Rozpoznawanie w oparciu o algorytmy CNN. Steganografia

WYKŁAD 8

Dla studiów niestacjonarnych 2021/2022

Dr hab. Anna Korzyńska, prof. IBIB PAN

Komputerowa analiza i rozpoznawanie obrazów

Jest to „sztuka" udzielania, **automatycznej** i mającej **matematyczne podstawy** odpowiedzi na pytanie:

Co ten obraz przedstawia (i o czym mówi) ?

Podejścia:

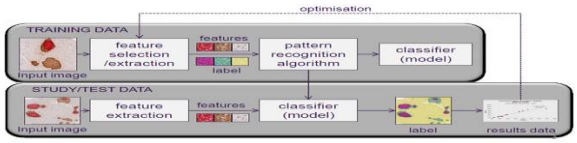
1. **Klasyczna** – oparta o cechy dobranych przez deweloperów systemu na podstawie rad ekspertów i ich doświadczenia
   - **Analiza obrazu** - zajmuje się ekstrakcją cech obiektów wysegmentowanych z obrazów na potrzeby rozpoznawania obrazów
   - **Właściwe rozpoznawanie obrazu** - zajmuje się tworzeniem i weryfikacją reguł, na podstawie których udziela się odpowiedzi na powyższe pytanie oraz stosowaniem tych reguł w konkretnych zagadnieniach praktycznych
2. **Oparta na sztucznej inteligencji** - bazuje na cechach wyznaczonych automatycznie w procesie uczenia

2

Podsumowanie metod klasycznych

Klasyczne algorytmy rozpoznawania obrazów zawsze **zależą** od skomplikowanego przetwarzania obrazu w celu poprawy początkowej jakości obrazu i umożliwienia **segmentacji / separacji obiektów**.

Jednak najważniejszy jest **proces inżynierii cech**, który jest kluczowy w klasycznej technice rozpoznawania. Tradycyjne „ręcznie" **dobierane cechy i funkcje** w dużej mierze **opierają się na wiedzy specjalistycznej** w dziedzinie. Najczęściej są zaproponowane deweloperom systemu przez specjalistów, którzy przygotowują adnotację do zestawu szkoleniowego.



Rozpoznawanie obrazów metodami opartymi na głębokich konwolucyjnych sieciach neuronowych

Wymagają nauczenia sieci neuronowej rozwiązywania pewnego zadania stosując algorytmy głębokiego uczenia (ang. deep learning) należące do dziedziny zwanej sztuczną inteligencją

4

Sztuczna inteligencja

Sformułowanie „sztuczna inteligencja" (ang. Artificial Intelligence; AI) jest używana do nazwania takich komputerów/maszyn, które naśladują „funkcje poznawcze", które ludzie kojarzą z ludzkim umysłem, takie jak „uczenie się" i „rozwiązywanie problemów".

Powstała jako dyscyplina akademicka w 1955. Termin zaproponował McCarthy (z Uniwersytetu Stanford) na konferencji w Dartmouth

1. inteligencja realizowana w procesie technicznym, a nie naturalnym, biologicznym;
2. dziedzina badań naukowych informatyki i kognitywistyki czerpiąca także z osiągnięć psychologii, neurologii, matematyki i filozofii.

Definicja z 2019 r: „.. zdolność systemu (maszyny, komputera dopisek) do prawidłowego interpretowania danych pochodzących z zewnętrznych źródeł, nauki na ich podstawie oraz wykorzystywania tej wiedzy, aby wykonywać określone zadania i osiągać cele poprzez elastyczne dostosowanie."

Cytat z: Andreas Kaplan; Michael Haenlein (2019) Siri, Siri in my Hand, who's the Fairest in the Land? On the Interpretations, Illustrations and Implications of Artificial Intelligence, Business Horizons, 62(1), 15-25

Algorytmy sztucznej inteligencji (AI)

Sztuczna inteligencja jako technologia aktualnie dostępna

potrafi:

- **Używać naturalnego werbalnego języka, tłumaczyć** z języka na język, ale nie potrafi prowadzić komunikacji pozawerbalnej; wyczuć intencji i nastroju mówiącego
- **Uczyć się na podstawie danych, na próbach i błędach przez powtarzanie procesy uczenia,** ale tylko w stosunku do konkretnego ściśle wyznaczonego zadania
- Otrzymuje bardzo **dokładne rezultaty** w specyficznych trudnych dla człowieka zadaniach obliczeniowych, w grach o ściśle określonych regułach, poszukiwaniu podobieństwa w danych and nie potrafi żadnej z tych czynności adoptować do nawet lekko zmienionych warunków zadania
- **nie ma celu, woli, emocji, poczucia humoru, zawziętości czy wyrozumiałości i innych cech charakteru (indywidualności)**

Zastosowanie AI w: wizji komputerowej, w rozpoznawaniu mowy, maszynowym tłumaczeniu, w filtrowaniu poczty, graniu w gry nawet trudne jak szachy czy Go . Ostania również odnosi wielkie sukcesy we wspomaganiu diagnostyki w medycynie.

6

Uczenie maszynowe czy uczenie się maszyn?

- **Uczenie z nauczycielem, nadzorowane** (supervised learning) – na podstawie przeszłych danych o wejściu i wyjściu dostarczonych przez dewelopera systemu (np. od ekspertów)
- **Uczenie bez nauczyciela, nienadzorowane** (unsupervised learning) – ukryte zależności i organizacja danych, ich podobieństwa i zaburzenia, czy odchylenia od normy są identyfikowane przez system bez pomocy ekspertów w dziedzinie lub dewelopera systemu
- **Nie ma treningu tylko następuje uczenie się dzięki próbom i błędom** (reinforcement learning) i naturalnej pochodzącej ze środowiska karze, która pozwala podnieść efektywność uczenia się np.: uczenie chodu i gry np.: szachy
- Uczenie częściowo nadzorowane (weeklly supervised), w którym na wstępie zastosowane jest uczenie z nauczycielem na tzn. danych etykietowanych, a następnie zasadnicza część uczenia to uczenie bez nauczyciela.

7

Przykłady możliwości systemów opartych na uczeniu maszynowym

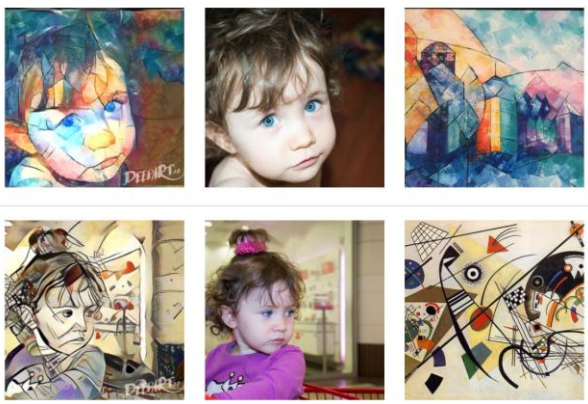
- Prowadzenie samochodów i pilotowanie samolotów oraz robotów
- Rozpoznawanie mowy (2012);
- Ekstrahuje i aplikuje cechy stylu (Deep Convolution Generative Adversarial Networks DCGAN)


Nvidia
https://www.youtube.com/watch?v=qhUvQiKec2U


Boston Dynamics: https://www.youtube.com/watch?v=-e9QzIkP5ql


„A Neural Algorithm of Artistic Style" Leon A. Alexander S. Ecker, Matthias Bethge

8

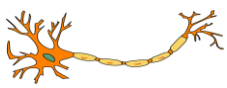


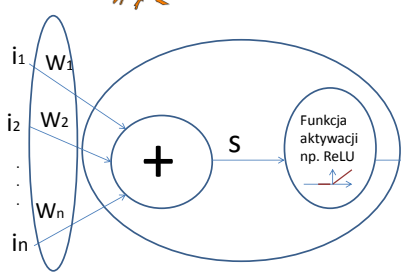
9

Jak są zbudowane sztuczne sieci neuronowe?

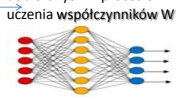
10

Sieci neurunowe





Elementem funkcjonalnym sieci neuronowej jest perceptron / neuron - matematyczna symulacja funkcji komórek nerwowych (odbiór i odesłanie przetworzonego sygnału). Sieć jest złożona z warstw neuronów połączonych w taki sposób, że wyjście warstwy wcześniejszej jest wejściem warstwy następującej po niej. Połączenia poszczególnych neuronów regulują przekazywany sygnał według dobieranych w procesie uczenia współczynników W

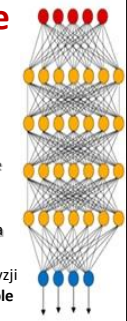


McCulloch and Walter Pitts w 1943 roku

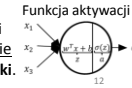
11

Głębokie sieci neuronowe

- Głębokie sieci neuronowe (NN) to sieci o **więcej niż trzech warstwach**.
- Sieci neuronowe uczą się z danych źródłowych i **mapują** „informacje/wiedze wciągniętą ze zbioru uczącego" **do współczynników sieci**.
- W zadaniach rozpoznawania sieci neuronowe potrafią **w procesie optymalizacji automatycznie dobrać cechy** konieczne do rozpoznawania pewnych klas obiektów w procesie treningu.
- Typowe sieci neuronowe w eksploatacji zachowują się jak „**czarna skrzynka**".
- Sieci, w których mamy dostęp do informacji o „przyczynach" decyzji o rozpoznawaniu i klasyfikacji nazywają się po angielski **eXplanable NN**; (XNN)
- Okazuje się, że kombinacja cech pochodzących od ekspertów (używanych w klasycznych metodach rozpoznawania) z cechami zidentyfikowanymi na etapie uczenia się maszyny daje szansę nie tyle bardzo dobre wyniki, ale **zrozumiałe dla użytkownika wyniki**.



Funkcja aktywacji



12

Sztuczne sieci neuronowe

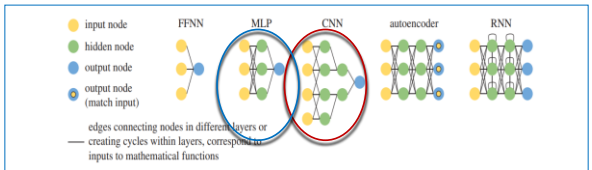


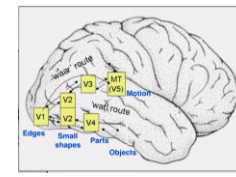
Figure 1. Neural networks come in many different forms. Left: A key for the various types of nodes used in neural networks. Simple FFNN: a feed-forward neural network in which inputs are connected via some function to an output node and the model is trained to produce some output for a set of inputs. MLP: the multi-layer perceptron is a feed-forward neural network in which there is at least one hidden layer between the input and output nodes. CNN: the convolutional neural network is a feed-forward neural network in which the inputs are grouped spatially into hidden nodes. In the case of this example, each input node is only connected to hidden nodes alongside their neighbouring input node. Autoencoder: a type of MLP in which the neural network is trained to produce an output that matches the input to the network. RNN: a deep recurrent neural network is used to allow the neural network to retain memory over time or sequential inputs. This figure was inspired by the Neural Network Zoo by Fjodor Van Veen.

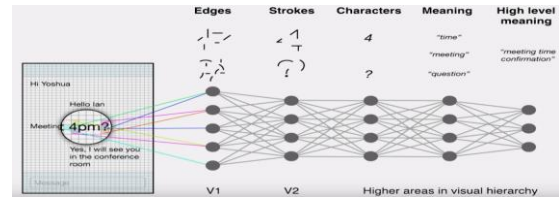
Headline review

Cite this article: Ching T *et al.* 2018 Opportunities and obstacles for deep learning in biology and medicine. *J. R. Soc. Interface* **15**: 20170387.
http://dx.doi.org/10.1098/rsif.2017.0387

13

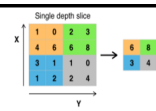
Praca głębokiej konwolucyjnej sieci (CNN) imituje proces rozpoznawanie w ludzkim mózgu





14

Warstwy



Podstawowe rodzaje warstw i procesów z nimi związanych:

- **Klasyczna** – Fully Connected (FC) - dense
- **Konwolucyjna** - W odróżnieniu od warstwy typu fully-connected, rozmiar wyniku działania warstwy konwolucyjnej zależy od rozmiaru danych wejściowych, rozmiaru filtrów i funkcji zmieniającej rozmiar, oraz wartości kroku. Sieci tego typu dobrze działają tam gdzie jest duże redundancja informacji np.: przy analizie obrazów, dźwięków lub zbioru cech.
- **Pooling.** Pooling to proces zmniejszeniu przestrzeni cech/rozmiaru sieci wewnątrz konwolucyjnej sieci neuronowej przez uśrednianie (average pooling) lub przyjmowanie wartości maksymalnej (max pooling) z określonego otoczenia.
- **Padding**, czyli uzupełnianie odpowiednimi wartościami, do określonego rozmiaru. Od liczby dodanych zer (stosowany zero padding) zależy, czy mapa wyjściowa będzie miała rozmiar większy, mniejszy czy taki sam, w porównaniu do rozmiaru mapy wejściowej na danym poziomie-warstwie.

Źródło: By Aphex34 - Own work, CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=45673581

15

Jakie matematyczne operacje umożliwiają uczenie się sieci

17

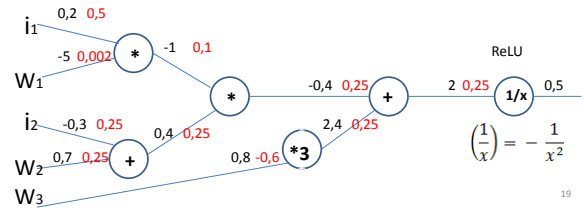
Funkcje aktywacji

Name	Plot	Equation	Derivative
Identity		$f(x) = x$	$f'(x) = 1$
Binary step		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x \neq 0 \\ ? & \text{for } x = 0 \end{cases}$
Logistic (a.k.a Soft step)		$f(x) = \frac{1}{1+e^{-x}}$	$f'(x) = f(x)(1-f(x))$
TanH		$f(x) = \tanh(x) = \frac{2}{1+e^{-2x}} - 1$	$f'(x) = 1 - f(x)^2$
ArcTan		$f(x) = \tan^{-1}(x)$	$f'(x) = \frac{1}{x^2+1}$
Rectified Linear Unit (ReLU)		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
Parameteric Rectified Linear Unit (PReLU) [2]		$f(x) = \begin{cases} \alpha x & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} \alpha & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
Exponential Linear Unit (ELU) [3]		$f(x) = \begin{cases} \alpha(e^x - 1) & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} f(x) + \alpha & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
SoftPlus		$f(x) = \log_e(1 + e^x)$	$f'(x) = \frac{1}{1+e^{-x}}$

18

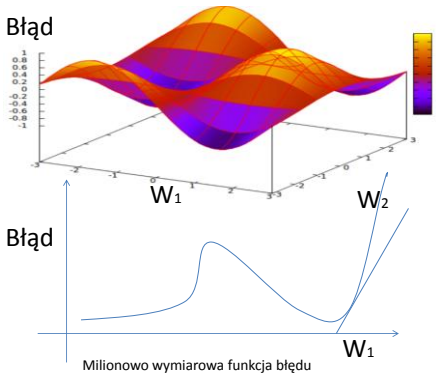
Algorytm propagacji wstecznej *ang. back propagation algorithm*

Algorytm oblicza wielkość korekty wag połączeń neuronów rozmieszczonych w sąsiednich warstwach sieci. Oparty jest on na minimalizacji funkcji błędu (np.: sumy kwadratów błędów) uczenia z wykorzystaniem optymalizacyjnej metody największego spadku.



19

Funkcja błędu



Wn = Wp -ηΔ
W – wektor of
η - współczynnik szybkości uczenia
Δ - gradient (wektor pochodnych cząstkowych)

20

Pojęcia związane z uczeniem sieci neuronowych

- **Przeuczenia (*ang. Overfiting*)**: zbyt mała ilość danych w porównaniu do liczby parametrów (zanikanie), zwiększenie zbioru uczącego, zmniejszenie liczby parametrów, losowe wykluczanie neuronów
- **Zanikanie gradientu (*ang. Gradients disappears*)** zamieniając funkcję aktywacji lub wzmacniając gradient przez regularyzację
- **Zbieżność/konwergencja procesu uczenia (*ang. Convergence of learning*)** zbieżność procesu optymalizacji zmiana funkcji błędu
- **Ocena poprawności uczenia (*ang. Correctness assessment*)**: dokładność i precyzja, Classical MSE / mutual entropy (CE -cross entropy; Averaged CE (ACE-averaged CE); F1
- **Generalizacja (ang. Generalization)** – cel uczenia

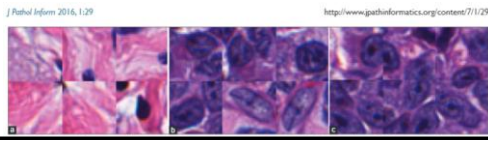
21

Augmentacja (*ang. augmentation*)

To proces powielenia danych uczących przez:

– **Transformacje obrazów** (ang. adversarial transformation); translacje, obroty, przekształcenia afiniczne?

– **Dołożenia pewnych zakłóceń** – rozjaśnienie, przyciemnienia, dodanie szumu zmiana tonacji, itp



22

Podsumowanie o sieciach CNN

23

Sztuczna sieć neuronowa

- Poprawa efektywności rozpoznawania, klasyfikacji i segmentacji obrazów przez żmudny **proces optymalizacji sieci – uczenia się**, który przypomina proces Darwinowskiego doboru naturalnego
- Na etapie korzystanie z nauczonej sieci mamy na **wejściu dane, a na wyjściu prawdopodobieństwo** że obiekt należy do którejś ze znanych kategorii **i tylko tyle**!
- **Nie ma „rozumienia”,** bo nie ma kontekstu; nie można wyobrazić sobie konsekwencji, itp.

CNN wady i zalety

Możliwości CNN:

- Nadzorowane podejście do uczenia maszynowego polega na **szkoleniu modelu statystycznego** przy użyciu zestawu obrazów ze zbioru uczącego, dla których istnieją etykiety i oznaczenia przygotowane przez ekspertów w dziedzinie. Wyuczony model odwzorowuje automatycznie wybrane obiekty na **klasy/kategorie**, czyli uczy się z doświadczeń.
- **Połączenie różnych informacji w modelu** np.: danych klinicznych i obrazów diagnostycznych daje lepsze wyniki predykcji klasyfikacji.

25

CNN wady i zalety

CNN mają pewne ograniczenia, które są do pokonania:

- Uczenie się sieci wymaga **ogromnej ilości danych** treningowych
- Uczenie się sieci wymaga dużej **mocy obliczeniowej**
- **Proces uczenia się wymagaja odpowiedniej architektury wydajnego systemu komputerowego** mogą być złożone i często muszą być ściśle dostosowane do konkretnej aplikacji (programowalne tablice bramek (FPGA), procesory graficzne (GPU) i specyficzne dla aplikacji układy scalone (ASIC)) są badane w celu wykorzystania równoległości struktury obliczeniowej sieci neuronowych, bardziej niż równoległe procesory)
- Powstałe **modele mogą nie być łatwo interpretowalne**
- **Długi czas uczenia się** (metoda prób i błędów),
- Występowanie **nadmiernego dopasowania** w przypadku niewystarczającej ilości/liczby danych.
- Trudności w zapewnieniu **konwergencji procesu uczenia się**.
- **Wymagają od użytkownika zrozumienia, jak należy interpretować wyniki oddawane przez wyuczony model zjawiska/wyuczoną sieć, aby wysunąć sprawdzalne hipotezy dotyczące badań**

26

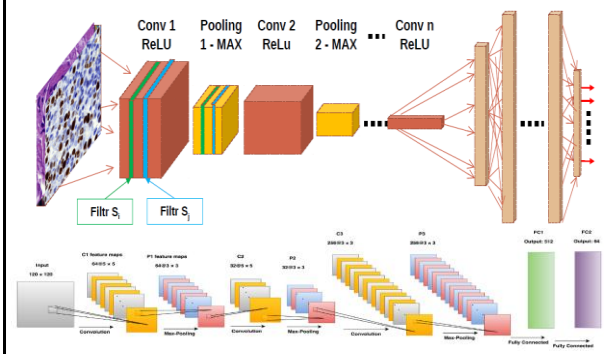
Przykłady sieci

27

Sieci, ich autorzy i rok ich powstania

Year	CNN	Developed by	Place	Top-5 error rate	No. of parameters
1998	LeNet(8)	Yann LeCun et al			60 thousand
2012	AlexNet(7)	Alex Krizhevsky, Geoffrey Hinton, Ilya Sutskever	1st		60 million
2013	ZFNet()	Matthew Zeiler and Rob Fergus	1st		
2014	GoogLeNet(19)	Google	1st		4 million
2014	VGG Net(16)	Simonyan, Zisserman	2nd		138 million
2015	ResNet(152)	Kaiming He	1st		

Podstawowa architektura sieci CNN



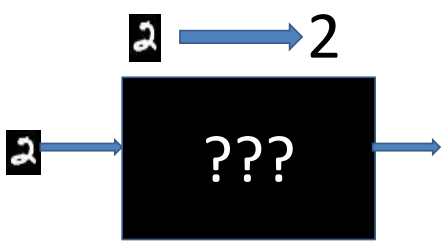
Hadi Rezaeilouyeh, Ali Mollahosseini, Mohammad H. Mahoor, "Microscopic medical image classification framework via deep learning and shearlet transform," J. Med. Imag. 3(4), 044501 (2016),doi: 10.1117/1.JMI.3.4.044501

29

Rozpoznawanie cyfr zapisanych ręcznie

np.: na czekach - Case study

Jak z obrazu uzyskać informację o cyfrze?



p(obiekt~0)
p(obiekt~1)
p(obiekt~**2**)
p(obiekt~3)
p(obiekt~4)
p(obiekt~5)
p(obiekt~6)
p(obiekt~7)
p(obiekt~8)
p(obiekt~9)

Baza obrazów MNIST

(*Modified National Institute of Standards and Technology*)

- 70 000 obrazów ręcznie pisanych cyfr
- 55 000 uczenie, 5 000 walidacja
- 10 000 testy
- 28x28 pikseli
- Wydany w 1999 roku
- 500 różnych osób
- Obraz + etykieta (jaka to cyfra)
- Skala kolorów znormalizowana do 0.0…1.0

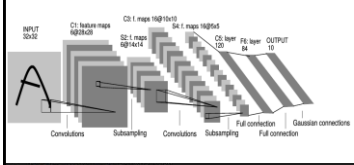
28 px

28 px

28 x 28 = 784 pikseli

Używamy do nauki:
THE MNIST DATABASE of handwritten digits
http://yann.lecun.com/exdb/mnist/

Sieć LeNet




```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_2 (Conv2D)            (None, 28, 28, 6)         156
_________________________________________________________________
average_pooling2d_2 (Average (None, 14, 14, 6)         0
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 10, 10, 16)        2416
_________________________________________________________________
average_pooling2d_3 (Average (None, 5, 5, 16)          0
_________________________________________________________________
flatten_1 (Flatten)          (None, 400)               0
_________________________________________________________________
dense_3 (Dense)              (None, 120)               48120
_________________________________________________________________
dense_4 (Dense)              (None, 84)                10164
_________________________________________________________________
dense_5 (Dense)              (None, 10)                850
=================================================================
Total params: 61,706
Trainable params: 61,706
Non-trainable params: 0
```

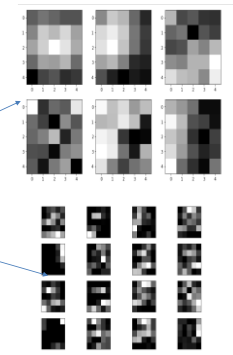
```
lenet5 = keras.models.Sequential()

lenet5.add(keras.layers.Conv2D(filters=6, kernel_size=(5, 5),
                               activation='relu', input_shape=(32,32,1)))
lenet5.add(keras.layers.AveragePooling2D())
lenet5.add(keras.layers.Conv2D(filters=16, kernel_size=(5, 5),
                               activation='relu'))
lenet5.add(keras.layers.AveragePooling2D())
lenet5.add(keras.layers.Flatten())

lenet5.add(keras.layers.Dense(units=120, activation='relu'))
lenet5.add(keras.layers.Dense(units=84, activation='relu'))
lenet5.add(keras.layers.Dense(units=10, activation = 'softmax')) # output
```

Praca zaliczeniowa na APO

studenta mgr inż. Brylewa

Ref. Y. Lecun, Gradient-Based Learning Applied to Document Recognition, PROCEEDINGS OF THE IEEE. 86 (1998) 47.

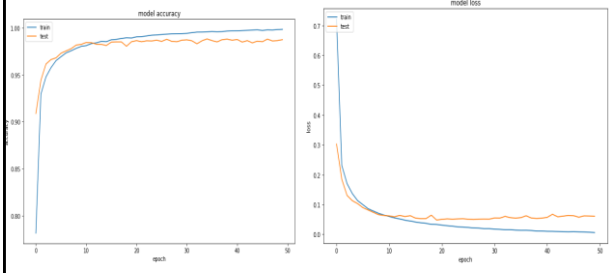
Uczenie się sieci to optymalizacja ze względu na cel



```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_2 (Conv2D)            (None, 28, 28, 6)         156
_________________________________________________________________
average_pooling2d_2 (Average (None, 14, 14, 6)         0
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 10, 10, 16)        2416
_________________________________________________________________
average_pooling2d_3 (Average (None, 5, 5, 16)          0
_________________________________________________________________
flatten_1 (Flatten)          (None, 400)               0
_________________________________________________________________
dense_3 (Dense)              (None, 120)               48120
_________________________________________________________________
dense_4 (Dense)              (None, 84)                10164
_________________________________________________________________
dense_5 (Dense)              (None, 10)                850
=================================================================
Total params: 61,706
Trainable params: 61,706
Non-trainable params: 0
```

[-0.0789532 -0.02418987 0.10222547 0.00066162 -0.04760809
0.09279951 -0.01421791 -0.07095297 0.05741353 -0.01717872]

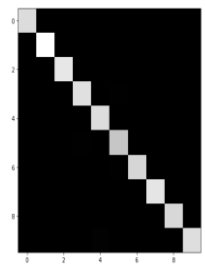
Parametry wyuczenia sieci



Najlepszego dopasowania

Minimalizacji błędu

Kontrola błędów



```
      0    1    2    3    4    5    6    7    8    9
0 [ 974    0    0    0    0    0    2    1    3    0]
1 [   0 1130    1    0    0    1    2    0    0    1]
2 [   4    1 1020    0    3    0    0    2    2    0]
3 [   1    0    2  993    0    6    0    4    2    2]
4 [   0    0    0    0  976    0    2    2    0    2]
5 [   2    0    0    7    0  875    2    2    3    1]
6 [   2    2    0    0    1    5  947    0    1    0]
7 [   0    3    4    1    1    0    0 1015    2    2]
8 [   1    0    1    4    3    3    1    2  955    4]
9 [   2    2    0    4   10    3    0    4    0  984]
```

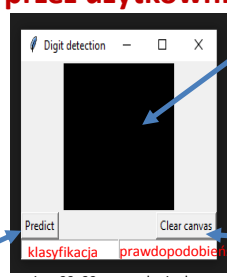
```
result = lenet5.evaluate(X_test, y_test)
dict(zip(lenet5.metrics_names, result))
```

```
313/313 [==============================] - 0s 1ms/step - loss: 0.0400 - accuracy: 0.9869
```

```
{'loss': 0.04001135712862015, 'accuracy': 0.9868999719619751}
```

Wizualizacja macierzy pomyłek

Moduł wykonawczy do rozpoznawania znaków narysowanych przez użytkownika

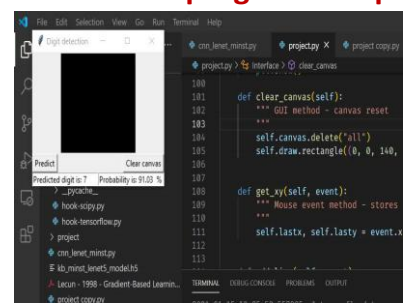


Powierzchnia na której można narysować cyfrę do rozpoznania

Czyszczenie powierzchni do rysowania

Zamiana obrazu na macierz 32x32 px, wysłanie do modelu, wyliczenia prawdopodobieństw przynależności do jednej z 10 klas i wybór prawdopodobieństwa maksymalnego

Działanie programu rozpoznającego



- Python 3.7
- Numpy
- Matplotlib
- Tensorflow 2.4 – wymaga AVX w zestawie instrukcji CPU!!!
- CUDA 10.1
- SciKit
- Tkinter - GUI

Moduł inferencji

U-net

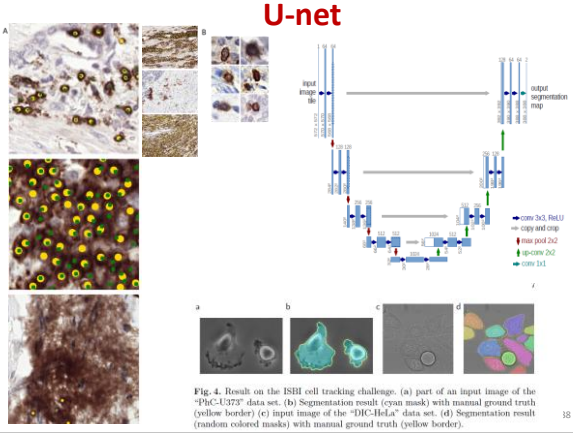
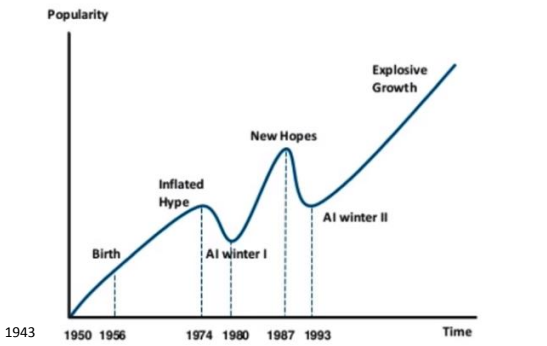


Fig. 4. Result on the ISBI cell tracking challenge. (a) part of an input image of the "PhC-U373" data set. (b) Segmentation result (cyan mask) with manual ground truth (yellow border) (c) input image of the "DIC-HeLa" data set. (d) Segmentation result (random colored masks) with manual ground truth (yellow border).

18

Historia i aktualne problemy NN

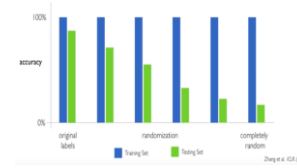
39



40

Ograniczenia DNN

- Można wytrenować sieć, która pasuje do kompletnie przypadkowych podziałów na klasy (na danych treningowych) da 100% dokładności

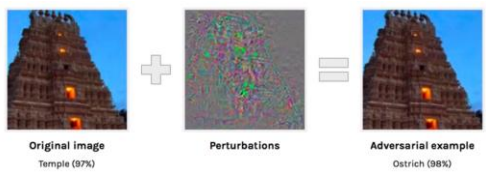


- Możemy wnioskowa o funkcji aproksymującej jedynie w zakresie danych zbioru uczącego.

41

Ograniczenia DNN

- Uwaga! na proces augmentacji bo czasem produkujemy „podobne przeciwieństwa" (*ang. adversarial examples*)



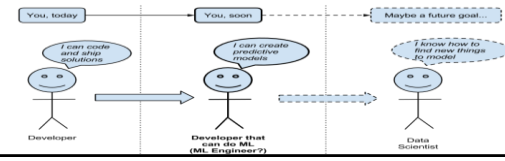
Despois. "Adversarial examples and their implications" (2017).

42

Ograniczenia DNN

- Dużo danych bo dużo współczynników do optymalizacji
- Obliczeniowo i wdrożeniowo (GPU) kosztowny
- Mogą być oszukiwane przez „podobne przeciwieństwa"
- Słabo można zbadać model, jego odporność i niepewność i inne parametry
- Model i oprogramowanie jest **czarną skrzynką** której trudno ufać – **szczególnie ekspertom/lekarzom**
- Nie każde podejście zawsze daje wynik w postaci dobrze optymalizowanej - potrzeba **prób i błędów**
- **Potrzebna wiedza eksperta** (wąskie gardło)

Praca dewelopera oprogramowania w dobie przyszłych mocno zaawansowanych sieci



43

Obawy społeczne związane z sztuczna inteligencją

- Przypominają obawy z XIX w sprawie **mechanizacji pracy** lub obawy o Puszkę Pandory
- **Czy technologia może coś chcieć**, być bezwzględnym megalomanem? **Nie ma woli i celu** – to ludzie mają chęć dominacji i gromadzenia zasobów i dokonują projekcji na AI
- Powody nieufności do technologii to głównie: **nieprzejrzystość i brak dostępu do procesu ich uczenia się sieci** (same wyszukują cechy ich zakresy i same się optymalizują wagi, a jak coś idzie nie tak to nadzorujący uczenie się maszyny właściwie nie wiedzą dlaczego, czy to można naprawić. Nadzorujący ponawia proces oparty na przypadkowym podawaniu przykładów ze zbioru uczącego).
- **Zbiory uczące przygotowuje człowiek i to jego wiedza jest w nich zawarta i przez niego wyselacjonowana** (postulat: **akredytacje dla deweloperów**).
- Technologia sztucznej inteligencji rozwijana się stopniowo, jest poprawiana pod względem bezpieczeństwa i skuteczności, projektowana, tak aby spełniać różne praktyczne kryteria
- Aktualne sukcesy AI wynikają z ich ogromnej siły obliczeniowej pozwalającej na przeanalizowanie ogromnej ilości danych uczących, czyli przypominanie **sawantka czyli genialnego idioty –** film „Rainmen" z Dustinem Hofmanem.
- Słowa Alisona Gopinka (University of Clifornia w Berkeley): „**Jak na razie znacznie więcej szkody narobić może naturalna głupota niż sztuczna inteligencja**."

44

Przykłady sieci

45

STEGANOGRAFIA

STEGANOGRAFIA

Nauka o komunikacji w taki sposób, by obecność komunikatu nie mogła zostać wykryta, czyli **ukrywanie/hermetyzacji informacji** (ang. *information hiding)*

Słowo „steganografia" pochodzi z języka greckiego i oznacza ukryte pismo

Zaleta steganografii w stosunku do szyfrowania informacji jest ukrycie samego faktu porozumiewania się stron.

Ukrywanie informacji obrazowej (lub tekstowej) w obrazie (*image watermarking*) w różnych celach:

59

Cele stosowania steganografii

1. Ochrona praw autorskich (ukryty obraz (lub tekst) pełni rolę znaku wodnego (*watermark*)), w tym przypadku konieczną własnością obrazu ukrytego jest (poza odpornością na usunięcie przez czynniki zewnętrzne) jego wystarczająca **odporność** na działanie standardowych operacji przetwarzania obrazów (filtracja, kompresja, zniekształcenia geometryczne itp.),

2. Ochrona autentyczności obrazu (ukryty obraz pełni rolę znaku wodnego (*watermark*)), w tym przypadku konieczną własnością obrazu ukrytego jest (poza odpornością na usunięcie przez czynniki zewnętrzne) jego wystarczająca **podatność** na działanie operacji przetwarzania obrazów (filtracja, kompresja, zniekształcenia geometryczne itp.).

3. Praktyczne przesyłania informacji

60

Historia steganografii

- Histiajos w niewoli króla perskiego Dariusza postanowił przesłać informację do swego zięcia Arystagorasa z Miletu stosując następujący fortel: na wygolonej głowie swego orasa niewolnika **wytatuował informację** , gdy niewolnikowi odrosły włosy posłał go z oficjalnym, mało istotnym listem
- Egipcie i Chinach powszechnie stosowano **atrament sympatyczny**.
- Demaratus ostrzegł Grecję o ataku, pisząc go bezpośrednio na drewnianym podłożu woskowej tabletki przed nałożeniem powierzchni wosku pszczelego.
- **Mikrokropki**: Pierwsze próby miniaturyzowania przesyłanych informacji za pomocą mikrofotografii podjęto w 1871 r. podczas wojny francusko-pruskiej, kiedy to przesyłano raporty do oblężonego przez Niemców Paryża w postaci prostokątów o wymiarach 3 cm × 4 cm. Udoskonalenia tej techniki dokonał wywiad niemiecki Abwehra na początku lat 40. XX wieku.
- punkt o średnicy 1 mm wykonany przez specjalne urządzenie będące połączeniem aparatu fotograficznego i mikroskopu, zawierający zminiaturyzowane dane tekstowe lub fotografie i szczegółowe rysunki techniczne. Skala miniaturyzacji wynosi ok. 1:300, oznacza to możliwość pomniejszenia kartki formatu A4 zawierającej np. tekst do wielkości pojedynczej kropki znajdującej się w tekście pisanego czcionką normalnej wielkości listu.
- Współcześnie technika mikrokropek (wykonywanych w tysiącach sztuk w technice grawerowania laserowego) jest wykorzystywana także komercyjnie np. do zabezpieczania żetonów w kasynach przed podrabianiem, znakowania samochodów lub cennych przedmiotów.
- Zapisy **kodem pod znaczkiem na liście**
- Podczas II Wojny Światowej Velvalee Dickinson nazywana **Doll Woman** japoński szpieg w Nowym Yorku z lalkami którymi handlowała przesyłała informacje szpiegowskie.
- **Jeremiah Denton** wielokrotnie mrugał oczami w alfabecie Morse'a podczas telewizyjnej konferencji prasowej w 1966 roku, po porwaniu przez porywaczy z Wietnamu Północnego, wypowiadając „**T-O-R-T-U-R-E**".
- W 1968 r. członkowie **załogi statku wywiadowczego USS Pueblo**, przetrzymywani przez Koreę Północną jako więźniowie, komunikowali się w języku migowym na zdjęciach.
- Obraz Koranu miał zaszyfrowane informacje o zamachu na World Trade Center

Ukrywanie obrazu w obrazie - nazwy obrazów
1. obraz ukrywający [**p**]
2. obraz ukrywany [**h**]
3. obraz ukrywany po przekształceniu [**hmod**]
4. obraz ukrywający wraz z obrazem ukrytym [**ph**]

Obrazy ukrywające z gradacją poziomów szarości
Obraz ukrywający: na 1 piksel obrazu ukrywanego przypada 1 bajt (8 bitów)
Obraz ukrywany może i powinien być uproszczony: na 1 piksel obrazu może przypadać 1, 2, 3, 4, 5, 6, 7, 8 bitów, co odpowiada 2,4,8,16,32,64,128,256 poziomom szarości (***M***)
W praktyce, ze względu na potrzebę ograniczenia wpływu obrazu ukrywanego na wygląd obrazu ukrywającego, stosowane są wartości ***M***<<256, a obraz ukrywany zapisywany jest na najmniej znaczących bitach obrazu ukrywającego.

Zajęcia badanie czy:
przy wzroście wartości ***liczby bitów obrazu ukrywanego*** następuje coraz większa zmiana wyglądu obrazu ukrywającego i jednocześnie coraz wyraźniejsze uwidocznienie obrazu ukrywanego w tym obrazie.

62

Obrazy ukrywające kolorowe np. w formacie RGB
Obraz ukrywający: na 1 piksel obrazu przypada 3 bajty (24 bity) odpowiadające 3 składowym R, G, B
Przykładowy zapis piksla obrazu ukrywanego: 2 najmłodsze bity składowej R oraz po jednym najmłodszym bicie składowych G i B (razem 4 bity, co odpowiada obrazowi o *M*=16 poziomach)
Zapis informacji tekstowej obrazie RGB: 1 znak – 8 bitów co oznacza że do zapisu 1 znaku można wykorzystać 2 piksle.

Dodatkowe kodowanie obrazów ukrywanych => lepsze ich ukrycie
(np. przemieszanie poszczególnych piksli)

Operacje najczęściej stosowane w procesie ukrywania i odtwarzania obrazu
- **Jednopunktowe jednoargumentowe**
  - **progowania, redukcji poziomów szarości, rozciągania, uniwersalne operacje punktowe (UOP)**
- **Jednopunktowe dwuargumentowe**
  - **arytmetyczne (dodawanie, odejmowanie)**
  - **logiczne (suma (OR), iloczyn (AND))**

63

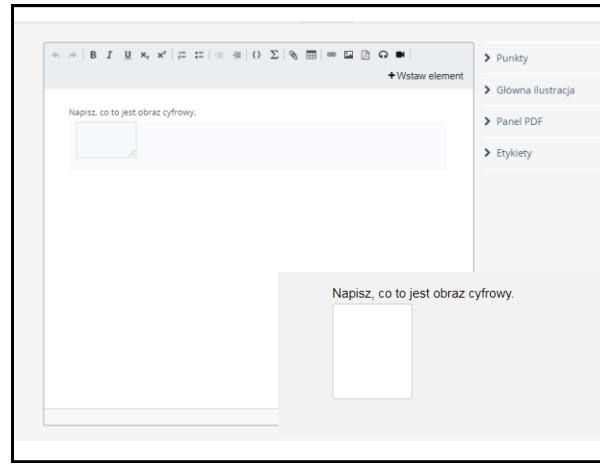
Koniec wykładu

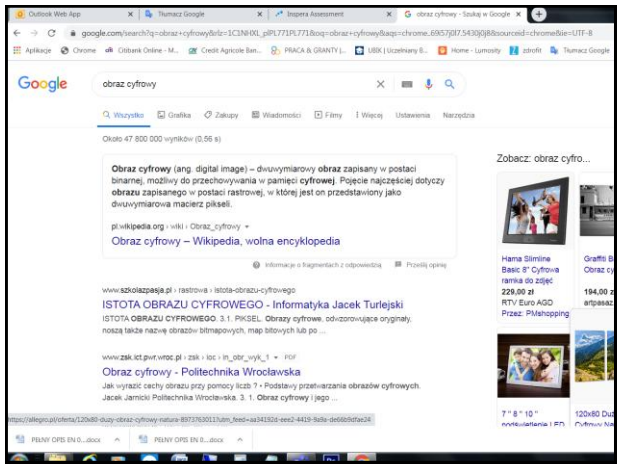
Egzamin będzie zdalny

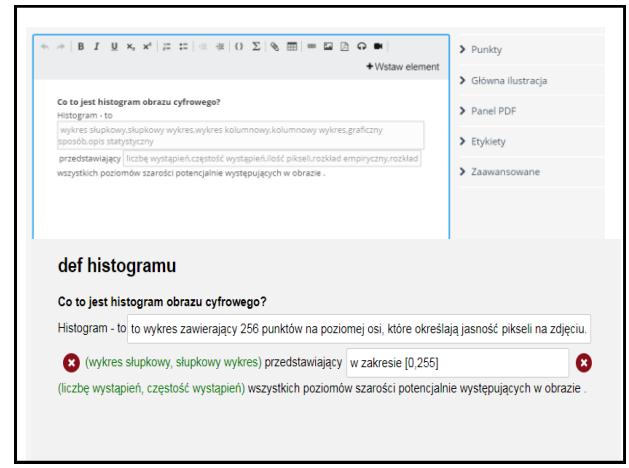
Terminy egzaminu: 02.07.2021 i 09.07.2021

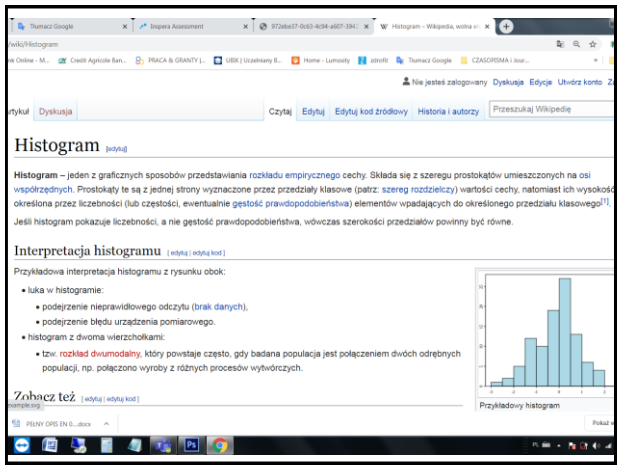
- Egzamin będzie na każdym etapie egzaminem testowym zdalnym.
- Będzie wykonywany przy pomocy oprogramowaniu **Inspera.**
- Kursy dla studentów na temat tego oprogramowania już się odbyły, ale można je jeszcze raz przejść rozwiązując udostępnione testy.

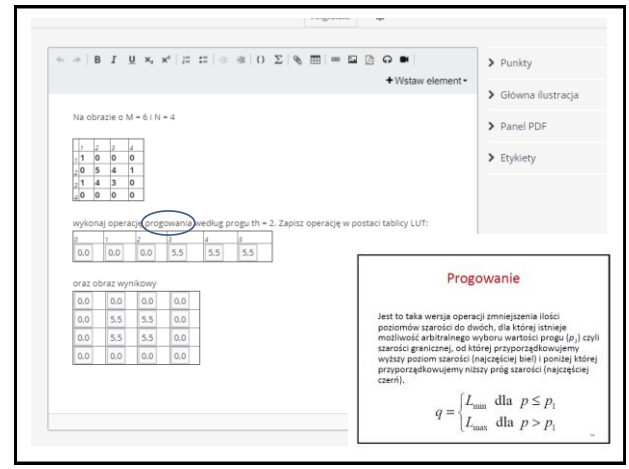
66

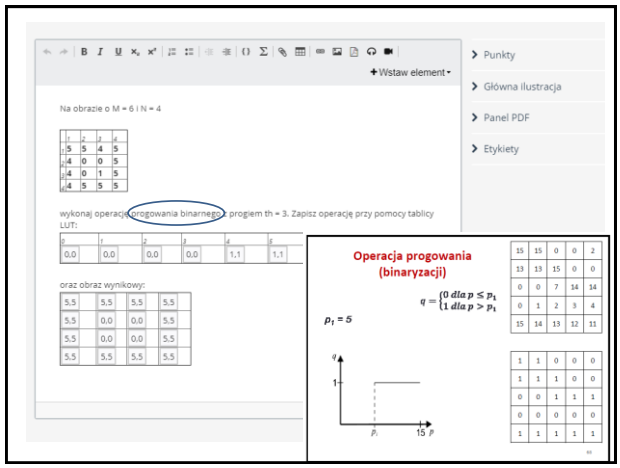


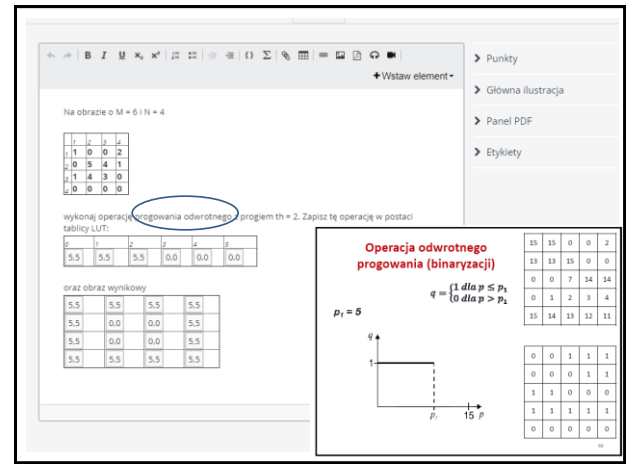

obraz cyfrowy
Obraz cyfrowy (ang. digital image) – dwuwymiarowy obraz zapisany w postaci binarnej, możliwy do przechowywania w pamięci cyfrowej. Pojęcie najczęściej dotyczy obrazu zapisanego w postaci rastrowej, w której jest on przedstawiony jako dwuwymiarowa macierz pikseli.
Obraz cyfrowy – Wikipedia, wolna encyklopedia
ISTOTA OBRAZU CYFROWEGO - Informatyka Jacek Turlejski
Obraz cyfrowy - Politechnika Wrocławska
Zobacz: obraz cyfro...


Punkty
Główna ilustracja
Panel PDF
Etykiety
Zaawansowane
def histogramu
Co to jest histogram obrazu cyfrowego?
Histogram - to wykres zawierający 256 punktów na poziomej osi, które określają jasność pikseli na zdjęciu.
(wykres słupkowy, słupkowy wykres) przedstawiający w zakresie [0,255]
(liczbę wystąpień, częstość wystąpień) wszystkich poziomów szarości potencjalnie występujących w obrazie .


Histogram
Interpretacja histogramu
Przykładowy histogram
Zobacz też


Punkty
Główna ilustracja
Panel PDF
Etykiety
Na obrazie o M = 6 i N = 4
wykonaj operację progowania według progu th = 2. Zapisz operację w postaci tablicy LUT:
oraz obraz wynikowy
Progowanie


Punkty
Główna ilustracja
Panel PDF
Etykiety
Na obrazie o M = 6 i N = 4
wykonaj operację progowania binarnego z progiem th = 3. Zapisz operację przy pomocy tablicy LUT:
oraz obraz wynikowy:
Operacja progowania (binaryzacji)

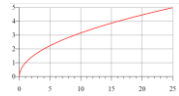

Punkty
Główna ilustracja
Panel PDF
Etykiety
Na obrazie o M = 6 i N = 4
wykonaj operację progowania odwrotnego z progiem th = 2. Zapisz tę operację w postaci tablicy LUT:
oraz obraz wynikowy
Operacja odwrotnego progowania (binaryzacji)

Jak rozwiązywać zadania obliczeniowe?

- Zadania z liczeniem odległości

Metryka Euklidesowa

$$\rho_1(\underline{x}^\mu, \underline{x}^\eta) = \sqrt{\sum_{\nu=1}^{n} (x_\nu^\mu - x_\nu^\eta)^2}$$



Metryka uliczna (Manhattan, city block distance):

$$\rho_2(\underline{x}^\mu, \underline{x}^\eta) = \sum_{\nu=1}^{n} |x_\nu^\mu - x_\nu^\eta|$$

Metryka Czebyszewa (maksymalna)

$$\rho_3(\underline{x}^\mu, \underline{x}^\eta) = \max_{1 \le \nu \le n} |x_\nu^\mu - x_\nu^\eta|$$

Jak rozwiązywać zadania problemowe?

- Czym różni się akwizycja w przypadku tych trzech obrazów dłoni?
- Czym różni się zapis informacji o obrazie dla tych trzech obrazów dłoni?