Convolutional Neural Networks (CNNs)

Convolutional Neural Networks (CNNs)

Introducción

Introducción

Motivación

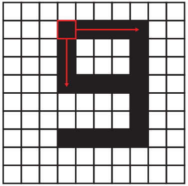
¿Es posible hacer que las redes neuronales sean capaces **ver**?

Propiedades necesarias:

1. **Localidad**
2. **Invarianza al movimiento**
3. **Composición jerárquica**

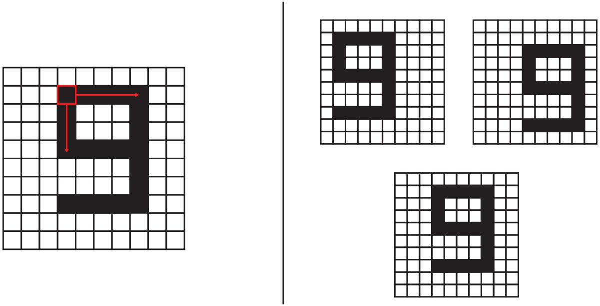
Propiedades necesarias:

1. **Localidad:** Fijarse en zonas concretas. Elementos cercanos están relacionados.
2. **Invarianza al movimiento**
3. **Composición jerárquica**



Propiedades necesarias:

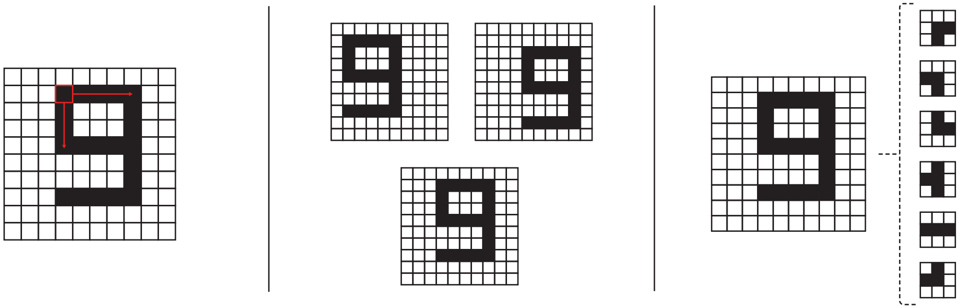
1. **Localidad:** Fijarse en zonas concretas. Elementos cercanos están relacionados.
2. **Invarianza al movimiento:** Misma salida si la posición del objeto de entrada cambia.
3. **Composición jerárquica**



Introducción

Propiedades necesarias:

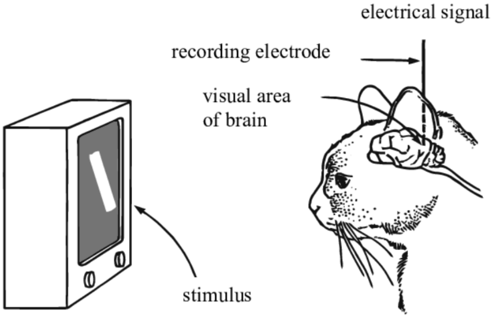
1. **Localidad:** Fijarse en zonas concretas. Elementos cercanos están relacionados.
2. **Invarianza al movimiento:** Misma salida si la posición del objeto de entrada cambia.
3. **Composición jerárquica:** Detectar patrones que definen un objeto.

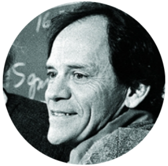


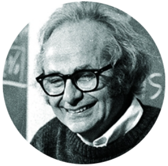
Introducción: Historia

¿Cómo funciona nuestra visión?

En 1981, David Hubel y Torsten Wiesel reciben el Nobel de medicina por sus contribuciones en el campo de la neurociencia y su investigación sobre el procesamiento visual en el cerebro.





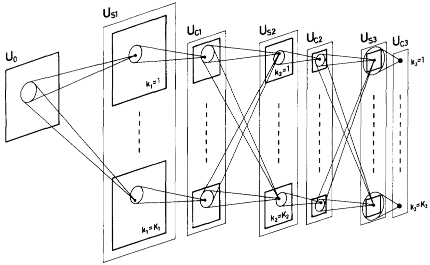


Introducción: Historia

Neocognitron

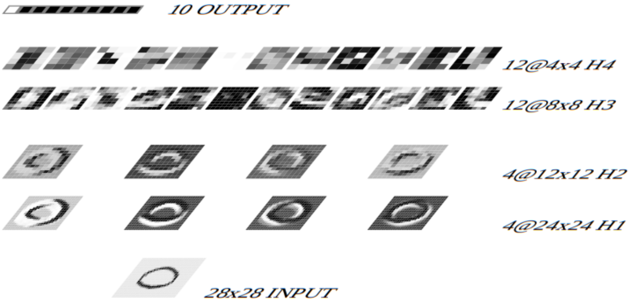
En 1980, Fukushima propone una forma de implementar el modelo jerárquico del sistema nervioso visual de Hubel y Wiesel utilizando redes neuronales.

- Creado a partir de **convoluciones**.
- Capaz de detectar composiciones jerárquicas.
- Algoritmo de entrenamiento ineficiente.



Convolutional networks

En 1990, LeCun entrena una red convolucional utilizando backpropagation. Aboga por el aprendizaje de características end-to-end en la clasificación de imágenes.



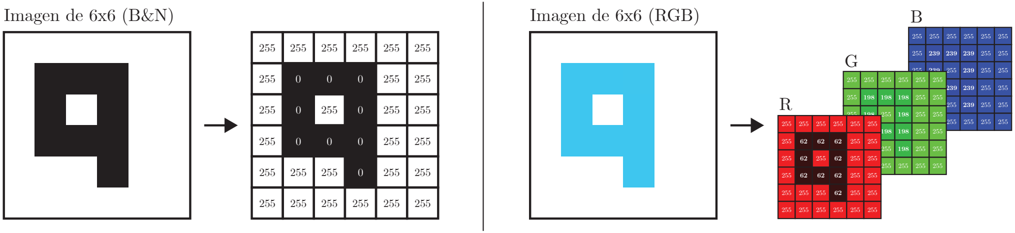
Convolutional Neural Networks (CNNs)

Convoluciones

Codificación de imágenes

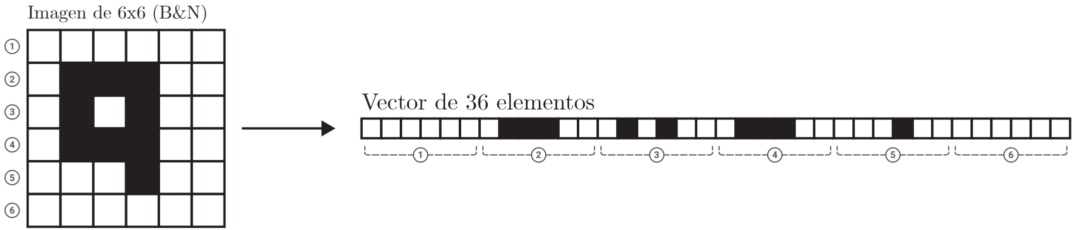
Las imágenes son matrices de **pixels**. Cada uno posee un valor entre $[0, 255]$ que representan la intensidad.

- Si la imagen es en blanco y negro, solo tendremos **una matriz**.
- Si es en color, tendremos **tres matrices**, una por cada **canal** (Rojo, Verde y Azul).



Procesamiento dentro de una red

No pueden ser tratados como vectores "no estructurados" normales, han de ser invariantes al movimiento.

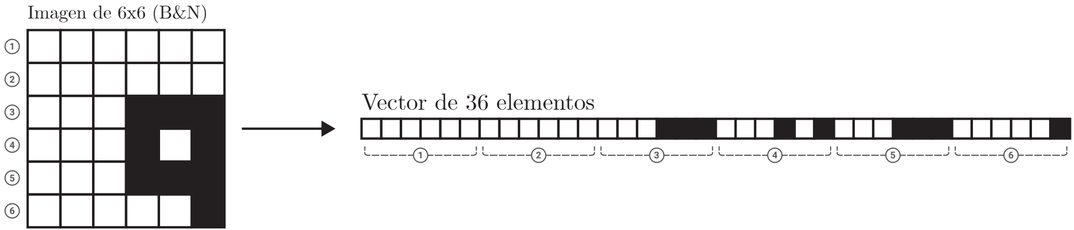


Además, los modelos resultantes tendrían un tamaño enorme:

- Una pequeña imagen en B&N de 100×100 generaría un vector de 10000.

Procesamiento dentro de una red

No pueden ser tratados como vectores "no estructurados" normales, han de ser invariantes al movimiento.



Además, los modelos resultantes tendrían un tamaño enorme:

- Una pequeña imagen en B&N de 100×100 generaría un vector de 10000.

Convoluciones

Definición

Operación matemática capaz de extraer características o patrones de unos datos de entrada, típicamente imágenes o señales.

Compuesta por:

- Datos de entrada $\mathbf{x}$.
- Uno o varios **kernels** o filtros $\mathbf{u}$.
- Salida $\mathbf{o}$.

Convoluciones: 1D

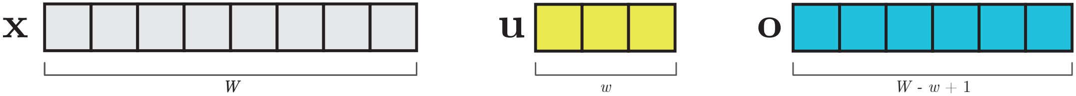
Si nuestros datos de entrada son de una dimensión (una señal, por ejemplo), tendremos que aplicar la **convolución 1D**.

Tendremos por tanto:

- Vector 1D de entrada $\mathbf{x} \in \mathbb{R}^W$.
- Vector 1D kernel $\mathbf{u} \in \mathbb{R}^w$.
- Vector 1D de salida $\mathbf{o} \in \mathbb{R}^{W-w+1}$.

La operación de convolución se define como:

$$(\mathbf{x} \circledast \mathbf{u})[i] = \sum_{m=0}^{w-1} x_{m+i} \cdot u_m$$



Utilizaremos el operador $\circledast$ para referirnos a la convolución.

Convoluciones: 1D

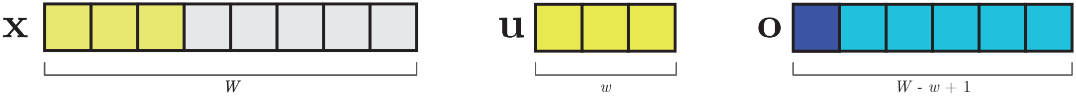
Si nuestros datos de entrada son de una dimensión (una señal, por ejemplo), tendremos que aplicar la **convolución 1D**.

Tendremos por tanto:

- Vector 1D de entrada $\mathbf{x} \in \mathbb{R}^W$.
- Vector 1D kernel $\mathbf{u} \in \mathbb{R}^w$.
- Vector 1D de salida $\mathbf{o} \in \mathbb{R}^{W-w+1}$.

La operación de convolución se define como:

$$(\mathbf{x} \circledast \mathbf{u})[i] = \sum_{m=0}^{w-1} x_{m+i} \cdot u_m$$



Utilizaremos el operador $\circledast$ para referirnos a la convolución.

Convoluciones: 1D

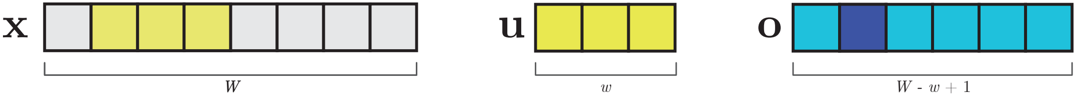
Si nuestros datos de entrada son de una dimensión (una señal, por ejemplo), tendremos que aplicar la **convolución 1D**.

Tendremos por tanto:

- Vector 1D de entrada $\mathbf{x} \in \mathbb{R}^W$.
- Vector 1D kernel $\mathbf{u} \in \mathbb{R}^w$.
- Vector 1D de salida $\mathbf{o} \in \mathbb{R}^{W-w+1}$.

La operación de convolución se define como:

$$(\mathbf{x} \circledast \mathbf{u})[i] = \sum_{m=0}^{w-1} x_{m+i} \cdot u_m$$



Utilizaremos el operador $\circledast$ para referirnos a la convolución.

Convoluciones: 1D

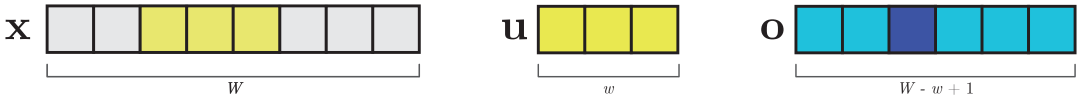
Si nuestros datos de entrada son de una dimensión (una señal, por ejemplo), tendremos que aplicar la **convolución 1D**.

Tendremos por tanto:

- Vector 1D de entrada $\mathbf{x} \in \mathbb{R}^W$.
- Vector 1D kernel $\mathbf{u} \in \mathbb{R}^w$.
- Vector 1D de salida $\mathbf{o} \in \mathbb{R}^{W-w+1}$.

La operación de convolución se define como:

$$(\mathbf{x} \circledast \mathbf{u})[i] = \sum_{m=0}^{w-1} x_{m+i} \cdot u_m$$



Utilizaremos el operador $\circledast$ para referirnos a la convolución.

Convoluciones: 1D

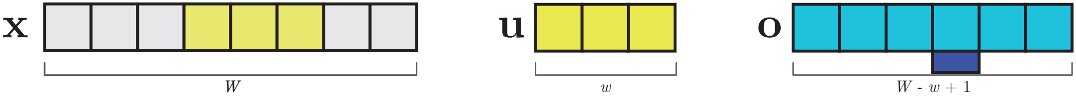
Si nuestros datos de entrada son de una dimensión (una señal, por ejemplo), tendremos que aplicar la **convolución 1D**.

Tendremos por tanto:

- Vector 1D de entrada $\mathbf{x} \in \mathbb{R}^W$.
- Vector 1D kernel $\mathbf{u} \in \mathbb{R}^w$.
- Vector 1D de salida $\mathbf{o} \in \mathbb{R}^{W-w+1}$.

La operación de convolución se define como:

$$(\mathbf{x} \circledast \mathbf{u})[i] = \sum_{m=0}^{w-1} x_{m+i} \cdot u_m$$



Utilizaremos el operador $\circledast$ para referirnos a la convolución.

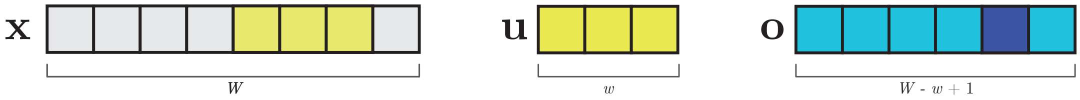
Si nuestros datos de entrada son de una dimensión (una señal, por ejemplo), tendremos que aplicar la **convolución 1D**.

Tendremos por tanto:

- Vector 1D de entrada $\mathbf{x} \in \mathbb{R}^W$.
- Vector 1D kernel $\mathbf{u} \in \mathbb{R}^w$.
- Vector 1D de salida $\mathbf{o} \in \mathbb{R}^{W-w+1}$.

La operación de convolución se define como:

$$(\mathbf{x} \circledast \mathbf{u})[i] = \sum_{m=0}^{w-1} x_{m+i} \cdot u_m$$



Utilizaremos el operador $\circledast$ para referirnos a la convolución.

Convoluciones: 1D

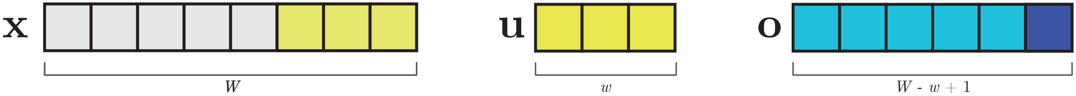
Si nuestros datos de entrada son de una dimensión (una señal, por ejemplo), tendremos que aplicar la **convolución 1D**.

Tendremos por tanto:

- Vector 1D de entrada $\mathbf{x} \in \mathbb{R}^W$.
- Vector 1D kernel $\mathbf{u} \in \mathbb{R}^w$.
- Vector 1D de salida $\mathbf{o} \in \mathbb{R}^{W-w+1}$.

La operación de convolución se define como:

$$(\mathbf{x} \circledast \mathbf{u})[i] = \sum_{m=0}^{w-1} x_{m+i} \cdot u_m$$



Utilizaremos el operador $\circledast$ para referirnos a la convolución.

Referencias

1. **Lecture 5: Convolutional networks**