

Sieci splotowe - wprowadzenie

Kamil Bogus

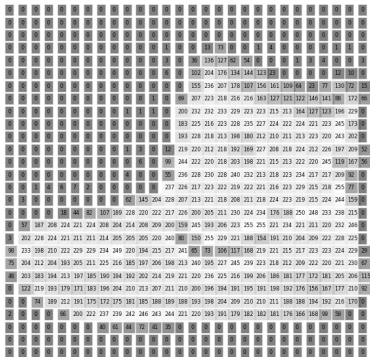
Wrocław, 22 listopada 2023

Interesuje nas przetwarzanie (potencjalnie dużych) obrazów tj. w szczególności ich

- rozpoznawanie (detekcja),
- przypisywanie obrazów do określonych klas oraz
- segmentacja obszaru.

Fashion MNIST

Zbiór danych Fashion-MNIST składa się z obrazów oraz przypisanego do nich opisu garderoby.

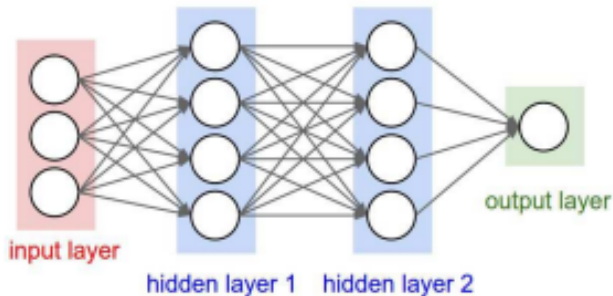


Rysunek: <https://miroslawmamczur.pl/jak-dzialaja-konwolucyjne-sieci-neuronowe-cnn/>

Obrazy mają rozmiar 28×28 pikseli i są w odcieniach szarości.

Klasyczne sieci neuronowe (gęste)

W klasycznej sieci neuronowej wszystkie neurony pomiędzy warstwami są w pełni połączone:



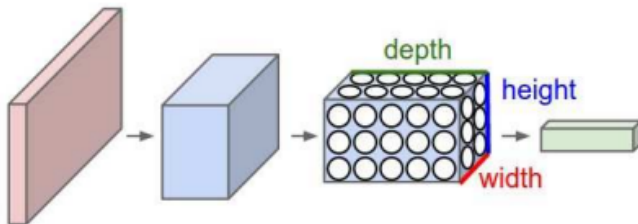
Rysunek: <https://miroslawmamczur.pl/jak-dzialaja-konwolucyjne-sieci-neuronowe-cnn/>

Problemy z klasycznymi sieciami neuronowe (gęste)

- Nawet proste sieci z np. dwiema warstwami mogą zawierać miliardy parametrów do wytrenowania...
- a to wymaga dużej mocy obliczeniowej, co wpływa na czas i koszty!
- Problemem jest również potencjalna utrata informacji wynikająca ze zmiany kształtu zdjęcia.

Sieci plotowe (konwolucyjne)

W sieci plotowej neurony połączone są tylko z pewnym obszarem w warstwie poprzedzającej:



Rysunek: <http://cs231n.github.io/convolutional-networks/>

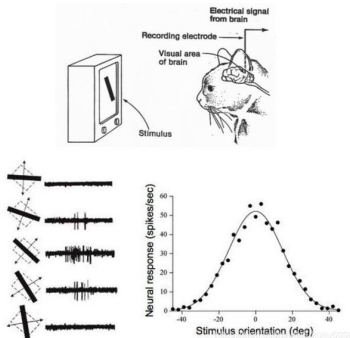
Obszar ten określony jest przez wielkość tzw. filtru, a głębokość wynika z liczby użytych filtrów do operacji splotu.

Sieci splotowe (inaczej konwolucyjne; Convolutional Neural Nets = CNN) znajdują zastosowanie głównie w przetwarzaniu obrazu, pomagając:

- określić prawdopodobieństwa przypisania obrazu do wyuczonych klas,
- wykrywać obiekty na obrazie i interpretować zależności pomiędzy nimi,
- dokonać podziału na obszary,

Historia - prace Hubela i Wiesel

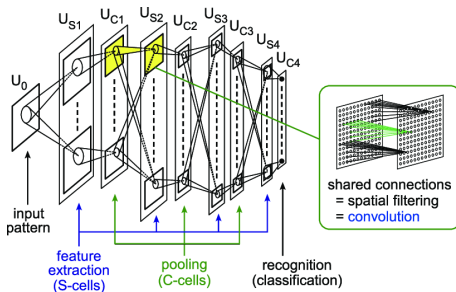
Na przełomie lat 50. i 60. **David Hubel i Torsten Wiesel** odkryli dwa typy komórek w pierwotnej korze wzrokowej zwane komórkami prostymi i komórkami złożonymi.



Rysunek: <https://www.programmersought.com/article/3392117538/>

Zaproponowali też kaskadowy model tych dwóch typów komórek do wykorzystania w rozpoznawaniu wzorców. Za swoje badania otrzymali w 1981 nagrodę Nobla z medycyny.

W 1979 **Kunihiko Fukushima** opracowuje neokognitron.



Rysunek: Kunihiko Fukushima, Recent advances in the deep CNN neocognitron

Ta hierarchiczna, wielowarstwowa sztuczna sieć neuronowa używana była do rozpoznawania znaków pisanych odręcznie w języku japońskim i innych zadań związanych z rozpoznawaniem wzorców.

- **Yann LeCun** wykorzystał propagację wsteczną, aby modyfikować współczynniki jądra splotu dla zbioru obrazków z ręcznie zapisanymi cyframi obrazów (MNIST).
- Od tego momentu uczenie się sieci stało się w pełni automatyczne, działało lepiej niż ręczne projektowanie współczynników i było dostosowane do szerszego zakresu problemów z rozpoznawaniem obrazu i typów obrazów.
- W 1998 powstaje sieć LeNet-5 (Y. LeCun, L. Bottou, Y. Bengio, P. Haffner) wykorzystująca wcześniej znane elementy (warstwy w pełni połączone, sigmoidalna funkcja aktywacji itd.) z nowymi elementami: **warstwami splotowymi i warstwami łączącymi**.

Czym jest konwolucja?

- Matematycznie: Konwolucja (splot) mierzy całkę z iloczynu punktowego funkcji

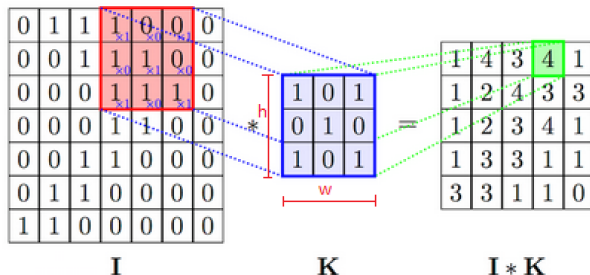
$$(f \star g)(y) = \int_{-\infty}^{\infty} f(y-x)g(x)dx.$$

- Obrazowo: jest to przekształcenie macierzowe fragmentów zdjęcia, które ma na celu wydobycie informacji o konkretnych cechach obrazu.
- Takie przekształcenie (macierz) nazywamy filtrem lub kernelem.
- Nieco ściślej: jedna z funkcji to dwuwymiarowa macierz zawierająca wartości pikseli obrazu (zazwyczaj o dużych wymiarach), a druga z funkcji to filtr (zdecydowanie mniejsza macierz), nazywana jądrem konwolucji.
- W wyniku konwolucji obrazu z filtrem otrzymujemy nowy obraz.

Przykład konwolucji - animacja

<https://miroslawmamczur.pl/jak-dzialaja-konwolucyjne-sieci-neuronowe-cnn/>

Przykład konwolucji



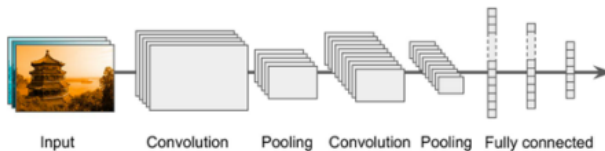
$$(I * K)_{xy} = \sum_{i=1}^h \sum_{j=1}^w K_{ij} \cdot I_{x+i-1, y+j-1}$$

Rysunek: <http://sciagaprogramisty.blogspot.com/2018/01/konwolucja-wstep-do-neuronowych-sieci.html>

Architektura sieci splotowych

W skład typowej sieci splotowej wchodzi m.in.

- warstwa wejściowa,
- warstwy splotowe,
- warstwy łączące.



Rysunek: Aurélien Géron. *Uczenie maszynowe z użyciem Scikit-Learn i TensorFlow*. Helion, 2018

Warstwa wejściowa (input layer)

- Reprezentuje ona obraz wejściowy do sieci.
- W przypadku kolorowych zdjęć będziemy używać trzech kanałów wejściowych RGB, odpowiadającym odpowiednio kanałom czerwonym, zielonym i niebieskim.
- W przypadku Fashion Mnist mamy skalę szarości i dlatego mamy jeden kanał.

Warstwa splotowa (convolutional layer)

- Zawierają wyuczone filtry (kernele), które wyodrębniają cechy odróżniające od siebie różne obrazy.
- Wartości w filtrach są dobierane i optymalizowane podczas trenowania sieci
- Filtry są współdzielone na całym zdjęciu tzn. wagi dobierane są do filtra, który jest następnie przesuwany po całym zdjęciu.
- Rozmiar samego zdjęcia nie wpływa na liczbę wag w warstwie konwolucyjnej.

<https://miroslawmamczur.pl/jak-dzialaja-konwolucyjne-sieci-neuronowe-cnn/>

Do pełnego zdefiniowania warstwy splotowej musimy określić:

- Kernel Size (rozmiar jądra/rozmiar filtra),
- Padding (wypełnienie),
- Strides (kroki),
- Funkcje aktywacji.

Rozmiar jądra/filtra

- Odnosi się do wymiarów przesuwanego okna nad wejściem,
- Małe jądra są w stanie wydobyć z danych wejściowych znacznie większą ilość informacji zawierających wysoce lokalne funkcje.
- Mniejszy rozmiar jądra prowadzi również do mniejszego zmniejszenia wymiarów warstw, co pozwala na głębszą architekturę.
- Duży rozmiar jądra wyodrębnia mniej informacji, co prowadzi do szybszego zmniejszenia wymiarów warstw, czego skutkiem często jest gorsza wydajność.
- Duże jądra lepiej nadają się do wyodrębniania większych elementów.

<https://miroslawmamczur.pl/jak-dzialaja-konwolucyjne-sieci-neuronowe-cnn/>

Padding (wypełnienie)

- Padding umożliwia otrzymanie rozmiaru wyjścia takiego samego jak rozmiar wejścia (przy założeniu, że strides jest przesunięciem o jeden).
- Osiąga się to kosztem dodania dodatkowych (sztucznych) wag na krawędziach (najczęściej z wartością zero).
- Dzięki dodanej ramce rozmiar wyjściowy (zielony kwadrat) ma rozmiar 5×5 , czyli tyle samo co obraz wejściowy (niebieski kwadrat).
- Gdyby nie było dodanej ramki, to wówczas rozmiar wyjściowy wynosiłby 3×3 .

<https://miroslawmamczur.pl/jak-dzialaja-konwolucyjne-sieci-neuronowe-cnn/>

Strides (kroki)

- Parametr strides oznacza krok przesunięcia okna filtra.
- Najczęściej używa się kroku wynoszącego 1 dla warstw splotowych.
- Oznacza to, że iloczyn skalarny jest wykonywany w oknie wejściowym np. 3×3 w celu uzyskania wartości wyjściowej, a następnie jest przesuwany o jeden piksel dla każdej kolejnej operacji.
- Poniżej przedstawiono przesuwanie jądra o wielkości 3×3 (szary kwadrat) po obrazie 5×5 pikseli z parametrem strides równym 2 i padding równym 1.

<https://miroslawmamczur.pl/jak-dzialaja-konwolucyjne-sieci-neuronowe-cnn/>

- Gdyby nie było nieliniowej funkcji aktywacji, głębokie sieci CNN przekształciłyby się w pojedynczą, równoważną warstwę splotową, która nie działałaby tak dobrze.
- Przykładem nieliniowej funkcji aktywacji jest m.in. ReLU (Rectified Linear Activation) określona wzorem

$$\text{ReLU}(x) = \max\{x, 0\}.$$

- Inna wykorzystywana powszechnie funkcja to Softmax:

$$\text{Softmax}(x_j) = \frac{\exp(x_j)}{\sum_j \exp(x_j)}.$$

Warstwa łącząca (pooling layer)

- Ma na celu zmniejszenie obrazu wejściowego w celu zredukowania obciążenia obliczeniowego, wykorzystania pamięci i liczby parametrów.
- Każdy neuron stanowiący część warstwy łączącej łączy się z wyjściami określonej liczby neuronów warstwy poprzedniej, mieszczącej się w obszarze niewielkiego pola recepcyjnego (podobnie jak w przypadku warstw spłotowych).
- Definiujemy rozmiar pola, wartość kroku, rodzaj wypełniania zerami, itd. (podobnie jak w przypadku warstw spłotowych).
- Warstwa łącząca nie zawiera wag.

Pooling - przykład

- Pooling polega na mapowaniu kilku pikseli np. rozmiaru 2×2 na 1 piksel
- Jednym ze rodzajów poolingu jest max pooling, gdzie bierzemy maksymalną wartość z danych pikseli
- Inny przykład to avg pooling, gdzie bierzemy średnią wartość z danych pikseli.
- Poniżej przedstawiono max pooling z parametrem (size) wynoszącym 2, co oznacza przechodzenie po obrazie wykorzystując rozmiar 2×2 i wzięcie maksymalnej wartości jako piksel wyjściowy.

<https://miroslawmamczur.pl/jak-dzialaja-konwolucyjne-sieci-neuronowe-cnn/>

Dropout Layer (wartwa porzucenia)

- Porzucanie to najpopularniejszy sposób radzenia sobie z overfittingiem.
- Dropout polega na losowym ustawieniu wychodzących krawędzi ukrytych jednostek (neuronów tworzących ukryte warstwy) na 0 przy każdej aktualizacji fazy treningu.
- W każdym przejściu losowo wyłączane są połączenia.
- Dzięki temu sieć nie nauczy się „na pamięć” zbyt szybko, ponieważ architektura co przeliczenie odrobinę się zmienia poprzez zerowanie losowych połączeń neuronów.
- Inne metody walki z overfittingiem to m.in. regularyzacja wag, metoda wczesnego zakończenia (early stopping) czy też batch normalization.

Flatten Layer (warstwa spłaszczająca)

- Ta warstwa przekształca naszą wielowymiarową warstwę w sieci w jednowymiarowy wektor.
- Pozwala to dopasować dane wejściowe w pełni połączonej warstwy do klasyfikacji.
- Przykładowo: tensor o wymiarze $20 \times 20 \times 3$ zostanie przekształcony w wektor długości 1200.
- Do klasyfikowania tych cech najczęściej używamy funkcji softmax, co wymaga jednowymiarowych danych wejściowych i to właśnie wymusza stosowanie spłaszczonej warstwy.

Przykładami architektur konwolucyjnych CNN są:

- LeNet (1998),
- AlexNet (2012),
- GoogLeNet (2014),
- VGGNet (2014),
- ResNet (2015),
- i wiele innych.

Do uczenia/porównań sieci spłotowych używane są m.in. następujące zbiory:

- MNIST,
- CIFAR-10,
- CIFAR-100,
- STL-10,
- SVHN (The Street View House Numbers).

Literatura - warto zajrzeć!

- Michel Nielsen "Neural Networks and Deep Learning",
<http://neuralnetworksanddeeplearning.com>
- Tianyi Liu, Shuangfang Fang, Yuehui Zhao, Peng Wang, Jun Zhang
"Implementation of Training Convolutional Neural Networks",
<https://arxiv.org/abs/1506.01195>
- <https://poloclub.github.io/cnn-explainer/>
- <https://miroslawmamczur.pl/jak-dzialaja-konwolucyjne-sieci-neuronowe-cnn/>
- <https://home.agh.edu.pl/~horzyk/lectures/ai/SztucznaInteligencja-UczenieGlebokichSieciNeuronowych.pdf>
- http://www.cs.put.poznan.pl/alawrynowicz/SI_ML_CNN_2020_lawrynowicz.pdf