

# Convolutional Neural Networks (CCNs)

# Convolution (splot)

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

↑  
sygnał

↓  
convolution kernel  
(jądro splotu)

↓  
sygnał uśredniony w czasie

# Convolution (splot): przypadek dyskretny

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

Na komputerze mamy do czynienia z sygnałami dyskretnymi, więc całkę zamieniamy na sumę:

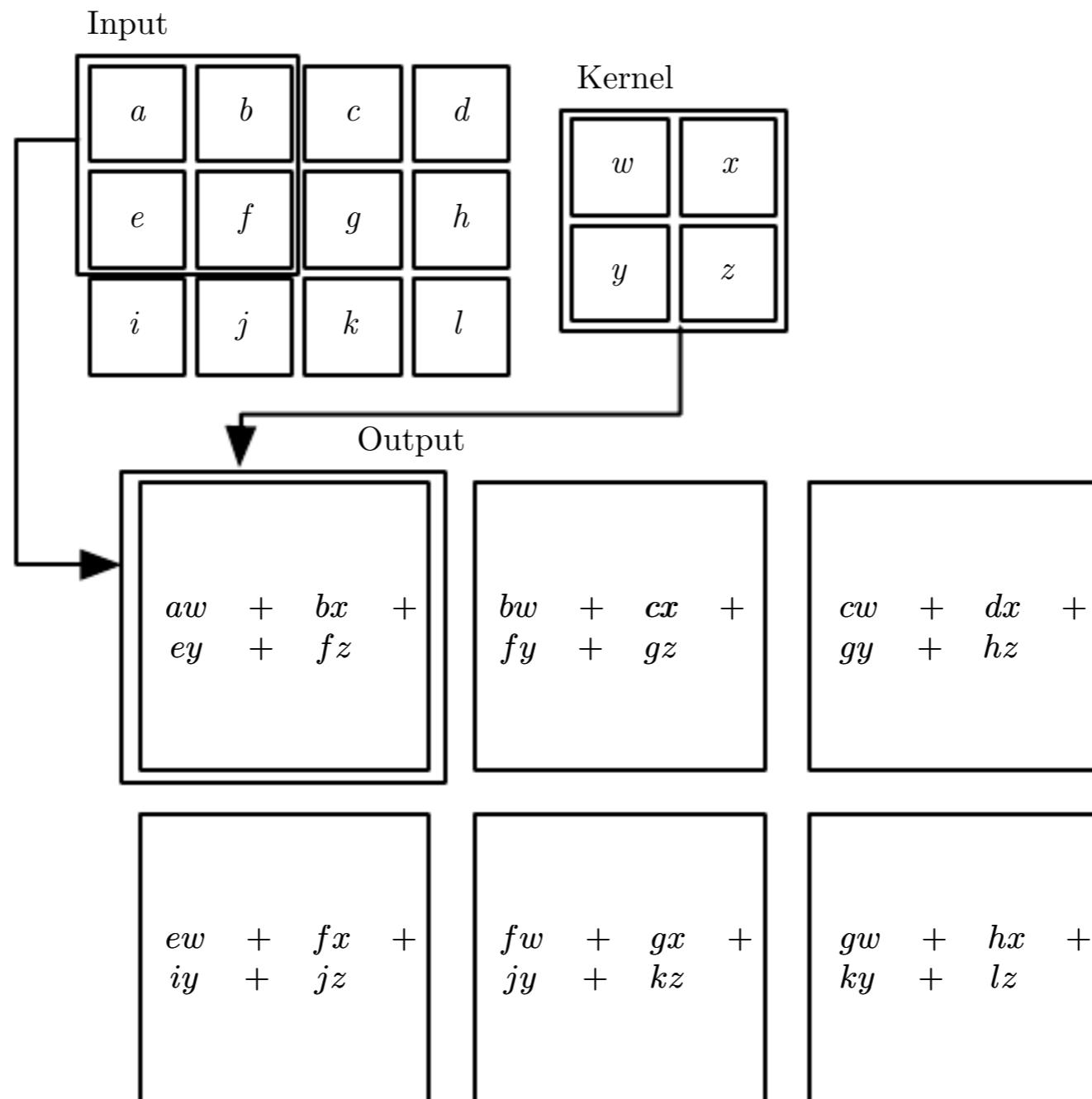
$$s(t) = (x * w)(t) = \sum_{a=-\infty}^{\infty} x(a)w(t-a)$$

Zazwyczaj będzie to suma skończona bo jądro splotu ma będzie miało tylko kilka niezerowych wyrazów.

W przypadku obrazków 2D:

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

# Działanie splotu

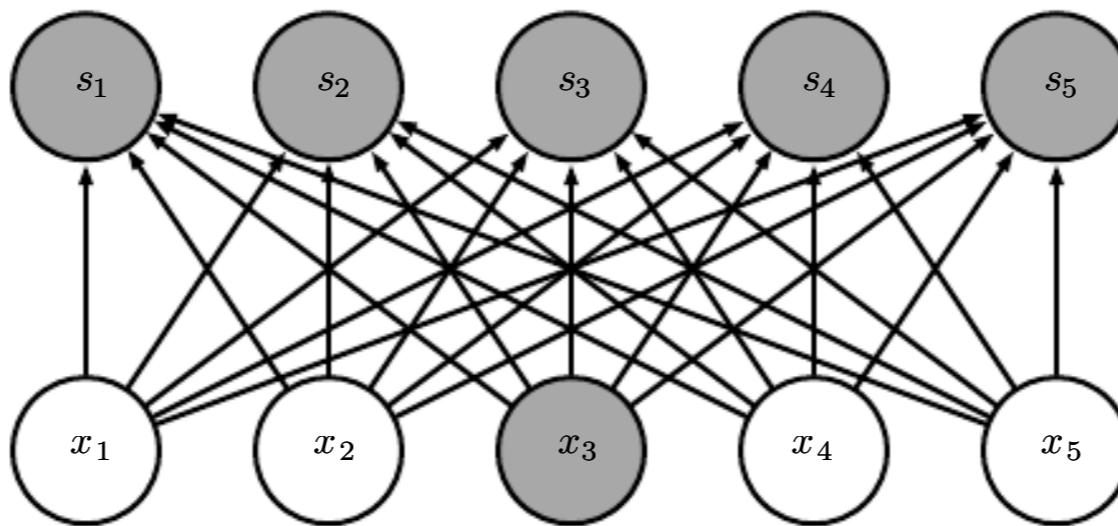
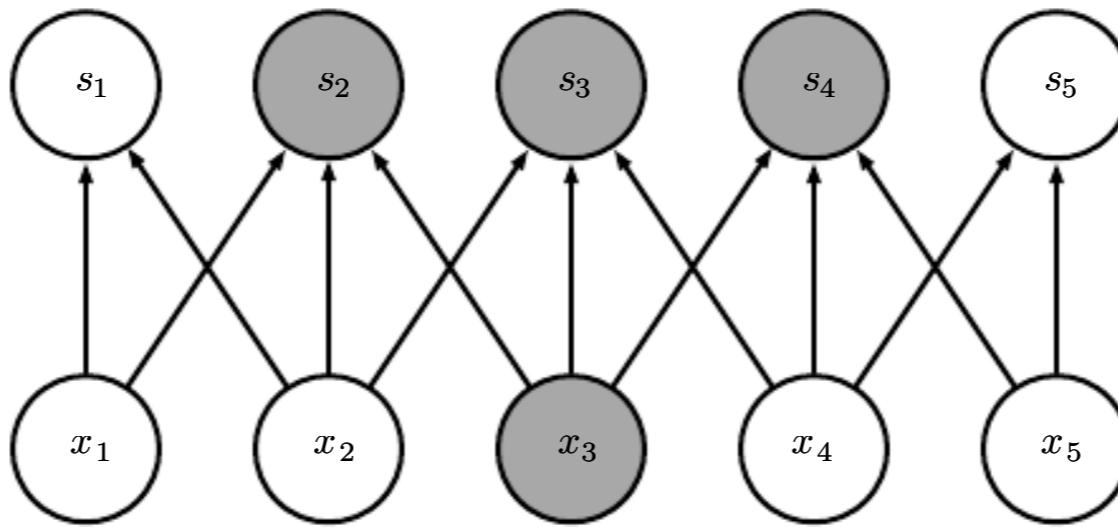


# Co to daje?

- Mniejsza liczba parametrów.
- Współdzielone parametry.
- Zwraca uwagę na lokalizację.
- Niezmienność ze względu na transformacje (np. translacje).

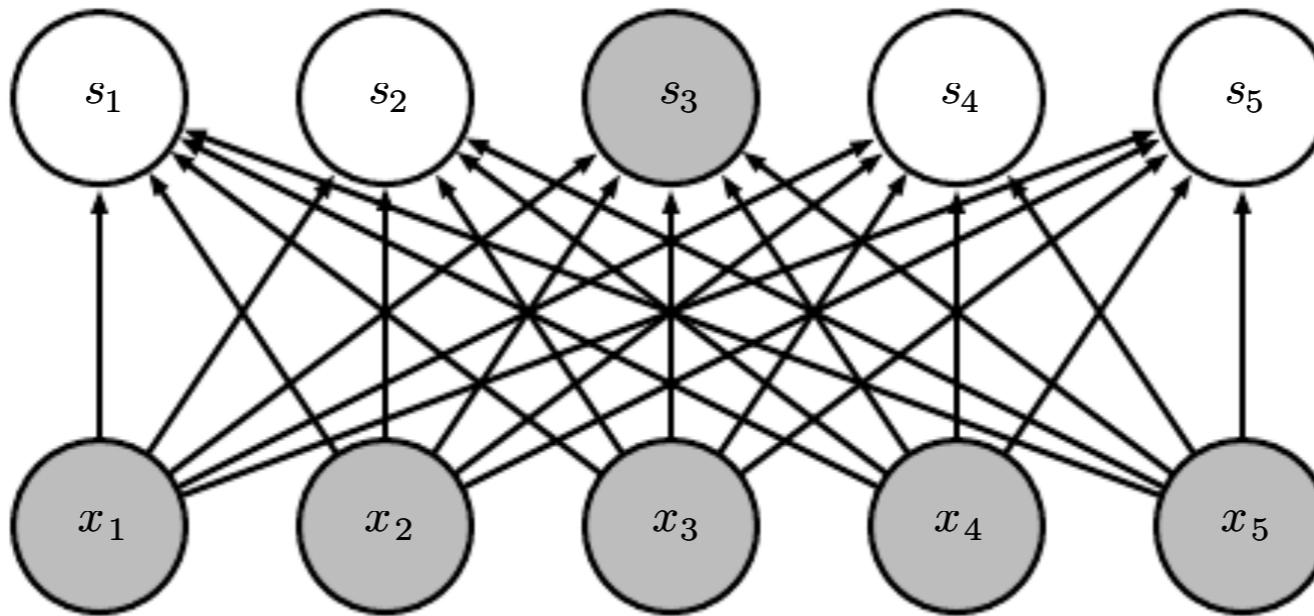
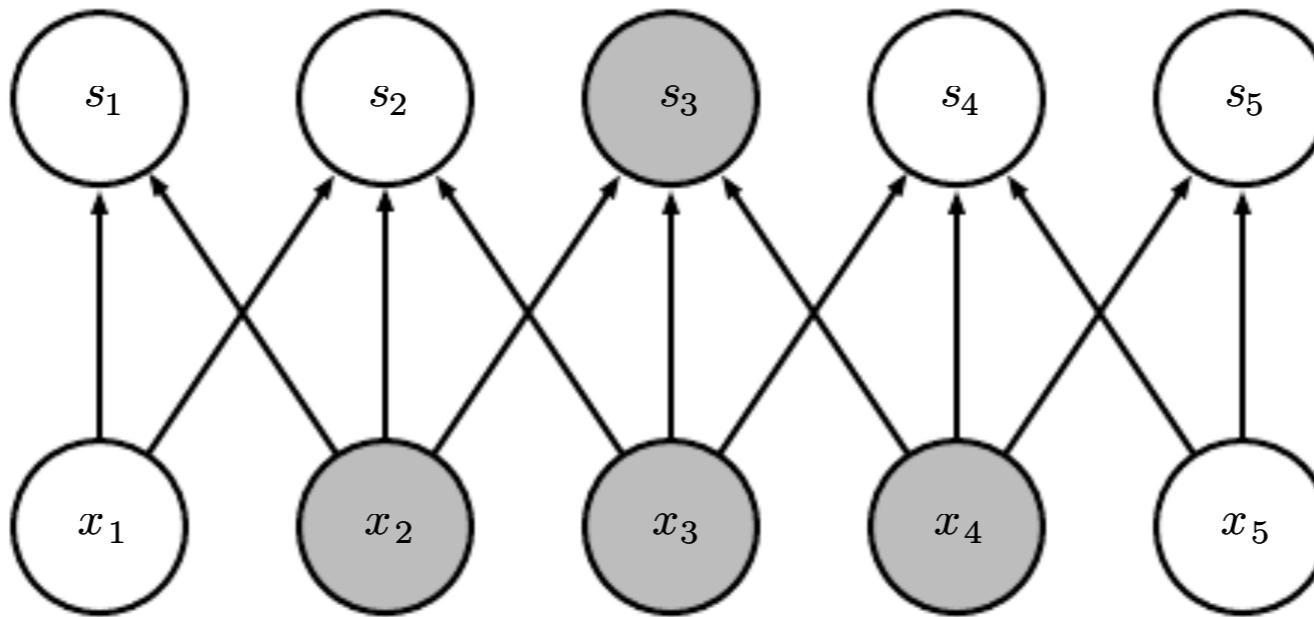
# Mniejsza liczba parametrów

Sparse connections



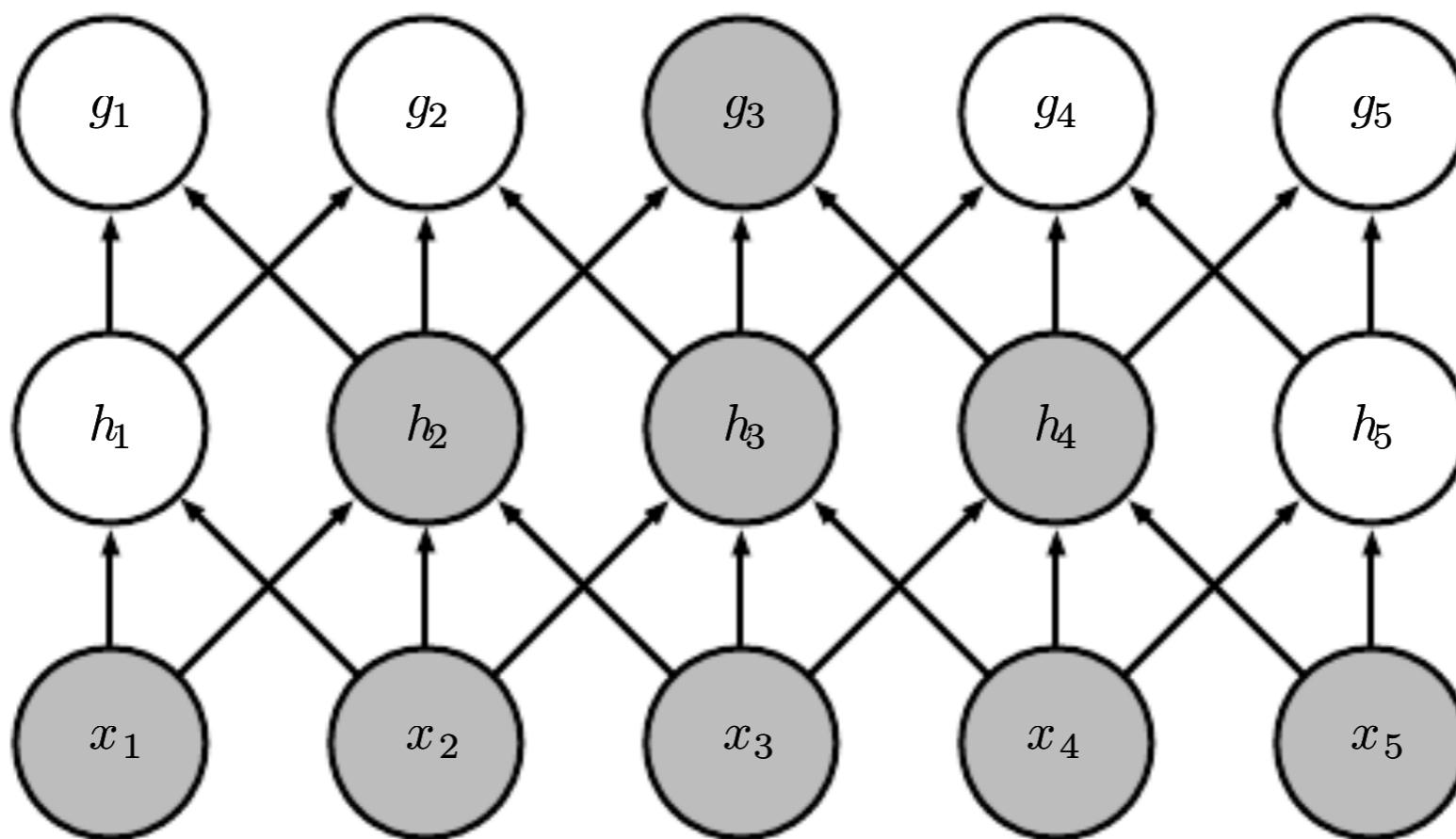
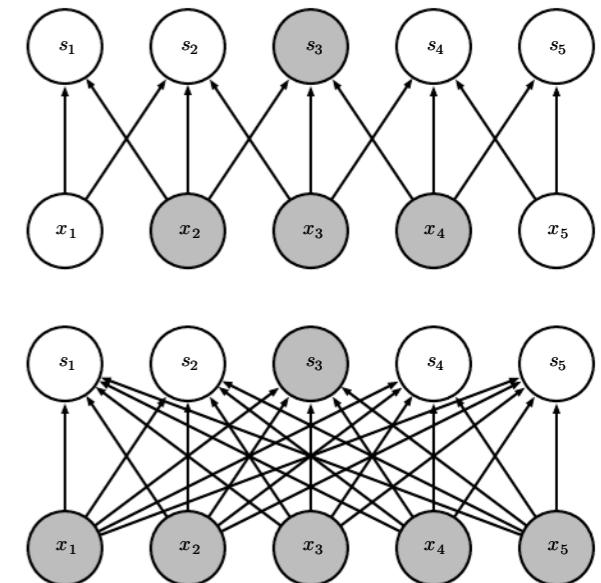
# Lokalizacja

Receptive field



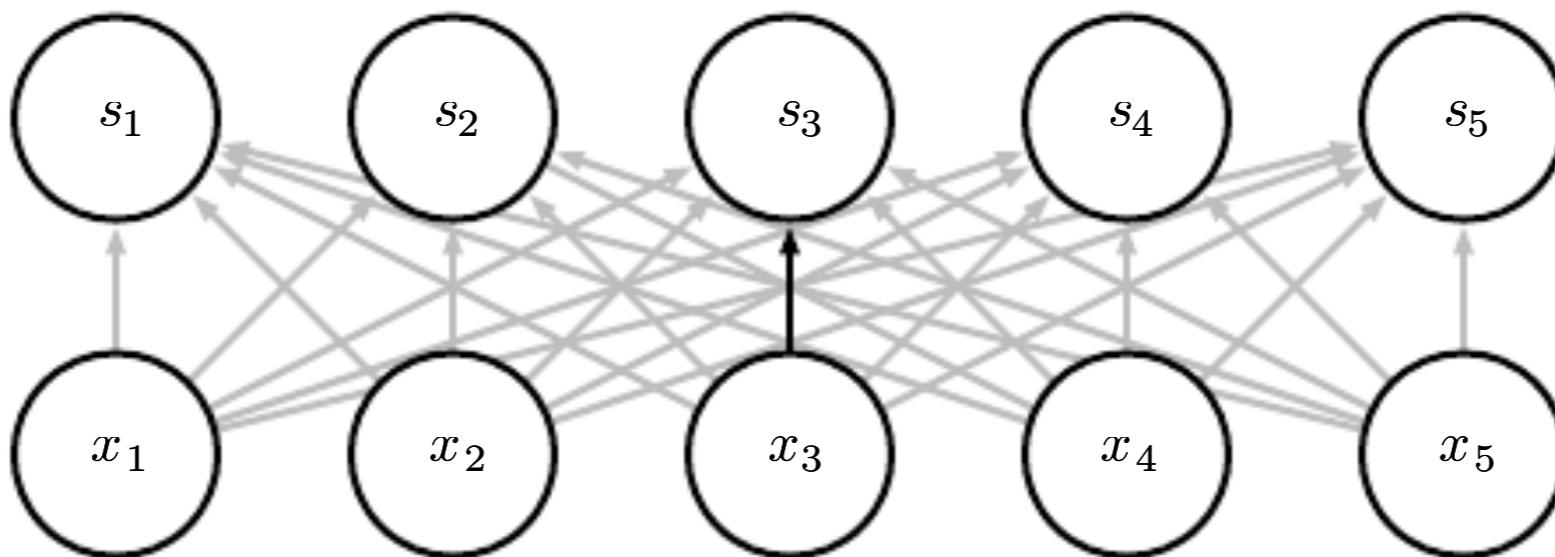
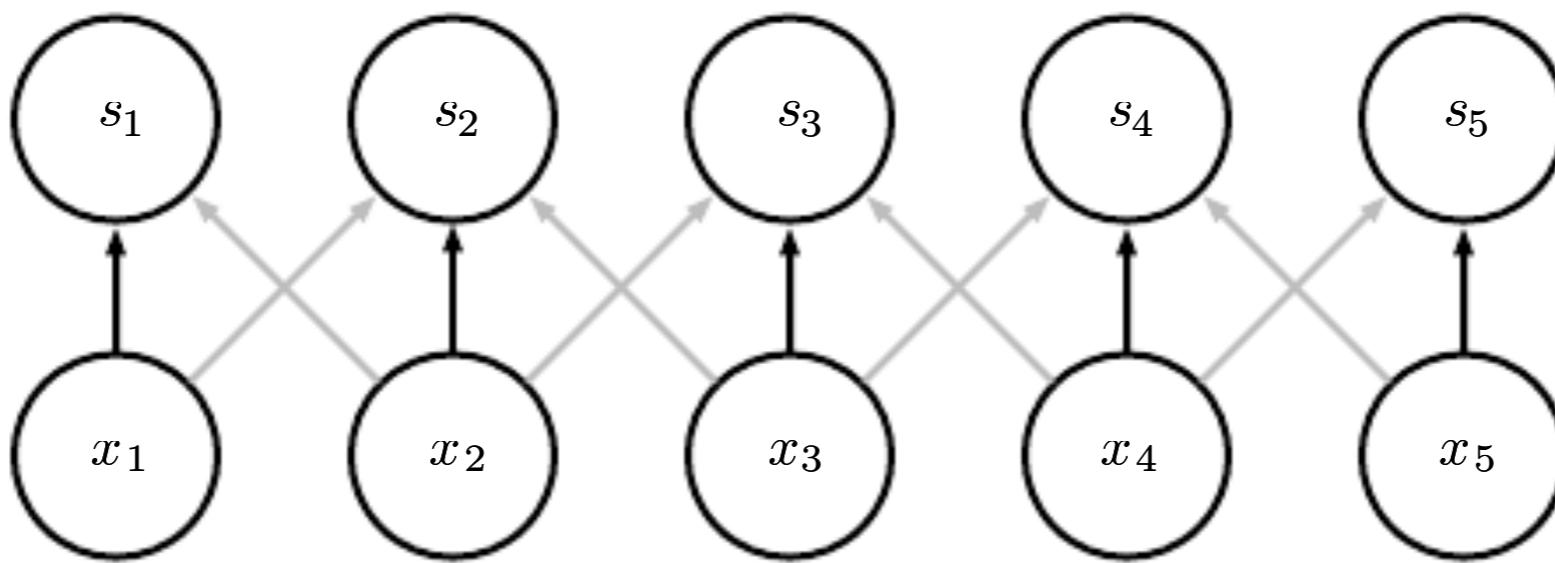
# Propagacja lokalizacji

Receptive field



# Współdzielenie parametrów

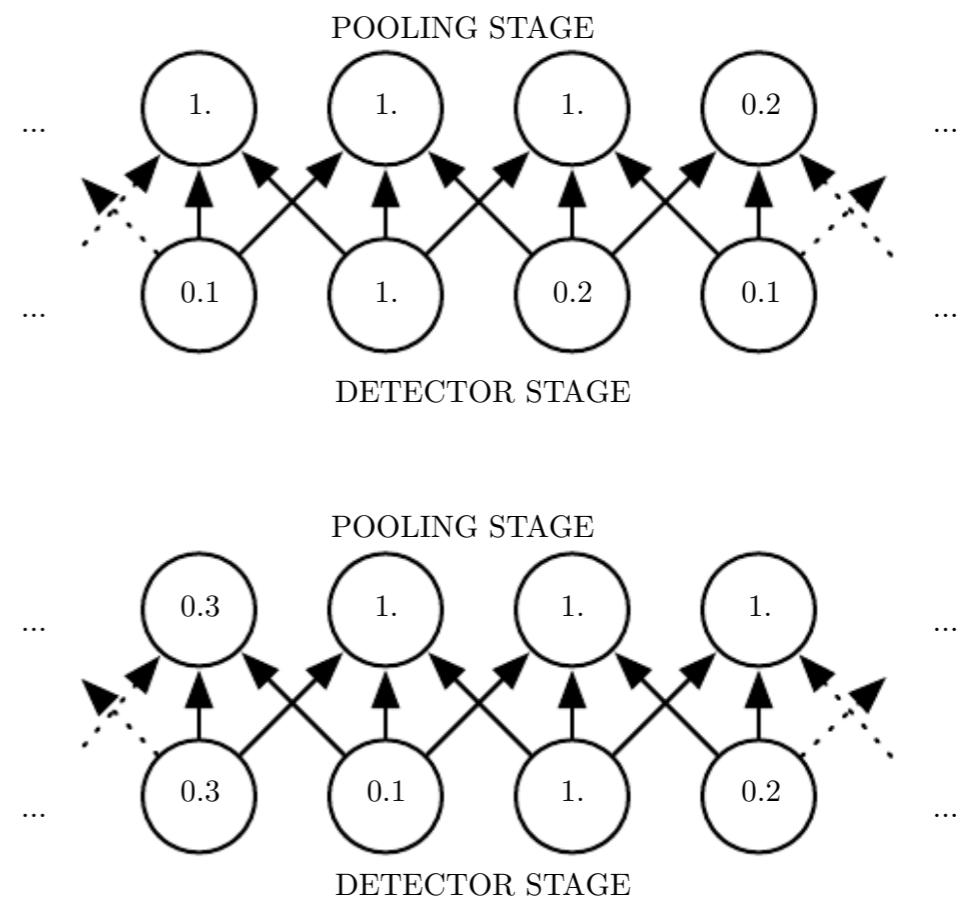
Parameter sharing



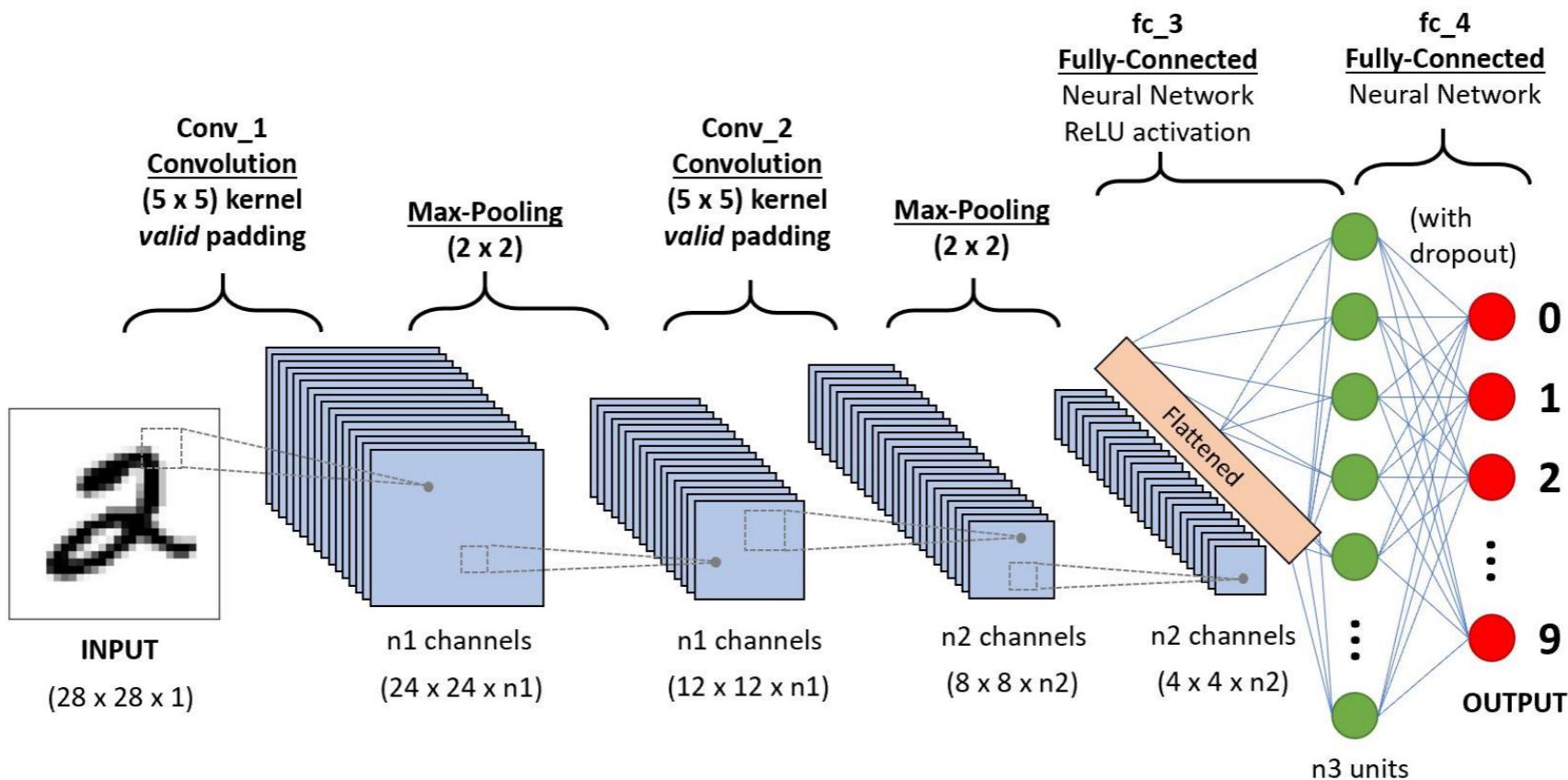
# Pooling

- Za warstwą Conv zazwyczaj znajduje się warstwa poolingu:

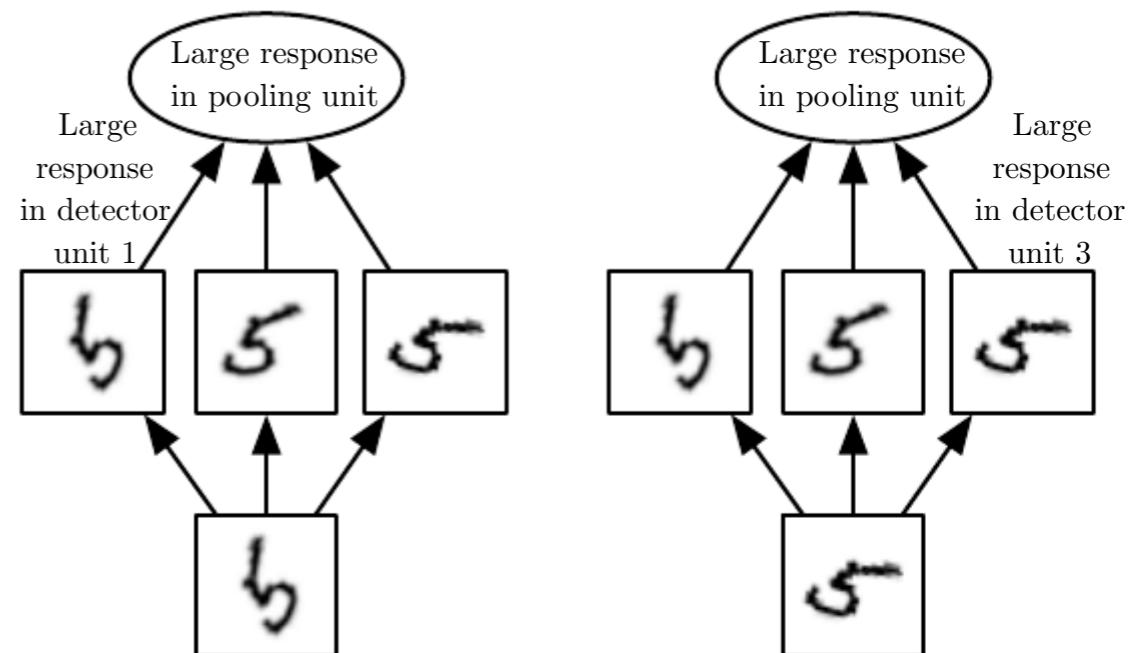
- MaxPool wybiera największą wartość z określonego “okna”.
- AveragePool wybiera wartość średnią.
- Przepuszcza dominujące własności
- MaxPooling zmniejsza szum
- Wprowadza niezmienność ze względu na przesunięcia.



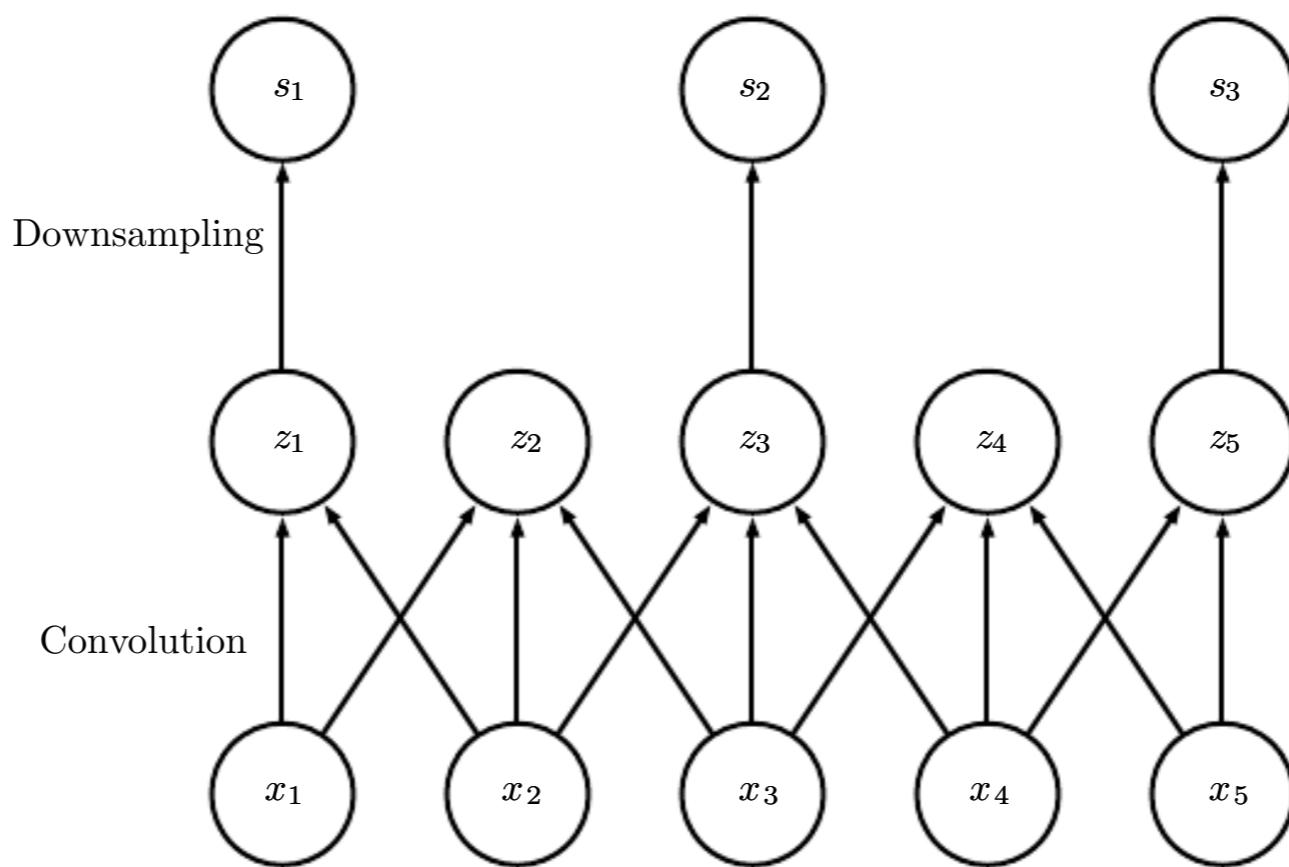
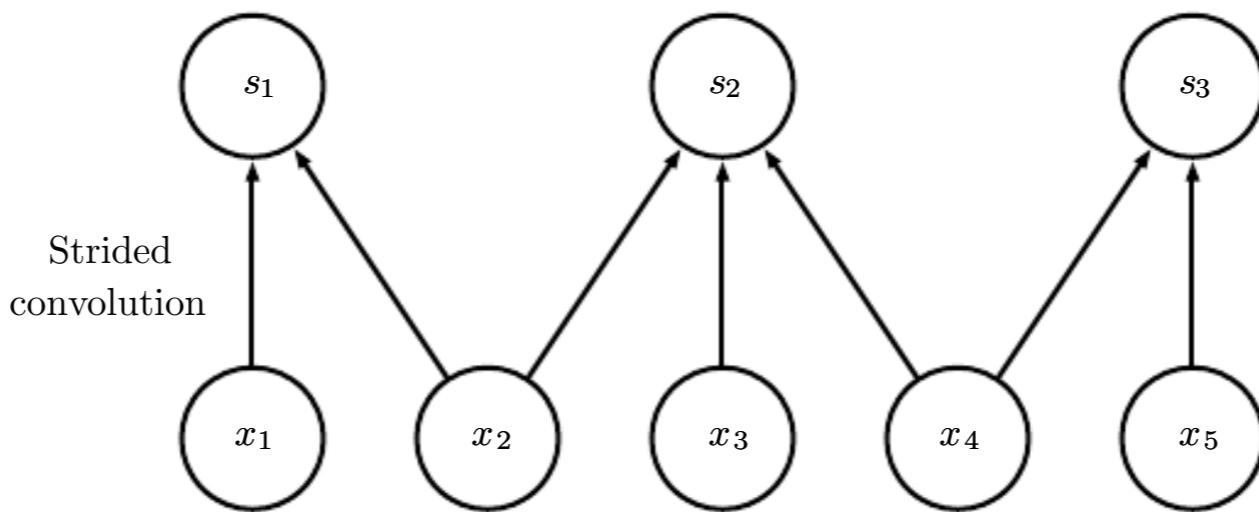
# Filtry



- na warstwę Conv zazwyczaj składa się wiele osobnych *filtrów*

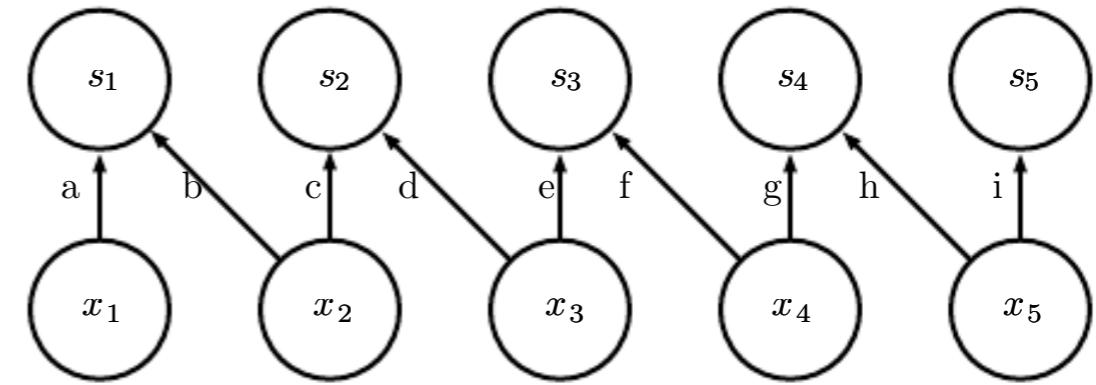
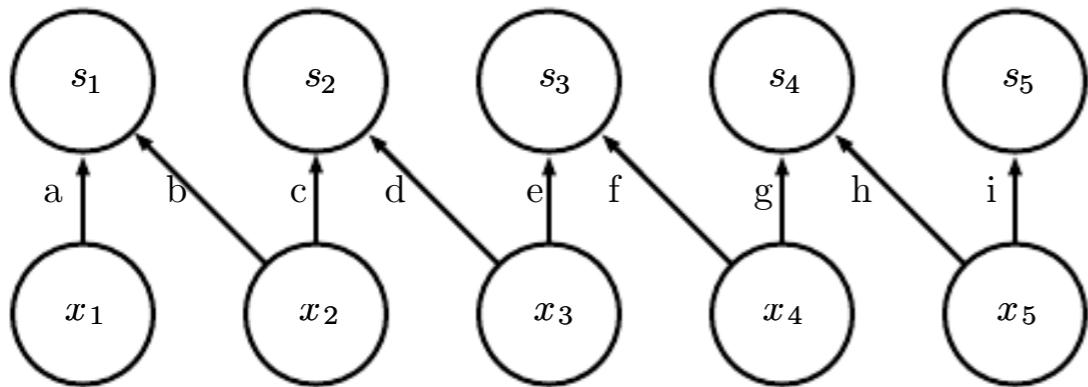


# Warianty CNN: strided convolution

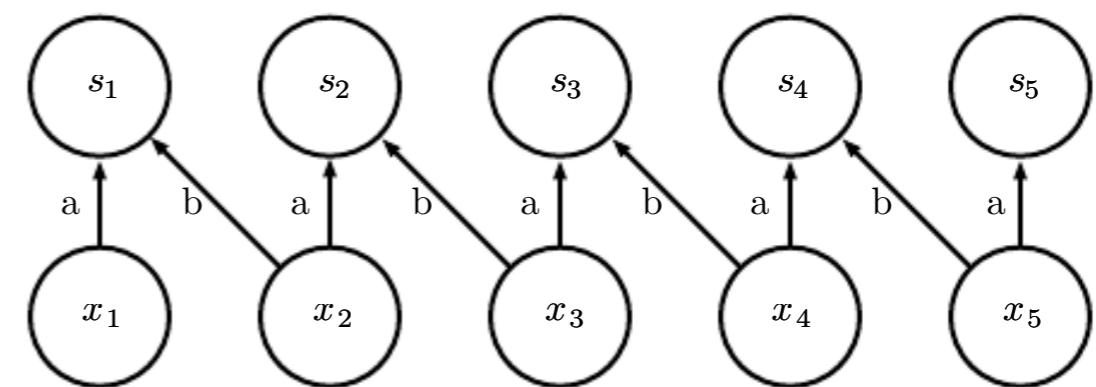
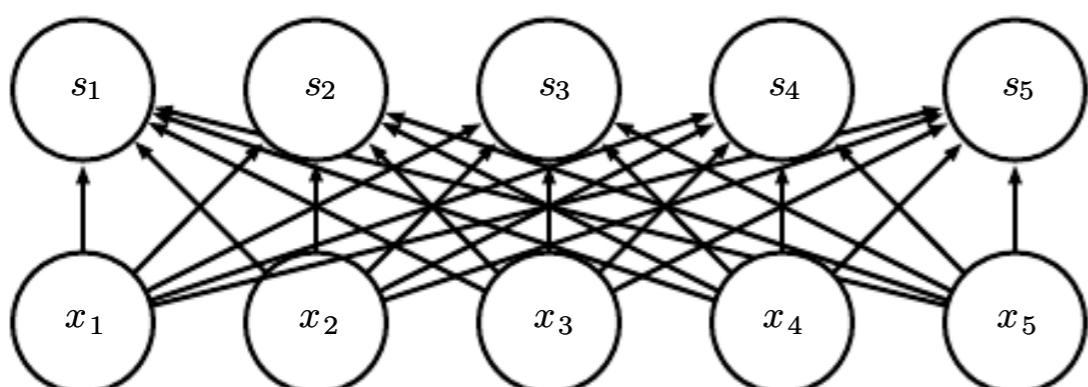
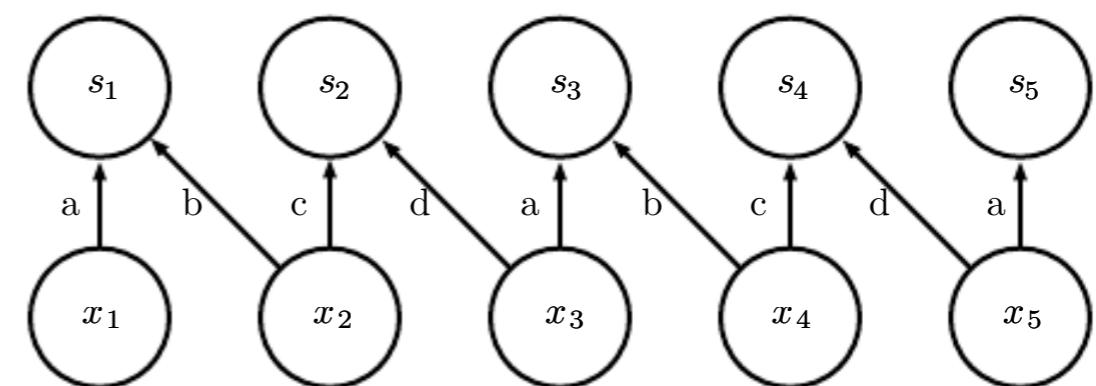
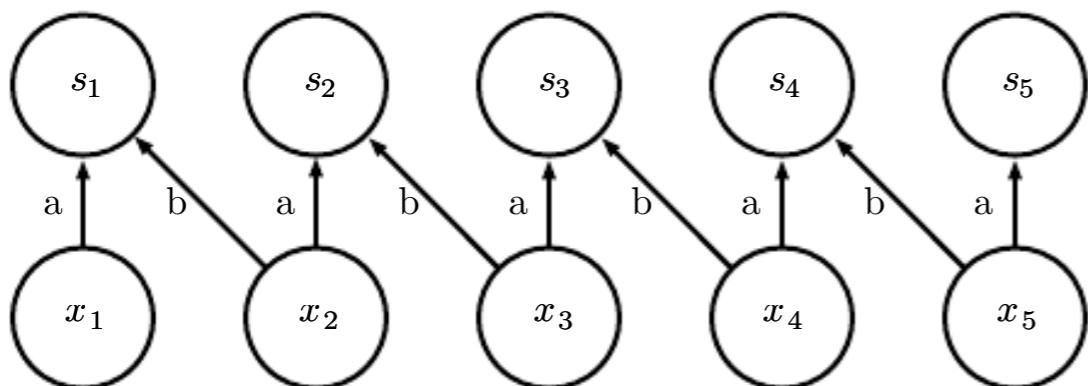


# Warianty CNN: locally connected layer and tiled convolution

locally connected layer



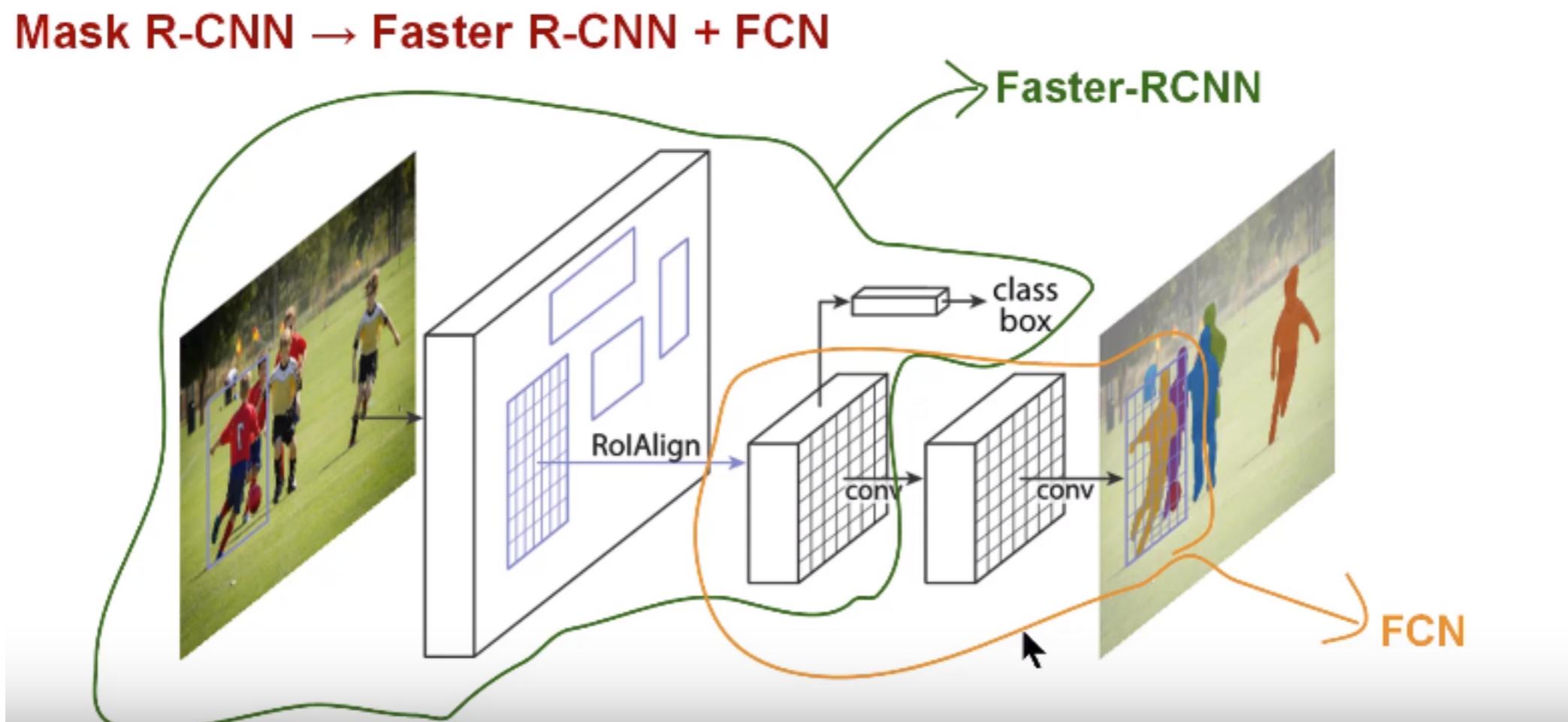
tiled convolution



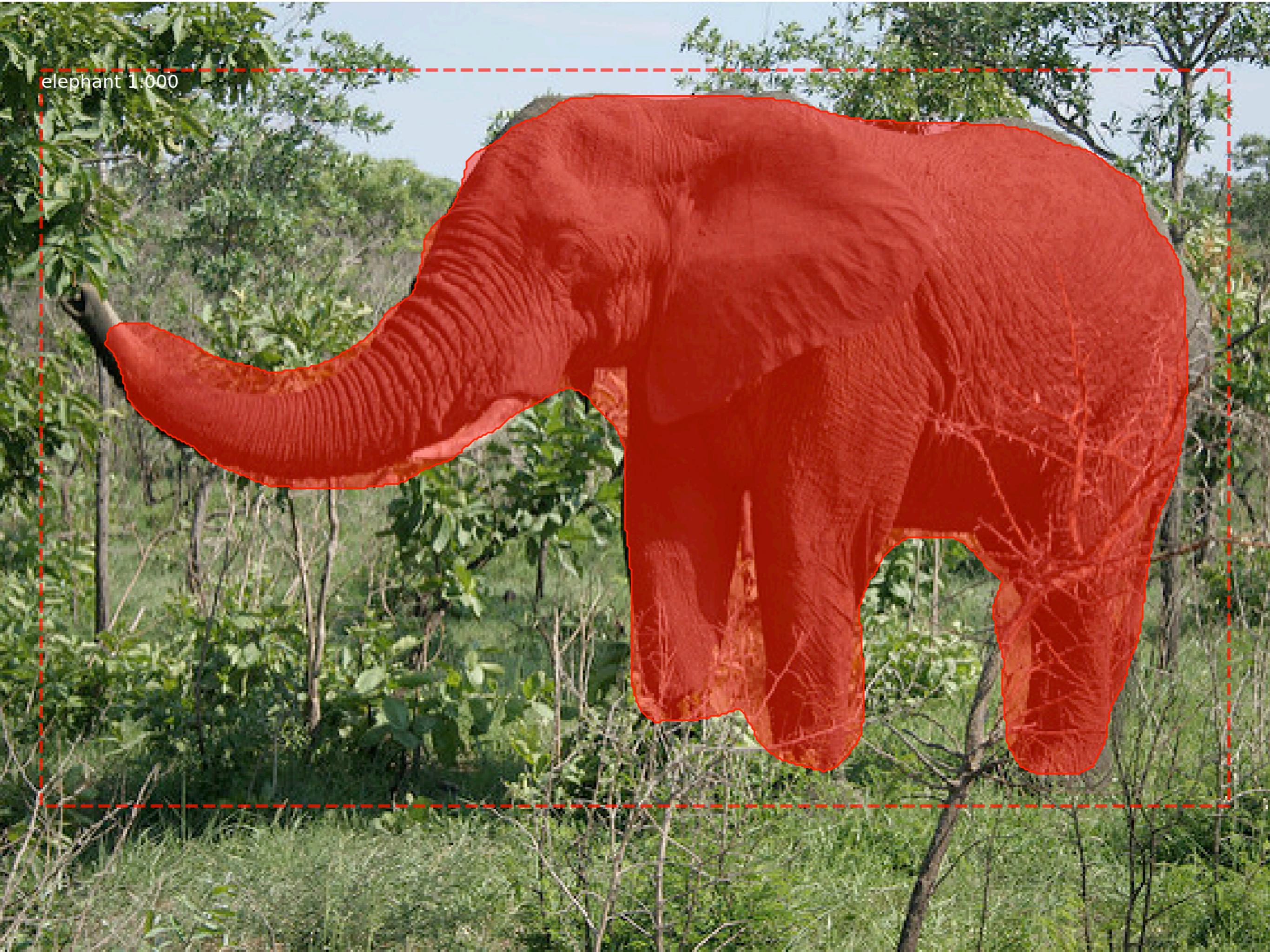
# Structured output

- Sieć może zwracać również tensor, np:

$\hat{y}_{ijk}$ :  $i, j$  – współrzędne pikela,  $k$  – przynależność do klasy



elephant 1.000

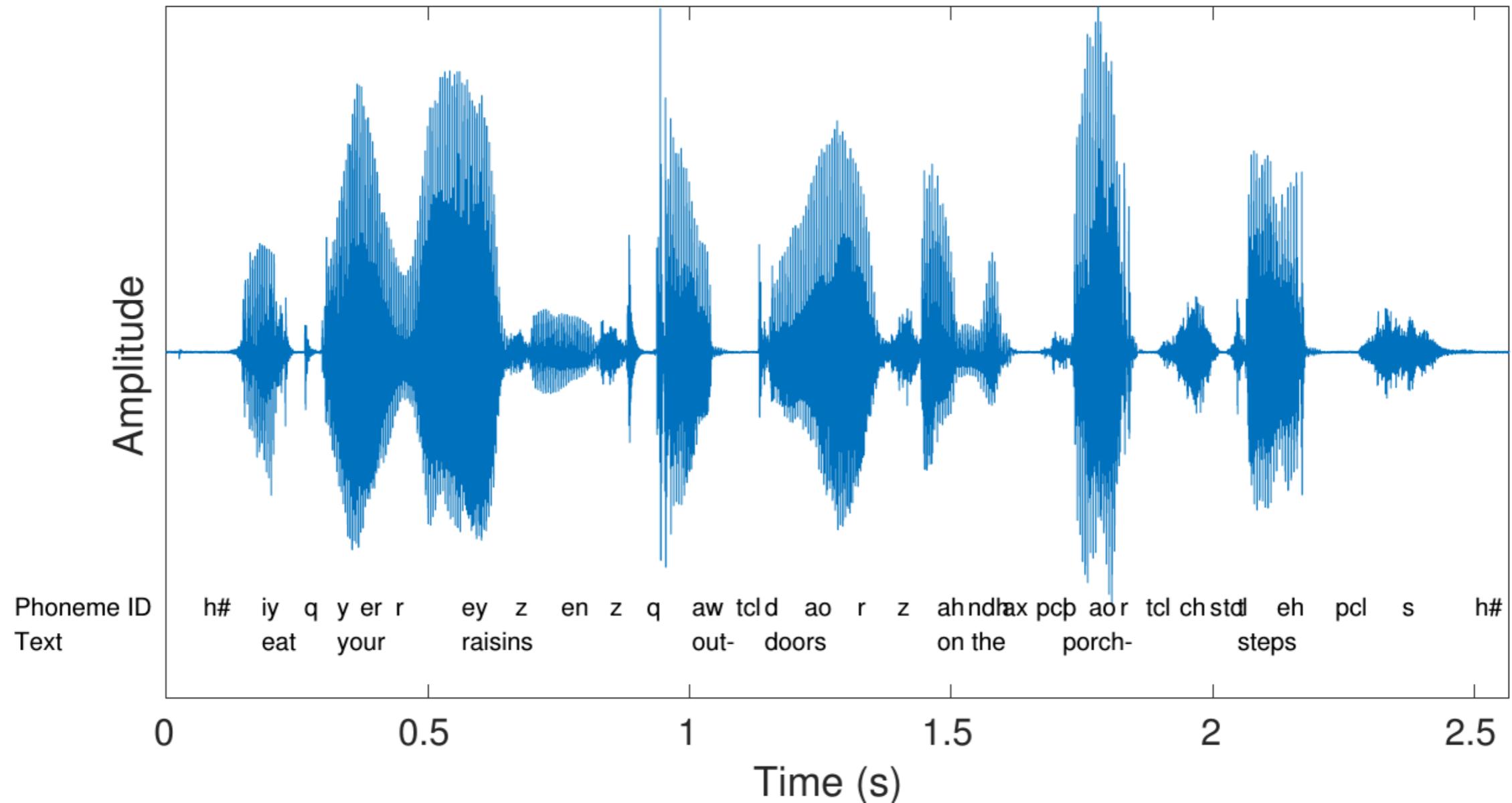


# Wejście sieci, zależnie od przypadku

- Wymiar sieci: liczba składowych tensora wejściowego dla których sąsiedztwo ma znaczenie.
- Liczba kanałów: wymiar składowej tensora wejściowego dla którego sąsiedztwo nie ma znaczenia.

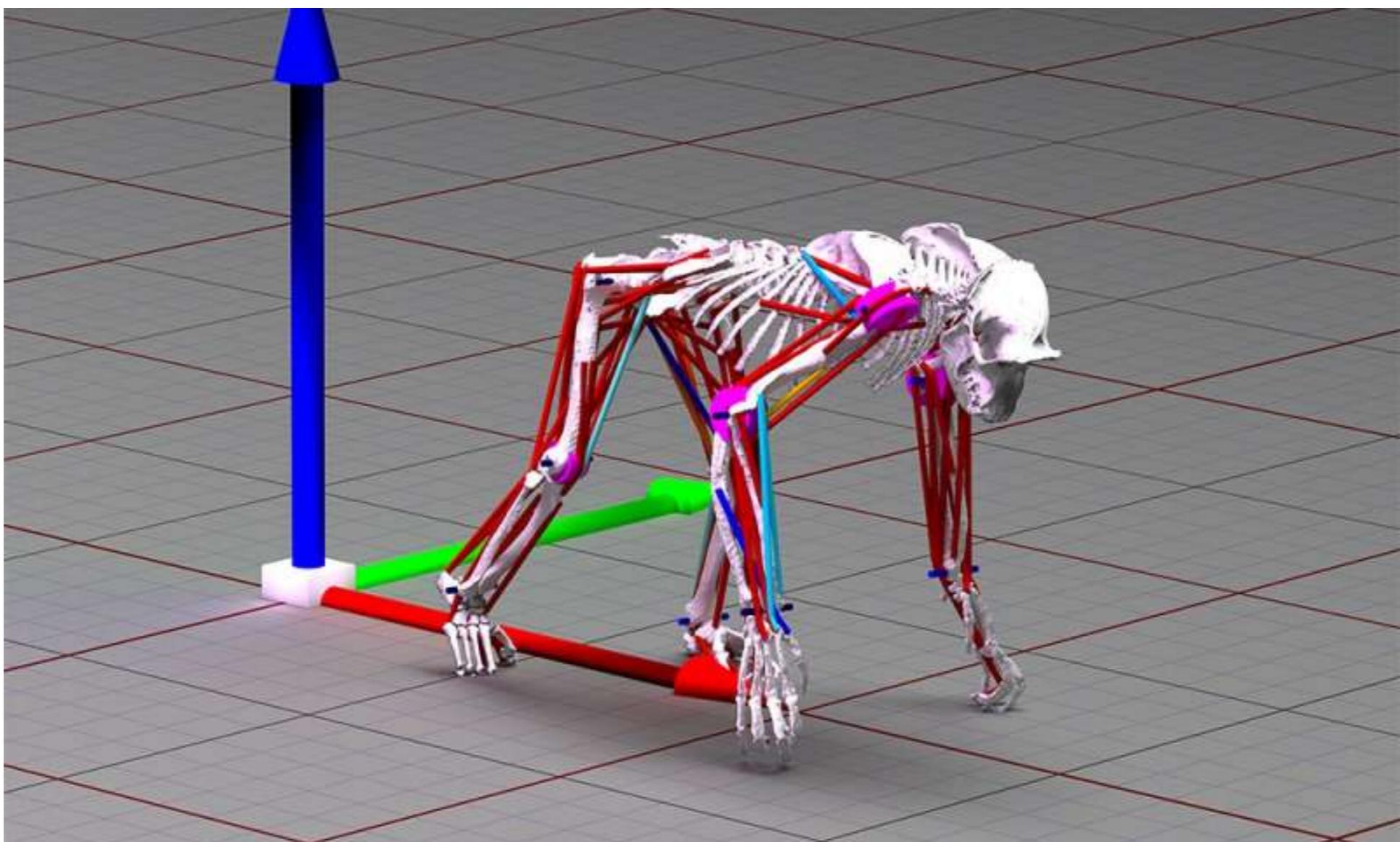
# 1D, 1 kanał

- Sygnał audio (waveform)



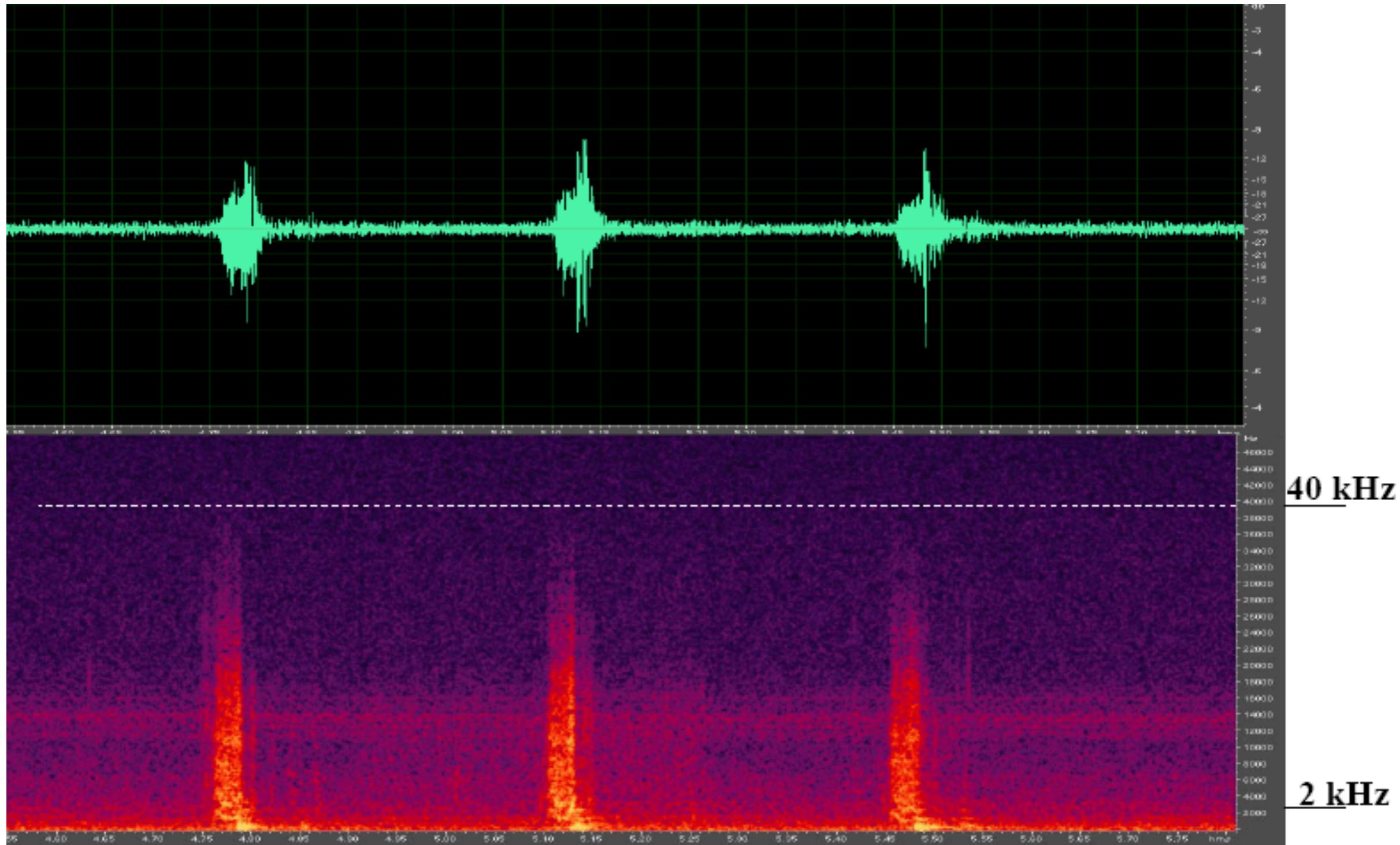
# 1D, wiele kanałów

- symulacja szkieletu  
poszczególne składowe to kąty opisujące wzajemne położenie kości w stawach.



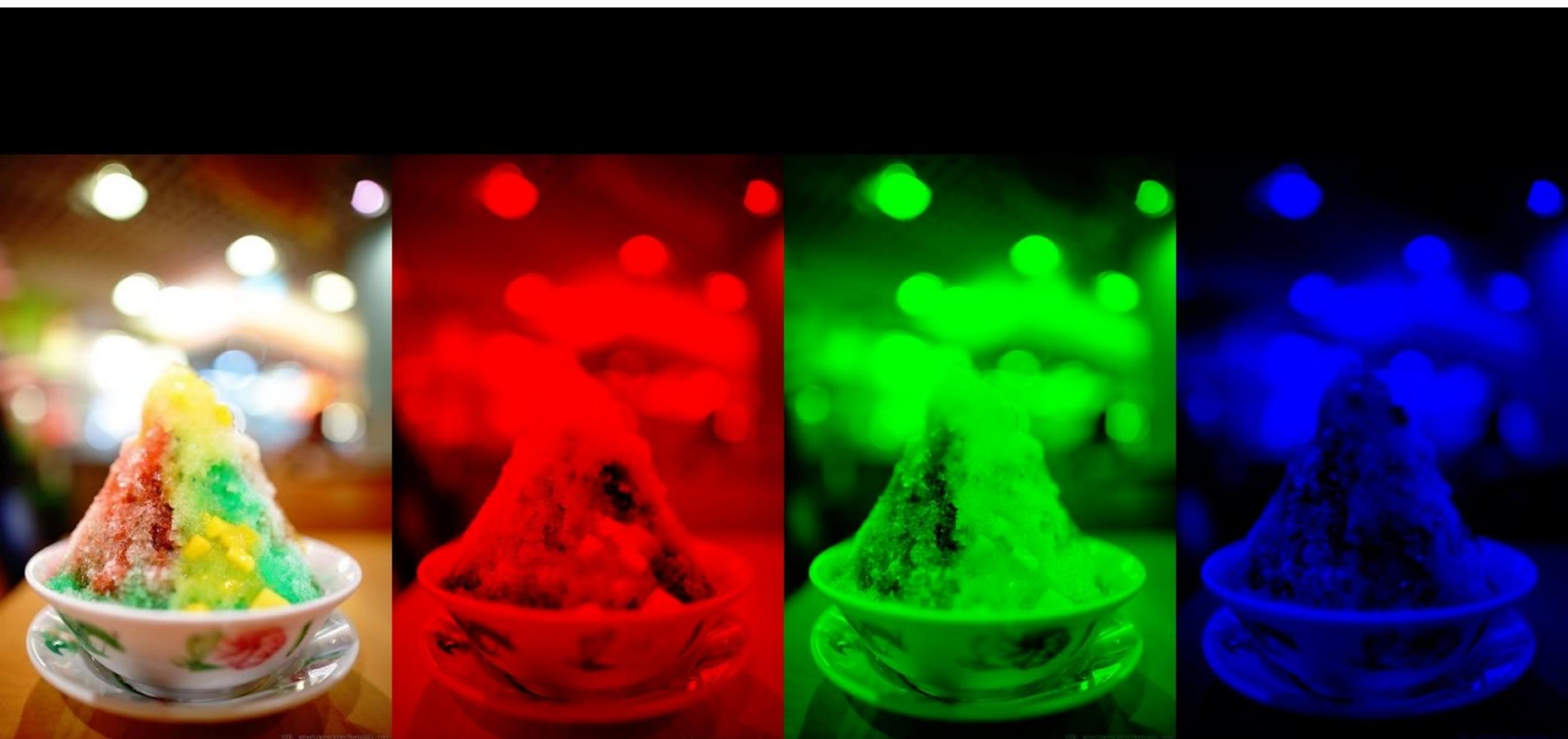
# 2D, 1 kanał

- Transformata Fouriera sygnału audio



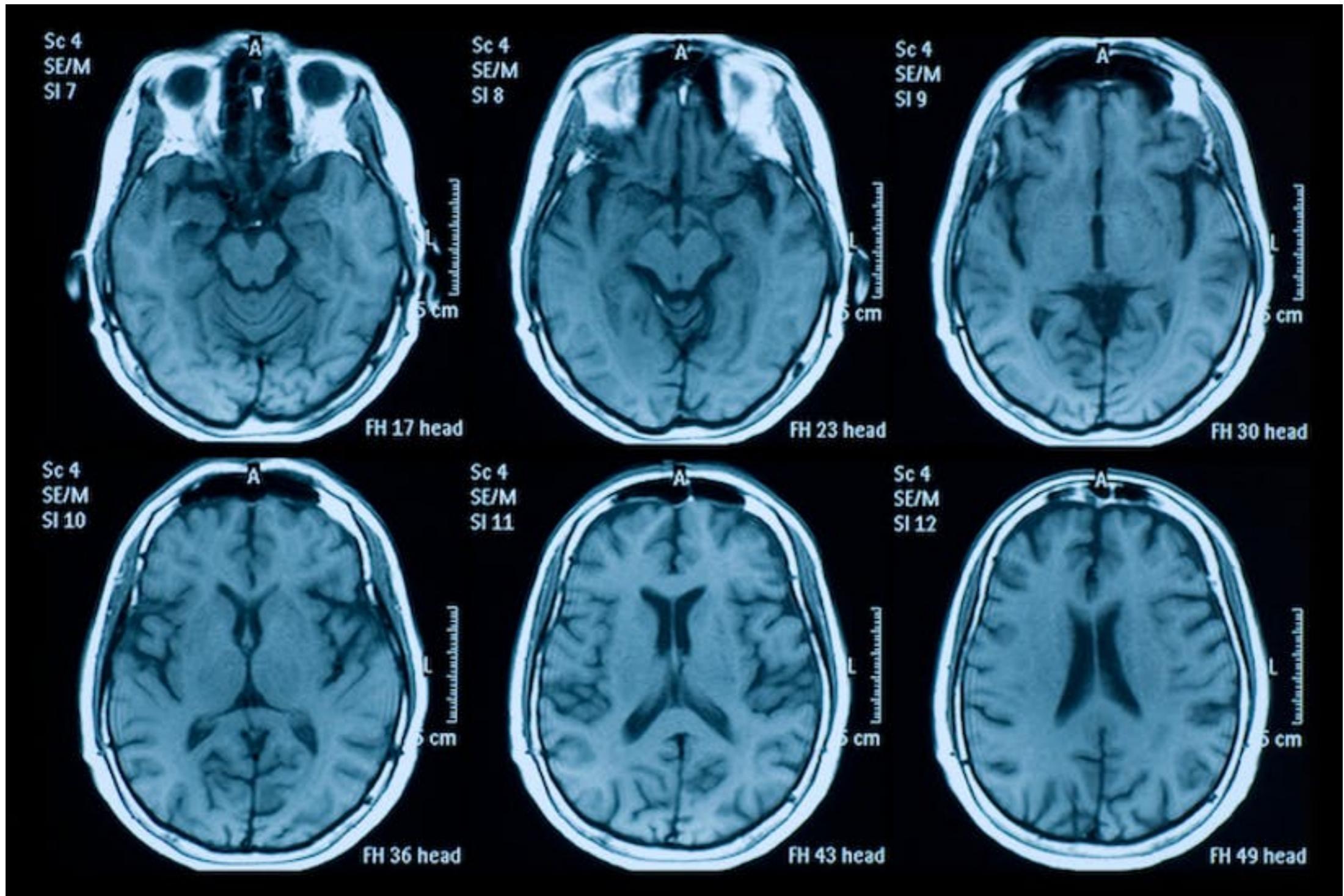
# 2D, wiele kanałów

- Kolorowy obraz (RGB, CMYK, HSV...)



# 3D, jeden kanał

- tzw. *volumetric data*, np. MRI



# 3D, wiele kanałów

- sekwencja video



# Rozmiar wejścia

Ze względu na warstwy typu Pooling oraz na współdzielenie parametrów CNN, **sieci o architekturze CNN dobrze radzą sobie z wejściami o różnych rozmiarach.**

- warstwa Pool musi się skalować
- podczas trenowania, rozmiar w ramach jednego batcha musi być stały
- przydatne przy klasyfikowaniu obrazów

# Przykładowe architektury “state-of-the-art”

**Table 1.** The comparison of different CNN architectures on model size, classification error rate, and model depth.

Model	Size (M)	Top-1/top-5 error (%)	# layers	Model description
AlexNet	238	41.00/18.00	8	5 conv + 3 fc layers
VGG-16	540	28.07/9.33	16	13 conv + 3 fc layers
VGG-19	560	27.30/9.00	19	16 conv + 3 fc layers
GoogleNet	40	29.81/10.04	22	21 conv + 1 fc layers
ResNet-50	100	22.85/6.71	50	49 conv + 1 fc layers
ResNet-152	235	21.43/3.57	152	151 conv + 1 fc layers

“conv” and “fc” indicates convolutional and fully-connected layers, respectively.

# Przykładowe architektury “state-of-the-art”

**Table 1.** The comparison of different CNN architectures on model size, classification error rate, and model depth.

Model	Size (M)	Top-1/top-5 error (%)	# layers	Model description
AlexNet	238	41.00/18.00	8	5 conv + 3 fc layers
VGG-16	540	28.07/9.33	16	13 conv + 3 fc layers
VGG-19	560	27.30/9.00	19	16 conv + 3 fc layers
GoogleNet	40	29.81/10.04	22	21 conv + 1 fc layers
ResNet-50	100	22.85/6.71	50	49 conv + 1 fc layers
ResNet-152	235	21.43/3.57	152	151 conv + 1 fc layers

“conv” and “fc” indicates convolutional and fully-connected layers, respectively.

