4ª Emisión

DIPLOMADO
Inteligencia Artificial Aplicada

Módulo 11: Introducción a las redes neuronales

3. Redes convolucionales

Instructor: Blanca Vázquez

DGTIC UNAM
DIRECCIÓN GENERAL DE CÓMPUTO Y DE TECNOLOGÍAS DE INFORMACIÓN Y COMUNICACIÓN

Dirección de Docencia en Tecnologías de Información y Comunicación

REDEC UNAM

Objetivo de la sesión

- Aprender a identificar los componentes y el funcionamiento de las redes neuronales convolucionales, así como las diferentes arquitecturas existentes.

Contenido

3.1. Motivación

3.2. La operación de convolución

3.3. Submuestreo

3.4. Retropropagación en capas convolucionales y de submuestreo

3.5. Arquitecturas de redes neuronales convolucionales (AlexNet,EfficientNet, ConvNext, etc.)

3.6. Acrecentamiento de datos

3.7. Aprendizaje por transferencia

Motivación

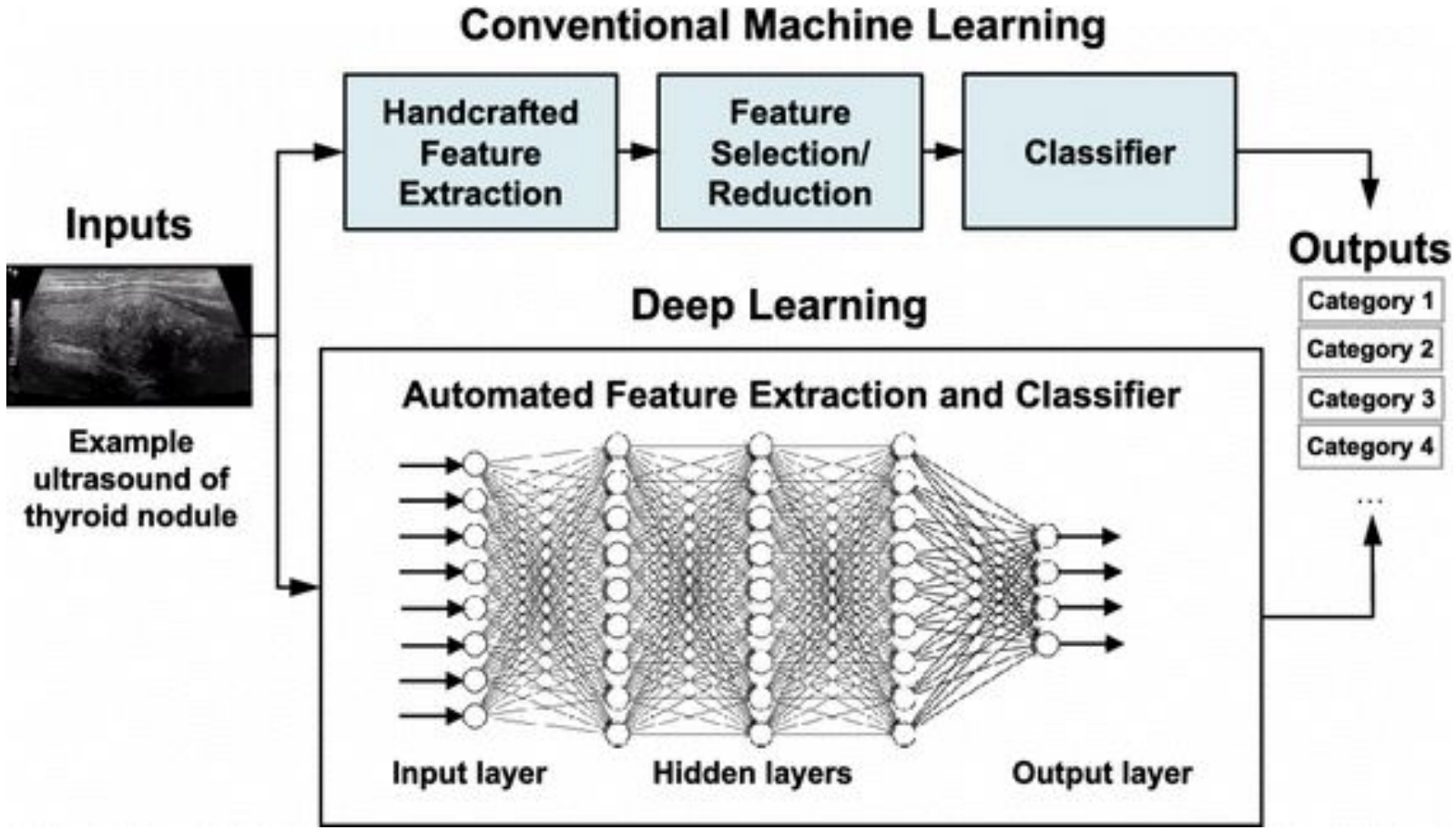


Imagen tomada de González, Woods, Digital Image Processing, 2018.

Motivación

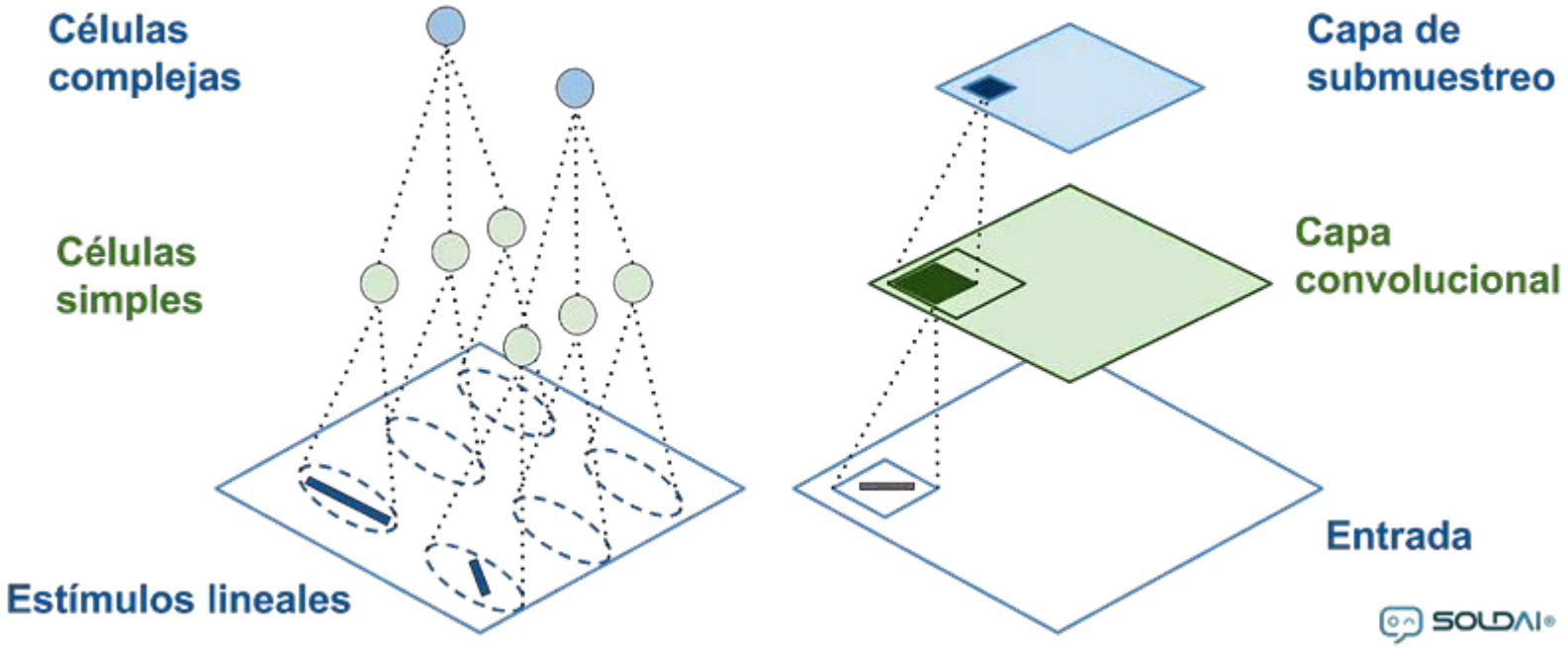


Figura. Relación entre las Redes Neuronales Convolucionales con el modelo biológico de Hubel y Wiesel.

Imagen tomada de Juan José Negron, 2020.

Motivación

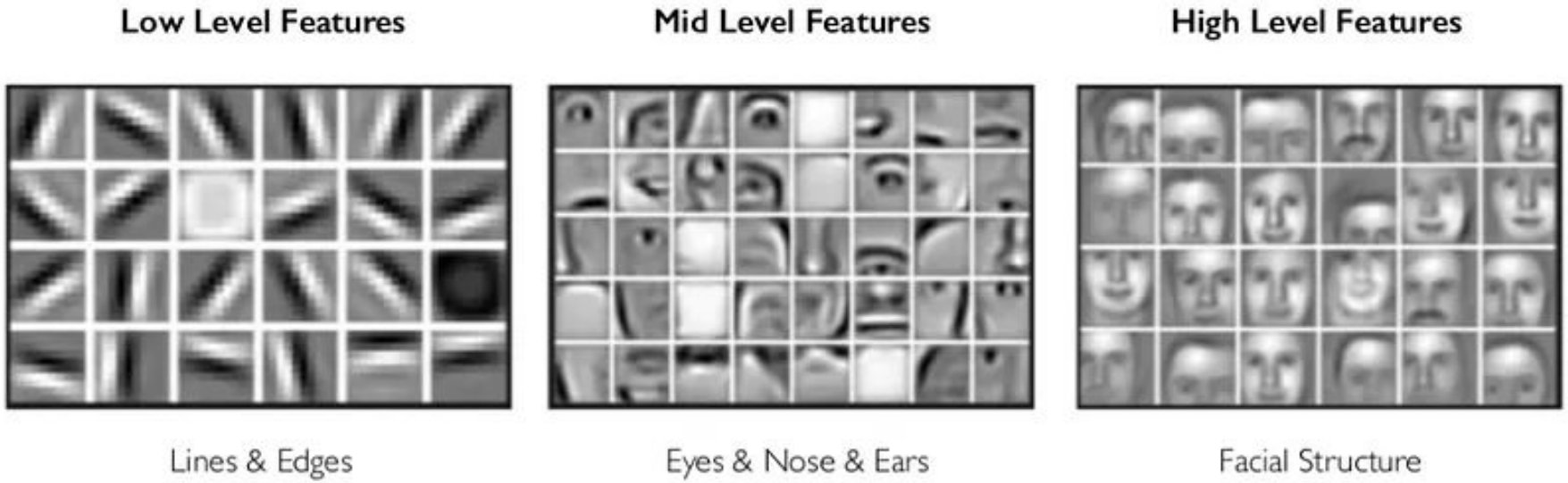


Imagen tomada de Juan José Negron, 2020.

Operación de convolución

Es una operación **matemática** que combina dos funciones (f, g) para describir la superposición entre ambas.

La convolución toma dos funciones:

- "**Desliza**" una función sobre la otra,
- **Multiplica** los valores de las funciones en todos los puntos de superposición, y
- **Suma** los productos para crear una nueva función.

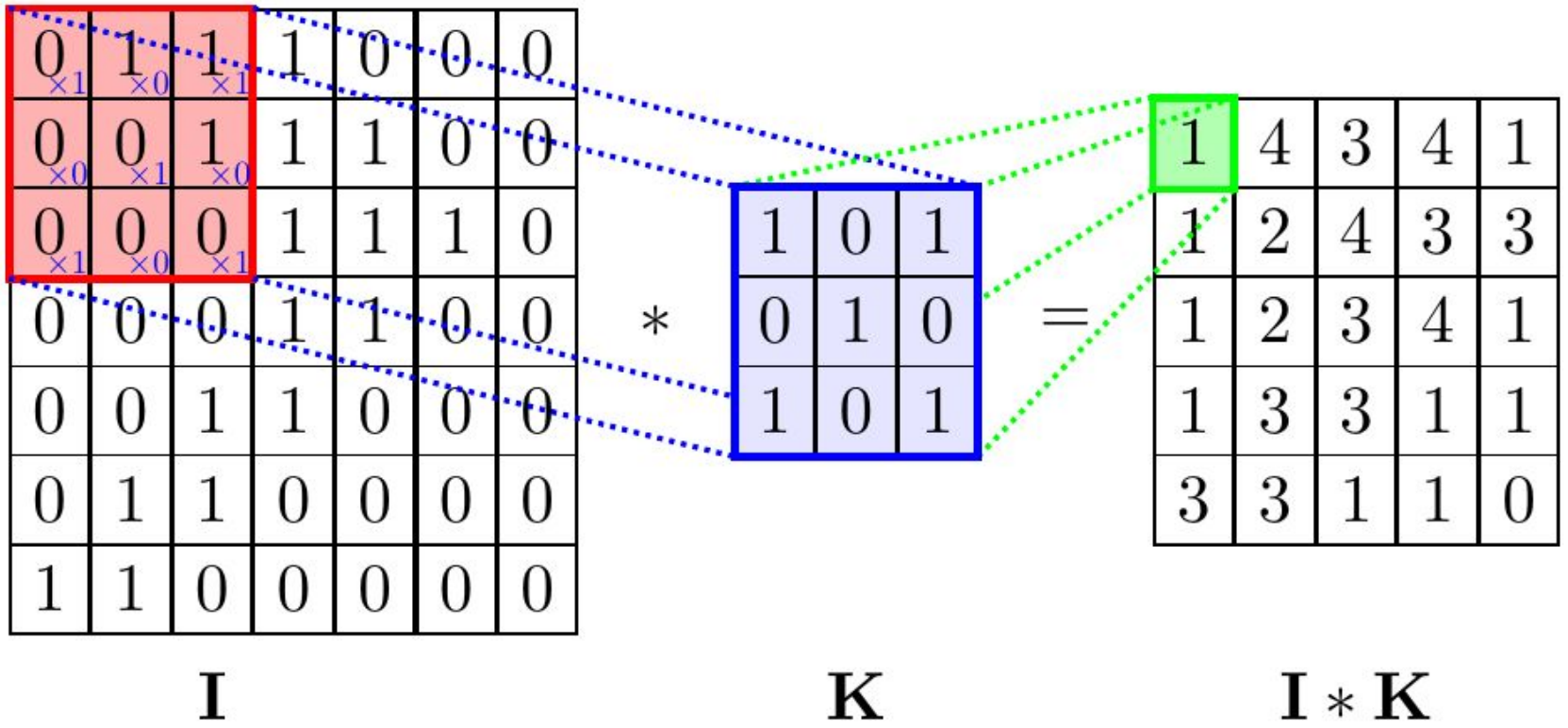
La nueva función representa cómo interactúan las dos funciones originales entre sí.

La convolución de f y g se denota $f * g$. Se define como la integral del producto de ambas funciones después de desplazar una de ellas una distancia t.

$$(f * g)(t) \doteq \int_{-\infty}^{\infty} f(\eta) g(t - \eta) d\eta$$

Imagen tomada de Wikipedia, 2024.

Convolución en el procesamiento de imágenes



La salida de una convolución se le conoce como: mapa de características

Imagen tomada de Fuentes, 2023.

Convolución en el procesamiento de imágenes

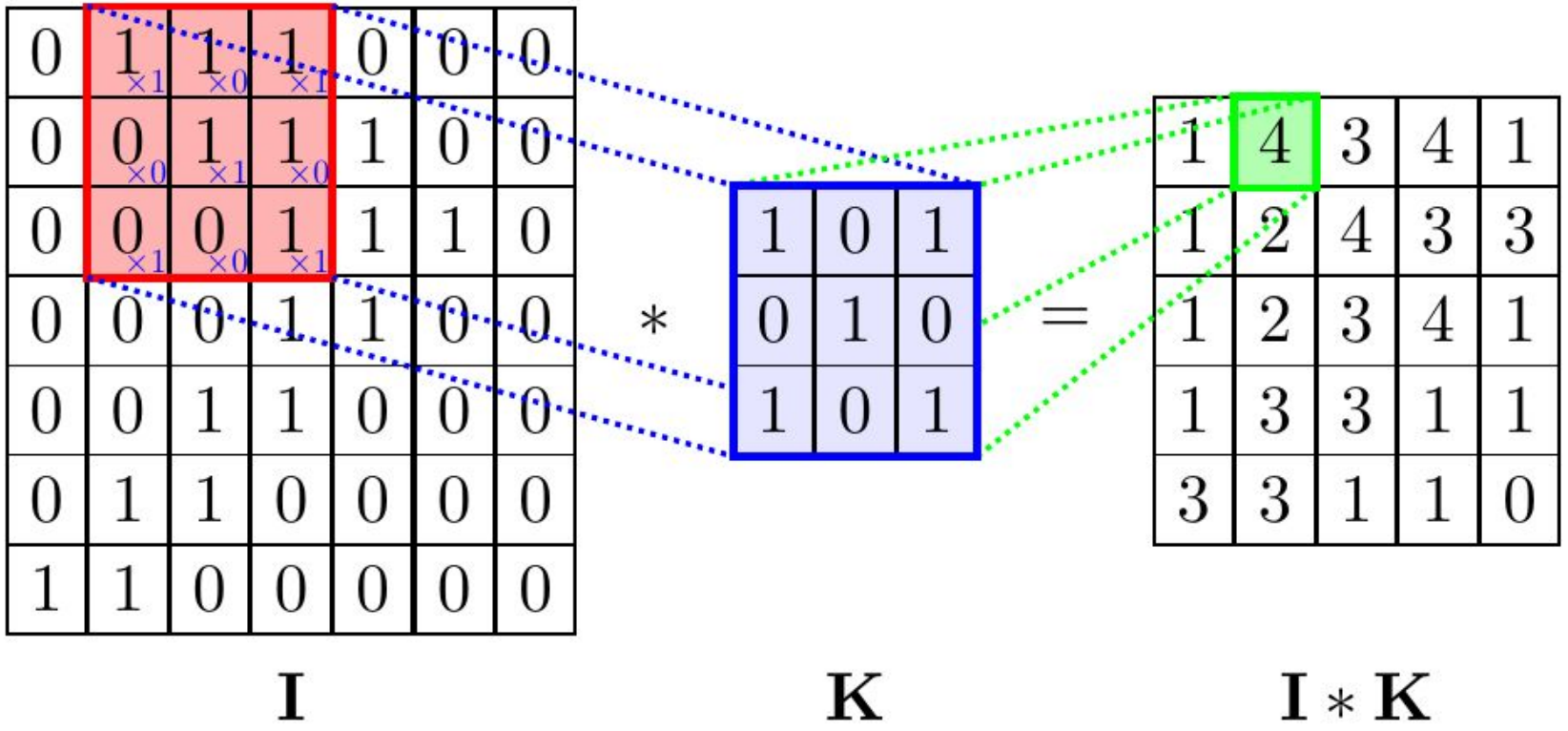


Imagen tomada de Fuentes, 2023.

Convolución en el procesamiento de imágenes

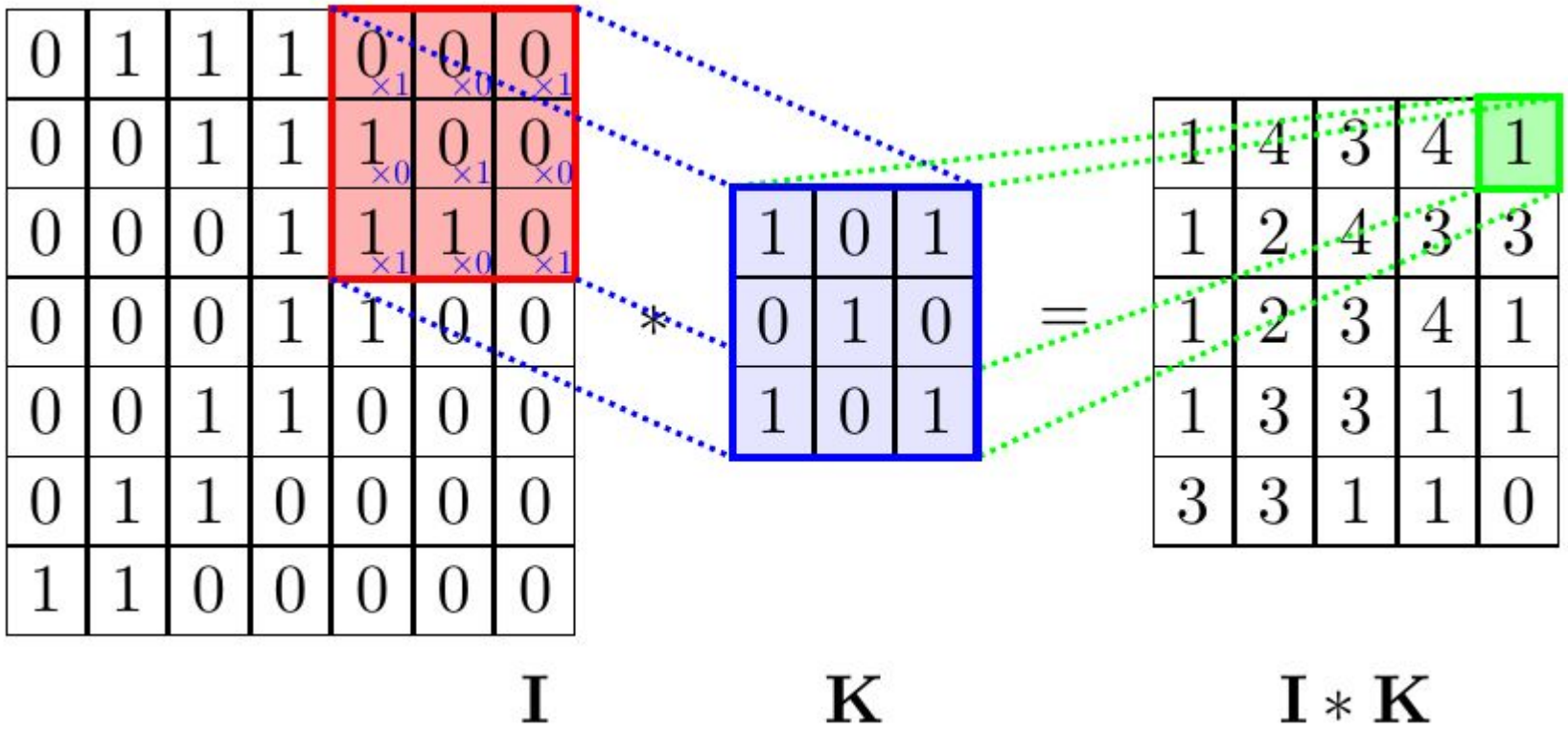


Imagen tomada de Fuentes, 2023.

Convolución en el procesamiento de imágenes

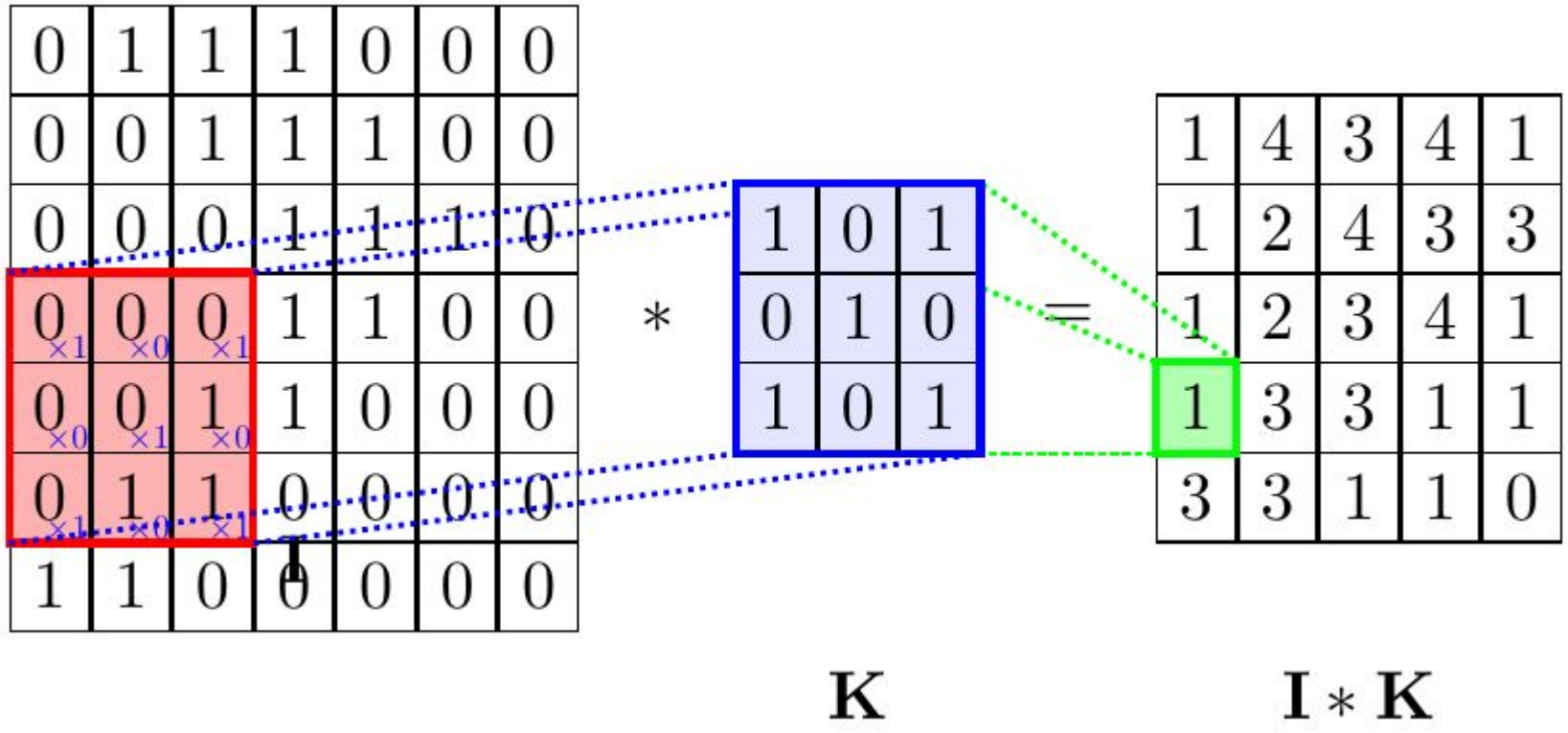
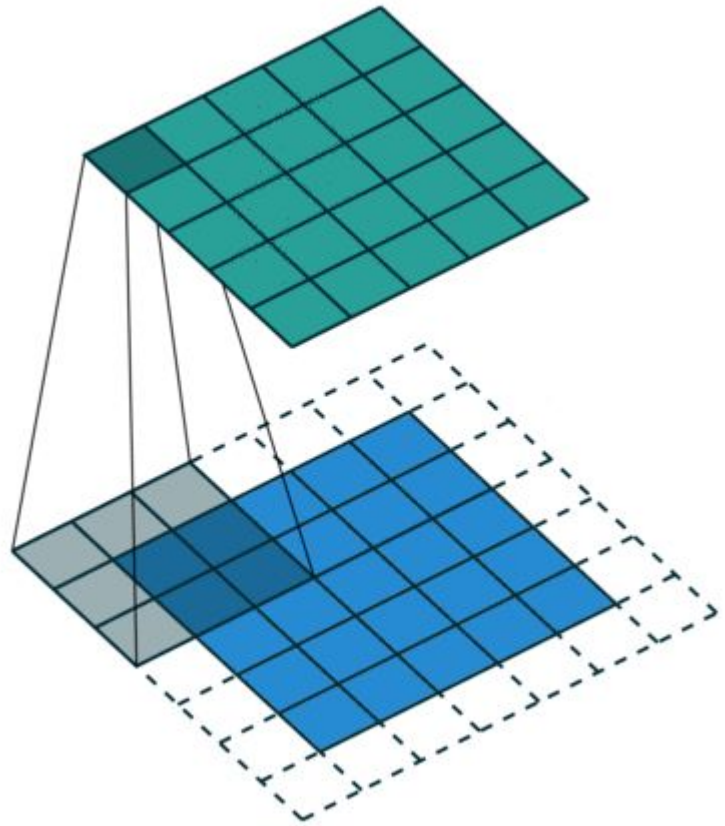


Imagen tomada de Fuentes, 2023.

Stride y padding

- **Stride**: define el tamaño del desplazamiento del kernel a través de los datos de entrada.
- **Padding**: determina si existe o no un aumento en la resolución del mapas de segmentación de entrada. Normalmente este aumento se consigue añadiendo píxeles de valor nulo.
- **Kernel**: define el tamaño del campo de visión de la convolución. Un tamaño común es k = 3.



Convolución 2D usando un kernel de tamaño 3, stride de 1 and padding.

Imagen tomada de Paul-Louis Pröve, 2017.

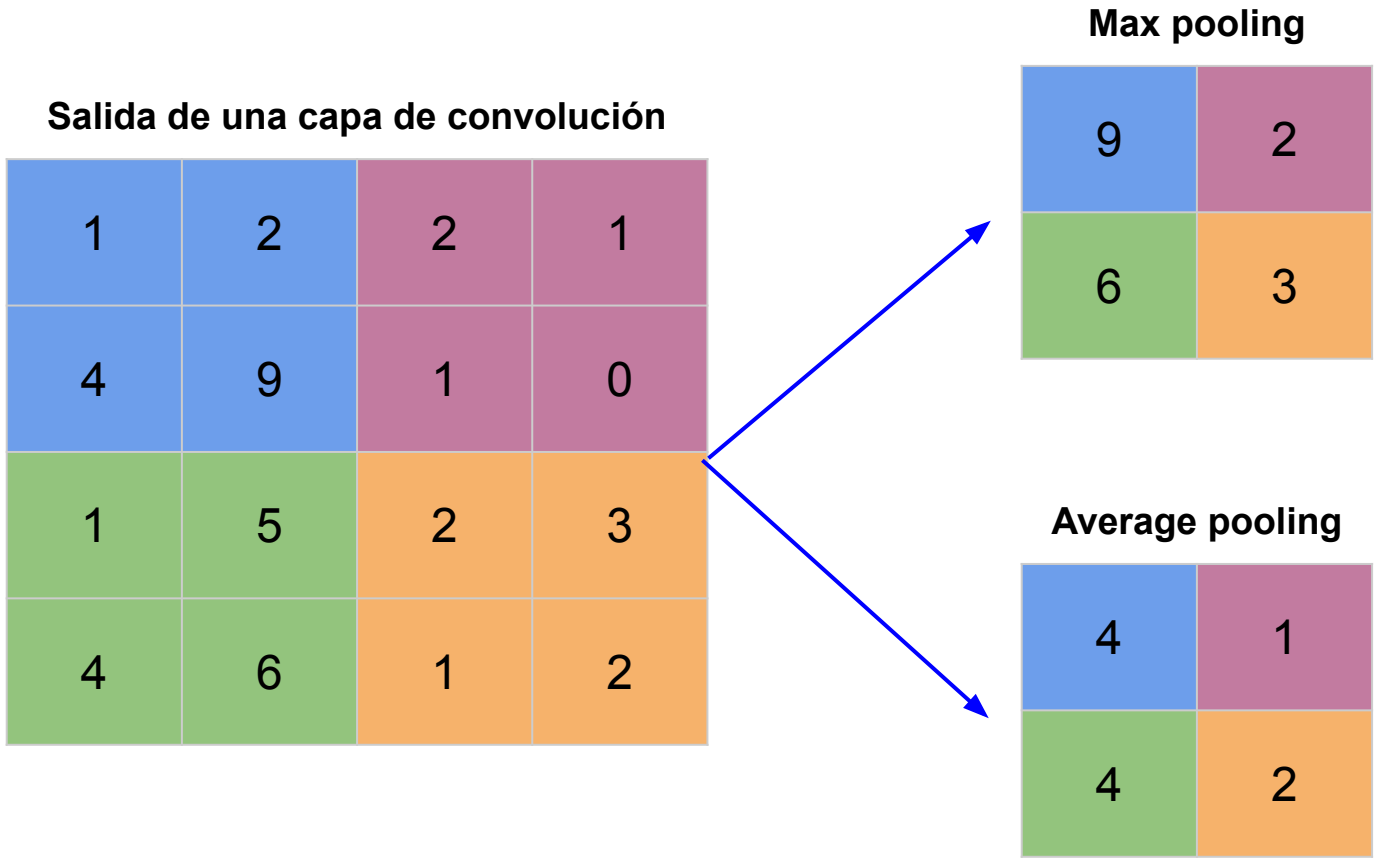
Submuestreo: capas de agrupación

Son utilizadas para sintetizar / reducir la información proveniente de capas anteriores mediante una operación en particular.

Existen dos tipos principales de capas de agrupación:

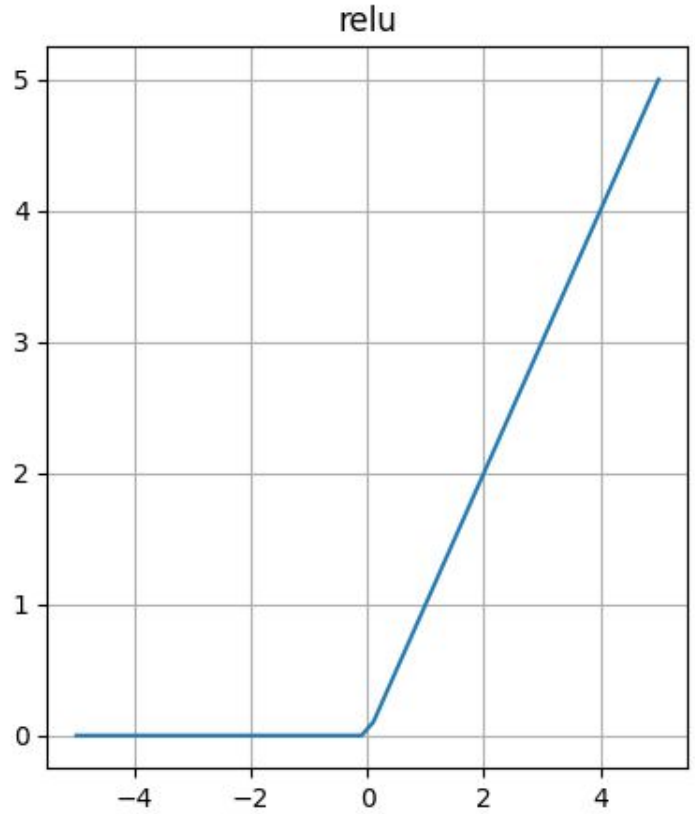
- **Max Pooling**: Toma el máximo elemento dentro de la ventana de aplicación.
- **Average Pooling**: Toma el promedio de los elementos dentro de la ventana de aplicación.

Submuestreo: capas de agrupación



Función de activación

- Se aplica una función de activación ReLU después de cada operación de convolución.
- Esta función ayuda a la red a aprender relaciones no lineales entre las características de la imagen, lo que hace la red más robusta para identificar distintos patrones.



Propagación en capas convolucionales y de submuestreo

Propagación hacia adelante:

- Se aplican las operaciones de convolución con su correspondiente activación.

Propagación hacia atrás:

- En la convolución cada neurona actualiza los gradientes por separado y al final se suman para actualizar los pesos compartidos.

Arquitecturas de redes neuronales convolucionales

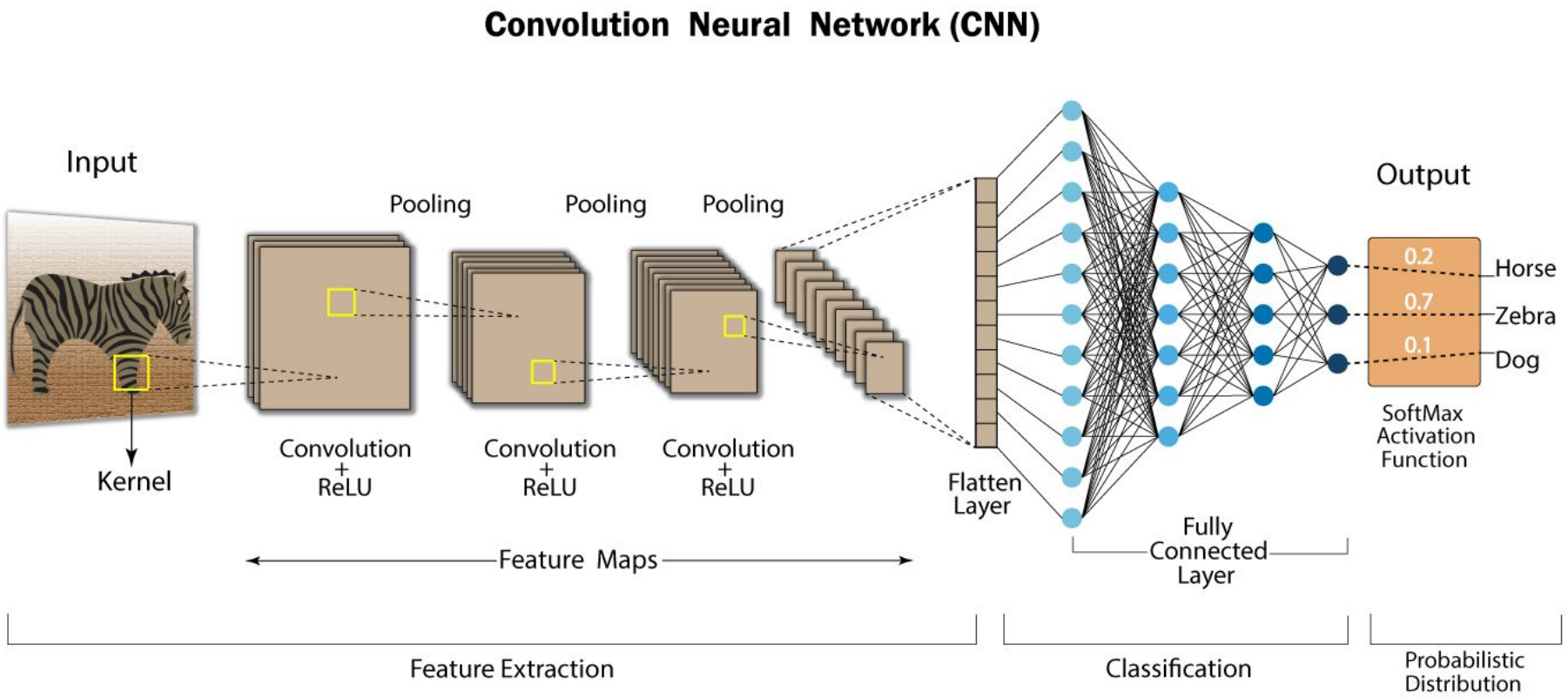


Imagen tomada de Debasish Kalita, 2024.

Time to Code

Motivación

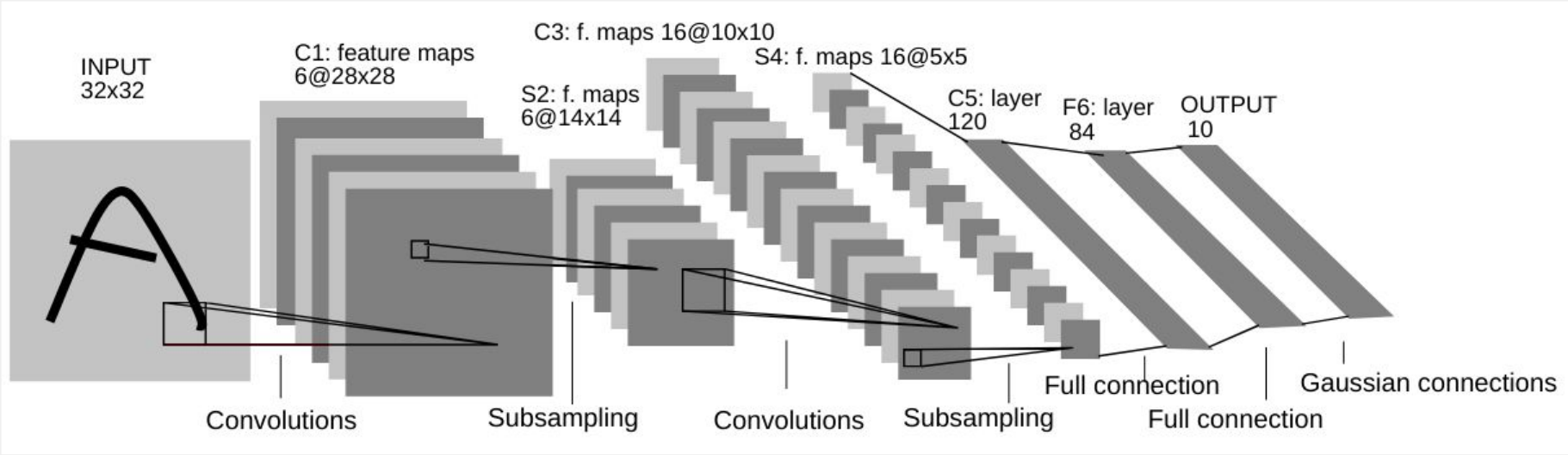
Time to Code

Visualización de filtros

Time to Code

Fashion MNIST

Arquitecturas de redes neuronales convolucionales: LeNet



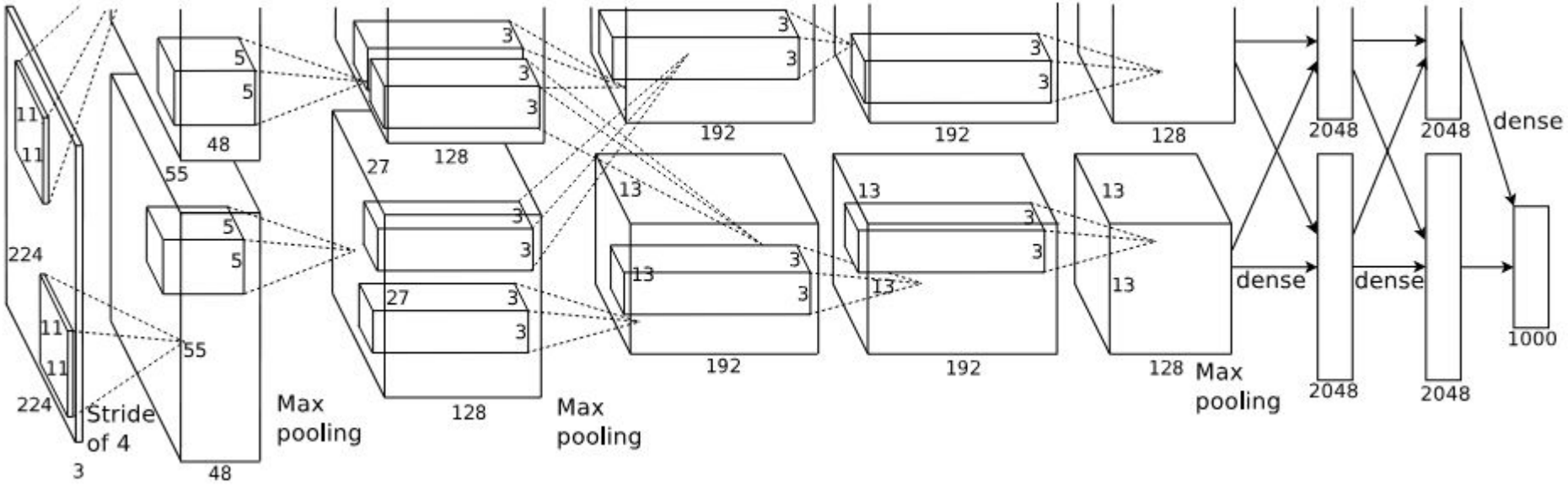
Total de parámetros: 60,000

Características:

- Propuesta por LeCun et al. en 1998.
- Conjunto de entrenamiento: imágenes de dígitos a mano (MNIST).
- Entrada: imágenes de tamaño 32 x 32 en escala de grises.
- Salida: 10 clases (una por cada dígito)

Imagen tomada de Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. (1998). "Gradient-based learning applied to document recognition" (PDF). Proceedings of the IEEE. 86 (11): 2278–2324

Arquitecturas de redes neuronales convolucionales: AlexNet



GPU

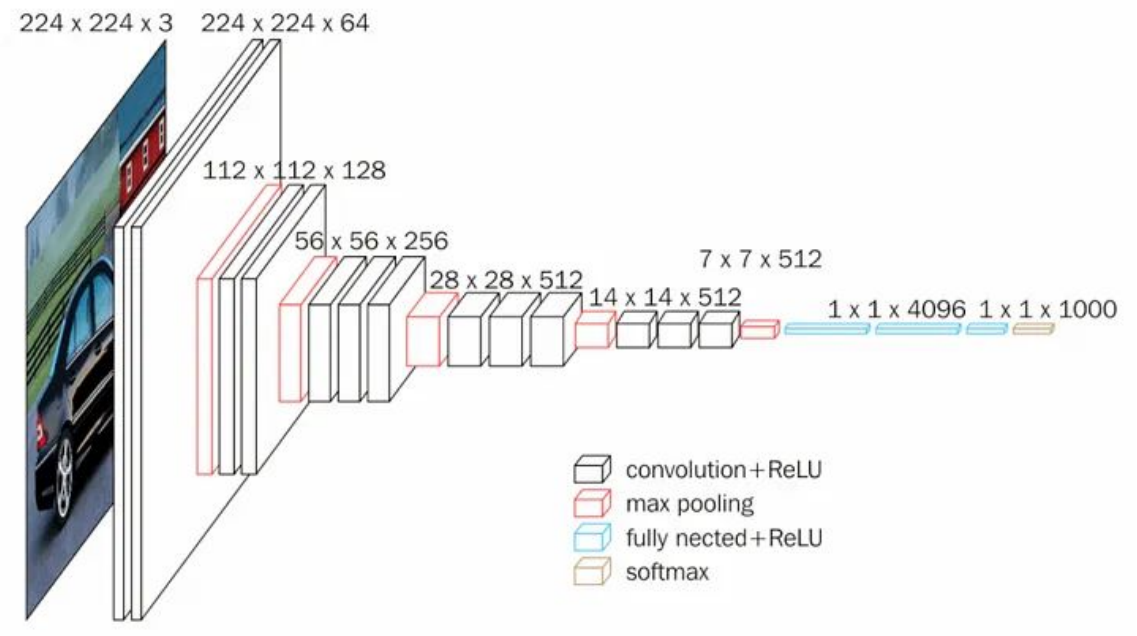
Total de parámetros: 62.3 millones

Características:

- Propuesta por Krizhevsky et al. en 2012.
- Conjunto de entrenamiento: ImageNet LSVRC-2010 (1.2 millones de imágenes).
- Entrada: imágenes de tamaño 227 x 227 a color.
- Salida: 1000 clases.

Imagen tomada de Krizhevsky, A., Sutskever, I., & Hinton, G.E. (2012). ImageNet classification with deep convolutional neural networks. Communications of the ACM, 60, 84 - 90.

Arquitecturas de redes neuronales convolucionales: VGGNet-16



Características:

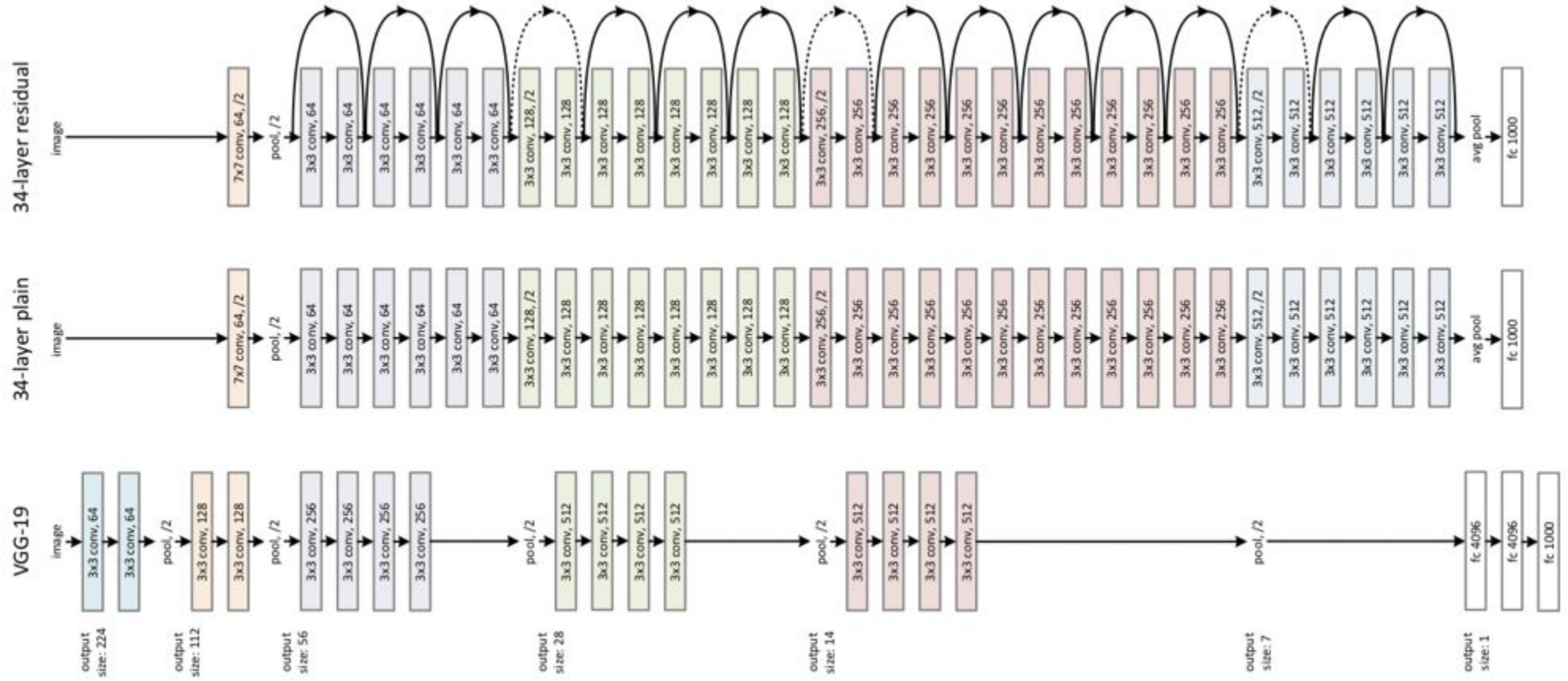
- Propuesta por Visual Geometry Group en el 2014.
- Conjunto de entrenamiento: ImageNet (14 millones de imágenes).
- Entrada: imágenes de tamaño 224 x 224 a color.
- Salida: 1000 clases.

Total de parámetros: 138.4 millones

Total de capas: 16 capas (13 convolucionales y 3 completamente conectadas).

Imagen tomada de neurohive.io

Arquitecturas de redes neuronales convolucionales: ResNet



VGG-19 con 19.6 billones de FLOPS (abajo). Red convolucional con 3.6 billones de FLOPS (centro). ResNet con 3.6 billones de FLOPS (arriba).

Imagen tomada de He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep Residual Learning for Image Recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 770-778.

Arquitecturas de redes neuronales convolucionales: ResNet

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$
conv3_x	28×28	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times8$
conv4_x	14×14	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times23$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times36$
conv5_x	7×7	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

Imagen tomada de He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep Residual Learning for Image Recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 770-778.

Arquitecturas de redes neuronales convolucionales: EfficientNets

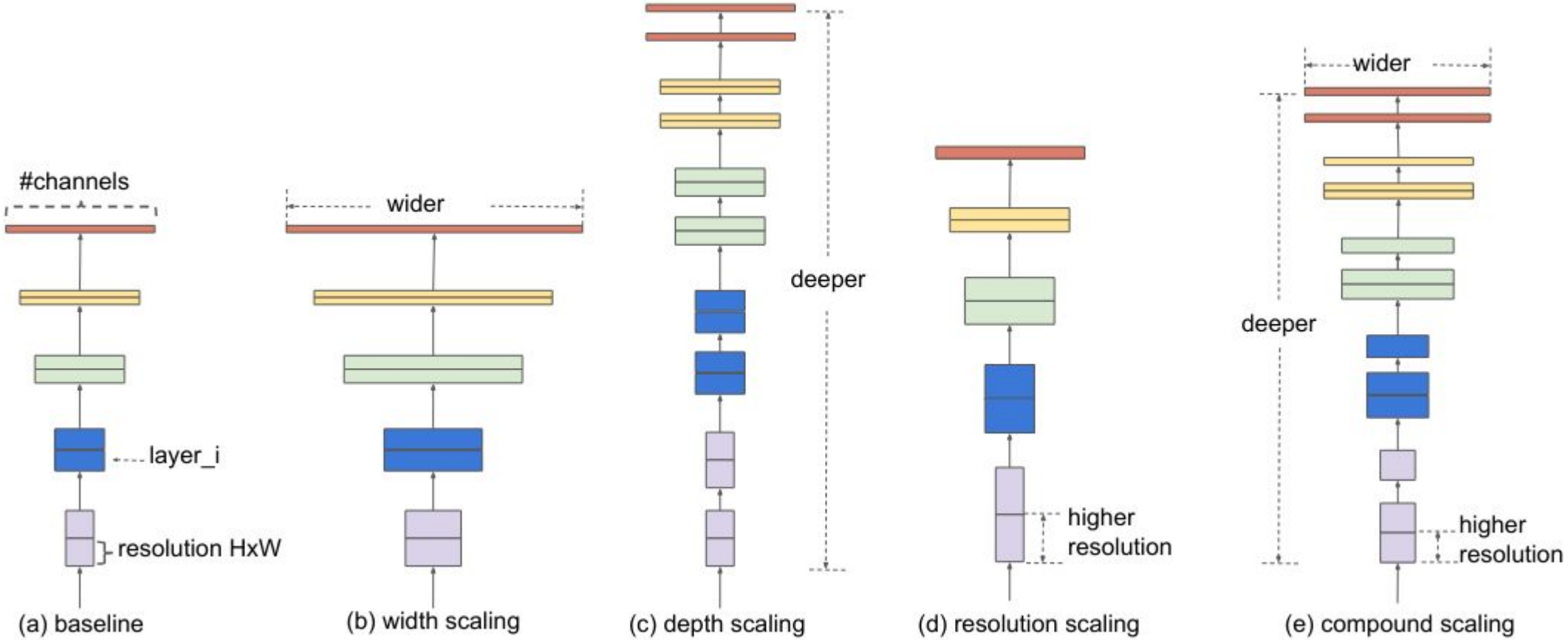
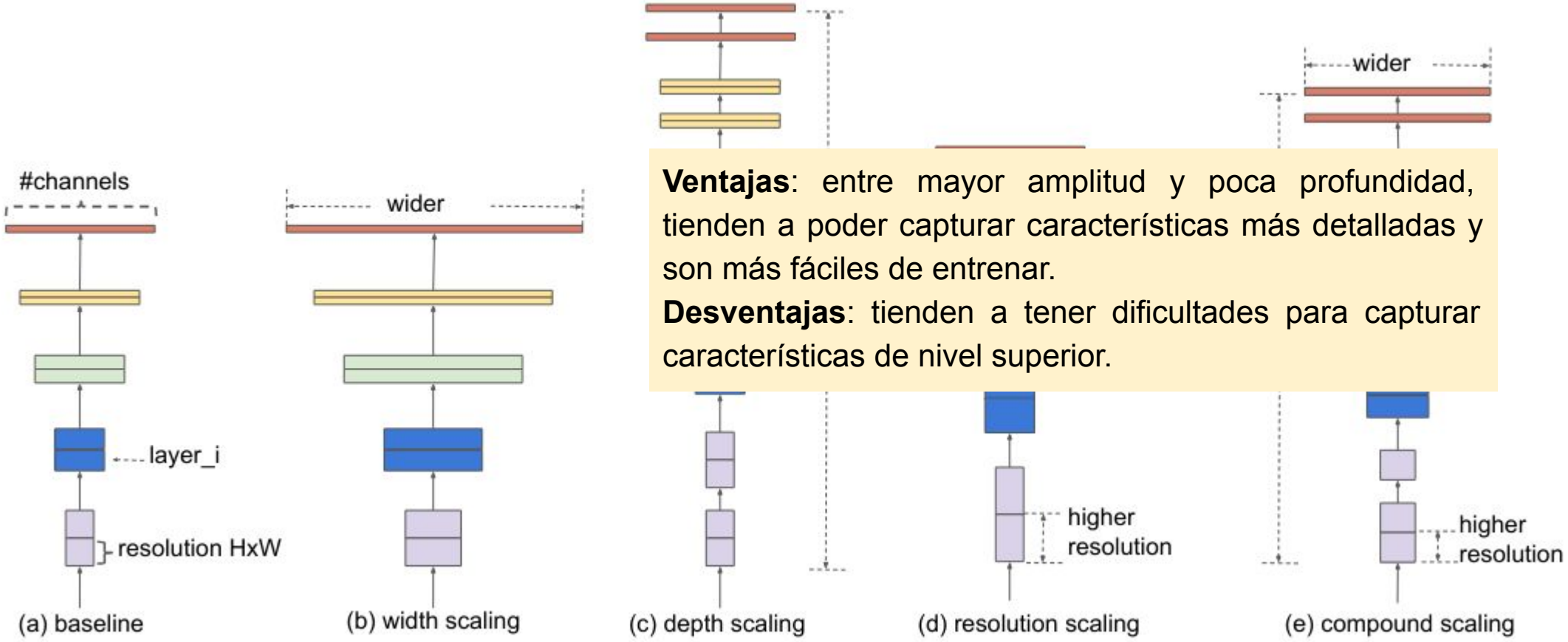


Imagen tomada de Tan, M. & Le, Q.. (2019). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. Proceedings of the 36th International Conference on Machine Learning, 97:6105-6114.

Arquitecturas de redes neuronales convolucionales: EfficientNets

Intuición



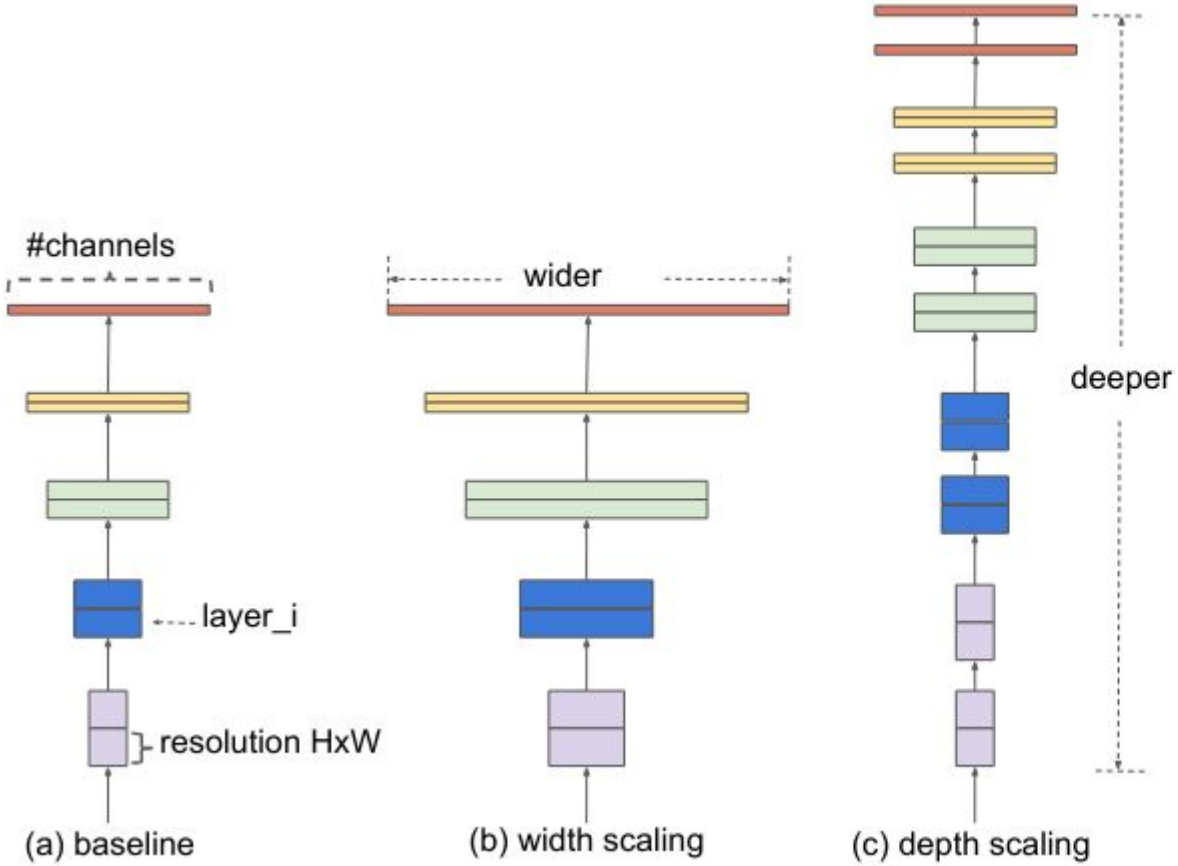
Ventajas: entre mayor amplitud y poca profundidad, tienden a poder capturar características más detalladas y son más fáciles de entrenar.

Desventajas: tienden a tener dificultades para capturar características de nivel superior.

Imagen tomada de Tan, M. & Le, Q.. (2019). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. Proceedings of the 36th International Conference on Machine Learning, 97:6105-6114.

Arquitecturas de redes neuronales convolucionales: EfficientNets

Intuición



Ventajas: a mayor número de capas (profundidad) mayor número de características (complejas) pueden capturar, y son capaces de generalizar bien en nuevas tareas.

Desventajas: son más difíciles de entrenar debido al problema del gradiente.

Imagen tomada de Tan, M. & Le, Q.. (2019). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. Proceedings of the 36th International Conference on Machine Learning, 97:6105-6114.

Arquitecturas de redes neuronales convolucionales: EfficientNets

Intuición

Ventajas: a mayor resolución, potencialmente puede capturar patrones más detallados (224x224, 299x299, 331x331, 480x480, 600x600).

Desventajas: aunque tienden a lograr una mayor exactitud (accuracy), esta se satura rápidamente después de alcanzar el 80%.

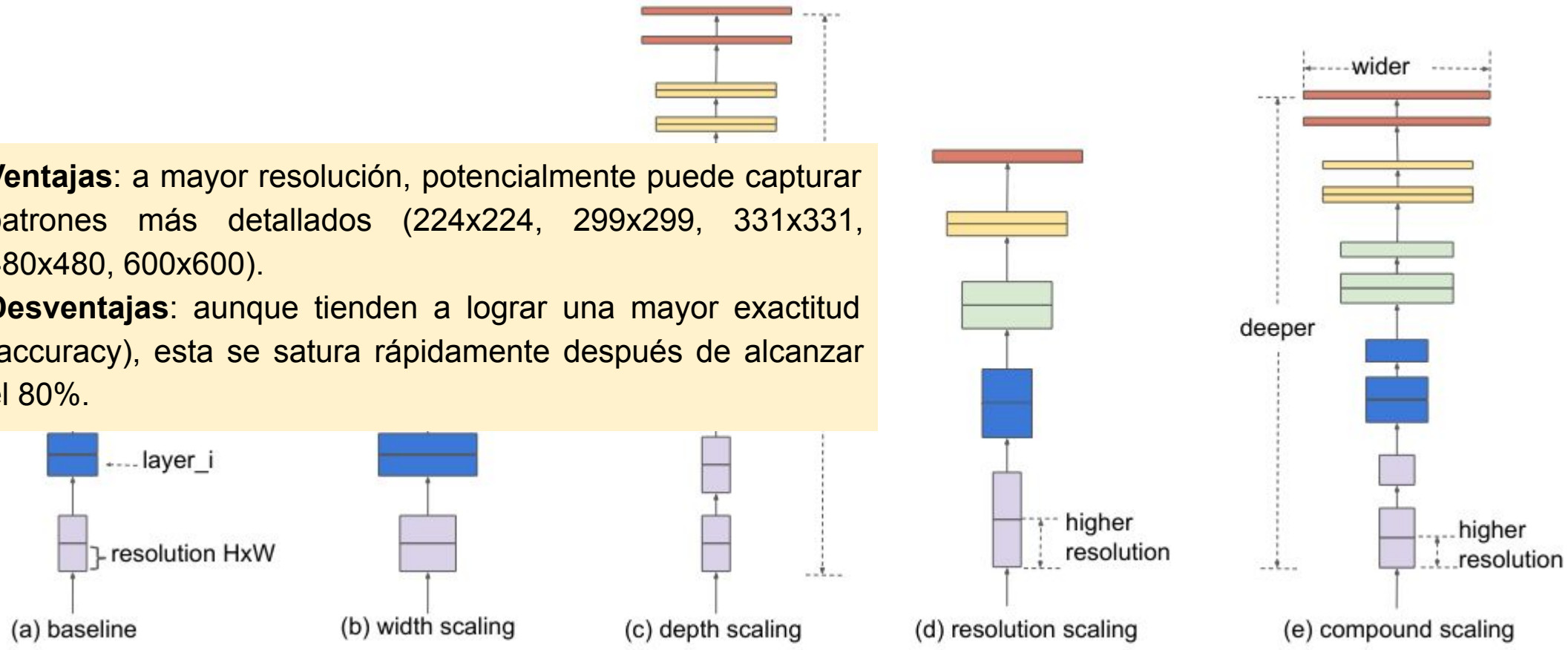


Imagen tomada de Tan, M. & Le, Q.. (2019). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. Proceedings of the 36th International Conference on Machine Learning, 97:6105-6114.

Arquitecturas de redes neuronales convolucionales: EfficientNets

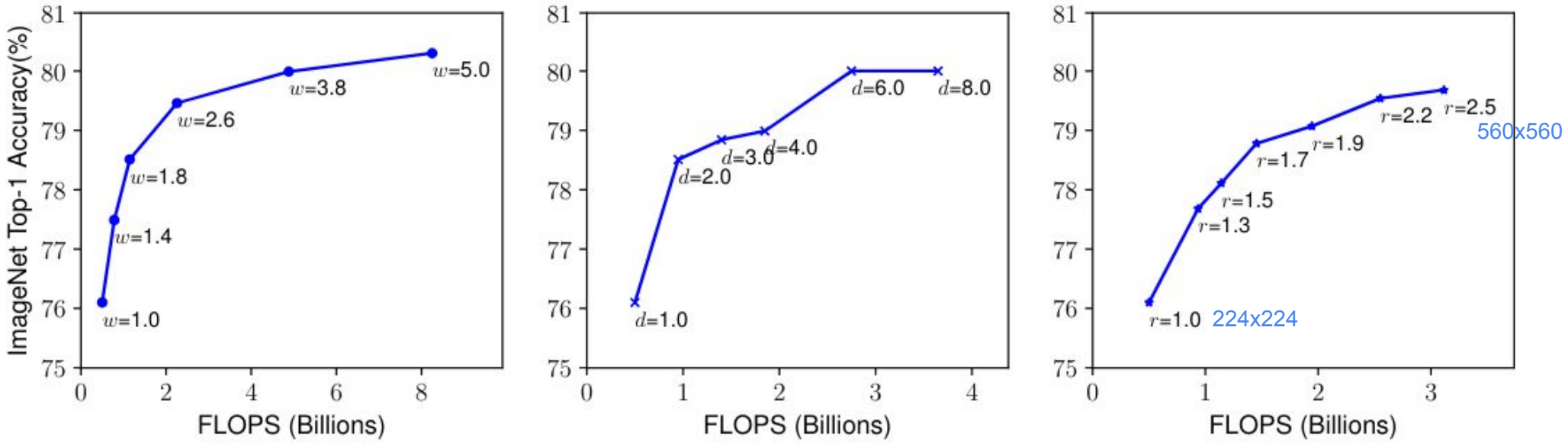


Figure 3. **Scaling Up a Baseline Model with Different Network Width (w), Depth (d), and Resolution (r) Coefficients.** Bigger networks with larger width, depth, or resolution tend to achieve higher accuracy, but the accuracy gain quickly saturate after reaching 80%, demonstrating the limitation of single dimension scaling. Baseline network is described in Table 1.

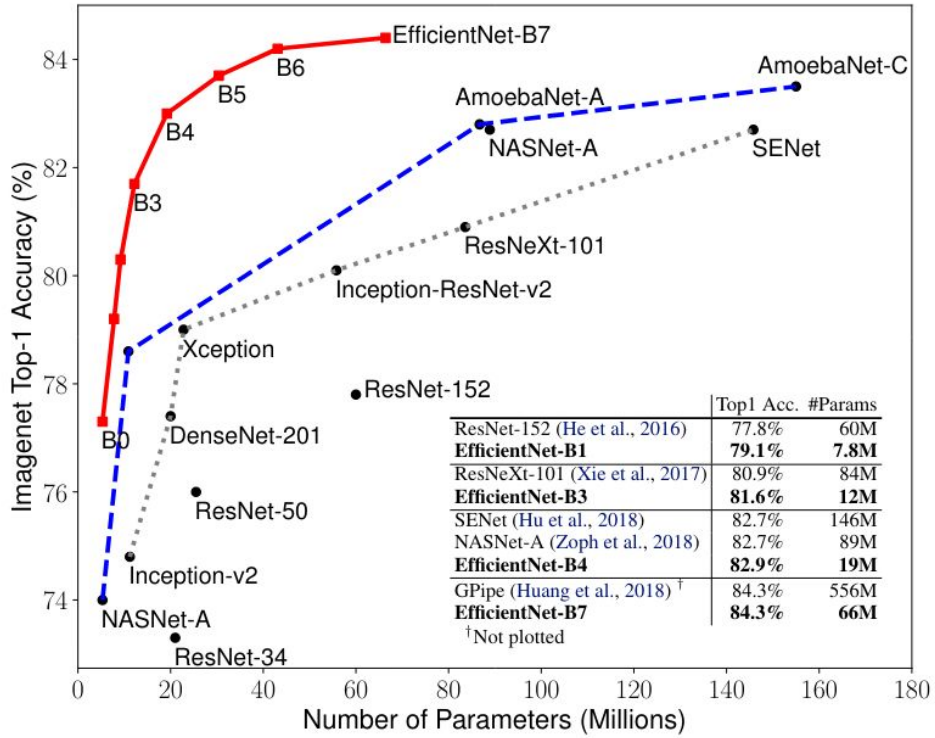
Imagen tomada de Tan, M. & Le, Q.. (2019). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. Proceedings of the 36th International Conference on Machine Learning, 97:6105-6114.

Arquitecturas de redes neuronales convolucionales: EfficientNets

Se propone un nuevo método de escalado compuesto, donde un **coeficiente de escalado** uniformemente escala amplitud, profundidad y resolución de la red.

$$\text{depth: } d = \alpha^\phi$$
$$\text{width: } w = \beta^\phi$$
$$\text{resolution: } r = \gamma^\phi$$
$$\text{s.t. } \alpha \cdot \beta^2 \cdot \gamma^2 \approx 2$$
$$\alpha \geq 1, \beta \geq 1, \gamma \geq 1$$

donde cada coeficiente puede ser determinada usando *gridsearch.*



	Top1 Acc.	#Params
ResNet-152 (He et al., 2016)	77.8%	60M
EfficientNet-B1	**79.1%**	**7.8M**
ResNeXt-101 (Xie et al., 2017)	80.9%	84M
EfficientNet-B3	**81.6%**	**12M**
SENet (Hu et al., 2018)	82.7%	146M
NASNet-A (Zoph et al., 2018)	82.7%	89M
EfficientNet-B4	**82.9%**	**19M**
GPipe (Huang et al., 2018) †	84.3%	556M
EfficientNet-B7	**84.3%**	**66M**

†Not plotted

Imagen tomada de Tan, M. & Le, Q.. (2019). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. Proceedings of the 36th International Conference on Machine Learning, 97:6105-6114.

Convoluciones dilatadas

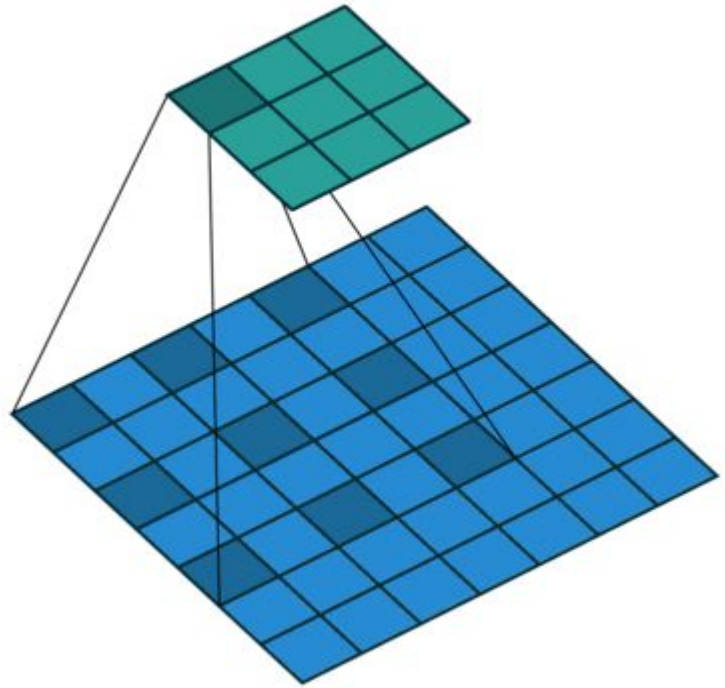
En estas convoluciones se añade el parámetro **tasa de dilatación,** definiendo un espacio entre los valores en un kernel, por ejemplo:

- Un kernel de 3x3 con una tasa de dilatación de 2 tendrá el mismo campo de visión que un kernel de 5x5, aunque solo utilizará 9 parámetros.

Ventajas:

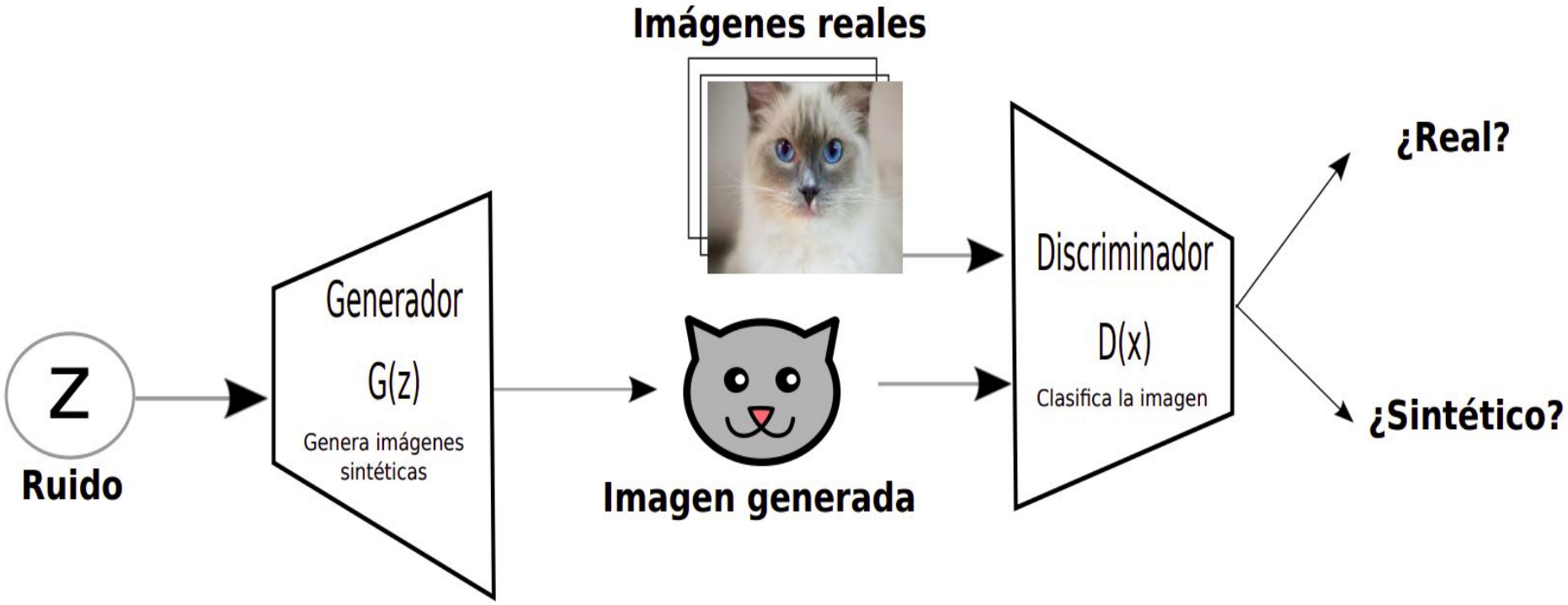
- Ofrece un campo de visión más amplio al mismo costo computacional.

Frecuente son populares en el campo de la segmentación en tiempo real. Estas capas son ideales si se requiere un campo de visión amplio y no puede permitirse múltiples convoluciones o kernels más grandes.



Imagina tomar un kernel de 5x5 y eliminar cada segunda fila y columna.

Acrecentamiento de datos
Arquitectura de las redes generativas antagónicas (GAN)



$$\min_G \max_D V(G,D) = \mathbb{E}_{x\sim p_{data}}[logD(x) + \mathbb{E}_{z\sim p_z}[log(1-D(G(z)))]$$

Acrecentamiento de datos
Evolución en la generación de imágenes sintéticas

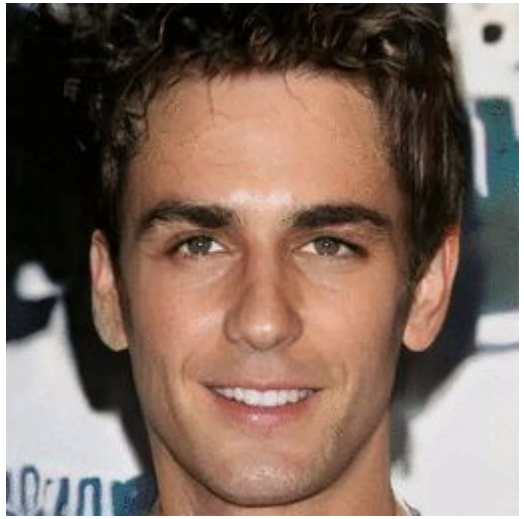
Progresión en la generación de rostros en las redes GAN.

Imagen tomada de Brundage et al., 2018.

Imagen tomada de A Style-Based Generator Architecture for Generative Adversarial Network, 2018.

Acrecentamiento de datos
Redes GAN

Lentes



Expresión

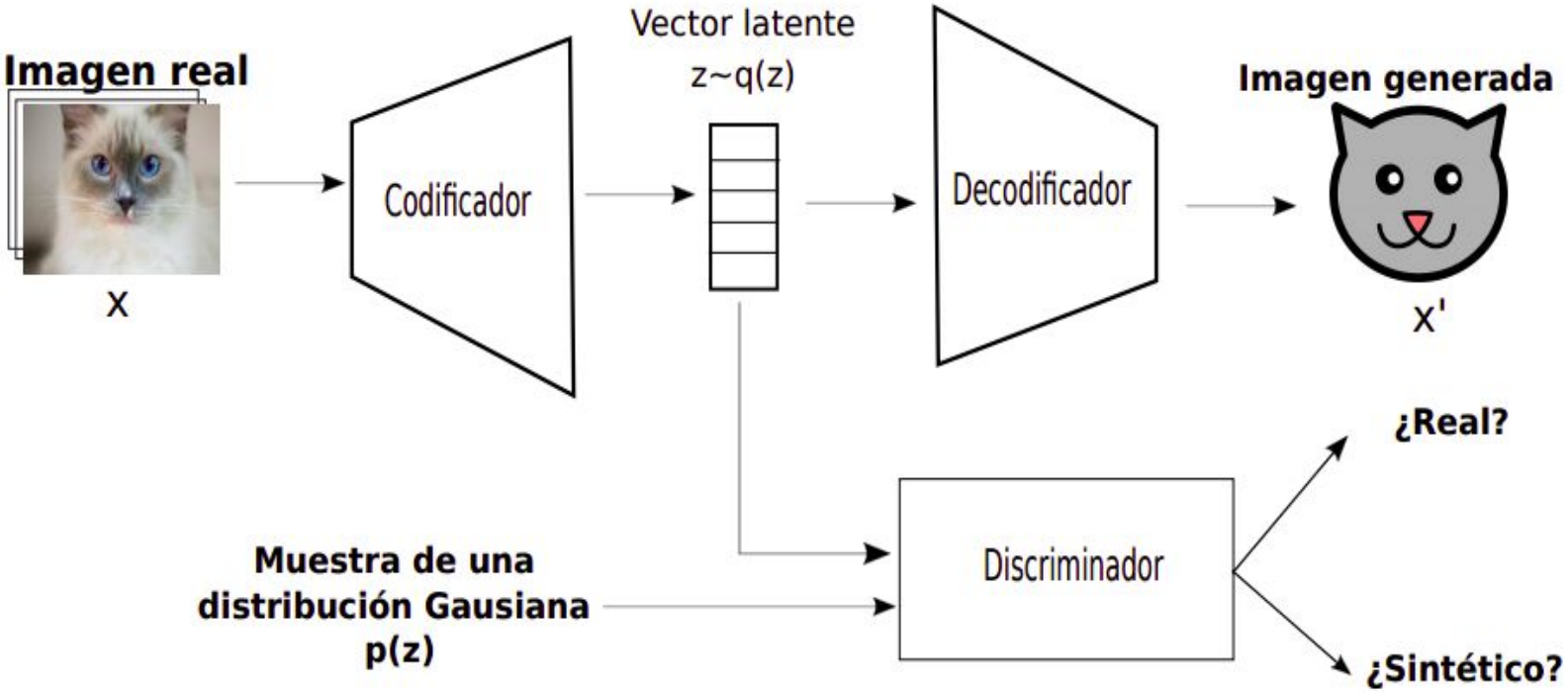
Edad

Pose

Interpretando el espacio latente en la edición de rostros

Imagen tomada de https://genforce.github.io/interfacegan/

Acrecentamiento de datos
Arquitectura de las redes auto-codificadoras variacionales (VAE)



$$q(z) = \int_x q(z|x)p_d(x)dx$$

Acrecentamiento de datos
Generación sintética usando el vector latente

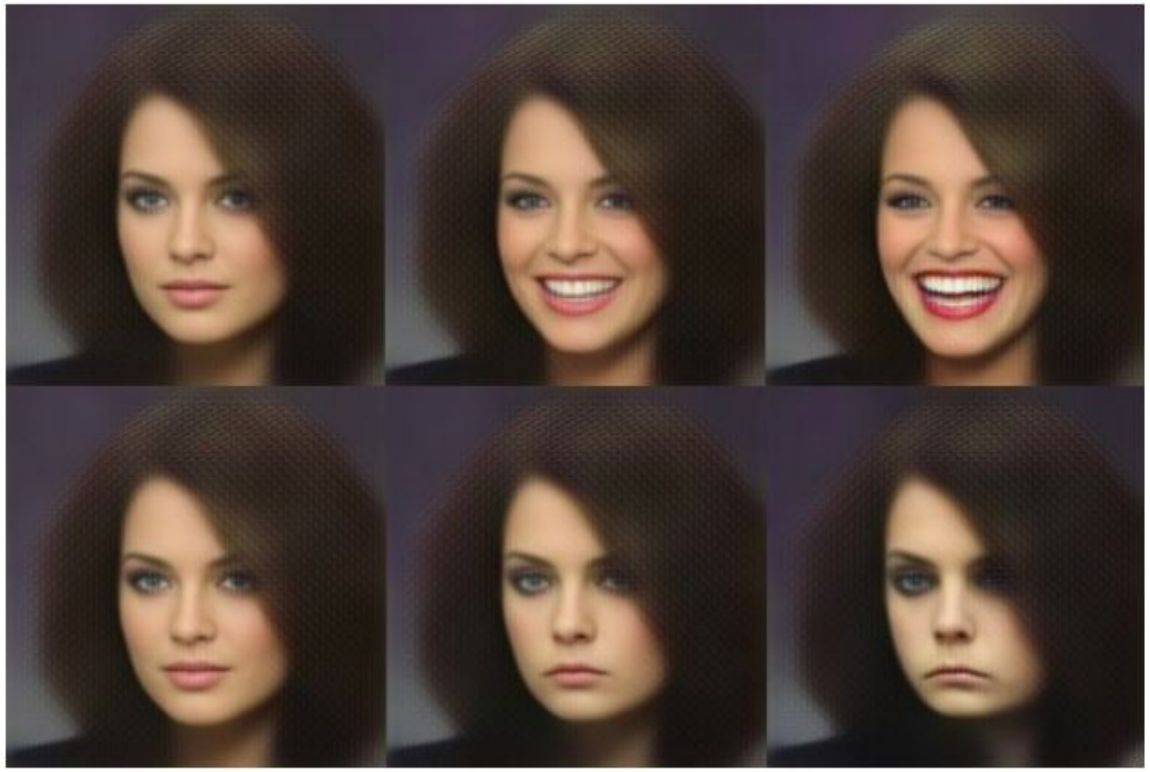


Imagen tomada de Sampling Generative Networks, 2016.

Aprendizaje por transferencia

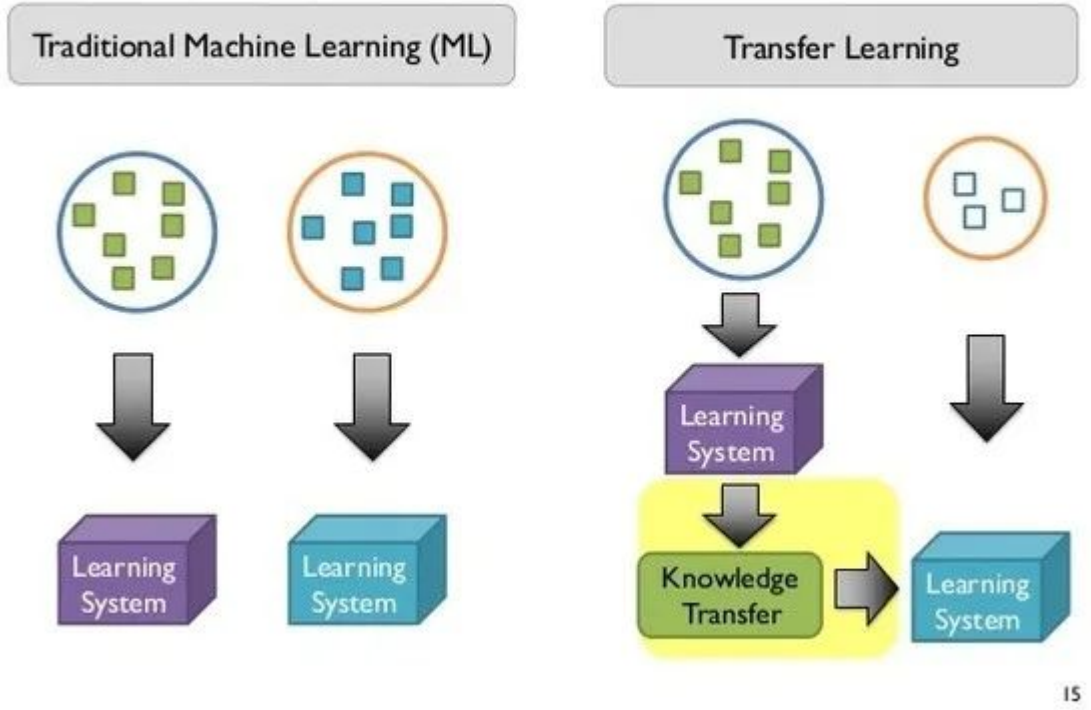


Imagen tomada de Anchit Jain, 2018.

Sitios importantes a considerar

- Hugging Face
- Pytorch
- Vision group
- SAM
- Robotic Instrument Segmentation Sub-Challenge

Time to Code

Aumentado de datos

Time to Code

ResNet

Utiliza otra red pre-entrenada

Repaso

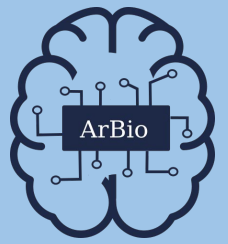
- Aprendimos las características de las redes convolucionales y los parámetros a considerarse en el entrenamiento.
- Analizamos las diferentes arquitecturas existentes para el procesamiento de imágenes usando redes convolucionales.
- Revisamos algunas arquitecturas para el acrecentamiento de datos.

Referencias

- Zhang A, Lipton Z, Li M, and Smola J. Dive into Deep Learning. 2020. Disponible en https://d2l.ai/
- Murphy, K. P. (2022). Probabilistic Machine Learning: An introduction. MIT Press. Capítulo 8, 10 y 11. Disponible en https://probml.github.io/pml-book/book1.html
- Nielsen, M. (2019). Neural Networks and Deep Learning. Capítulo 1. Disponible en http://neuralnetworksanddeeplearning.com/index.html
- Rafael C. Gonzalez, Richard Eugene Woods (2018). Digital Image Processing. Capítulo 12. Disponible en https://dl.icdst.org/pdfs/files4/01c56e081202b62bd7d3b4f8545775fb.pdf

Contacto

Dra. Blanca Vázquez
Investigadora Postdoctoral
Unidad Académica del IIMAS
en el estado de Yucatán, UNAM.
Correo: blanca.vazquez@iimas.unam.mx
Github: https://github.com/blancavazquez



Artificial Intelligence in
Biomedicine Group (ArBio)

https://iimas.unam.mx/arbio