Uniwersytet im. Adama Mickiewicza w Poznaniu
Wydział Matematyki i Informatyki

Informatyka
Informatyka kto wie jaka to będzie

Magdalena Mozgawa
Nr albumu: **389479**

Ataki na systemy przetwarzania obrazu

Attacks on computer vision systems

Praca magisterska

Promotor:
dr inż. Michał Ren

2021 (oby)

Poznań, dnia .....

OŚWIADCZENIE

Ja, niżej podpisany IMIONA NAZWISKO student Wydziału Matematyki i Informatyki Uniwersytetu im. Adama Mickiewicza w Poznaniu oświadczam, że przedkładaną pracę dyplomową pt: „TYTUŁ PRACY" napisałem samodzielnie. Oznacza to, że przy pisaniu pracy, poza niezbędnymi konsultacjami, nie korzystałem z pomocy innych osób, a w szczególności nie zlecałem opracowania rozprawy lub jej części innym osobom, ani nie odpisywałem tej rozprawy lub jej części od innych osób.

Oświadczam również, że egzemplarz pracy dyplomowej w wersji drukowanej jest całkowicie zgodny z egzemplarzem pracy dyplomowej w wersji elektronicznej.

Jednocześnie przyjmuję do wiadomości, że przypisanie sobie, w pracy dyplomowej, autorstwa istotnego fragmentu lub innych elementów cudzego utworu lub ustalenia naukowego stanowi podstawę stwierdzenia nieważności postępowania w sprawie nadania tytułu zawodowego.

Wyrażam zgodę na udostępnianie mojej pracy w czytelni Archiwum UAM.

Wyrażam zgodę na udostępnianie mojej pracy w zakresie koniecznym do ochrony mojego prawa do autorstwa lub praw osób trzecich.

.....

(czytelny podpis studenta)

Tutaj będą podziękowania dla sąsiadów
za siedzenie cicho.

I coś dla promotora.

Spis treści

Spis rysunków

Spis tablic

Spis kodów źródłowych

Streszczenie

Streszczenie wstępu jest dobrym pomysłem na początek abstraktu. Dobre praktyki tworzenia abstraktów znajdują się np. na stronie[1]. Wczuj się w rolę informatyka, który będzie czytał sam abstrakt, żeby zdecydować, czy reszta pracy mu się przyda. Zwięzłość jest w cenie, niemniej jednak trzeba się starać opisać o czym głównie jest praca, jak również co jest w tej pracy szczególnego, czego nie można znaleźć w innych. Zwykle abstrakt pisze się po napisaniu pracy, myśląc o takich kwestiach jak np. „jaki problem próbowano rozwiązać", „jaka była motywacja skupienia się nad tym problemem", „za pomocą jakich środków cel został osiągnięty". Wiele abstraktów różnego rodzaju prac można obejrzeć w Internecie.

Słowa kluczowe: praca dyplomowa, wzór, przewodnik

Abstract

Translation of your Polish abstract. Some leeway is allowed, but make sure it is a translation, not a completely different abstract. If you have a problem with English, ask your supervisor to help you translate. Machine translations (e.g. Google translate) are not good enough (yet...) to be acceptable.

Keywords: thesis, template, guide

[1] `http://www.editage.com/insights/how-to-write-an-effective-title-and-abstract-and-choose-a`

Wstęp

Krótkie omówienie tego, o czym będzie praca. (Czyli co zostanie w pracy powiedziane.)

Rzeczy które tu można ująć to np.

- mini-przewodnik po własnych wynikach, czy że zrobiono to, tamto i owamto
- motywacja do pracy, czyli dlaczego się tym zajęto i dlaczego masa rzeczy już na ten temat napisanych nie wystarczyła do szczęścia
- omówienie struktury pracy, czyli w tym rozdziale jest to, a tym owo
- historia badań na podobnymi tematami i aktualny stan wiedzy

Ilu autorów, tyle wstępów... Nie traktuj powyższych elementów jako obowiązkowe.

1

Rozdział o systemach przetwarzania obrazu nieopartych o głębokie uczenie maszynowe

(KOMENTARZ W tym rozdziale opisano systemy przetwarzania obrazu nieoparte o głębokie uczenie maszynowe – czyli w szczególności te na bazie algorytmu Viola-Jonesa oraz oparte o histogramy zorientowanych gradientów. Czy w tym rozdziale warto też od razu opisywać ataki? Na razie to robię, ale być może warto będzie zrobić jakieś przetasowanie. Czas pokaże. Czego NIE opisuję w tym rozdziale (póki co): dlaczego redukować w ogóle wymiarowość obrazów ("przekleństwo wymiarowości")). Co będę opisywać w tym rozdziale, ale będzie obejmować co najmniej te dwa algorytmy, więc nie opisuję dla każdego z osobna: jak wygląda pipeline'a przetwarzania danych (ekstrakcja cech-deskryptorów, trenowanie modelu, testowanie modelu)).

1.1 Histogramy zorientowanych gradientów

Histogramy zorientowanych gradientów (ang. *histograms of oriented gradients*, dalej: HOG) to deskryptory pozwalające na opisanie zawartości danego obrazu za pomocą wielkości i orientacji gradientów. Technika ta pozwala na redukcję wymiarowości obrazu oraz zniwelowanie wpływu lokalnych różnic na całość deskryptora.[1]

1.1.1 Opis algorytmu

Algorytm tworzenia histogramów zorientowanych gradientów składa się z dwóch faz: wyliczenia gradientów oraz głosowania histogramów. Pierwsza pozwala na pozyskanie dla n x m-wymiarowego obrazu w skali RGB dwóch n x m-wymiarowych macierzy opisujących gradienty w tym obrazie. Druga korzysta z tych macierzy do wyznaczenia histogramów zorientowanych gradientów w celu dalszej redukcji wymiarowości obrazu. Fazy te zostały bardziej szczegółowo opisane poniżej.

Wyliczenie gradientów. Plikiem wejściowym jest analizowany obraz o wymiarach n x m pikseli, który jest przetwarzany do 8-bitowej skali szarości i dzielony na nakładające komórki (ang. *cells*), np. 3x3 piksele. W każdej komórce korzystając z równań: TUTAJ WKLEIĆ TE RÓWNANIA wyznaczany jest wielkość i kierunek gradientu względem jej centralnego piksela. Uzyskuje się w ten sposób dwie macierze n x m z wartościami odpowiadającymi wielkościom gradientów oraz ich kątom *modulo* 180°.

Głosowanie histogramu. Elementem wejściowym są macierze uzyskane w kroku pierwszym. Macierze są dzielone na nienakładające się bloki (ang. *blocks*) np. 8x8 wartości. Następie w obrębie każdego bloku odbywa się głosowanie, którego wynikiem jest histogram danego bloku. Szczegółowy algorytm głosowania pokazano w pseudokodzie.

Algorithm 1 Głosowanie histogramu w bloku

Wejście: B_k – blok kątów w postaci listy i-elementowej, B_w – blok wielkości gradientów w postaci listy i-elementowej.

Wyjście: H – 9-elementowa lista definiująca histogram o klasach $0 - 20°$, $20 - 40°$, ..., $160 - 0°$.

$H \leftarrow [0, 0, 0, 0, 0, 0, 0, 0, 0]$
for $i \leftarrow 1, n$ **do**
    $c \leftarrow \lfloor B_{k_i} \rfloor$
    $h \leftarrow c \div 20$
    if $c == B_{k_i}$ **then**
        $H[h] \leftarrow B_{w_i} \div 2$
        $H[h-1] \leftarrow B_{w_i} \div 2$
    else
        $H[h] \leftarrow B_{w_i}$

Efekt końcowy. Wynikiem działania zastosowanej metody są histogramy zorientowanych gradientów. Warto zauważyć, że użycie tej metody prowadzi do

znacznego zmniejszenia wymiarowości danych. Przykładowo, można policzyć, że 24-bitowy obraz o wymiarach 64x128 pikseli, do którego opisania pierwotnie potrzeba by 24 576 wartości, może być podzielony na 27 bloków 16x16 pikseli, co daje 243 wartości w histogramach opisujące cały obraz. Jest to ponad stukrotne zmniejszenie liczby wartości opisujących obraz, które wciąż pozwala na stworzenie stosunkowo skutecznego deskryptora.[1]

1.1.2 Klasyfikatory oparte o histogramy zorientowanych gradientów

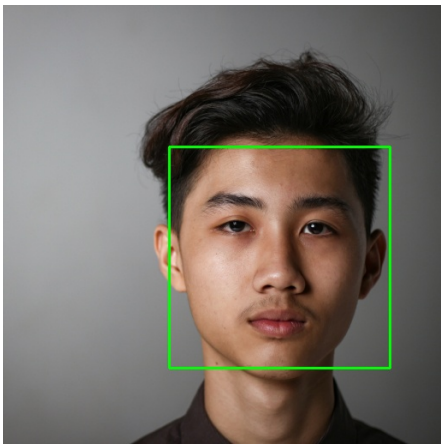
Wyliczenie deskryptora dla wielu obrazów to pierwszy krok do stworzenia klasyfikatora danej cechy opartego o uczenie maszynowe. Popularność HOG przypada na wczesne lata 2000, kiedy znacznym poważaniem cieszyły się m.in. maszyny wektorów podpierających (ang. *Support Vector Machines*, dalej: SVM). Jest to model uczenia maszynowego, zaproponowany przez Vladimira Vapnika w 1995, pozwalający na wyznaczenie optymalnej hiperpłaszczyzny, która odziedzialałaby dane z różnych klas zachowując największy możliwy margines zaufania.[6, 11] Ten model został zastosowany również przez autorów metody histogramów zorientowanych gradientów[1], jednak jego bardziej szczegółowe omówienie wykracza poza zakres tej pracy i nie jest konieczne do zrozumienia sposobu działania metody ani możliwych dróg ataku na nią. (KOMENTARZ Chyba że promotor bardzo będzie tego chciał albo zabraknie materiału.)

Wytrenowany klasyfikator może zostać następnie zastosowany do wykrywania danej cechy na różnych zdjęciach, także takich, które nie były użyte do jego wytworzenia. Przykładem klasyfikatora, którego można użyć do wykrycia twarzy *en face* jest ten załączony do biblioteki *dlib*, otwartoźródłowej biblioteki z narzędziami do uczenia maszynowego, napisanej w języku C++ i posiadającej też API do języka Python.[7] Poniżej zaprezentowano przykładową implementację wspomnianego klasyfikatora w celu rozpoznania twarzy na zdjęciu oraz jego efekt.

```
import numpy as np
import cv2
import matplotlib.image as mpimg
import matplotlib.pyplot as plt
import dlib

image_colour = mpimg.imread('pictures/face.jpg') # load the image
image_gray = cv2.cvtColor(image_colour, cv2.COLOR_BGR2GRAY) # grayscale
    conversion
detector = dlib.get_frontal_face_detector() # load the classifier
```

Rysunek 1.1: Oryginalne zdjęcie.[9]



Rysunek 1.2: Efekt działania klasyfikatora.

```
faces = detector(image_gray, 0) # search for all potential faces in the
     grayscale image

res = image_colour.copy() # create a copy of the original image (to
    draw on it later)

for face in faces:
    # for each detected face, draw a green bounding box around it
    cv2.rectangle(res, (face.left(), face.top()), (face.right(), face.
        bottom()), (0, 255, 0), 2)

cv2.imwrite('pictures/face_detected.jpg', cv2.cvtColor(res, cv2.
    COLOR_BGR2RGB)) # save results
```

Kod źródłowy 1.1: Wyszukiwanie twarzy z użyciem biblioteki dlib

1.1.3 Metody ataku

1.2 Algorytm Viola-Jonesa

1.3 Podrozdział o bibliografii

Podrozdziały mogą czytelnikowi ułatwić przyswajanie pracy – hierarchicznie uporządkowaną treść lepiej się czyta.

Dobrze jest cytować artykuły naukowe tak, żeby stwierdzenia zawarte w pracy, które nie są wynikiem oryginalnych myśli autora zawierały do nich odniesienie.

Rysunek 1.3: To jest logo UAM. Źródło: System Identyfikacji Wizualnej UAM [5]

Są różne szkoły cytowania, ale w informatyce przyjęło się, że bibliografię umieszczamy na końcu pracy numerując, a w tekście piszemy numer w nawiasie kwadratowym, np. tak [99]. Nie umieszczamy bibliografii w przypisach dolnych[1] – one służą raczej wyjaśnianiu różnych pojęć itp.

Na szczęście LaTeX dużo sam robi – w bibliografii (patrz plik bibliografia.bib) umieszczamy swoje pozycje bibliograficzne, a w pracy odwołujemy się do nich przez nazwy które sami im nadaliśmy. LaTeX sam się zatroszczy w jakim stylu zacytować. W obecnej bibliografii jest kilka przykładów, np. książka „*Kryptografia i bezpieczeństwo sieci komputerowych. Matematyka szyfrów i techniki kryptologii*"[**?**], strona internetowa „*Seminarium ZATABEDA*"[2], artykuł zamieszczony w Internecie „*How to write an effective title and abstract and choose appropriate keywords*"[10], wiadomość z grup dyskusyjnych „*Random numbers for C: The END?*"[8] i prezentacja lub wykład zamieszczone w Internecie „*Szumy pseudolosowych map.*"[3].

Wiele źródeł internetowych, szczególnie prac naukowych, już zawiera informację bibliograficzną w formacie LaTeXa, którą można wkleić do swojego pliku z bibliografią.

Jeśli samemu znajdujesz źródła w Internecie, pamiętaj że trzeba podać autora, tytuł, rok publikacji, informacje wystarczające do znalezienia źródła, etc. Czasem „autor" jest umowny, np. może to być instytucja.

[1] To jest przypis dolny.

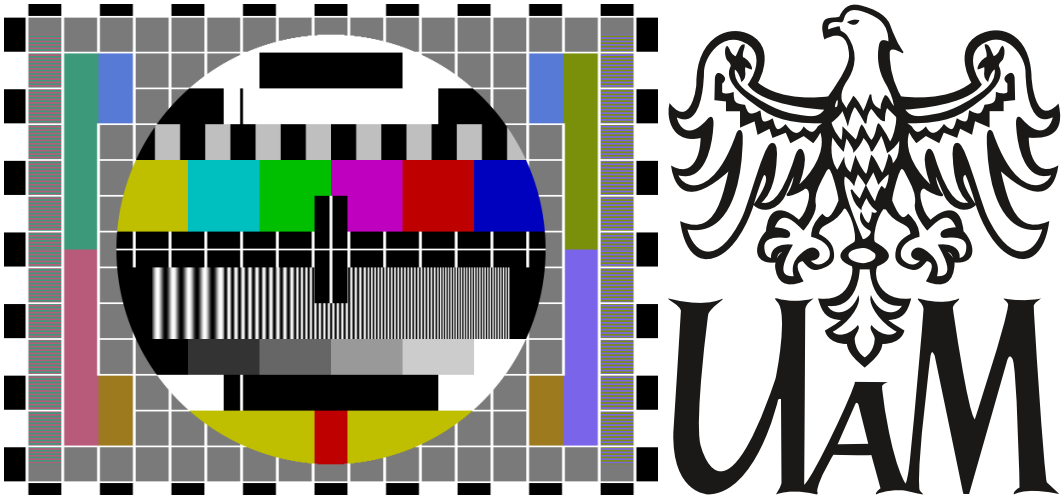
1.4 Kilka przykładów typografii

- **Wymieniany** – element może mieć zagnieżdżenia:
  - **Przykład**, faktycznie tu jest zagnieżdżenie.

Definicja 1. ***Można pogrubić nazwę określanego pojęcia*** *aby było jasne co definiujemy.*

2

Inny rozdział



Rysunek 2.1: Obrazki rastrowe i wektorowe

Rysunek 2.1 pochodzi z takiej tam strony[1].

2.1 Znów podrozdział

Przykład inline'owego trybu matematycznego: $X_{n+1} = aX_n + c \pmod{m}$.

[1] `http://siw.amu.edu.pl/siw/strona-glowna/strona-glowna`

2.1.1 Ostatni poziom zagłębienia uwzględniany w domyślnym spisie treści.

Gracze na szczęście nie przejmowali się dotarciem do „końca świata" [3]
– Marcin Mateusz Hanc

Nagłówek	Coś innego
Pierwszy rząd	100
Rząd podwójny	TAK

Tablica 2.1: Przykładowa tabela

```
//for each pixel, get the value
total = 0.0f;
frequency = 1.0f/(float)octaves;
amplitude = gain;

for (i = 0; i < octaves; ++i)
{
    total += noise(
        (float)x * frequency,
        (float)y * frequency
    ) * amplitude;
    frequency *= lacunarity;
    amplitude *= gain;
}

//now that we have the value, put it in
map[x][y]=total;
```

Kod źródłowy 2.1: Ułamkowy Ruch Browna

2.2 Sekcja ze znakiem _ działa

1. Elementy mogą być dłuższe niż jedna linia.

   FAKTYCZNIE.

2. Drugi element.

3

Typografia – dobre rady

Należy w pracy uważać, żeby jednoliterowe wyrazy takie jak „i", „a", etc. nie kończyły linii. Można to łatwo uzyskać pisząc ~(znak tyldy) za taką literką. Znak tyldy oznacza spację, której nie wolno dzielić; w innych przypadkach w których chcielibyśmy tego uniknąć, też można ten trick stosować. W tym akapicie jest to praktykowane, żeby pokazać technikę, a w całej reszcie szablonu – nie. Niestety, póki LaTeX nie zmądrzeje, trzeba to wszędzie ręcznie robić.

Proszę pamiętać o różnicy różnicy między dywizem, myślnikiem i półpauzą. Otóż – jak to zresztą widać w innych miejscach w szablonie – trzeba używać dwóch kresek do oznaczenia „myślnika" w zdaniu. W tradycyjnej polskiej typografii myślnik to —, a znak – to tzw. półpauza, jednak zaczyna ona wypierać myślnik i jest dziś powszechnie używana zamiast niego, więc tak radzę pisać. Pojedynczej kreski w LaTeXu używamy np. w zakresach (strony 1-10), albo złożeniach typu czerwono-czarne.

Wyrazy obcojęzyczne, można oznaczać kursywą – przykładowo kiedy się pisze np. o własności *non-repudiation*. Niektóre terminy można też doprecyzować podając w nawiasie źródło (ang. *source*).

W języku polskim, cudzysłów otwierający pisze się inaczej niż zamykający, a w LaTeXu pisze się je „tak".

Zanim pracę odda się promotorowi, warto ją przeczytać. Niestety, mózg autora buntuje się często wobec próby czytania po raz n-ty czegoś, co sam wymyślił. Radzę więc czytać na głos i się nagrać, a promotorowi wysłać audiobooka. Niektóre błędy wynikające z wielokrotnej edycji można wtedy wyłapać.

Na wszelki wypadek, należy też zajrzeć na stronę wydziału [4] i sprawdzić, czy nie zmieniły się wymagania dotyczące pracy dyplomowej. (W razie wątpliwości

można też dziekanat pytać o różne szczegóły.) Wyrocznią jest dziekanat, a nie ten szablon!

Zakończenie

Tak jak we wstępie pisało się o czym będzie praca, tak w zakończeniu pisze się o czym praca była. W zakończeniu podaje się często pomysły na dalsze badania w danym kierunku, ale jeśli takich pomysłów jest więcej lub są szczegółowe, to czasami robi się to w całym rozdziale przed zakończeniem.

Bibliografia

[1] N. Dalal and B. Triggs. *Histograms of oriented gradients for human detection*, volume 1, pages 886–893 vol. 1. 2005.

[2] M. Gogolewski. Seminarium ZATABEDA, część: bezpieczeństwo danych i kryptografia. `http://marcing.faculty.wmi.amu.edu.pl/seminarium.html`, 2013. Dostęp: 2015-07-22.

[3] M.M. Hanc. Szumy pseudolosowych map. `http://marcing.faculty.wmi.amu.edu.pl/prezentacje/szumy`, March 2015. Dostęp: 2015-07-22.

[4] Wydział Matematyki i Informatyki UAM Poznań. Wymagania dotyczące prac dyplomowych. `https://www.wmi.amu.edu.pl/pl/prace-dyplomowe#uko%C5%84czenie-studi%C3%B3w`, 2018. Dostęp: 2019-04-03.

[5] Uniwersytet im. Adama Mickiewicza w Poznaniu. System Identyfikacji Wizualnej. `http://siw.amu.edu.pl/`, 2012. Dostęp: 2019-04-03.

[6] N. Jankowski. *Ontogeniczne sieci neuronowe. O sieciach zmieniających swoją strukturę*. 1st edition, 2003.

[7] Davis E. King. dlib. a toolkit for making real world machine learning and data analysis applications in c++.

[8] G. Marsaglia. Random numbers for C: The END? Message-ID 36A5FC62.17C9CC33@stat.fsu.edu w grupach dyskusyjnych sci.math i sci.stat.math, January 1999. Dostęp: 2015-07-22.

[9] Imansyah Muhamad Putera. boy's face. `https://unsplash.com/photos/n4KewLKFOZw`, 2020. Dostęp: 2020-06-17.

[10] V. Rodrigues. How to write an effective title and abstract and choose appropriate keywords. http://www.editage.com/insights/how-to-write-an-effective-title-and-abstract-and-choose-appropriate-keyworc November 2013. Dostęp: 2015-07-22.

[11] Vladimir N. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, Berlin, Heidelberg, 1995.

Dodatek A: Cośtam dodatkowego

Dodatki zawierają treści, których zrozumienie nie jest konieczne do zrozumienia pracy i które mogłyby niepotrzebnie zajmować miejsce. Czasem są to listingi programów – może np. w pracy wystarczą krótkie fragmenty do których się akurat odnosimy, a w dodatku może być cały kod źródłowy, itp.

Akronimy

KISS Keep It Simple Stupid

IT Information Technology