sabe que las redes neuronales son aproximadores universales.
 - Se sabe que las redes tienen el potencial de aprender su propia representación.
 - Problemas irresolubles: limitaciones para resolver problemas reales, dificultad para utilizarlas, falta de garantías de convergencia, lentitud de los entrenamientos

“Revolution of Depth”

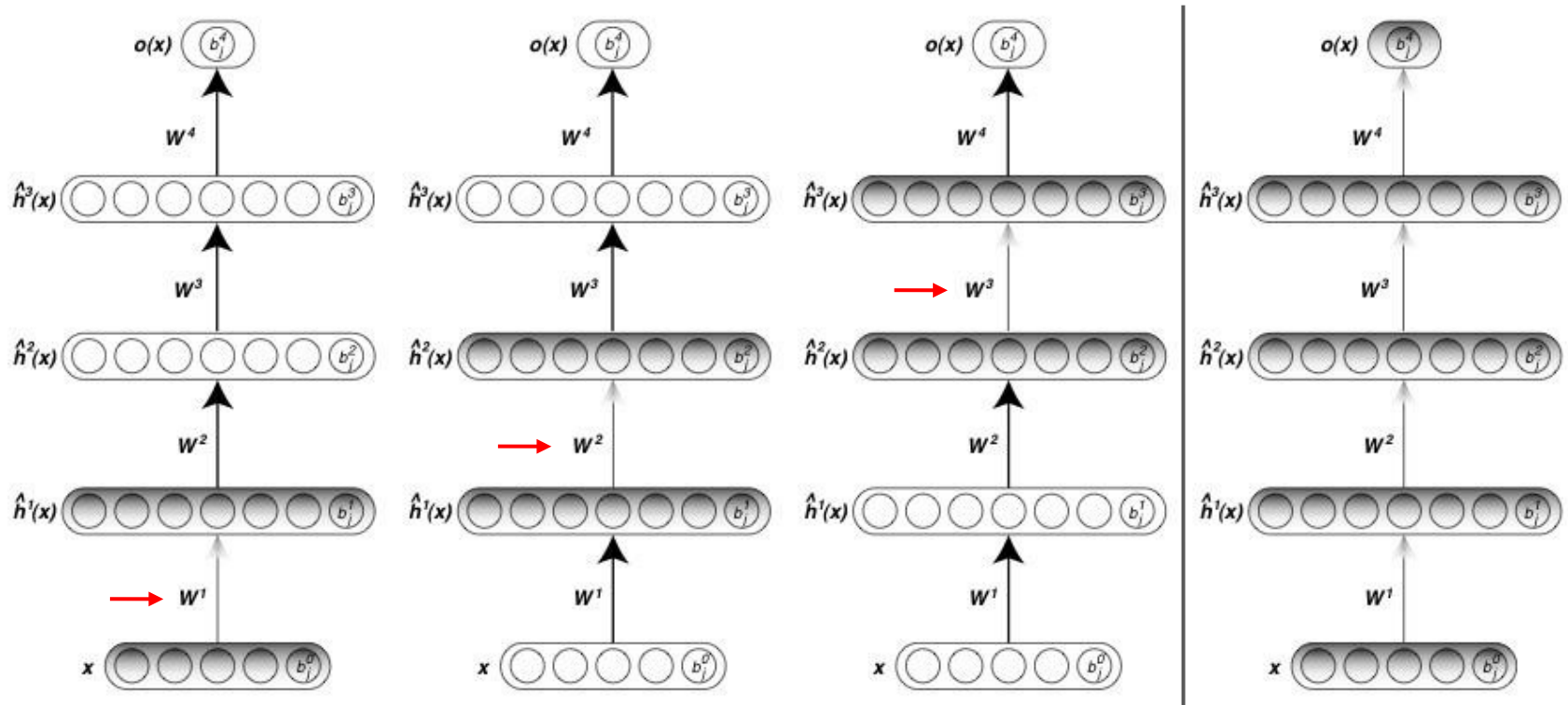
Por qué no seguir el mismo camino?

- “Aprender” las reglas produjo sistemas que solucionan las deficiencias de los sistemas expertos tradicionales.
- “Aprender” los features podría solucionar los problemas actuales.
- Si los métodos actuales no aprenden bien, por qué no usar métodos que puedan ir de lo simple a lo complejo?



No se sabía cómo hacer que las redes aprendan de esta forma!

2006: Greedy layer-wise training of deep networks



Hinton, Geoffrey E, and Ruslan R Salakhutdinov. "Reducing the dimensionality of data with neural networks." *Science* 313.5786 (2006): 504-507.

Hinton, Geoffrey, Simon Osindero, and Yee-Whye Teh. "A fast learning algorithm for deep belief nets." *Neural computation* 18.7 (2006): 1527-1554.

Bengio, Yoshua et al. "Greedy layer-wise training of deep networks." *Advances in neural information processing systems* 19 (2007): 153.

“Revolution of Depth”

Primera solución al problema de entrenar una red profunda.

- Hinton muestra que se puede encontrar una primera solución entrenando de manera no supervisada, y afinando la solución (fine tuning) después, pero en redes muy particulares.
- Resurgen las redes como método, y hay un avance exponencial que lleva a la actual revolución tecnológica.
- En pocos años se encuentra como entrenar redes generales, y como extender todo lo que ya se sabía hacer a modelos profundos.

3 claves: Datos (internet), Potencia (GPUs), Métodos (activaciones, regularización).

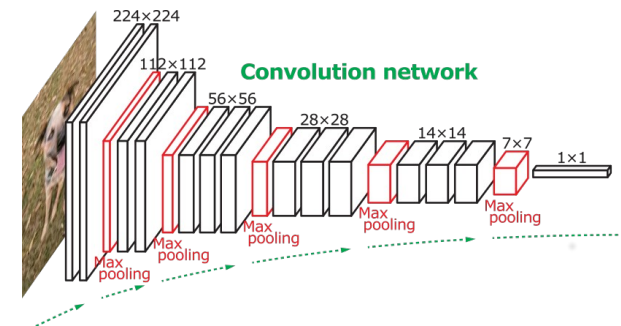
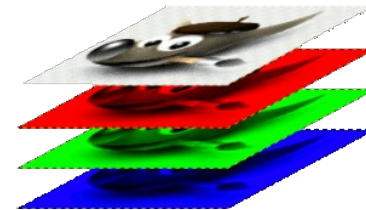
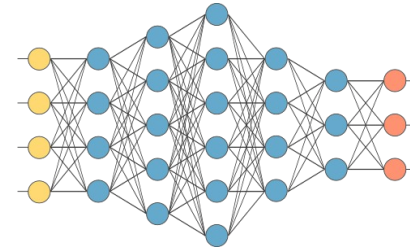
REDES NEURONALES



IMÁGENES



REDES NEURONALES CONVOLUCIONALES



Algo de historia



Primeras redes neuronales

THE 80s



Geoffrey E. Hinton



A164 453

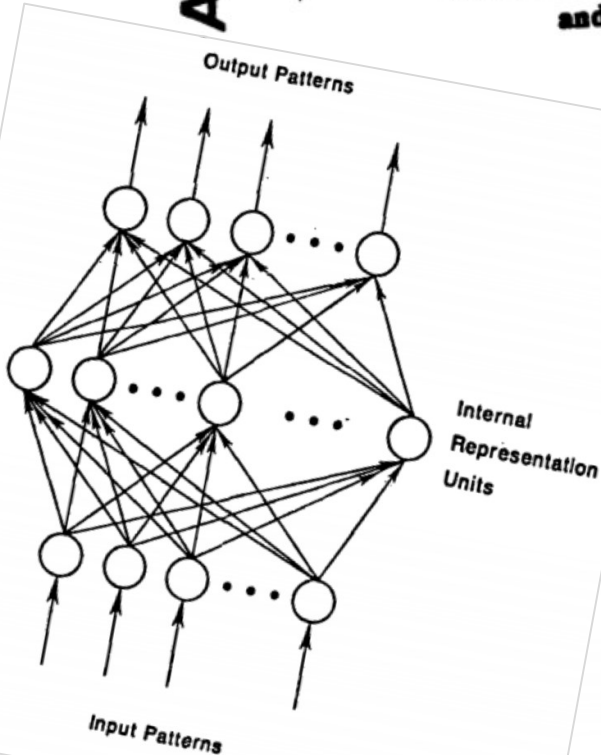
LEARNING INTERNAL REPRESENTATIONS BY ERROR PROPAGATION

David E. Rumelhart, Geoffrey E. Hinton,
and Ronald J. Williams

September 1985

Report 8506

SELF

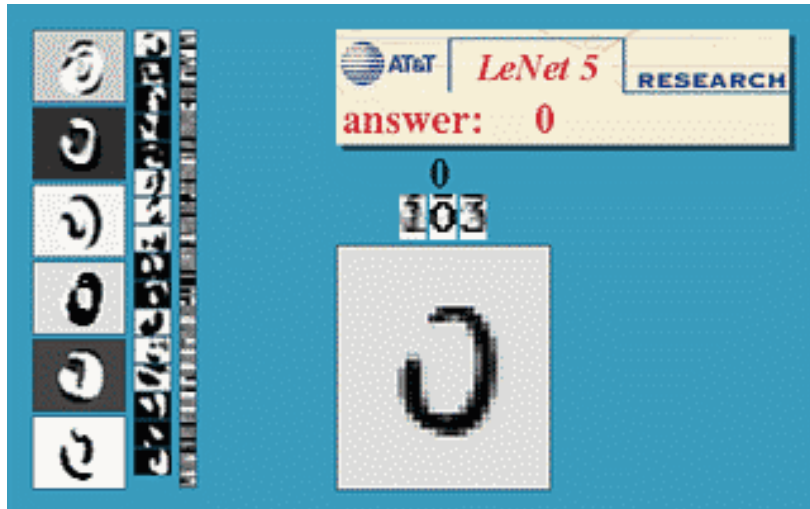


COGNITIVE
SCIENCE

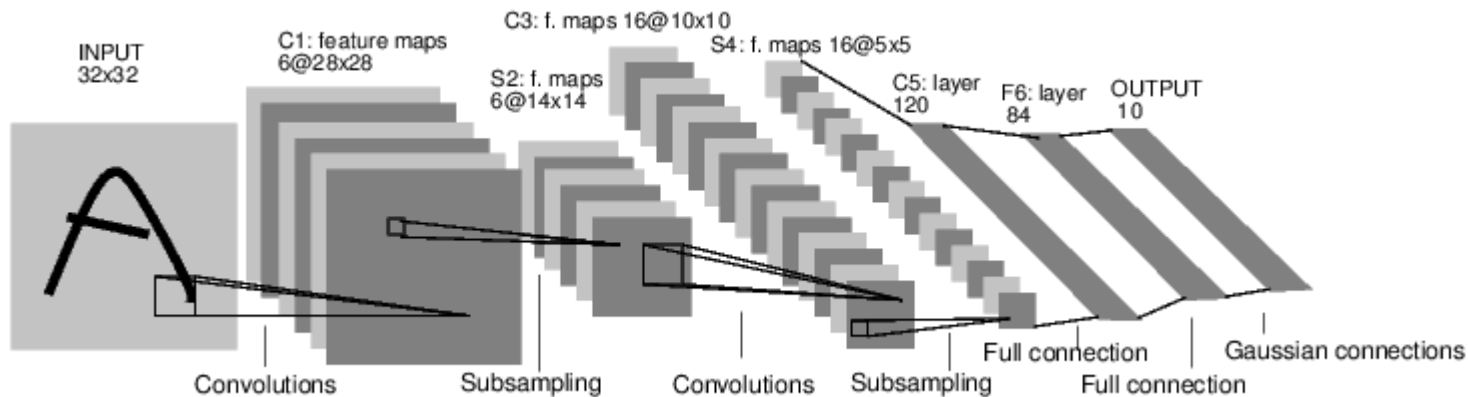


DTIC
ELECTE
FEB 2 Q 1986
D

Primera red convolucional



the90s



Yann LeCun



Lectura de código postal

60624

86 - (60626)

97222

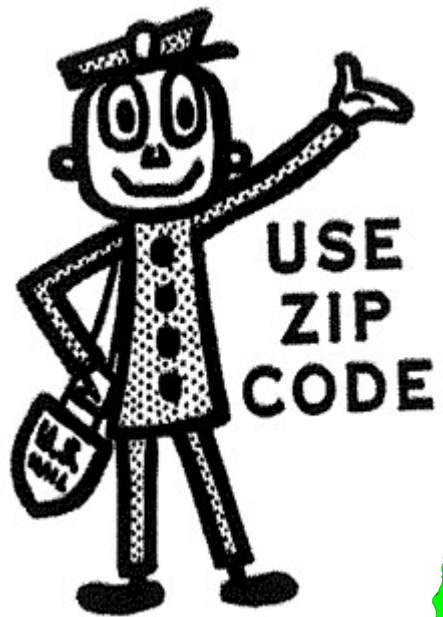
92 - (97222)

82071

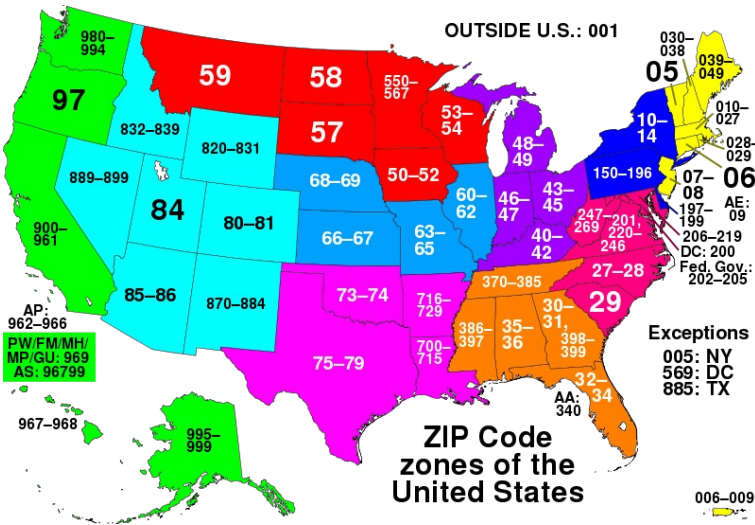
119 - (82071)

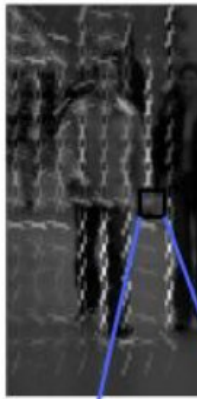
19801

469 - (19801)

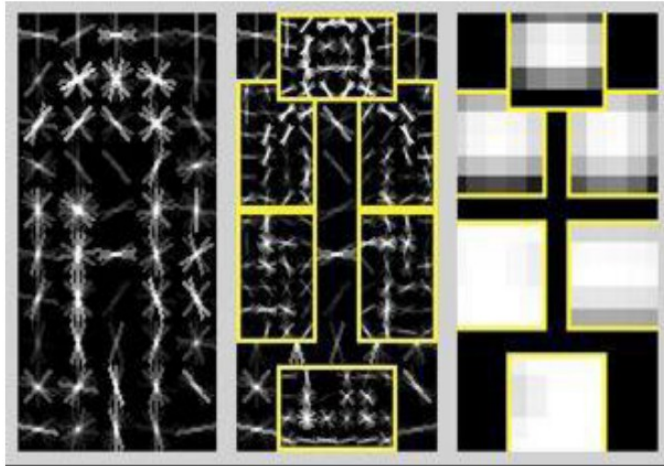
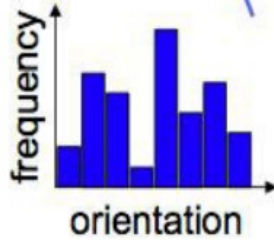


the90s



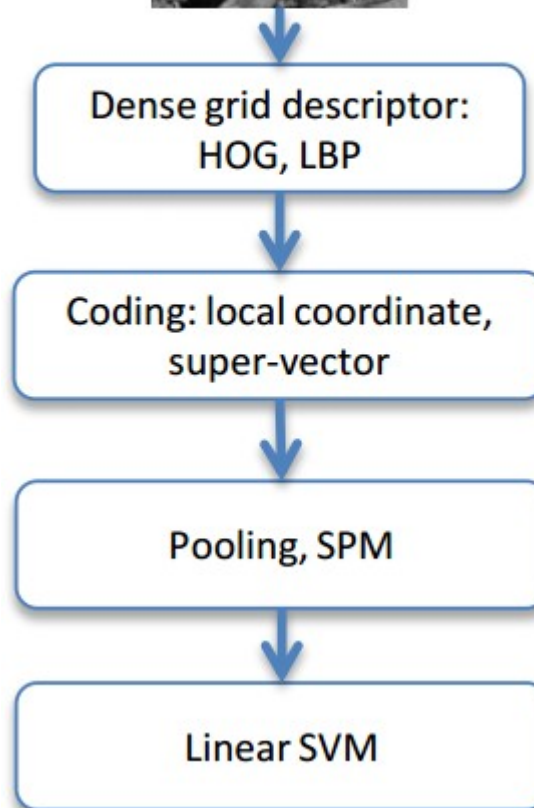


Histogram of
Gradients
(HoG)
Dalal & Triggs,
2005



Deformable Part Model Felzenswalb,
McAllester, Ramanan, 2009

The 2000s



2010

IMAGENET Large Scale Visual Recognition Challenge

Steel drum

The Image Classification Challenge:
1,000 object classes
1,431,167 images



Output:
Scale
T-shirt
Steel drum
Drumstick
Mud turtle

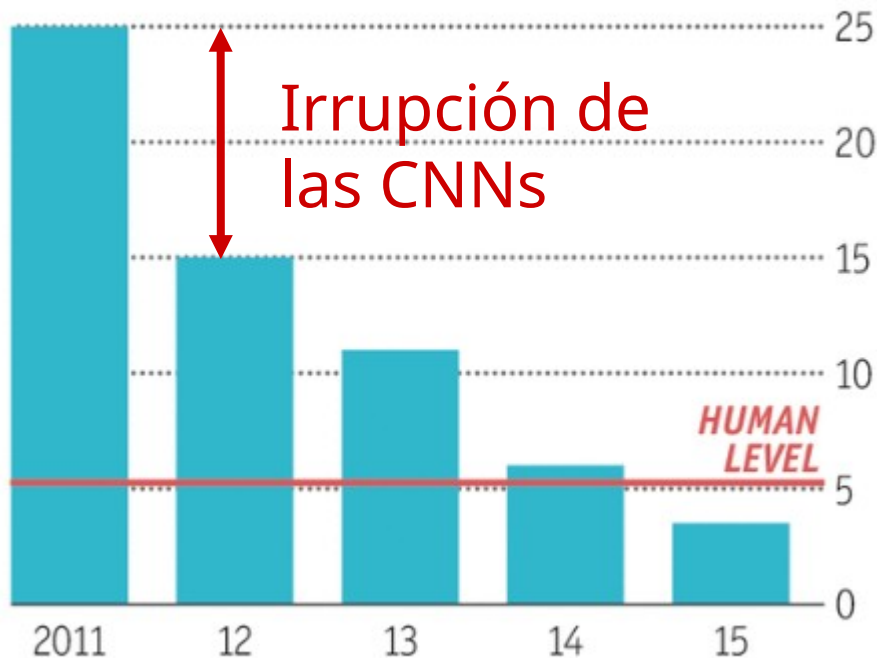


Output:
Scale
T-shirt
Giant panda
Drumstick
Mud turtle



Las redes convolucionales ganan la carrera

Error rates on ImageNet Visual Recognition Challenge, %



Sources: ImageNet; Stanford Vision Lab

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

Yann LeCun



Geoffrey E. Hinton



“Revolution of Depth”

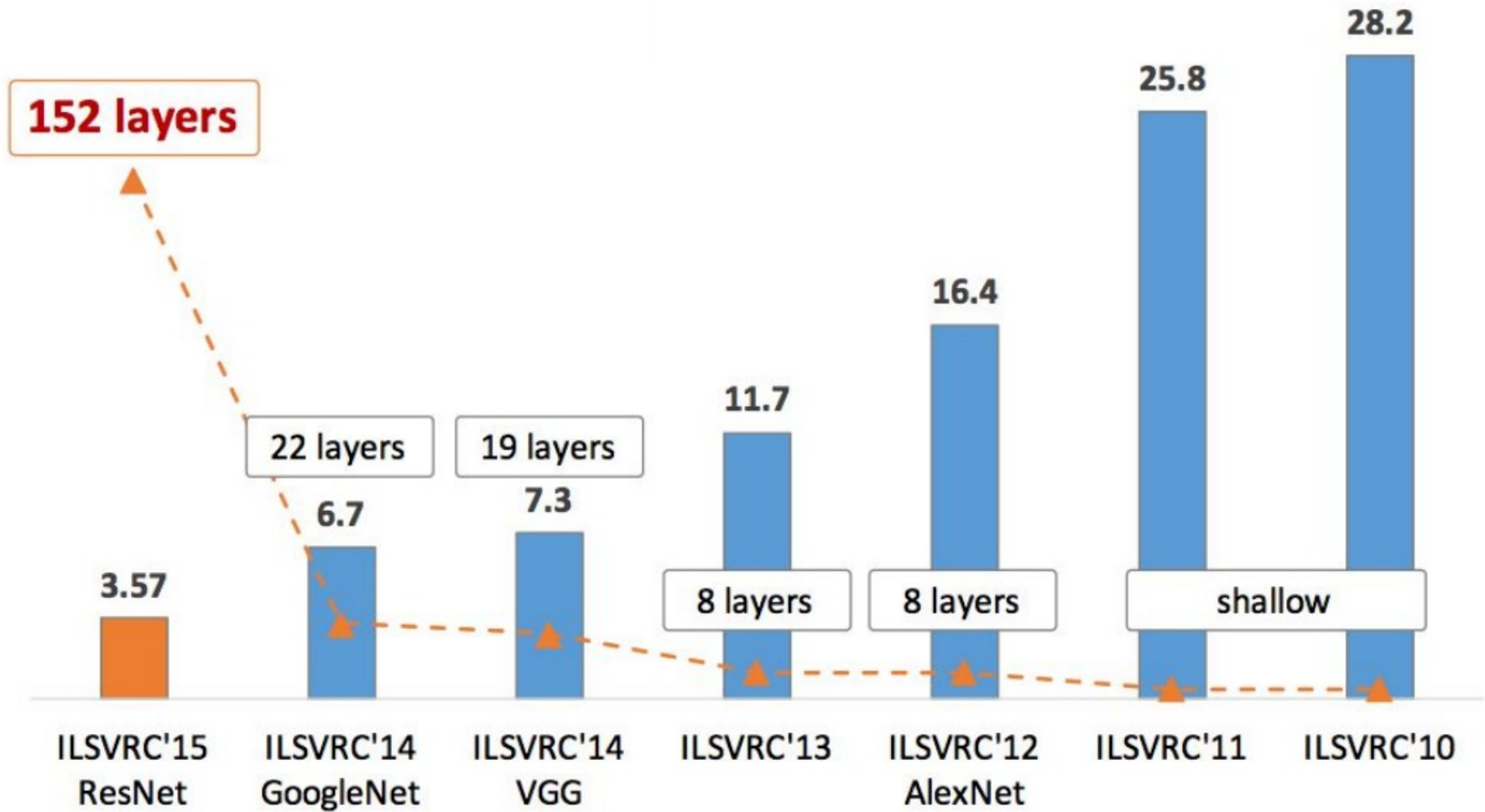


Figure copyright Kaiming He, 2016.

Entendiendo imágenes

Que necesitamos?



Redes convolucionales

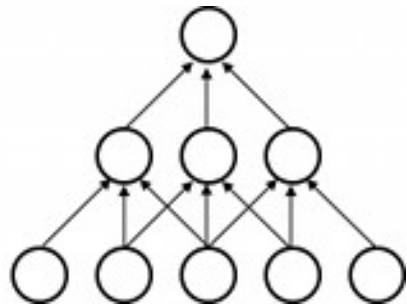
Convolutional Neural Networks

Sparse connectivity

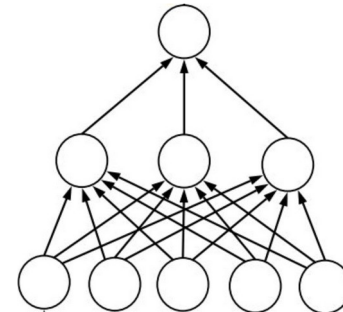
layer $m+1$

layer m

layer $m-1$



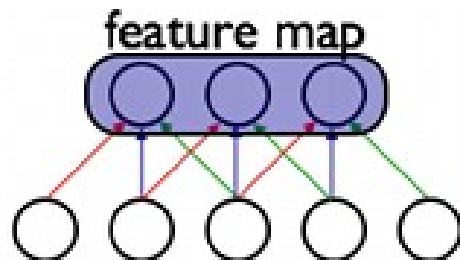
Dense connectivity



Shared weights

layer m

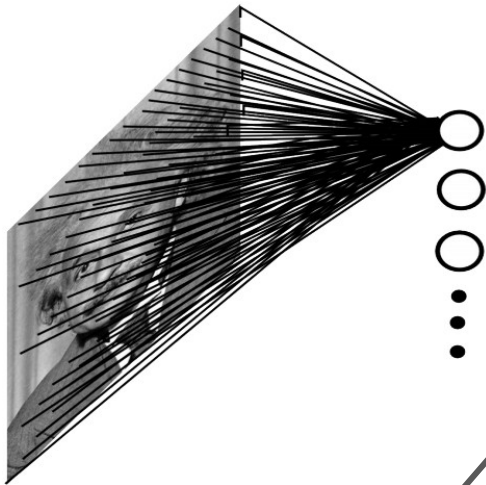
layer $m-1$



Convolutional Neural Networks

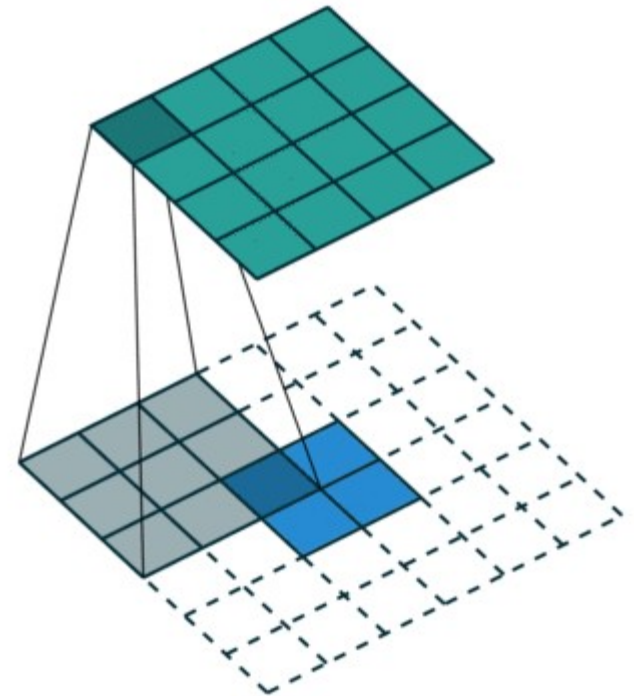
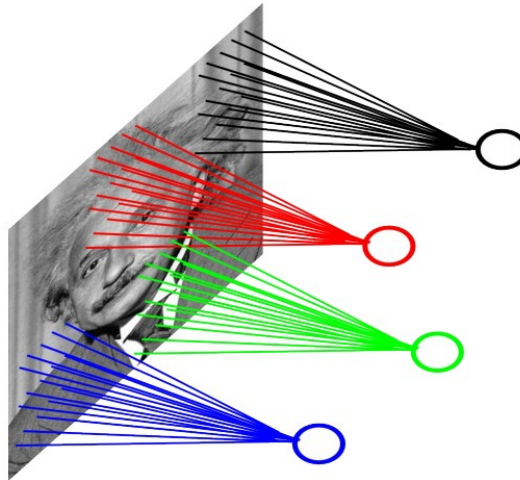
dot product + bias

$$h_k = \text{ReLU}(W_k \cdot x + b_k)$$



convolution + bias

$$h_{ij}^k = \text{ReLU}((W_k * x)_{ij} + b_k)$$



Convolutional Neural Networks

Por qué?

- **Invariancia traslacional.** La visión responde igual en cualquier lugar, una cara es una cara en cualquier lugar de la imagen
- **Localidad.** Entender qué hay en una imagen depende del lugar en que se concentra la atención.
- **Reducir sobreajuste.** Al tener menos pesos ajustables, se reduce la posibilidad de aprender particularidades.

Convoluciones en procesamiento de imágenes



Original



Sharpen



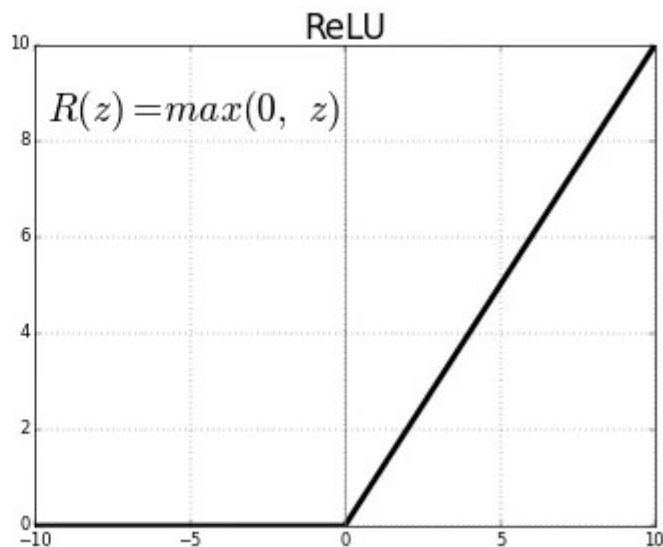
Edge Detect



"Strong" Edge
Detect

Convolutional Neural Networks

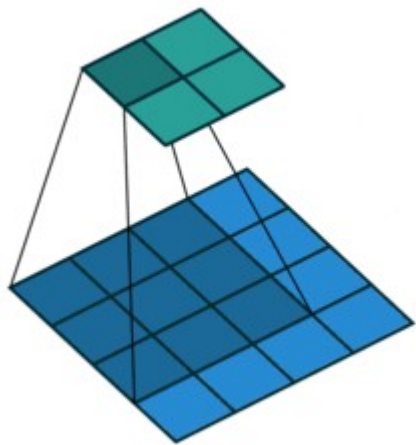
$$h_{ij}^k = \text{ReLU}((W_k * x)_{ij} + b_k)$$



$$(W * x)_{i,j} = \sum_{l,m} W_{l,m} x_{i-l,j-m}$$

Convolutional Neural Networks

Ejemplo



Input

0	1	2
3	4	5
6	7	8

*

Kernel

0	1
2	3

=

Output

19	25
37	43

Padding y Stride

Input Kernel Output

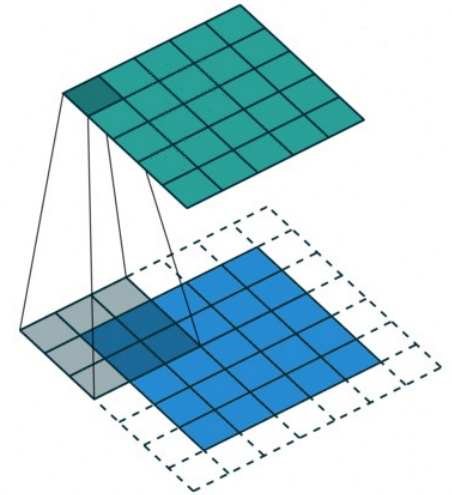
0	0	0	0	0
0	0	1	2	0
0	3	4	5	0
0	6	7	8	0
0	0	0	0	0

*

0	1
2	3

=

0	3	8	4
9	19	25	10
21	37	43	16
6	7	8	0



Stride is the number of “unit” the kernel shifted per slide over rows/columns.

E.g., Strides of 3 for height and 2 for width

Input Kernel Output

0	0	0	0	0
0	0	1	2	0
0	3	4	5	0
0	6	7	8	0
0	0	0	0	0

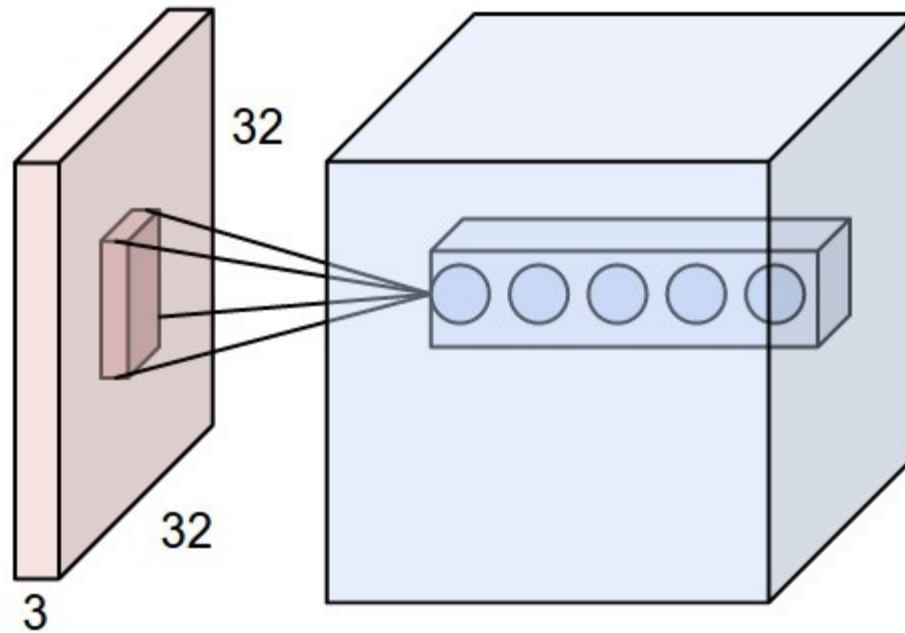
*

0	1
2	3

=

0	8
6	8

Una capa convolucional transforma el volumen



Entrada: imagen, 32x32, 3 canales RGB: 32x32x3

Filtro: tamaño 3x3, x 3 canales.

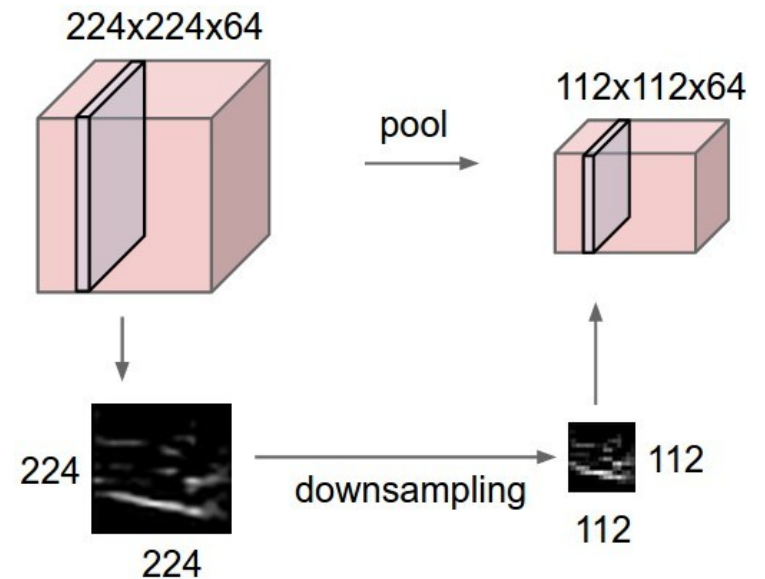
Salida: aplico 5 filtros convolucionales distintos,
obtengo por cada uno una “imagen” de 32x32: 32x32x5

Reducción de volumen: Pooling

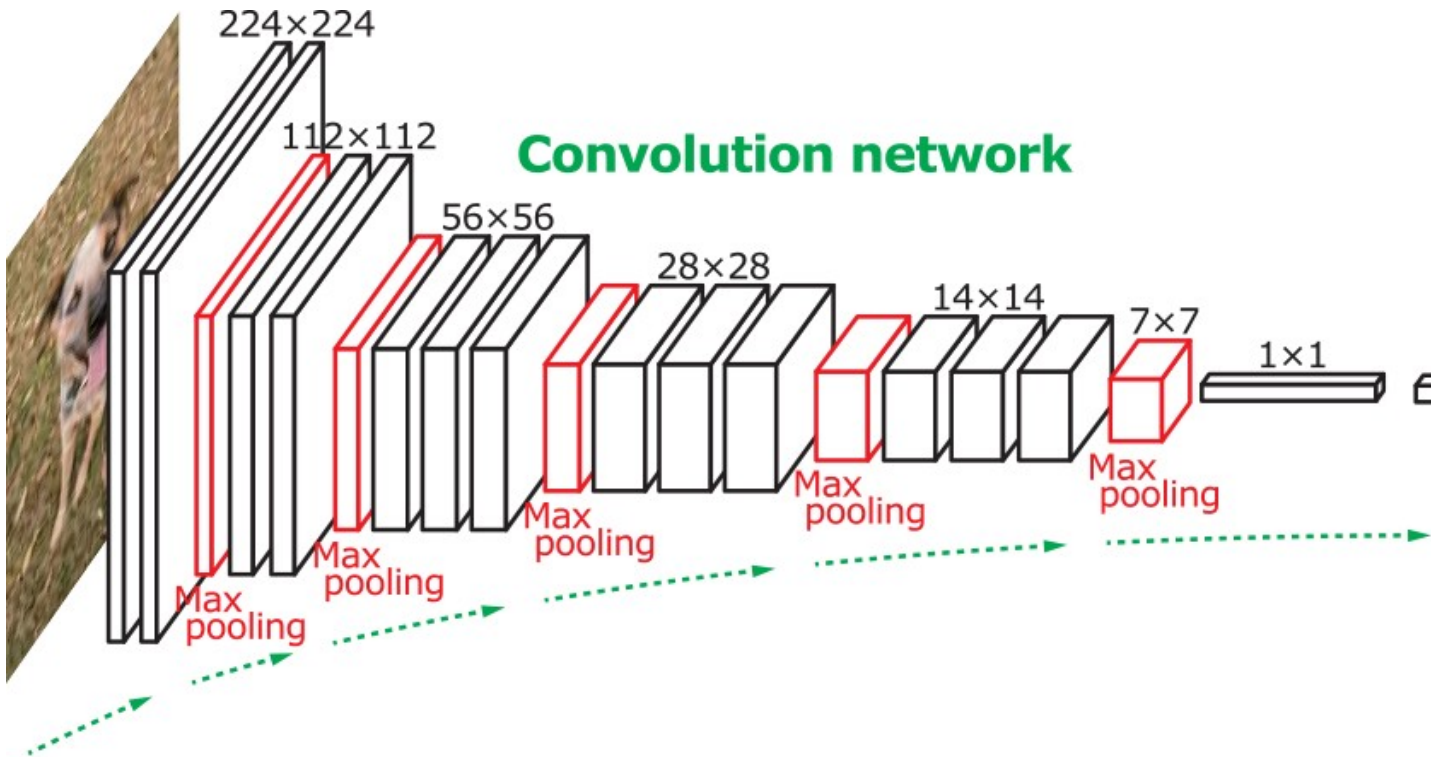
1	1	2	4
5	6	7	8
3	2	1	0
1	2	3	4



6	8
3	4



Ejemplo: VGG net ("versión D")



dataset: ImageNet (ILSVRC 2013)

Input 224x224

conv3-64

conv3-64

maxpool

conv3-128

conv3-128

maxpool

conv3-256

conv3-256

conv3-256

maxpool

conv3-512

conv3-512

conv3-512

maxpool

conv3-512

conv3-512

conv3-512

maxpool

FC-4096

FC-4096

FC-1000

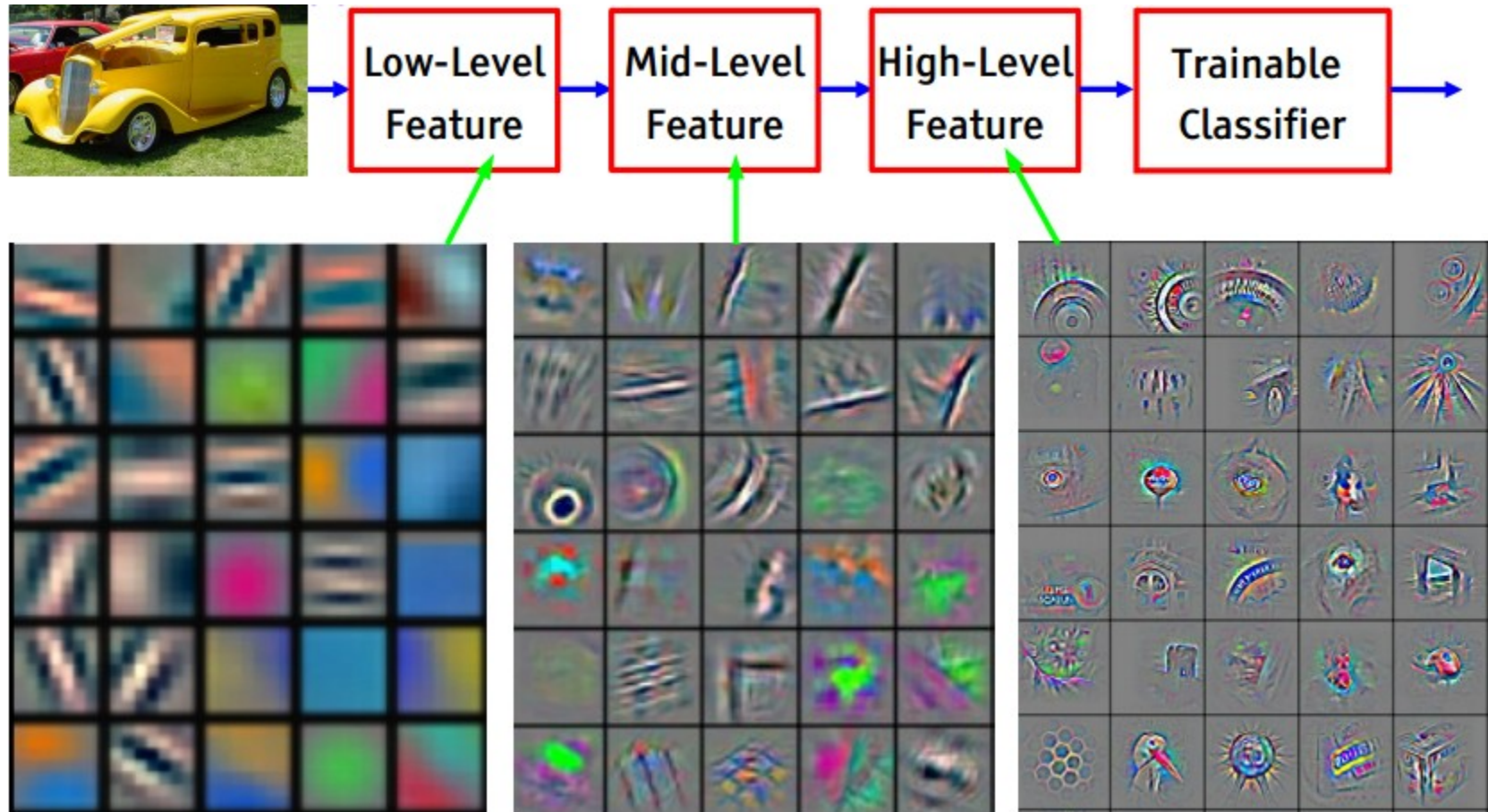
softmax

¿Cómo se entrena una CNN?

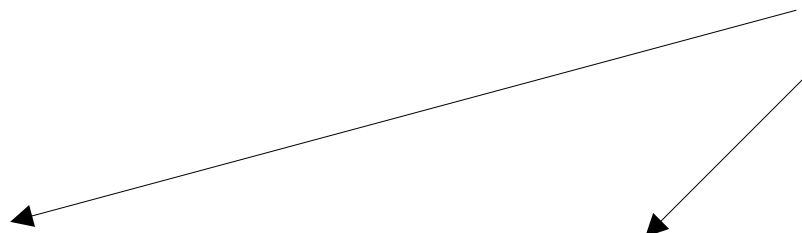
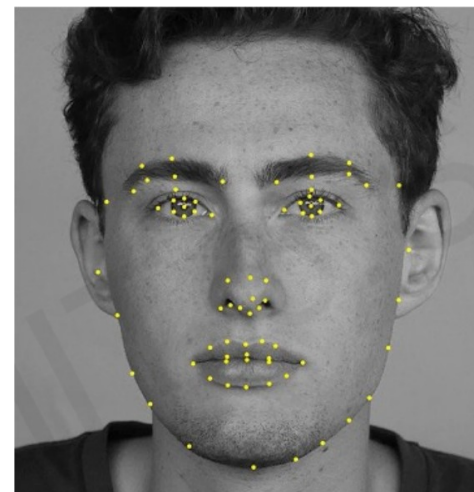
Backpropagation.

- **Derivación automática.**
- **Learning rate adaptativos.**
- **Minibatch.**
- **Unidades Relu para mantener el gradiente.**

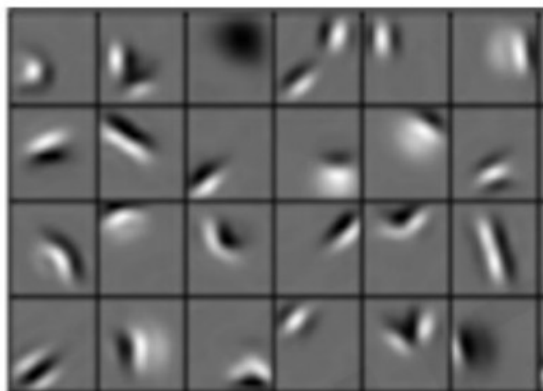
¿Qué ve una CNN ya entrenada?



¿Qué ve una CNN ya entrenada?



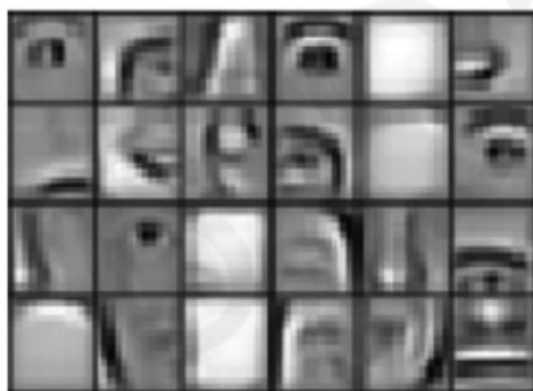
Low level features



Edges, dark spots

Conv Layer 1

Mid level features



Eyes, ears, nose

Conv Layer 2

High level features



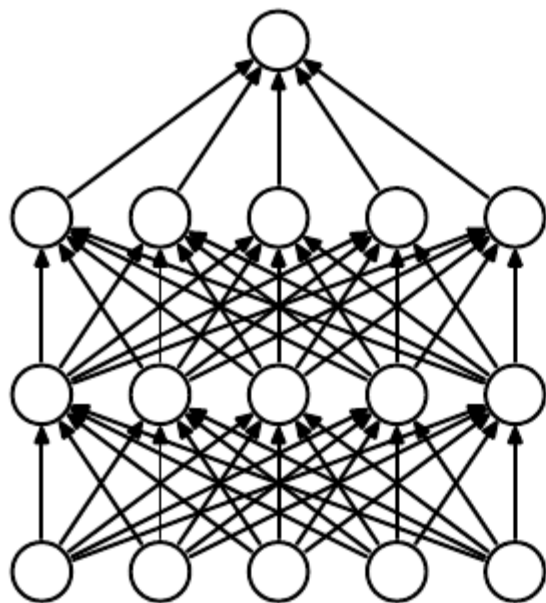
Facial structure

Conv Layer 3

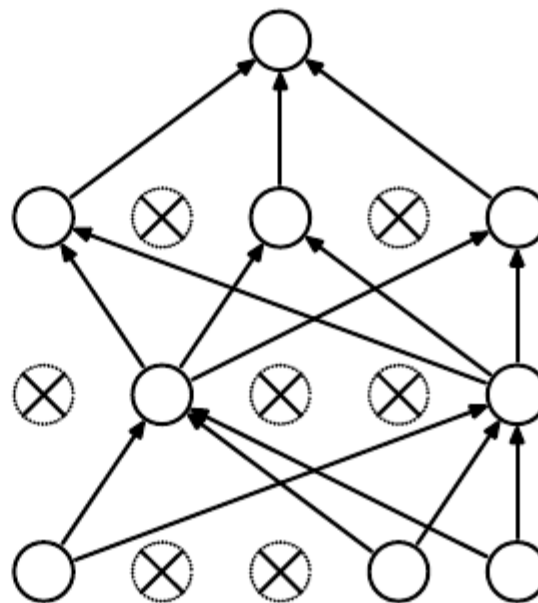
Regularización

Dropout y Data augmentation

Dropout Neural Net Model



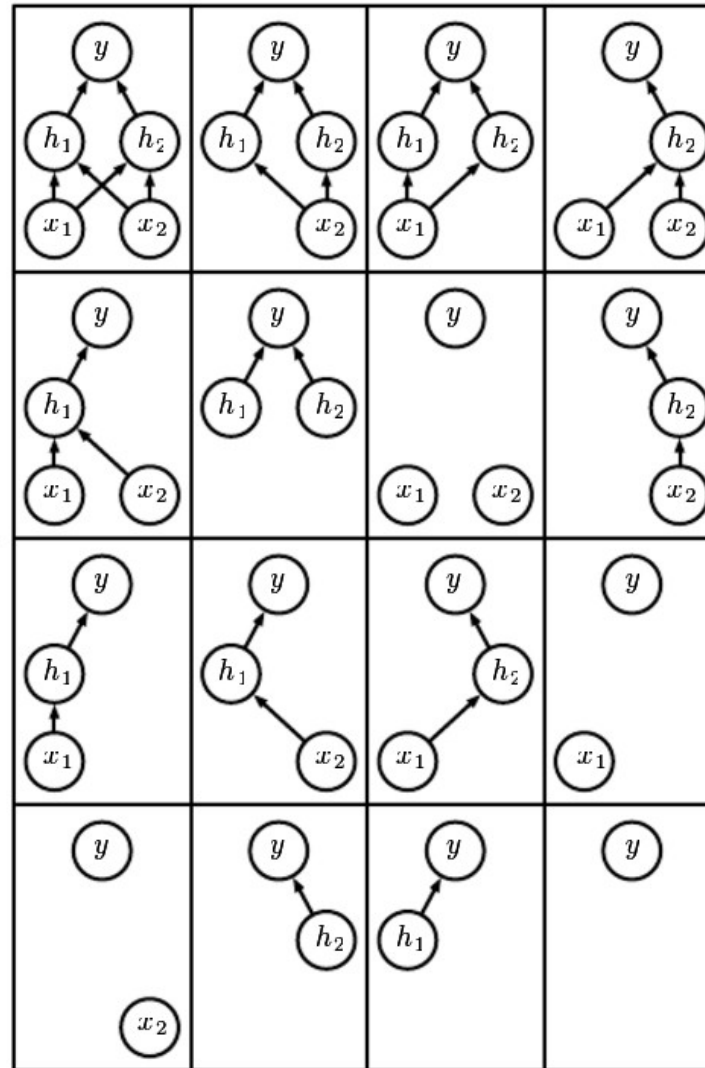
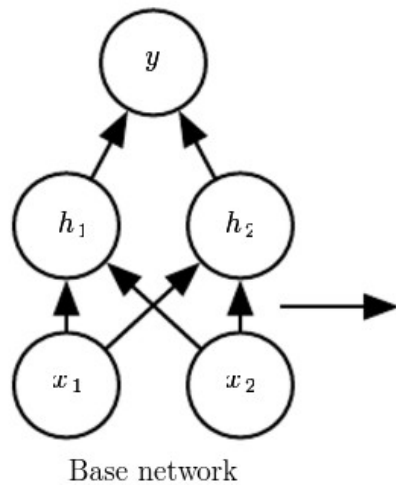
(a) Standard Neural Net



(b) After applying dropout.

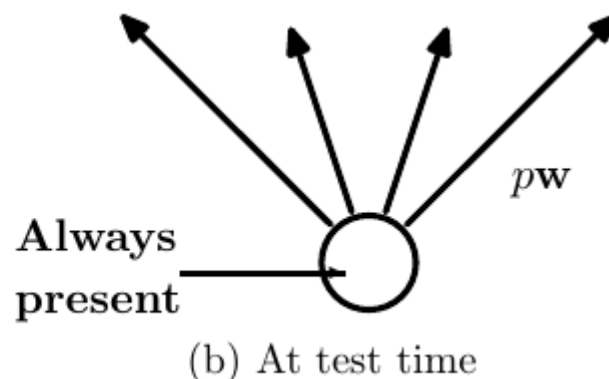
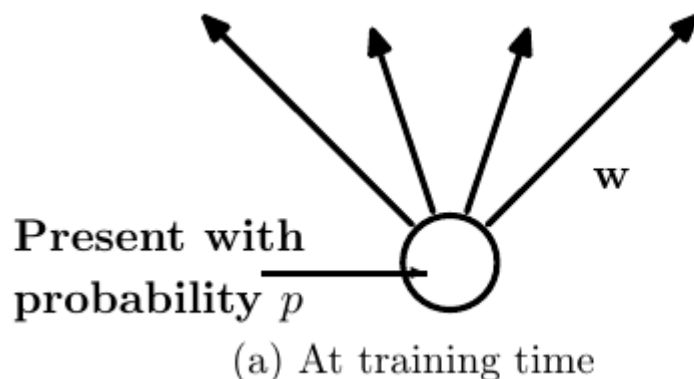
Dropout Neural Net Model. **Left:** A standard neural net with 2 hidden layers. **Right:** An example of a thinned net produced by applying dropout to the network on the left. Crossed units have been dropped.

Dropout Neural Net Model



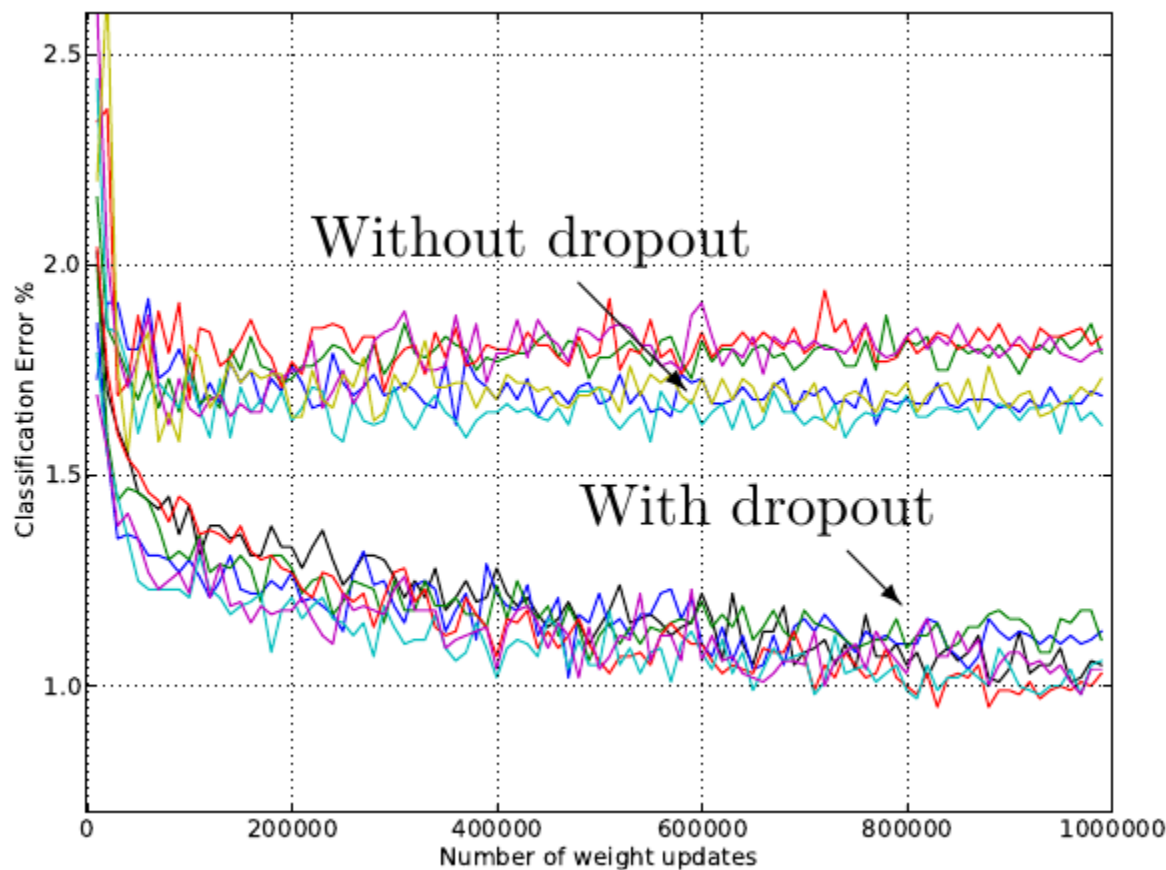
Ensemble of subnetworks

Dropout units



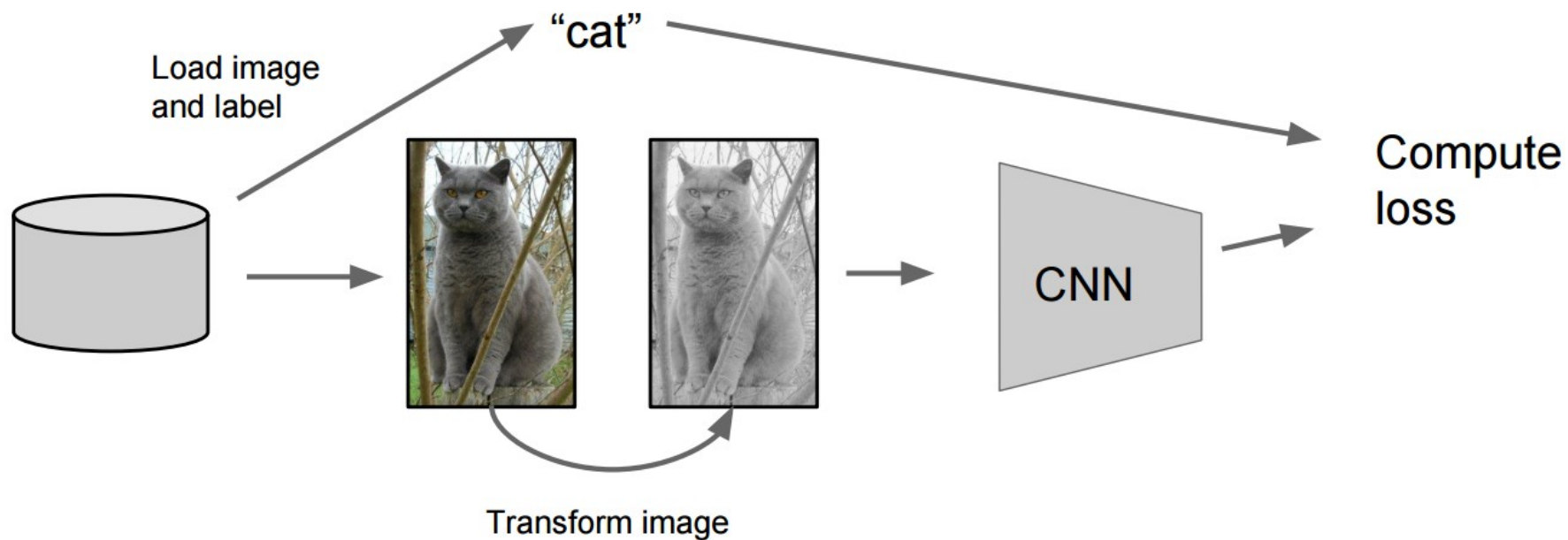
Left: A unit at training time that is present with probability p and is connected to units in the next layer with weights \mathbf{w} . **Right:** At test time, the unit is always present and the weights are multiplied by p . The output at test time is same as the expected output at training time.

Dropout: Robustness



Test error for different architectures with and without dropout. The networks have 2 to 4 hidden layers each with 1024 to 2048 units.

Data Augmentation



Data Augmentation: Horizontal flips



Data Augmentation: Random crops/scales

Training: sample random crops / scales

Por ejemplo [ResNet]:

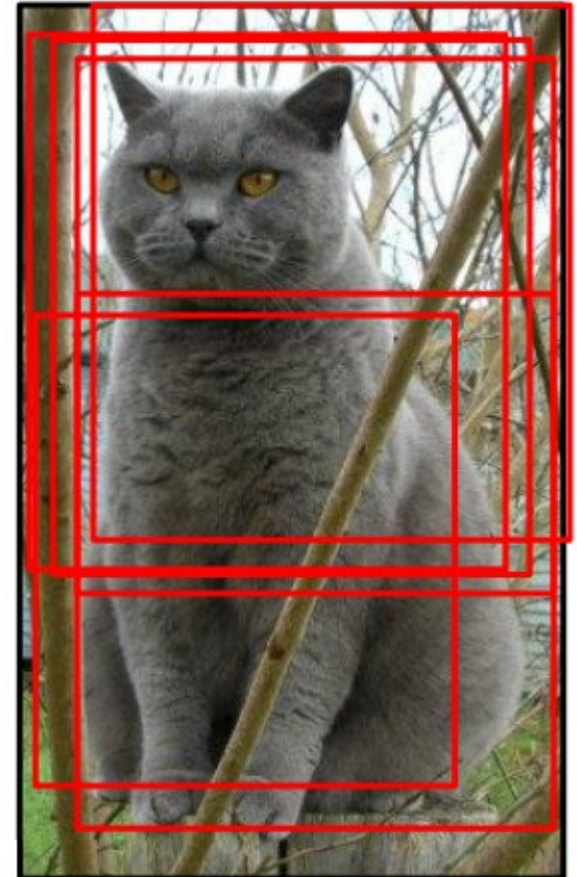
1. Pick random L in range [256, 480]
2. Resize training image, short side = L
3. Sample random 224 x 224 patch

Testing: average a fixed set of crops

Por ejemplo [ResNet]

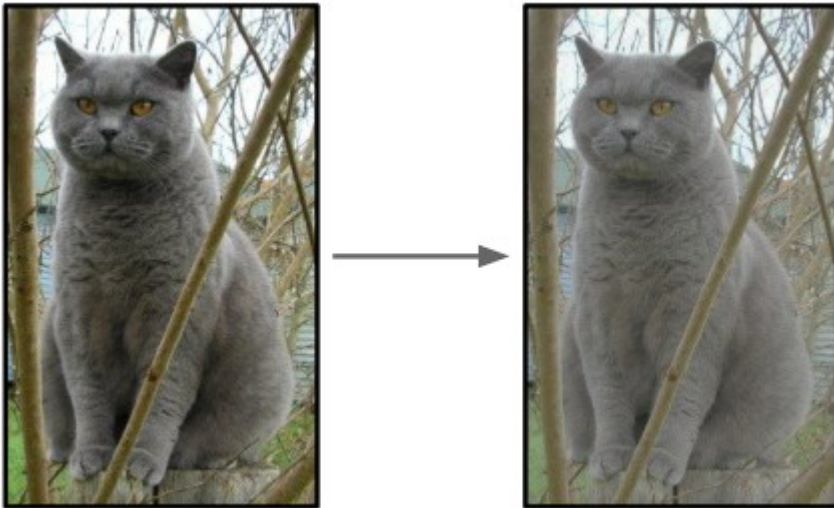
4. Resize image at 5 scales: {224, 256, 384, 480, 640}
5. For each size, use 10 224 x 224 crops: 4 corners + center, + flips

[ResNet] He, Kaiming, et al. "Deep residual learning for image recognition." arXiv preprint arXiv:1512.03385 (2015).



Versión simple:

Alterar el contraste aleatoriamente



Versión compleja:

1. Aplicar PCA a todos los píxeles [R,G,B] del training set
2. Samplear un "desplazamiento de color" a lo largo de las direcciones principales
3. Aplicarle este desplazamiento a todos los píxeles de una imagen de entrenamiento.