Politechnika Śląska
Wydział Automatyki, Elektroniki i Informatyki
kierunek: informatyka

Mateusz Trzeciak

Określenie wieku twarzy na podstawie tekstury

praca dyplomowa magisterska

promotor: dr hab. inż. Karolina Nurzyńska

Gliwice, listopad 2019

załącznik nr 2 do zarz. nr 97/08/09

Oświadczenie

Wyrażam zgodę / Nie wyrażam zgody* na udostępnienie mojej pracy dyplomowej / rozprawy doktorskiej*.

Gliwice, dnia 4 listopada 2019

.....
(podpis)

.....
(poświadczenie wiarygodności
podpisu przez Dziekanat)

* podkreślić właściwe

Oświadczenie promotora

Oświadczam, że praca „Określenie wieku twarzy na podstawie tekstury" spełnia wymagania formalne pracy dyplomowej magisterskiej.

Gliwice, dnia 4 listopada 2019

.....

(podpis promotora)

Spis treści

1 Wstęp **1**

2 Metoda bazowa - wskaźnik zmarszczek **5**

2.1 Metoda wykrywania twarzy 8

2.1.1 Przestrzenie barw oraz skala szarości 12

2.1.2 Metoda detekcji twarzy 14

2.2 Wyznaczanie stref 19

2.3 Wykrywanie zmarszczek - detektor Canny 25

2.4 Wyliczanie współczynnika zmarszczek 29

2.5 Algorytm analizy danych ze zbioru trenującego 31

2.6 Grupowanie danych - Fuzzy C-Means oraz wyznaczanie wieku 32

3 Modyfikacje metody bazowej **37**

3.1 Odjęcie wybranej strefy 37

3.2 Opis metody HOG 38

3.3 Metoda HOG oraz algorytm KNN 44

3.3.1 Zastosowanie w projekcie 45

3.3.2 Grupowanie KNN 45

4 Badania **51**

4.1 Środowisko, metodyka badań oraz statystyki z działania algorytmów 51

4.2 Wykrywanie zmarszczek 54

4.3 Skuteczność poszczególnych metod szacowania wieku 58

4.3.1 Metoda bazowa 59

4.3.2 Odjęcie strefy 2 62
4.3.3 Metoda HOG 64
4.3.4 Metoda HOG + KNN 67

5 Podsumowanie **69**

Rozdział 1

Wstęp

Wiek jest cechą, którą niełatwo człowiekowi odczytać z czyjejś twarzy. Dla komputera rozpoznawanie wieku jest trudniejsze niż dla człowieka. Dlatego do wyznaczania wieku z pomocą programu komputerowego należy podchodzić z dystansem. Mimo trudności programiści i naukowcy udoskonalają algorytmy, tak aby ocena wieku danej osoby była coraz dokładniejsza.

Istnieje wiele sposobów wyznaczania wieku. Większość metod skupia się na analizie tekstury twarzy. Idąc dalej - z obrazu danej osoby musi zostać wykryta twarz. Wykrycie twarzy na obrazie jest możliwe dzięki algorytmom segmentacji obrazu. Segmentacja obrazu jest stosowana w wizji komputerowej i polega na wyodrębnieniu z obrazu jakichś szczegółów. Mogą to być na przykład osoby, pojazdy, przedmioty itp. (rys. 1.1)

Można znaleźć wiele witryn internetowych, które udostępniają interfejsy programistyczne umożliwiające zaimplementowanie rozpoznawania wieku z obrazu. Istnieją algorytmy przetwarzania obrazu, które oprócz wieku wyznaczają z pewnym prawdopodobieństwem płeć danej osoby. Oprócz płci mogą one także wyznaczyć emocję oraz czy dana osoba nosi okulary.

Z weryfikacją wieku danej osoby można się spotkać przed wejściem do niektórych miejsc, takich jak klub nocny. Większość osób musi okazać ważny dowód osobisty, co generuje duże kolejki do wejścia. Aplikacje analizujące wiek na podstawie obrazu twarzy z kamery przed wejściem do takich miejsc znacząco usprawniłyby weryfikację wieku. Rozpoznawanie wieku może być

Rysunek 1.1: Przykład rozpoznawania osób na obrazie [28].

wykorzystywane przy analizie średniego wieku ludzi w jakimś miejscu np. podczas demonstracji.

Wiele gier posiada treści nieodpowiednie dla młodszych użytkowników. Możliwe jest stosowanie technologii wykrywania wieku użytkownika przed udostępnieniem mu treści, która wymaga odpowiedniego wieku.

Można znaleźć o wiele więcej potencjalnych zastosowań przetwarzania obrazu oraz rozpoznawania wieku na podstawie tekstury (obrazu) twarzy. Z biegiem lat z pewnością będzie można zauważyć dalszy rozwój tej dziedziny, która opiera się w głównej mierze na sztucznej inteligencji [30].

Cel i zakres pracy

Celem niniejszej pracy jest stworzenie algorytmu do szacowania wieku. Bazą dla przeprowadzonych badań było odtworzenie pracy „Age Estimation from Face Image using Wrinkle Features" [26]. Metodę w niej zawartą poddano następnie kilku modyfikacjom.

Zakres pracy obejmuje następujące zagadnienia:

- Implementacja algorytmu z pracy [26], w dalszej części pracy określa-

nego jako metoda oryginalna lub metoda bazowa;

- Zaproponowanie kilku modyfikacji oryginalnej metody w celu polepszenia jakości szacowania wieku;
- Opis algorytmów oraz zmian w stosunku do oryginalnej metody;
- Przeprowadzenie badań w celu rozstrzygnięcia, który algorytm jest najskuteczniejszy w działaniu;
- Wyciągnięcie wniosków z przeprowadzonych badań.

W rozdziale 2 zaprezentowano algorytm szacowania wieku oparty na oryginalnej metodzie. W sekcji 2.1 skupiono się na metodach wykrywania twarzy wraz z opisem metody wykrywania w bazowym algorytmie. Następnie w sekcji 2.2 zaprezentowano sposób wyznaczania stref zmarszczkowych z oryginalnej metody. W kolejnej sekcji 2.3 przedstawiono sposób detekcji zmarszczek na twarzy implementowany w metodzie bazowej. Przedostatnia sekcja rozdziału drugiego 2.5 prezentuje sposób analizy zależności pomiędzy wskaźnikiem zmarszczek, a wiekiem danej osoby. W ostatniej sekcji 2.6 tego rozdziału przedstawiono algorytm grupowania danych niezbędny do oszacowania wieku.

W kolejnym rozdziale 3 opisuje 3 modyfikację oryginalnej metody. Oprócz sposobu implementacji danych modyfikacji czytelnik zostaje zapoznany z teoretycznymi podstawami wykrywania zmarszczek za pomocą metody HOG (ang. Histogram of Oriented Gradients) oraz szacowania wieku dzięki algorytmowi KNN (ang. k-nearest neighbors).

W przedostatnim rozdziale 4 przedstawiono sposób realizacji badań oraz ich wyniki. W ostatnim rozdziale 5 podsumowano niniejsza pracą.

Rozdział 2

Metoda bazowa - wskaźnik zmarszczek

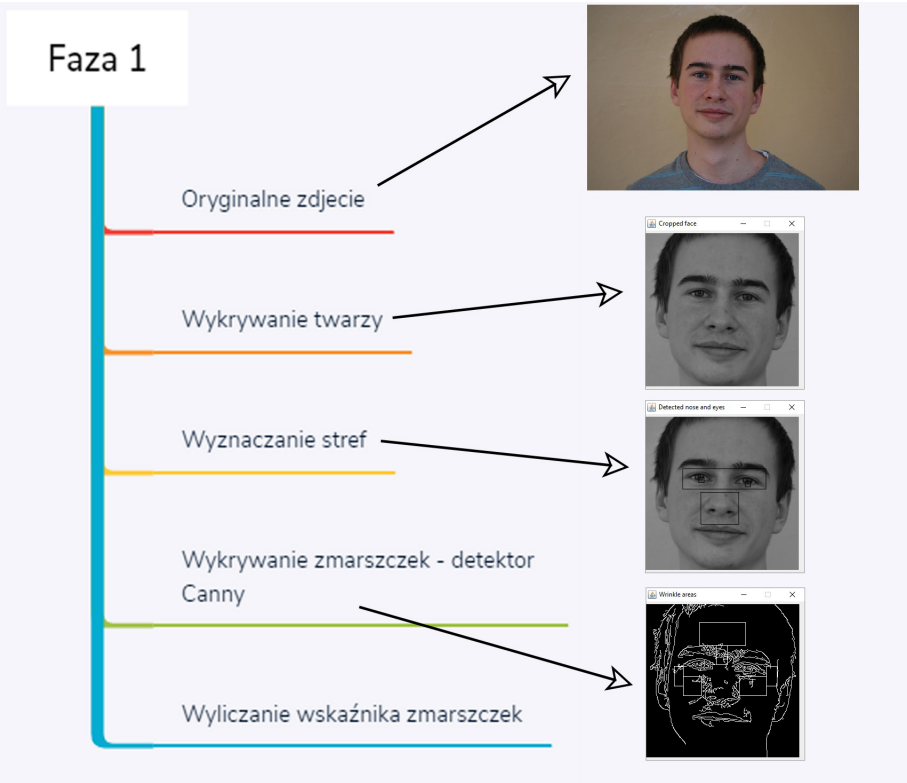
Istnieje wiele metod wyznaczania wieku z obrazu twarzy. Jedna z pierwszych metod szacowania wieku opierała się na wyznaczaniu proporcji twarzy, a następnie na detekcji i interpretacji zmarszczek. Była ona w stanie ze stuprocentową poprawnością wyznaczyć, czy dana osoba jest osobą dorosłą, czy dzieckiem [33].

W kolejnych latach algorytmy i techniki szacowania wieku były udoskonalane. Badano wpływ starzenia się osób na wygląd skóry. Oprócz naturalnych zmian skóry pod wpływem jej starzenia sie należało uwzględnić także inne czynniki, do których należą między innymi płeć, poziom stresu, czy ekspozycja na działanie środowiska zewnętrznego. Badania tego rodzaju opisano w pracy [1] A. Lanitis, Ch. J. Taylor oraz T. F. Cootes. Ponadto w tej pracy zaprezentowano algorytm rozpoznawania płci, emocji czy osób na podstawie obrazu twarzy.

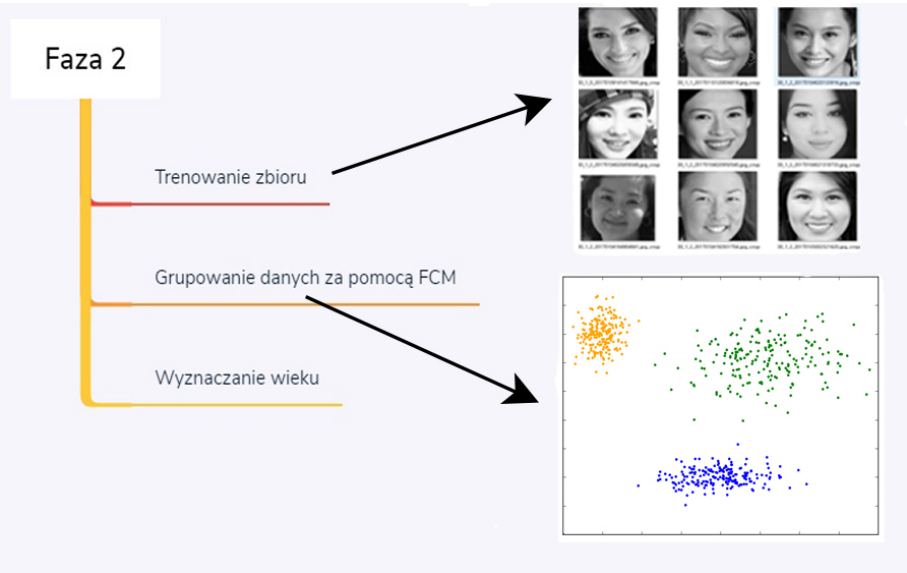
W kolejnych latach zaczęto przeprowadzać porównania cech twarzy tej samej osoby w różnym wieku. Różnice w powyższych cechach posłużyły do zbudowania statystyki zmian cech twarzy wraz ze starzeniem się. Taka analiza została zaprezentowana w pracy [22] N. Ramanathan oraz R. Chellappa. Rozwinięciem tego pomysłu była praca [32] X. Geng, Z. Zhou i K. Smith-Miles. W tym artykule porównywano sekwencje wielu zdjęć twarzy jednej

osoby. Zdjęcia przestawiały twarz w różnym wieku. Te badania pozwoliły na zbudowanie wzorca starzenia się twarzy. Z kolei M.M. Dehshibi oraz A. Bastanfard w swojej pracy [8] analizują proporcje twarzy oraz ilość zmarszczek przy szacowaniu wieku. Praca [15] autorstwa Vladimira Khryashcheva, Alexandra Ganina, Olgi Stepanovej oraz Antona Lebedeva zawiera podsumowanie technik szacowania wieku. Wynika z niego, że do wyodrębniania cech z twarzy stosuje się BIF(ang. biologically inspired features). Powyższa metoda została zaprezentowana w książce [12] Guodong Guo i in. Mniej popularne metody analizujące cechy twarzy to filtry Gabora oraz LBP (ang. local binary patterns) [25].

W metodzie bazowej, która została opisana w artykule [26], wykrywanie wieku dzieli się na kilka faz. Na początku należy wykryć twarz. Zastosowany algorytm wykrywania został opisany w sekcji 2.1. Następnie należy wyznaczyć strefy zmarszczkowe na twarzy. W artykule [26] udowodniono, że istnieje kilka konkretnych stref, w których następuje znacząca zmiana ilości zmarszczek wraz z wiekiem. Powyższe strefy zostały wymienione w sekcji 2.2. Sekcja 2.3 przedstawia technikę wykrywania zmarszczek znajdujących się w strefach. Wykryte zmarszczki pozwalają na obliczenie wskaźnika zmarszczek dla danej twarzy, zgodnie z opisem w sekcji 2.4. W tym miejscu kończy się faza wyznaczania wskaźnika zmarszczek dla danej osoby (rysunek 2.1). Kolejna faza jest potrzebna do znalezienia relacji pomiędzy wskaźnikiem zmarszczek a wiekiem. Do tego celu należy zastosować algorytm trenujący, który został opisany w sekcji 2.5. Wynikiem algorytmu trenującego jest zbiór danych, który należy pogrupować, tak jak to opisano w sekcji 2.6. Ostatnią fazą algorytmu jest wykrywanie wieku na podstawie wyników działania FCM (ang. Fuzzy C-Means) - sekcja 2.6 (rysunek **??**).



Rysunek 2.1: Faza 1 algorytmu



Rysunek 2.2: Faza 2 algorytmu

2.1 Metoda wykrywania twarzy

W literaturze można odnaleźć wiele metod wykrywania twarzy. Istnieje kilka podejść aby skutecznie wykrywać twarz na danym obrazie [27]:

- metoda oparta na nauce (ang. knowledge - based method):

  Metoda oparta na nauce kieruje się wiedzą na temat wyglądu twarzy, a bardziej precyzyjnie chodzi o charakterystyczne cechy, dzięki którym na zdjęciu można wyodrębnić obszar twarzy. Mowa tutaj o cechach takich jak kształt twarzy, kolor, miejsca o różnej jasności czy specyficzne krawędzie tworzone np. przez usta.

- metoda niezmienności cech (ang. features invariant method):

  W metodzie niezmienności cech wyszukuje się strukturalne cechy twarzy, które są widoczne w każdych warunkach oświetleniowych. Ponadto te cechy są widoczne bez względu na punkt widzenia, nawet jeśli twarz jest przedstawiona z profilu czy przechylona pod kątem.

- metoda dopasowania szablonu twarzy (ang. template matching method):

  Metoda dopasowania szablonu twarzy wykorzystuje kilka standardowych wzorów opisujących twarz. Na wejściu algorytmu obraz jest porównywany z tymi wzorami, a na wyjściu dostajemy informację, w jakim stopniu obraz jest dopasowany do ogólnego szablonu twarzy.

- metoda bazująca na wyglądzie (ang. appearance - based method):

  Ideą metody bazującej na wyglądzie jest analiza dużego zbioru obrazów twarzy, tak aby wychwycić zmienność ich cech. Tak wytrenowany model jest później wykorzystywany do wykrywania twarzy.

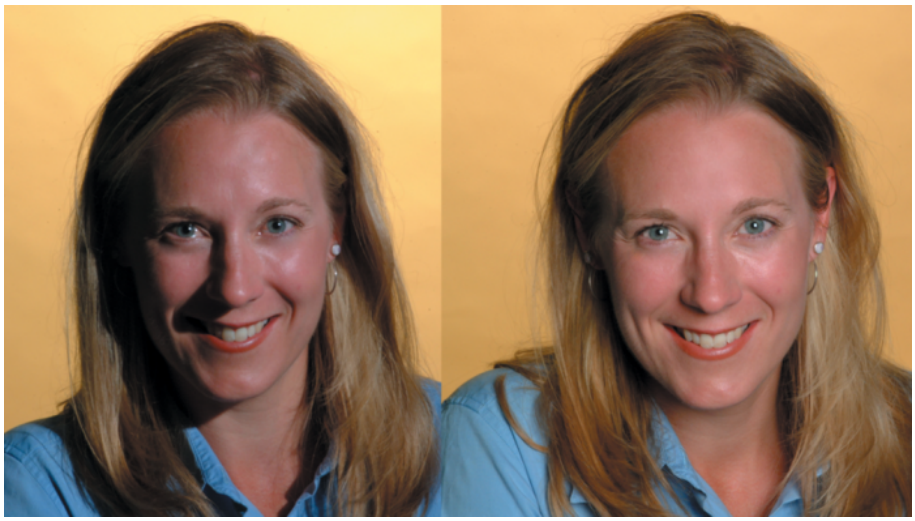
Należy jednak zauważyć, że w procesie ekstrakcji twarzy z obrazu mogą wystąpić liczne problemy. [27]. Jednym z nich jest nieodpowiednia poza. Wiąże się to z różnymi ustawieniami twarzy wobec aparatu fotograficznego lub kamery. Twarz może być nachylona, przechylona lub odchylona. Inaczej

mówiąc, może mieć różne położenie w trzech wymiarach, a niektóre części twarzy lub jej cechy mogą zostać przysłonięte. Im mniej cech widocznych na twarzy, tym mniej danych, które algorytm może z niej wyodrębnić, a im mniej danych, którymi algorytm operuje, tym mniejsze prawdopodobieństwo prawidłowego wykrycia twarzy. Niektóre twarze mogą zawierać pewne wyróżniające je cechy takie jak broda, blizny czy okulary. Różnorodność tych cech także wpływa na efektywność wykrywania twarzy.

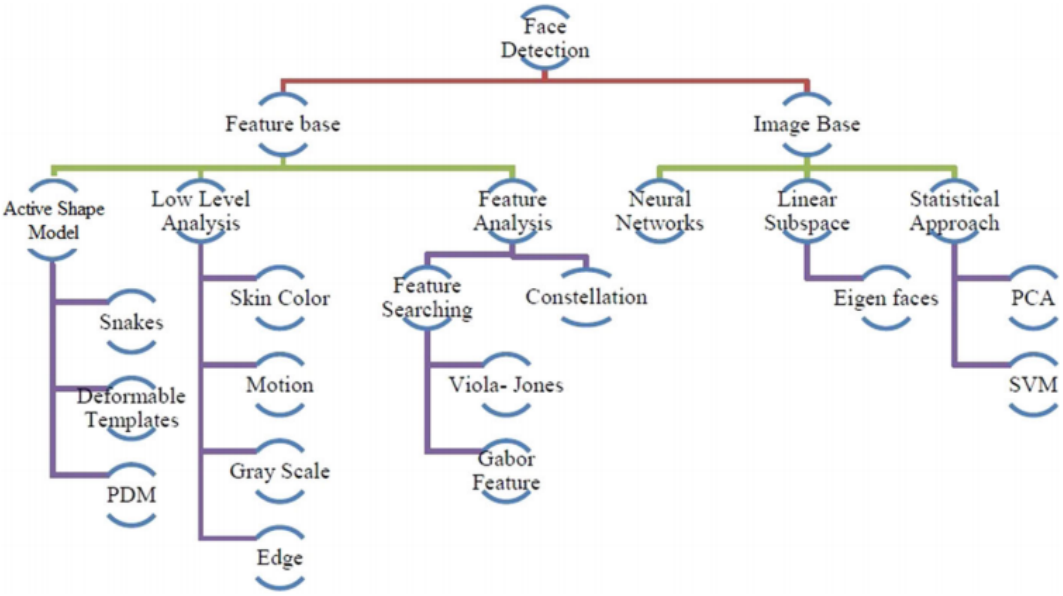
Zdarza się, że twarz zostaje częściowo przysłonięta przez jakiś inny obiekt. Na przykład na zdjęciu, które obejmuje grupę wielu osób, część danej twarzy może być przysłonięta przez inną twarz. Takie przysłonięcie przez inny obiekt wiąże się z utratą informacji o części twarzy, co zmniejsza prawdopodobieństwo prawidłowego jej wykrycia.

Ilość zmarszczek na twarzy jest zmienna, w zależności od wyrazu mimicznego. Przy różnych minach zmienia się kształt ust, a czasem w wyniku pracy mięśni uwydatniają się na twarzy ostre krawędzie. Widoczne mogą być różne pofałdowania skóry.

Kolejnym istotnym elementem jest oświetlenie twarzy. Gdy twarz oświetlona jest tzw. ostrym światłem, występują na niej cienie i światła określane jako ostre (obraz 2.3). W tym przypadku ryzyko utraty szczegółów oświe-



Rysunek 2.3: Przykład twarzy oświetlonej ostrym (twarz po lewej) oraz miękkim światłem [10].



Rysunek 2.4: Różne techniki wykrywania twarzy [3].

tlanej twarzy jest większe. Kiedy twarz jest skierowana na wprost słońca, z dużym prawdopodobieństwem można stwierdzić, że zostanie oświetlona ostrym światłem. Z kolei miękkie światło jest generowane na przykład przez zachmurzone niebo. Istotne jest także źródło światła, które może być punktowe lub rozproszone. Przy punktowym źródle światła cała twarz jest okryta jednolitym cieniem, którego intensywność zależy od rodzaju światła. Natomiast przy świetle rozproszonym intensywność cieni jest mniejsza. Na rys. 2.4 przedstawione są różne techniki wykrywania twarzy.

Jak widać metod wykrywania twarzy jest sporo, a szczegółowe omówienie każdej z nich nie byłoby merytorycznie uzasadnione. Poniżej zostaną przytoczone dwie metody wykrywania twarzy. Dodatkowo zostanie omówiona metoda, która posłużyła do wykrywania twarzy w niniejszej pracy.

W pracy [16] przedstawiono opisaną w skrócie poniżej technikę wykrywania twarzy. W pierwszym etapie procesu obszary, gdzie może znajdować się ludzkie oko, są wykrywane przez przeprowadzenie testów na zacienionych rejonach obrazu. Pary takich obszarów wyodrębnia się na podstawie algorytmu genetycznego, aby następnie wyznaczyć możliwy obszar twarzy. Dla

każdego obszaru mierzy się wartość dopasowania na podstawie jego projekcji na wektory własne, obliczone dla obszarów twarzy, tzw. eigenfaces. Aby wiarygodność wykrywania była wyższa, każdy możliwy obszar twarzy normalizuje się pod kątem oświetlenia. Proces ten powtarza się pewną ilość razy, a następnie do dalszej weryfikacji są wybierane możliwe obszary twarzy o wysokiej wartości dopasowania. Na tym etapie mierzy się symetrię twarzy oraz sprawdza się, czy na każdym wybranym obszarze istnieją rysy twarzy. Rysy określa się przez ewaluację rzeźby topograficznej - wystających i wklęsłych elementów różnych regionów obszaru twarzy, poddanego uprzednio normalizacji. Algorytm jest w stanie wykryć obszar twarzy także wtedy, gdy głowa jest przechylona.

W roku 1997 w pracy [6] opisano metodę wykrywania twarzy bazującą na wykrywaniu cechy, jaką jest kolor skóry. Zarówno dla człowieka, jak i dla maszyny kolor skóry jest najbardziej widoczną cechą twarzy. Ponadto kolor jest przetwarzany znacznie szybciej niż inne cechy. Przy dobrych warunkach oświetleniowych ustawienie twarzy nie ma wpływu na skuteczność wykrywalności twarzy opisywaną metodą.

Niezależnie od wybranej metody mogą wystąpić problemy z poprawnym wykryciem twarzy. Jednym z nich jest problem wykrywalności twarzy przy nierównomiernym oświetleniu. Problem pojawia się także, gdy na obrazie widoczny jest obszar skóry spoza twarzy, np. z rąk. Warto zaznaczyć, że kolor twarzy na obrazie jest zależny od względnego kierunku oświetlenia. Obszar twarzy w omawianym algorytmie jest wykrywany poprzez normalizacje histogramu kolorów. Normalizacja jest potrzebna do redukcji wpływu natężenia oświetlenia na kolor.

Algorytm Haar Cascade jest jedynym algorytmem do wykrywania twarzy zaimplementowanym w bibliotece OpenCV, dlatego został wykorzystany w procesie badawczym. Algorytm został bliżej przedstawiony w książce [24] w 2001 roku i składa się z trzech podstawowych faz, których dokładny opis znajduje sie w sekcji 2.1.2.

2.1.1 Przestrzenie barw oraz skala szarości

Każdy piksel w trybie kolorowym ma określoną reprezentację barwy z określonego modelu. Najczęściej są to trzy lub cztery wartości [13]. Do przestrzeni barw zaliczają się:

- CIEXYZ,
- CMYK,
- RGB.

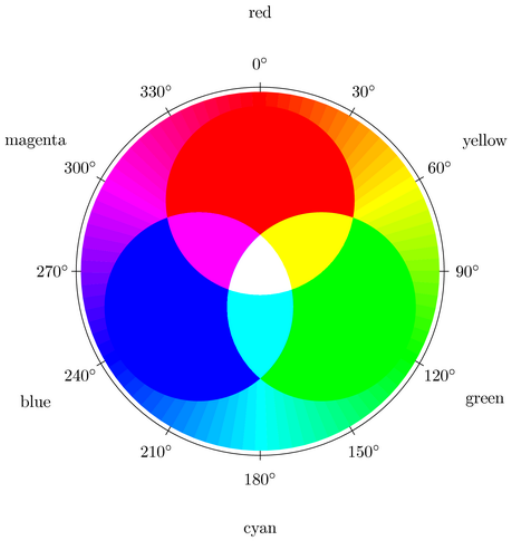
W dziedzinie rozpoznawania obrazów najczęściej barwy są reprezentowane przez przestrzeń barw RGB [13]. Przestrzeń kolorów RGB składa się z trzech kanałów [14]:

- R - czerwonego (z ang. Red),
- G - zielonego (z ang. Green),
- B - niebieskiego (z ang. Blue).

Barwy mieszane są poprzez syntezę addytywną. W przeciwieństwie do syntezy subtraktywnej, barwa wynikowa powstaje tu poprzez sumowanie wiązek światła widzialnego o różnych długościach [13]. Każdy piksel opisany za pomocą przestrzeni barw RGB ma trzy 8-bitowe wartości reprezentujące każdy kanał. Spotykane są 12- lub 16-bitowe reprezentacje kanałów, jednak 8-bitowa jest najpopularniejsza. Dla 8-bitowych kanałów wartość „0” danego kanału oznacza brak jasności, natomiast „255” maksymalną jasność. Poprzez mieszanie jasności tych trzech kanałów można uzyskać szerokie spektrum barw (rysunek 2.5).

Przykładowo kolor o reprezentacji R=153, G=217, B=234 przedstawiono na rysunku 2.6

Kolor (rysunek 2.6) może być też reprezentowany w kodzie szesnastkowym #99D9EA. Każda wartość heksadecymalna odpowiada kolejno kanałowi R, G, B.



Rysunek 2.5: Mieszanie kanałów RGB [21].

Rysunek 2.6: Kolor R=153 G=217 B=234.

Obraz może też być przedstawiony za pomocą odcieni jednej barwy. Taki obraz nazywa się obrazem monochromatycznym. Najczęściej stosowaną barwą w takich obrazach jest szarość [13].

Istnieją 3 metody konwersji obrazu z przestrzeni RGB na monochromatyczny [21]:

- największej jasności,
- średnia,
- luminancji.

Metoda największej jasności konwertuje na skalę szarości wg wzoru 2.1.

$$\frac{(max(R,G,B) + min(R,G,B))}{2} \tag{2.1}$$

Metoda średnia bazuje na wzorze 2.2, natomiast metodę luminancji ilustruje wzór 2.3.

$$\frac{(R+G+B)}{3} \tag{2.2}$$

$$0,21R + 0,72G + 0,07B \tag{2.3}$$

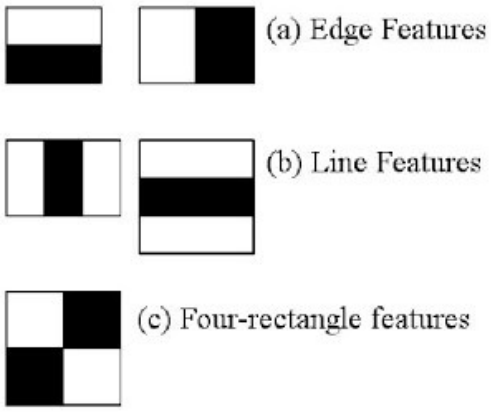
W pracy zastosowano konwersję z przestrzeni RGB na skale szarości wg wzoru 2.4 zaimplementowanego w bibliotece OpenCV, który jest modyfikacją metody luminancji.

$$0,299R + 0,587G + 0,114B \tag{2.4}$$

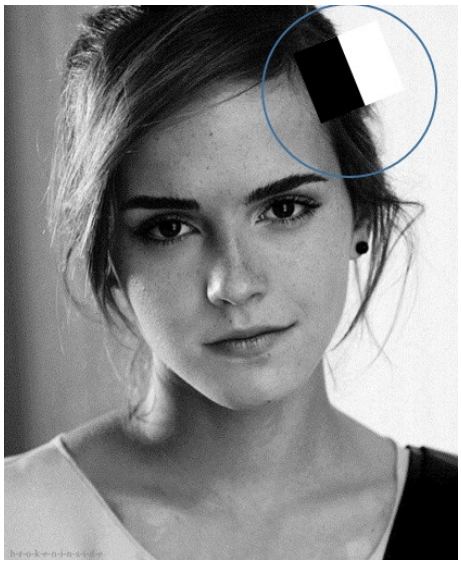
2.1.2 Metoda detekcji twarzy

Haar Cascade jest algorytmem służącym do wykrywania obiektów na obrazach. Został stworzony przez Paula Violę oraz Michaela Jonesa w 2001 roku [24]. Opiera się na zbudowaniu kaskadowej funkcji za pomocą analizy wielu zdjęć twarzy. Obrazy są dzielone na dwie kategorie - pozytywne oraz negatywne. Na zdjęciach klasyfikowanych jako pozytywne istnieje obiekt, który ma zostać wykryty, natomiast na zdjęciach negatywnych nie ma tego obiektu. Ekstrakcja cech w algorytmie Violi i Jonesa jest realizowana przez filtry Haara. Przed zastosowaniem filtru obraz musi zostać przekształcony do skali szarości. Filtry mają postać białych i czarnych prostokątów pogrupowanych w okna (rysunek 2.7). Wyznaczana jest suma jasności pikseli w obu rodzajach prostokątów, a następnie dla każdego okna obliczana jest różnica pomiędzy białymi a czarnymi. Na krawędzi istnieje różnica w jasności pikseli, dlatego opisywany algorytm ma zastosowanie w ich wykrywaniu. (rysunek 2.8).

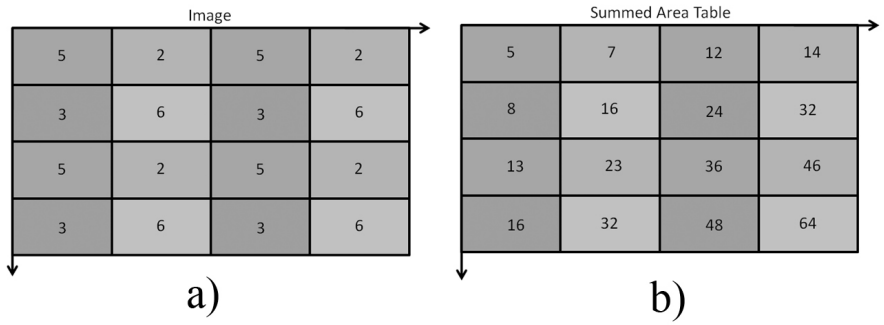
W celu poprawy efektywności sumowania pikseli stosowane są rozwiązania zwane w literaturze obrazem scałkowanym (ang. Integral image lub summed-area table) [30]. Jego ideą jest, aby każdy obraz został przekształcony w tabelę, w której każdy element x, y tej tabeli odpowiada sumie ja-



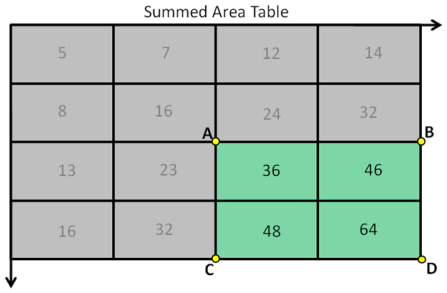
Rysunek 2.7: Filtr Haara a) krawędziowy b) liniowy c) szachownica [2]



Rysunek 2.8: Filtr Haara nałożony na krawędź twarzy [2]



Rysunek 2.9: Tabela jasności poszczególnych pikseli: a) przed zastosowaniem całkowania b) po zastosowaniu całkowania [4]



Rysunek 2.10: Sumowanie okna [4]

sności wszystkich pikseli według wzoru 2.5.

$$I(x,y) = \sum_{x' \leq x \cap y' \leq y} i(x',y') \tag{2.5}$$

gdzie I(x,y) jest wartością na pozycji x,y w tabeli(tabela obrazu scałkowanego), natomiast i(x,y) oznacza jasność piksela o współrzędnych x,y na obrazie. Na rysunku 2.9 przedstawiona jest tabela prezentująca jasność pikseli przed i po zastosowaniu całkowania obrazu.

Sumowanie przykładowego okna (rysunek 2.10) wymaga czterech operacji (wzór 2.6).

$$\sum_{x_0 \leq x \leq x_1 \cap y \leq y \leq y_1} i(x,y) = I(D) + I(A) - I(B) - I(C) \tag{2.6}$$

gdzie lewa część równania oznacza sumę jasności pikseli zaznaczonego okna tj. na rysunku 2.10, I(A) - wartość scałkowanego obrazu przy punkcie A (analogicznie I(B), I(C), I(D)) - (rysunek 2.10).

W związku z powyższym obliczenie wartości dla krawędziowego filtru Haara wymaga obliczenia różnicy dwóch sum, zatem do uzyskania wyniku konieczne jest wykonanie ośmiu operacji. Reprezentacja obrazu za pomocą obrazu scałkowanego znacznie zwiększa efektywność obliczania wartości w filtrze Haara.

Liczba cech wykrywanych w danym zdjęciu za pomocą filtru Haara jest znacznie większa od liczby pikseli na obrazie [24]. Dla obrazu o wymiarach 24x24 pikseli liczba cech wynosi ponad 180000. Autorzy algorytmu stwierdzili, że dla zwiększenia jego szybkości należy wybrać małą grupę istotnych cech, które razem mogą stworzyć jeden efektywny klasyfikator obiektu, zwany również silnym klasyfikatorem. Ten klasyfikator składa się z wielu słabych klasyfikatorów. Każdy z nich na podstawie jednej istotnej cechy dostarcza informacji, czy na obrazie znajduje się twarz, czy nie. Liczba tych istotnych cech została ustalona na 2500. W celu wyodrębnienia tych istotnych cech zastosowano algorytm Adaboost, który został opisany poniżej.

Zbiór n obrazów do trenowania można oznaczyć tak jak we wzorze 2.7:

$$(O_{x_1}, O_{y_1}), (O_{x_2}, O_{y_2}), \ldots, (O_{x_n}, O_{y_n}) \tag{2.7}$$

O_{x_1} oznacza obraz z bazy treningowej o indeksie 1, natomiast O_{y_1} określa, czy dany obraz przedstawia twarz ($O_{y_1} = 1$), czy nie ($O_{y_1} = 0$). Obraz przestawiający twarz nazywany jest obrazem pozytywnym, natomiast obraz negatywny nie przedstawia twarzy.

Następnym krokiem jest inicjalizacja wag (wzór 2.8).

$$w_{1,i} = \frac{1}{2m}, \frac{1}{2l} \tag{2.8}$$

Parametr $\frac{1}{2m}$ jest wagą dla zdjęć pozytywnych, a $\frac{1}{2l}$ oznacza wagę dla obrazów negatywnych. Parametry m, l oznaczają odpowiednio liczbę negatywnych oraz pozytywnych zdjęć.

Następnie inicjalizowany jest parametr t oznaczający istotną cechę, natomiast T oznacza liczbę wszystkich takich cech na obrazie. Dla każdej cechy t = 1, ..., T, przeprowadzany jest poniższy proces:

1. Normalizowane są wagi (wzór 2.9):

$$w_{t,i} = \frac{w_{t,i}}{\sum_{j=1}^{n} w_{t,j}} \tag{2.9}$$

w_t jest rozkładem prawdopodobieństwa

2. Dla każdej cechy j, trenowany jest klasyfikator h_j, który używa tylko jednej cechy wyliczonej z filtru Haara. Błąd jest liczony następująco (wzór 2.10):

$$w_t, \epsilon_j = \sum_i w_i h_j(x_i) - y_i \tag{2.10}$$

3. Wybierany jest klasyfikator h_t z najmniejszym błędem ϵ_t.

4. Następuje aktualizacja wag (wzór 2.11):

$$w_{t+1,i} = w_{t,i}\beta_t^{1-e_i} \tag{2.11}$$

gdzie $e_i = 0$. Jeśli x_i jest sklasyfikowane prawidłowo, wtedy $e_i = 1$, w innym wypadku $\beta_t = \frac{\epsilon_t}{1-\epsilon_t}$.

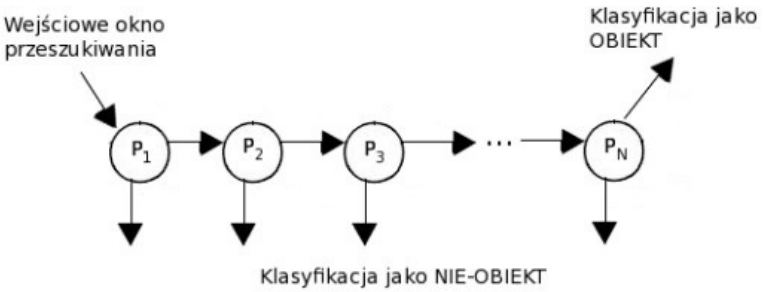
Silny klasyfikator $h(x)$ jest opisany równaniem:

$$h(x) = \begin{cases} 1 & \text{gdy } \sum_{t=1}^{T} \alpha_t h_t(x) >= \frac{1}{2} \sum_{t=1}^{T} \alpha_t \\ 0 & \text{w przeciwnym wypadku} \end{cases} \tag{2.12}$$

gdzie $\alpha_t = \lg \frac{1}{\beta_t}$

Algorytm Adaboost zmniejsza ilość cech Haara z ponad stu tysięcy do kilkuset - do tych najistotniejszych cech.

Ostatnim etapem jest wytworzenie kaskady klasyfikatorów. Zwiększa ona znacznie szybkość wykrywania pożądanego obiektu na obrazie. Ideą kaskady jest zgrupowanie klasyfikatorów, które powstały w poprzednim procesie -



Rysunek 2.11: Kaskada klasyfikatorów [31].

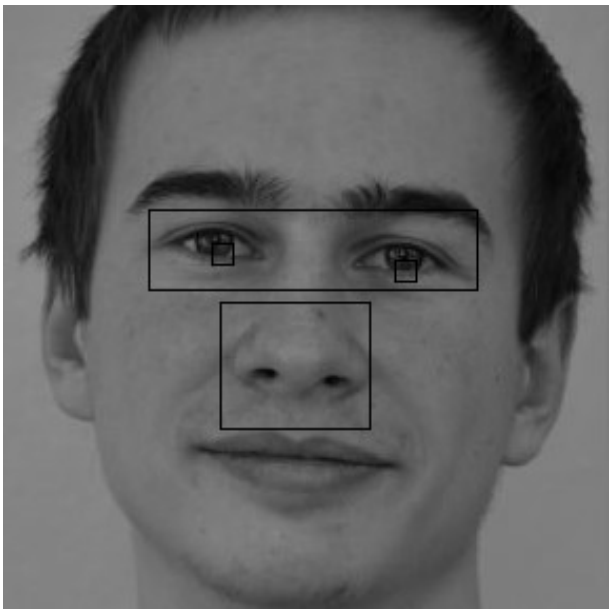
tzw. procesie boostingu. Klasyfikatory są grupowane w okna, które są połączone ze sobą tak jak na rysunku 2.11. Okna są oznaczone jako P1, P2, ..., Pn. Gdy dane okno wykryje obiekt, przechodzi do kolejnego okna w kaskadzie. W przeciwnym wypadku algorytm przerywa działanie i na danym obrazie nie zostaje zidentyfikowany obiekt. Okna są poustawiane tak, aby każde z nich klasyfikowało obiekt z różnym prawdopodobieństwem wykrycia oraz prawdopodobieństwem błędu.

Pierwsze okna posiadają klasyfikatory o słabszym TPR oraz FPR niż kolejne. Oznacza to, że prawdopodobieństwo TPR w oknie P_{x-1} jest mniejsze od tego w P_x. Natomiast prawdopodobieństwo FPR w oknie P_{x-1} jest większe od tego w P_x. Ostatnie okna mają największy współczynnik TPR oraz najmniejszy FRP ze wszystkich. Takie ustawienie okien ma na celu wstępne przepuszczenie przez okna obrazów, które z dużym prawdopodobieństwem zawierają szukany obiekt. Natomiast ostatnie okna w kaskadzie analizują niewielką część obrazu wejściowego.

Biblioteka OpenCV zawiera wytrenowane klasyfikatory, które zostały użyte w tej pracy magisterskiej. Wykorzystano je do wykrycia twarzy, ust oraz oczu. Modele stworzone w wyniku treningu klasyfikatora mają postać plików xml, które można znaleźć na oryginalnym repozytorium projektu OpenCv.

2.2 Wyznaczanie stref

Przed wyznaczeniem stref zmarszczkowych należy zidentyfikować na twarzy oczy oraz nos (rysunek 2.12). Należy podkreślić, że algorytm przerywa



Rysunek 2.12: Wykryty nos oraz oczy.

działanie jeśli nie zostanie wykryta twarz, oczy lub nos. Gdy zostanie wykryty obszar twarzy, oczu oraz nosa wyznaczone zostaje sześć stref zmarszczkowych (rysunek 2.13).

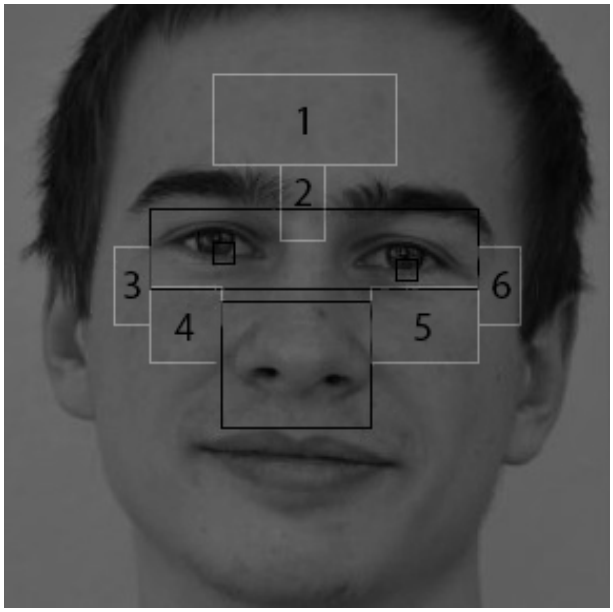
Strefy zmarszczkowe są umiejscowione na czole (Strefa „1"), w górnej części nosa (Strefa „2"), górnej części policzków (Strefa „4" i „5") oraz w okolicach powiek (Strefa „3" i „6"). To właśnie te miejsca zostały uznane przez autorów książki [26] za najbardziej znaczące w detekcji wieku.

Po detekcji oczu należy zmierzyć odległość pomiędzy środkiem lewego oka (x_l, y_l), a prawego (x_p, y_p) (wzór 2.13).

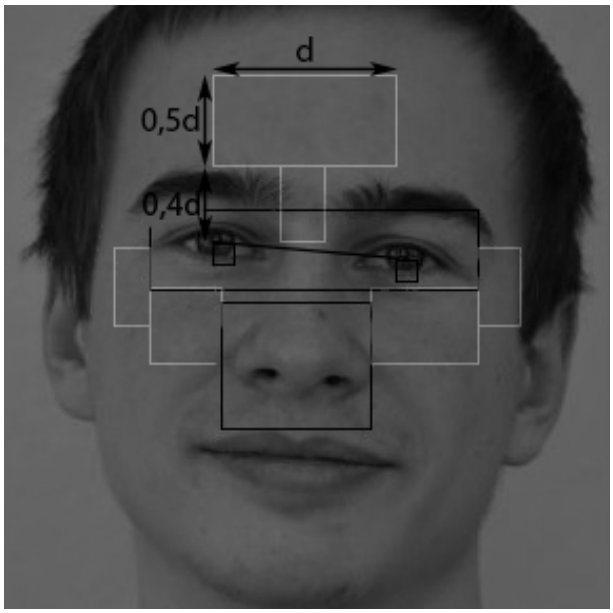
$$d = \sqrt{(x_r - x_l)^2 + (y_r - y_l)^2} \tag{2.13}$$

Odległość d służy do wyznaczanie strefy znajdującej się na czole (rysunek 2.14).

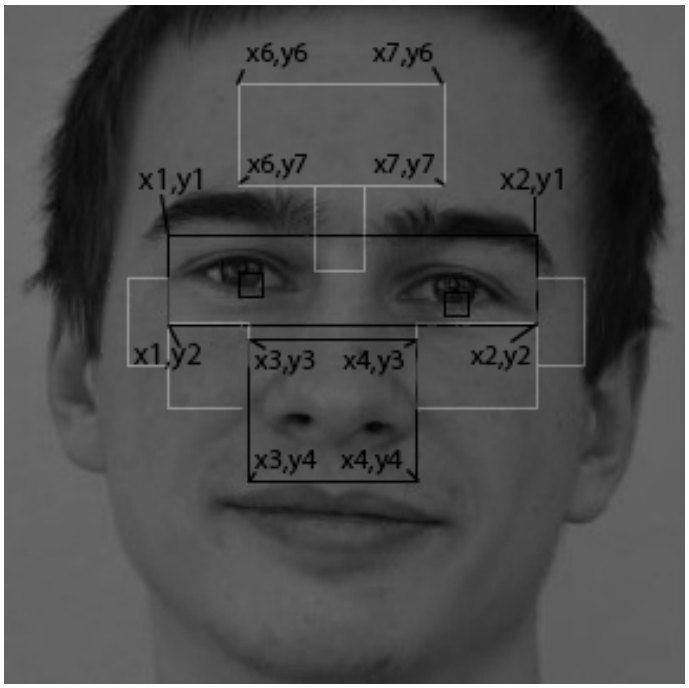
Autorzy [26] algorytmu założyli, że odległość od linii oczu do linii brwi wynosi $0,4 \cdot d$, natomiast wymiary „strefy czoła" wynoszą $d \times 0,5d$. Na rysunku 2.15 przedstawiono współrzędne, które są pomocne w wyznaczaniu



Rysunek 2.13: Strefy zmarszczkowe widoczne w białych prostokątach.



Rysunek 2.14: Wyznaczenie strefy znajdującej się na czole.



Rysunek 2.15: Pomocnicze współrzędne do wyliczania stref zmarszczkowych.

stref zmarszczkowych.

Strefa 2 została wyznaczona dzięki znajomości położenia prawego oka, dystansu między oczami oraz strefy „1". Każda strefa może być wyznaczona przez jeden punkt (rysunek 2.16) oraz jej wymiary.

Współrzędna x punktu A została wyznaczona ze wzoru 2.14

$$x_A = x_6 + 0,375 \times d \tag{2.14}$$

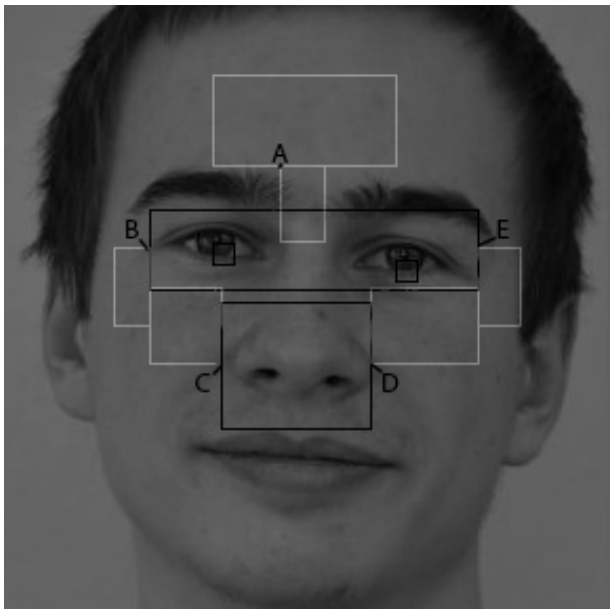
gdzie x_6 to współrzędna z rysunku 2.15. Natomiast d jest odległością pomiędzy oczami (wzór 2.13).

Współrzędną y punktu A opisuje wzór 2.15:

$$y_A = y_7 \tag{2.15}$$

Współrzędna y_7 analogicznie jak x_6 znajduje się na rysunku 2.15.

Zakładając, że prawe oko ma współrzędne x_o oraz y_o wysokość strefy 2



Rysunek 2.16: Punkty wyznaczające położenie stref.

wyznaczana jest w poniższy sposób (wzór 2.16).

$$h_A = y_o - y_7 \tag{2.16}$$

Natomiast szerokość strefy 2 oblicza się następująco: (wzór 2.17):

$$w_A = 0,25 \cdot d \tag{2.17}$$

Wartość 0,25 ze wzoru 2.17 została dobrana empirycznie.

Współrzędne punktu C (strefa 4) zostają wyznaczone ze wzoru 2.18 oraz 2.19.

$$x_C = x_3 \tag{2.18}$$

$$y_C = \frac{y_4 - y_3}{2} \tag{2.19}$$

Wysokość oraz szerokość strefy 4 (wzór 2.20 oraz 2.21)

$$h_C = y_C - y_3 \tag{2.20}$$

$$w_C = x_3 - x_1 \tag{2.21}$$

Strefa 5 zawiera punkt D - wzór 2.22 oraz 2.23.

$$x_D = x_4 \tag{2.22}$$

$$y_D = y_C \tag{2.23}$$

Dla strefy 5 wyznacza się wysokość (wzór 2.24) oraz szerokość na podstawie wzoru (Wzór 2.25).

$$h_D = h_C \tag{2.24}$$

$$w_D = x_2 - x_4 \tag{2.25}$$

Strefa 3 zawiera punkt B wyznaczany ze wzorów 2.26 i 2.27

$$x_B = x_1 \tag{2.26}$$

$$y_B = \frac{y_2 - y_1}{2} \tag{2.27}$$

Wysokość (wzór 2.28) oraz szerokość (wzór 2.29) strefy 3.

$$h_B = y_C - y_B \tag{2.28}$$

$$w_B = w_C * 0,4 \tag{2.29}$$

Na samym końcu zostaje wyznaczona strefa 6. Wyznaczona zostaje współrzędna E - wzór 2.30 i 2.31.

$$x_E = x_2 \tag{2.30}$$

$$y_E = y_B \tag{2.31}$$

Szerokość oraz wysokość strefy 6.

$$h_E = h_B \tag{2.32}$$

$$w_E = w_D \cdot 0,4 \tag{2.33}$$

Należy zauważyć, że szerokość stref 3 i 6 jest pomnożona przez 0,4 szerokości odpowiednio stref 4 i 5. Wartość „0,4” została dobrana empirycznie.

2.3 Wykrywanie zmarszczek - detektor Canny

Następnym krokiem algorytmu po wyznaczeniu stref zmarszczkowych jest wyodrębnienie zmarszczek poprzez detekcję krawędzi. Istnieje wiele metod do tego celu, jednak detektor Canny jest jedną z najbardziej dokładnych i niezawodnych, dlatego został zastosowany w toku badań. Metoda ta została opracowana przez Johna F. Canny w 1986 roku. Oprócz samej implementacji algorytmu jego twórca zaprezentował teorię obliczeniową, która wyjaśnia działanie tej metody. Ponadto Canny zauważył, że wymagania dotyczące implementacji detekcji krawędzi są podobne do siebie w wielu systemach wizyjnych. W związku z powyższym jego algorytm może zostać skutecznie zastosowany w różnych systemach tego typu. Poniżej przedstawiono najważniejsze zasady, które są niezbędne do dobrej detekcji krawędzi w opisywanym

algorytmie:

- Detekcja krawędzi z niskim prawdopodobieństwem błędu: Oznacza to, że algorytm powinien wykryć jak najwięcej krawędzi, które rzeczywiście istnieją. Natomiast ilość krawędzi wykrytych błędnie powinna być jak najmniejsza.
- Precyzja detekcji: Algorytm powinien precyzyjnie zlokalizować krawędź dokładnie w jej środku.
- Brak redundantnych detekcji: Krawędź powinna być zlokalizowana jednokrotnie, a szum na obrazie nie powinien generować dodatkowych krawędzi.

W algorytmie detekcji Canny używa rachunku wariacyjnego. Rachunek wariacyjny to dziedzina analizy matematycznej, która analizuje przestrzenie funkcyjne i znajduje w nich ekstrema funkcjonałów. Funkcjonały natomiast przekształcają przestrzeń wektorową na liczby rzeczywiste. Zatem rachunek wariacyjny ma za zadanie pomóc w znalezieniu charakterystycznej funkcji, dla której funkcjonał przyjmuje wartość ekstremalną. Algorytm detektora jest podzielony na kilka kroków [18]:

- Redukcja szumów na obrazie filtrem Gaussa.
- Szukanie gradientów jasności.
- Zastosowanie techniki zmniejszania grubości krawędzi.
- Likwidacja krawędzi o małym gradiencie jasności.
- Filtracja poprzez histerezę.

Redukcja szumów na obrazie filtrem Gaussa

Szum na obrazie to piksele o losowym kolorze, jasności oraz umiejscowieniu (współrzędnych na obrazie). Takie piksele są nadmiarowymi informacjami o obrazie oraz stanowią efekt uboczny przetwarzania obrazu przez matrycę cyfrową (rysunek 2.17).

Rysunek 2.17: a) Obraz z nieznaczną ilością szumów b) Obraz ze znaczną ilością szumów [29].

Efekt szumu jest spotykany również w fotografii analogowej. W fotografii cyfrowej szum wzrasta na skutek zwiększania czułości matrycy lub przez wzrost jej temperatury. Redukcję szumów można uzyskać stosując filtr Gaussa. Splot filtru Gaussa z obrazem daje w wyniku wygładzony obraz ze zmniejszoną ilością szumów. Filtr Gaussa może mieć różne wymiary. Rozmiar filtra ma wpływ na wydajność wygładzania obrazu, a co za tym idzie na czułość wykrywania szumu [18].

Szukanie gradientów jasności

Istotnym parametrem w wykrywaniu krawędzi jest gradient jasności. Określa on jak bardzo zmienia się jasność danego piksela względem sąsiadujących pikseli. Gradient jasności otrzymany zostaje ze wzoru 2.34:

$$G = \sqrt{G_x^2 + G_y^2} \tag{2.34}$$

gdzie G_x jest zmianą jasności danego piksela w kierunku poziomym, a G_y w kierunku pionowym. Kierunek gradientu opisuje wzór 2.35

$$\Theta = atan2(G_y, G_x) \tag{2.35}$$

Krawędzie na obrazie charakteryzują się pewną zmianą jasności na danym obszarze obrazu. Krawędź na obrazie może być położona pod różnym kątem. Opisywany detektor używa czterech filtrów w celu wykrycia pionowych, poziomych oraz ukośnych krawędzi.

Kierunek gradientu jest zaokrąglony do jednego z filtrów kierunku: poziomego ($\Theta = 0°$), pionowego ($\Theta = 90°$) lub ukośnego ($\Theta = 45°$ lub $\Theta = 135°$ stopni). Każdy piksel obrazu otrzymuje wartość gradientu (wzór 2.34) oraz kierunek - zgodny z powyższym filtrem kierunku. W wyniku tego działania otrzymuje się obraz gradientowy [18].

Zastosowanie techniki zmniejszania grubości krawędzi

Zmniejszanie grubości krawędzi służy do filtracji krawędzi powstałych po poprzednim kroku. Krawędzie wykryte za pomocą gradientu są rozmyte. Dzięki technice zmniejszania grubości można odszukać na małym obszarze gradienty o największej wartości , które identyfikują najostrzejsze krawędzie. Wszystkie inne, o mniejszej wartości, są ignorowane. Algorytm operuje na obrazie gradientowym powstałym jako wynik działania algorytmu opisanego w sekcji 2.3 Działa on następująco:

Odczytana zostaje wartość gradientu oraz jego kierunek dla danego piksela. Następnie wartość gradientu porównuje się z wartościami gradientu dla dwóch sąsiednich pikseli. Sąsiednie piksele leżą na jednej linii wyznaczonej przez kierunek gradientu. Jeśli wartość danego piksela jest największa spośród pikseli wzdłuż wyżej wymienionych linii, to dany piksel należy do najostrzejszej krawędzi [18].

Filtracja krawędzi o małym gradiencie jasności

Zastosowanie techniki zmniejszania grubości krawędzi pozostawia na obrazie wiele krawędzi, które są wygenerowane przez szum i zmiany kolorów. Kolejnym krokiem jest filtracja powyższych krawędzi, gdyż są one nadmiarowe. W tym celu ustawiany jest pewien próg, zwany progiem małym T_l. Oprócz progu małego ustawiany jest również próg duży T_h w celu wyodrębnienia krawędzi o wysokim gradiencie jasności. Jeśli wartość gradientu dla danej krawędzi jest mniejsza od progu małego, to zostaje ona usunięta. W przypadku, gdy wartość jest większa od progu małego, ale mniejsza od progu dużego, dana krawędź zostaje oznaczona jako słaba. Krawędź oznacza się jako mocną, gdy wartość gradientu dla niej jest większa od wartości progu dużego [18].

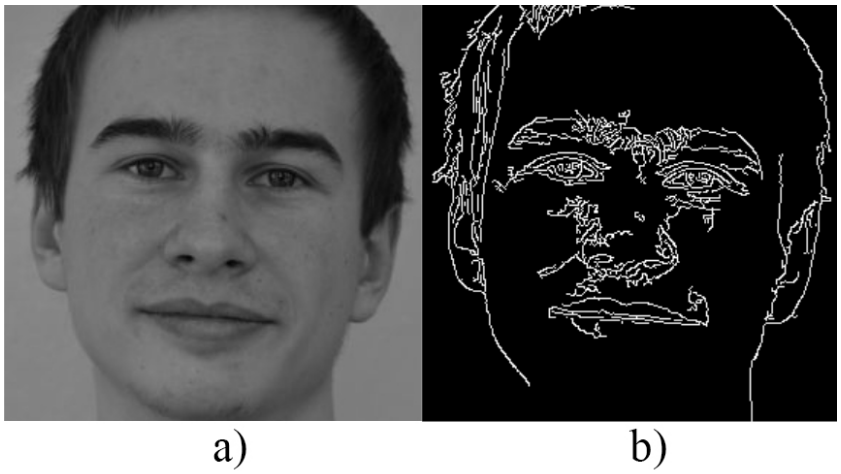
Filtracja poprzez histerezę

W wyniku działania algorytmu do tej pory zostały uzyskane słabe oraz mocne krawędzie. Mocne krawędzie zostaną oznaczone jako prawdziwe krawędzie znajdujące się na obrazie. Słabe krawędzie mogą być częścią mocnych krawędzi, ale mogą również zostać wygenerowane przez szum lub zmiany kolorów. Krawędzie tego typu nie są prawdziwymi krawędziami w obrazie, więc w ostatnim kroku algorytmu powinny zostać przefiltrowane. Słabe krawędzie, które są powiązane z mocnymi znajdują się w najbliższym sąsiedztwie mocnych krawędzi. W celu odnalezienia tych powiązań pomiędzy krawędziami zostaje zastosowana analiza spójności krawędzi. Jeśli zostaje zidentyfikowane powiązanie pomiędzy mocną, a słabą krawędzią, to słabą krawędź pozostawia się w obrazie. W przypadku, gdy słaba krawędź nie jest powiązana z żadną mocną krawędzią - zostaje usunięta [18].

2.4 Wyliczanie współczynnika zmarszczek

W wyniku działania detektora Cannego, który został opisany w sekcji 2.3 wygenerowany zostaje obraz binarny z wyodrębnionymi krawędziami. Obraz binarny zawiera piksele o dwóch wartościach jasności pikseli. Piksel może

być albo biały albo czarny. Krawędzie można rozpoznać jako białe piksele, natomiast czarne piksele oznaczają brak krawędzi (rysunek 2.18).



Rysunek 2.18: a) Oryginalny obraz b) Obraz z wykrytymi krawędziami.

Na rysunku 2.18 widoczne są krawędzie, które identyfikują owal twarzy. Ponadto widoczne są poszczególne części twarzy tj. nos, oczy, brwi. Można także zauważyć dodatkowe krawędzie w strefach zmarszczkowych (sekcja 2.2), które identyfikują zmarszczki. W metodzie bazowej ilość białych pikseli w strefach zmarszczkowych jest wprost proporcjonalna do ilości zmarszczek na twarzy danej osoby.

Z każdej strefy zmarszczkowej obliczany jest stosunek ilości białych pikseli do wszystkich pikseli (wzór 2.36).

$$W_{s1} = \frac{PB_{s1}}{PW_{s1}} \tag{2.36}$$

gdzie W_{s1} jest stosunkiem białych pikseli do wszystkich w strefie 1, PB_{s1} - suma białych pikseli w strefie 1, PW_{s1} - suma wszystkich pikseli w strefie 1. Analogicznie obliczane są stosunki pikseli dla pozostałych stref. Ostatnim etapem jest sumowanie wszystkich stosunków pikseli (wzór 2.37).

$$WZ = W_{s1} + W_{s2} + W_{s3} + W_{s4} + W_{s5} + W_{s6} \tag{2.37}$$

gdzie WZ to wskaźnik zmarszczek - parametr określający ilość zmarszczek dla danej osoby.

2.5 Algorytm analizy danych ze zbioru trenującego

W celu skonstruowania programu wyznaczającego wiek na podstawie tekstury należy wcześniej zbadać zależność ilości zmarszczek od wieku. W tym celu należy posiadać odpowiednią bazę zdjęć, na których można przeprowadzić wyżej wspomniane badania.

Jest wiele darmowych źródeł obrazów twarzy. Najczęściej używane bazy to FG-NET oraz MORTH II, które zawierają dziesiątki tysięcy zdjęć. Zdjęcia są różnej jakości i nie każda baza zdjęć nadaje się do konkretnych badań [15]. Przy realizacji tej pracy wykorzystana została baza UTKFace, ponieważ wszystkie zgrupowane w niej obrazy przedstawiają twarze w pozycji frontalnej. Dodatkowo ścieżka pliku każdego obrazu zawiera informacje na temat rzeczywistego wieku, co pozwala w łatwy sposób zweryfikować poprawność algorytmu. W związku z powyższym z każdego zdjęcia z tej bazy otrzymywano dwie informacje - rzeczywisty wiek danej osoby oraz współczynnik zmarszczek. Następnie dla całej bazy wygenerowano zbiór danych, który w dalszej kolejności był odpowiednio analizowany w celu sprawdzenia zależności pomiędzy współczynnikiem zmarszczek, a wiekiem. Autorzy algorytmu w celu analizy uprzednio wygenerowanego zbioru danych wykorzystali algorytm grupowania - Fuzzy C-Means. W wyniku działania tego algorytmu otrzymuje się informacje, dzięki którym na podstawie współczynnika zmarszczek można oszacować wiek [26].

2.6 Grupowanie danych - Fuzzy C-Means oraz wyznaczanie wieku

Grupowanie danych, zwane również klasteryzacją, jest szeroko stosowane w uczeniu maszynowym, rozpoznawaniu wzorców, analizie obrazu, bioinformatyce, kompresji danych czy w grafice komputerowej. Polega na podzieleniu dużego zbioru danych na grupy. Powyższe grupy zawierają dane, które są podobne do siebie [9].

Klasteryzacja jest zadaniem, które może zostać wykonane na wiele sposobów. Jest również iteracyjnym procesem odkrywania danych w celu odnalezieniu relacji pomiędzy nimi.

Odpowiedni algorytm grupowania i ustawienia parametrów (w tym parametrów takich jak funkcja odległości lub liczba oczekiwanych grup) zależą od zestawu danych i sposobu wykorzystania wyników. Dane nieraz muszą zostać przefiltrowane lub potrzebna jest zmiana parametrów grupowania w celu osiągnięcia zamierzonego efektu [9].

Najczęściej grupa jest definiowana przez jak najmniejszą odległość pomiędzy jej członkami. Zarazem odległość pomiędzy członkami grupy jest głównym parametrem klasteryzacji hierarchicznej. Istnieje również wiele innych modeli grupowania. Jednym z nich jest model centroidowy, który zakłada, że każda grupa jest zdefiniowana przez jeden wektor, posiadający wartość średnią. Istnieją też modele, które opierają się na sieciach neuronowych. Ich podstawą jest założenie, że dane grupują się za pomocą nienadzorowanych sieci neuronowych. Spotykany jest także model oparty o rozkłady statystyczne.

Klasteryzacja może być twarda lub miękka. Twarda klasteryzacja oznacza, że każdy element danych może należeć tylko do jednej grupy. Natomiast w przypadku miękkiej klasteryzacji każdy element danych może w pewnym stopniu należeć do każdej z grup. Autorzy algorytmu szacowania wieku będącego podstawą przeprowadzonych badań zastosowali grupowanie metodą Fuzzy C-means. Należy ona do grupy modeli centroidowych [9]. W Fuzzy C-means zastosowano klasteryzacje miękką.

Algorytm Fuzzy C-means został stworzony w 1973 przez J. C. Dunn. Po raz pierwszy został opisany w książce [**?**].

Klasteryzacja Fuzzy C-means dzieli zbiór S na podzbiory s_i zgodnie ze wzorem 2.38):

$$S := \{s_1, s_2, s_3, \ldots, s_N\} \subset \mathbb{R} \tag{2.38}$$

W parametrze algorytmu zostaje ustalona liczba grup N. Każda grupa posiada charakterystyczną wartość nazywaną centroidem $p_j \subset \mathbb{R}$. Jak zostało wspomniane na początku tego rozdziału, w algorytmie C-means zastosowana jest miękka klasteryzacja. Każdy element danych x_i ma przypisywany wektor U_i przynależności do grupy (wzór 2.39).

$$U_i = (u_{i1}, u_{i2}, u_{i3}, \ldots, u_{ic}) \tag{2.39}$$

gdzie u_{i1} oznacza stopień przynależności elementu x_i do grupy 1. Analogicznie u_{i2} oznacza stopień przynależności elementu x_i do grupy 2.

W pierwszym kroku zbiór centroidów P jest inicjalizowany losowo (wzór 2.40).

$$P^0 = (p_1^0, p_2^0, p_3^0, \ldots, p_c^0) \tag{2.40}$$

W k-tym kroku algorytmu wyliczona jest macierz funkcji przynależności $U^k = u_{ij}^k$ (wzór 2.41).

$$u_{ij}^k = \frac{1}{\sum_{k=1}^{c} \left(\frac{\left\| x_i - p_j^k \right\|}{\left\| x_i - p_k^k \right\|} \right)^{\frac{2}{m-1}}} \tag{2.41}$$

Wyrażenie $\|x_i - p_j\|$ oznacza odległość Euklidesową pomiędzy elementem x_i a centroidem p_j. Parametr „m” jest nazywany współczynnikiem rozmycia. Współczynnik ten może być w zakresie od 1 do nieskończoności, jednak w większości przypadków jest równy 2.

W kroku $(k+1)^{th}$ centroid $p_j^{(k+1)}$ jest uaktualniany według wzoru 2.42.

$$p_j^{(k+1)} = \frac{\sum_{i=1}^{n} u_{ij}^{(k+1)m} x_j}{\sum_{i=1}^{n} u_{ij}^{(k+1)m}} \tag{2.42}$$

Algorytm w kolejnych iteracjach minimalizuje funkcję przynależności $J(U, P)$ podaną wzorem 2.43:

$$J(U, P) = \sum_{i=1}^{n} \sum_{j=1}^{c} u_{ij}^{m} \left\| x_i - p_j \right\|^2 \tag{2.43}$$

Iteracje kończą się, gdy zostanie osiągnięty warunek opisany wzorem 2.44

$$\left\| J^{(k+1)}(U, P) - J^{(k)}(U, P)) \right\| < \epsilon \tag{2.44}$$

gdzie $\epsilon > 0$. Parametr ϵ jest parametrem ustawianym przed uruchomieniem klasteryzacji. W niektórych implementacjach istnieje możliwość zakończenia działania algorytmu po określonej liczbę iteracji, jeśli wcześniej nie zostanie osiągnięte kryterium ze wzoru 2.44.

Po zakończeniu działania wyżej opisanego algorytmu zostaje wyliczony wektor przynależności U (wzór 2.39). Wektor ten opisuje stopień przynależności elementu f_i do poszczególnych klastrów. Suma wektora przynależności U_i, dla elementu x_i jest opisana wzorem 2.45

$$\sum_{j=0}^{j=c} u_{ij} = 1 \tag{2.45}$$

gdzie c oznacza liczbę klastrów. Element f_i będzie należał do klastra k, jeśli u_{ik} ma największą wartość w wektorze U_i.

W tym akapicie zostanie przedstawiony algorytm trenowania za pomocą Fuzzy C-means. Z każdego zdjęcia „i" otrzymywane są dane dotyczące rzeczywistego wieku A_i oraz parametr wskaźnika zmarszczek WF_i. W algorytmie Fuzzy C-means są grupowane zbiory danych zawierające wartości wskaź-

nika zmarszczek WF_i. Jak wiadomo z powyższego opisu algorytmu Fuzzy C-means, każdy element (w tym przypadku WF_i) ma określony stopień przynależności do każdego klastra. Dany element WF_i będzie należał do klastra o największym stopniu przynależności. Każde zdjęcie jest grupowane według wyżej opisanego algorytmu FCM.

Każdy klaster ma określony centroid j, który posiada parametr P_{wfj} oraz dodatkowy parametr - średnią wartość wieku P_{aj}. P_{wfj} oznacza wartość wskaźnika zmarszczek dla danego centroida. P_{aj} jest wyznaczany ze wzoru 2.46.

$$P_{aj} = \frac{\sum A_i}{N_j} \tag{2.46}$$

gdzie A_i to rzeczywisty wiek osoby z obrazu i, który należy do klastru j. Natomiast N_j to liczba zdjęć zaklasyfikowanych do klastru j.

W powyższym akapicie został opisany algorytm klasteryzacji, który generuje N centroidów, reprezentujących każdą grupę. Natomiast osobnym etapem jest szacowanie wieku. W pierwszej fazie działanie algorytmu jest dokładnie takie same jak w sekcjach od 2.1 do 2.4: Algorytm szacowania wieku w pierwszej kolejności dokonuje detekcji twarzy, oczu, ust na danym obrazie O_i. Jeśli powyższe elementy zostały zidentyfikowane, wyznaczane są strefy zmarszczkowe, a następnie zostaje wyliczony parametr wskaźnik zmarszczek WF_i [26].

Dla parametru WF_i zostaje wyznaczony wektor przynależności do grupy według wzoru 2.39. Przy założeniu dla wektoru przynależności zgodnym ze wzorem 2.45 szacowany wiek PA wynosi (wzór 2.47).

$$PA = \sum_{j=1}^{N} (u_{ij} \cdot P_{aj}) \tag{2.47}$$

gdzie u_{ij} oznacza przynależność i-tego zdjęcia do j-tego klastra, N oznacza liczbę klastrów, podaną jako parametr wejściowy algorytmu FCM, natomiast P_{aj} jest średnim wiekiem dla j-tego centroida.

Rozdział 3

Modyfikacje metody bazowej

W sekcji 3.1 zostanie przedstawiona pierwsza modyfikacja algorytmu bazowego. Kolejna sekcja 3.2 zawiera opis kolejnej modyfikacji algorytmu, w której użyto algorytm HOG-a. W ostatniej sekcji 3.3 została zaprezentowana ostatnia modyfikacja, która oprócz algorytmu HOG do szacowania wieku wykorzystuje algorytm KNN.

3.1 Odjęcie wybranej strefy

Pierwsza modyfikacja polegała na odjęciu jednej wybranej strefy. Detektor Canny oprócz zmarszczek wykrywa także włosy. Podczas przeglądania wielu zdjęć zauważono, że w obrębie strefy zmarszczek nr 2 (obraz 2.13) wykrywane są części brwi oprócz samych zmarszczek. Generuje to błąd, dlatego wyżej wymienionej strefy nie uwzględniono w obliczeniach. W sekcji 3.1 porównano różnice w algorytmie przed opisaną wyżej modyfikacją i po niej.

Zmiana algorytmu względem metody bazowej

Przed modyfikacją metody bazowej parametr wskaźnika zmarszczek był obliczany według wzoru 2.37, który jest umieszczony w sekcji 2.4.

Po modyfikacji wskaźnik zmarszczek oblicza się według wzoru 3.1.

$$WZ = W_{s1} + W_{s3} + W_{s4} + W_{s5} + W_{s6} \tag{3.1}$$

Kolejną modyfikacją było zastosowanie metody HOG (ang. Histogram of Oriented Gradients), która została opisana w sekcji 3.2.

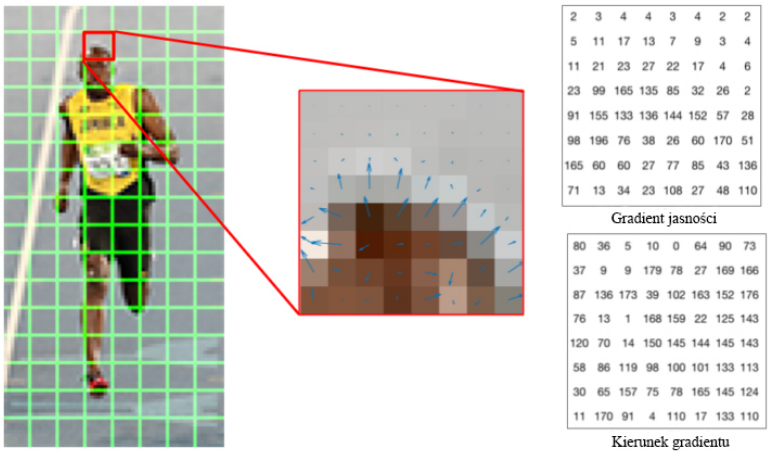
3.2 Opis metody HOG

Metoda HOG jest deskryptorem cech z obrazu. Deskryptor cech wyodrębnia z obrazu pewne cechy, do których należą między innymi kolory czy tekstury. W metodzie bazowej rolę deskryptora tego rodzaju spełnia detektor krawędzi Canny (sekcja 2.3), który wykrywa krawędzie, identyfikowane jako zmarszczki. Z kolei Histogram of Oriented Gradients jest techniką, która liczy wystąpienia kierunków gradientów jasności. Metoda HOG została zaimplementowana w 2005 roku przez Navneeta Dalala i Billa Triggsa [7]. Została wtedy zastosowana do detekcji pieszych na zdjęciach. Później jej zastosowanie rozszerzono o wykrywanie ludzi w filmach wideo, ponadto stosowano ją także do wykrywania zwierząt oraz pojazdów na zdjęciach.

Metoda Histogram of Oriented Gradients opiera się na założeniu, że wygląd i kształt obiektów na obrazie może zostać opisany przez gradient jasności oraz jego kierunek. Jest to dobrze widoczne na rysunku 3.1

Na granicy pomiędzy tłem, a głową sportowca widoczne są duże wartości gradientu jasności. W ten sposób zostaje zidentyfikowana krawędź głowy.

Działanie algorytmu można pokrótce opisać w następujący sposób. Obraz jest dzielony na małe obszary zwane komórkami, które zwykle mają wymiar kilka na kilka pikseli. Komórki są grupowane w bloki, mogące składać się z różnej ilości komórek. Dla wszystkich pikseli w komórce tworzony jest histogram kierunków gradientu. Następnie histogramy są łączone w jeden wspólny deskryptor. Dla zwiększenia dokładności lokalne histogramy normalizowane są pod względem kontrastu. Realizowane jest to przez pomiar jasności w bloku. Normalizacja zapewnia zmniejszenia niepożądanego efektu generowanego przez różnice w oświetleniu różnych obszarów zdjęcia. Wspomniana sy-



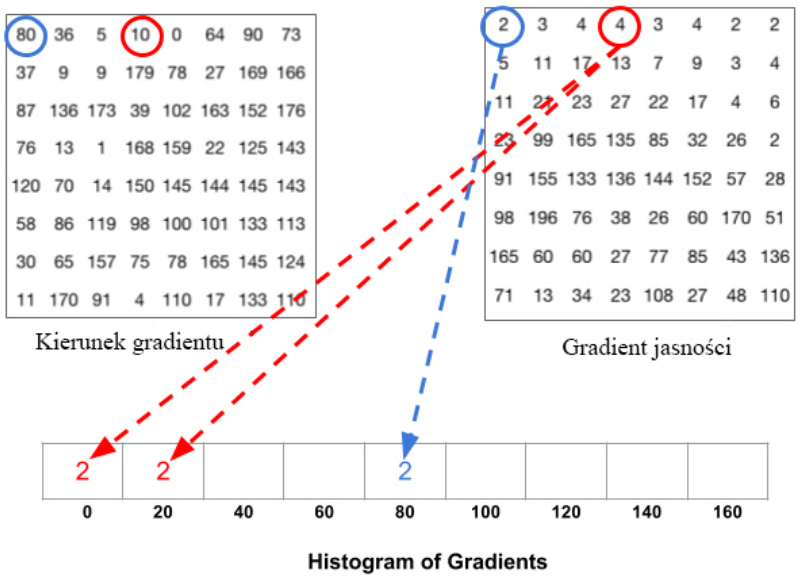
Rysunek 3.1: Gradient jasności identyfikujący granicę pomiędzy głową a tłem [19].

tuacja występuje na przykład, gdy zdjęcie wykonywane jest w nieprawidłowych warunkach oświetleniowych (rysunek 2.3).

W pierwszej fazie algorytmu HOG obraz powinien zostać przeskalowany, tak aby jego wymiary były podzielne przez rozmiar pojedynczej komórki. (rysunek 3.2). Warto wspomnieć, że w wielu metodach generujących deskryptory pierwszą fazą jest korekcja luminancji. Jednak ten krok może zostać pominięty, gdyż w późniejszym kroku stosowana jest normalizacja, która dokonuje korekcji luminancji cieni oraz świateł. W kolejnej fazie muszą zostać wyznaczone gradienty jasności. Wyznaczenie gradientów jasności zostało szczegółowo opisane w sekcji 2.3. W kolejnej fazie zostają wyznaczone histogramy gradientów dla każdej komórki. Na rysunku 3.1 został przedstawiony wynik działania tego kroku. Zostają wygenerowane dwie macierze. Pierwsza zawiera informację o gradiencie jasności każdego piksela komórki, natomiast druga - informację o kierunku każdego gradientu. Wartości kierunku gradientu wyrażone są w stopniach. W metodzie Histogram of Oriented Gradients zastosowano kierunki gradientu bez znaku. Oznacza to, że zakres kierunków mieści się w granicach od 0° do 180°. Użycie kierunku gradientu bez znaku jest uzasadnione tym, że wartość kierunku o takiej samej wartości, lecz różnych znakach reprezentuje gradient o tym samym kierunku, lecz różnych zwrotach. Dla detekcji kształtów krawędzi czy kształtów istotny jest kierunek, a nie

Rysunek 3.2: Komórki o rozmiarze 8x8 pikseli na zdjęciu [19].

zwrot. Na podstawie kierunku gradientu oraz wartości gradientu jasności zostaje wygenerowany histogram gradientów. W parametrze wejściowym algorytmu HOG ustawiana jest ilość kubełków histogramu. Najczęściej stosuje się 9 kubełków. Każdy kubełek histogramu odpowiada pewnemu zakresowi kąta kierunku gradientu. Przykładowo dla histogramu o szerokości 9 kubełków każdy kubełek odpowiada przedziałom 20° wycentrowanym dla podanych wartości: 0°, 20°, 40°, ..., 160°. Na rysunku 3.3 przedstawiono sposób przypisywania wartości do kubełków histogramu. Piksel zaznaczony na niebiesko posiada gradient o wartości 2 oraz kierunek o kącie 80°. Wartość gradientu zostaje dodana do aktualnej wartości kubełka histogramu reprezentującego kąt 80°. Piksel zaznaczony na czerwono posiada gradient o wartości 4 oraz kierunek o kącie 10°. Wartość gradientu równa 4 zostaje rozdzielona na dwie proporcjonalne wartości: 2 oraz 2, które są dodawane do kubełków reprezentujących kąt 0° oraz 20°. Jeśli wartość kąta wynosi 10°, wartości gradientu są przydzielane do kubełków reprezentujące dwa kąty (w tym przypadku przedziały reprezentujące kąt od 0° do 20°). W związku z tym, że powyższy kąt wynosi 10°, wartości dla kubełków histogramu są obliczane według wzoru 3.2 oraz wzoru 3.3:



Rysunek 3.3: Komórki o rozmiarze 8x8 pikseli na zdjęciu [19].

$$X_a = \frac{A_x - A_a}{A_b - A_a} \cdot G_x \tag{3.2}$$

gdzie X_a oznacza wartość przydzielaną do kubełka reprezentującego mniejszy kąt (tak jak 0° dla powyższego przykładu), A_x oznacza wartość kąta gradientu dla danego piksela (tak jak 10° dla powyższego przykładu), A_a oznacza kąt dla kubełka reprezentującego mniejszy kąt (tak jak 0° dla powyższego przykładu), A_b oznacza kąt dla kubełka reprezentującego większy kąt (tak jak 20° dla powyższego przykładu), a G_x oznacza wartość gradientu dla danego piksela (tak jak 4 dla powyższego przykładu).

$$X_b = \frac{A_b - A_x}{A_b - A_a} \cdot G_x \tag{3.3}$$

gdzie X_b oznacza wartość przydzieloną dla kąta o większej wartości (tak jak 20° dla powyższego przykładu).

Reszta zmiennych odpowiada zmiennym we wzorze 3.2. W przypadku,

gdy kąt kierunku gradientu jest większy od 160° to wartość danego gradientu jest przydzielana proporcjonalnie do komórki 160° oraz 0° zgodnie ze wzorem 3.3. Jednak wartość komórki 0° zostaje zastąpiona kątem 180°, a wyliczona wartość zostaje przydzielona do komórki reprezentującej kąt 0°. Powyższa procedura jest powtarzana, a kolejne wartości gradientów są dodawane do aktualnych wartości przechowywanych w komórkach histogramu. Końcowy efekt tej fazy dostarcza histogram reprezentujący komórkę. Wartości histogramu dla danej komórki reprezentują część wektoru deskryptora.

Kolejnym krokiem jest normalizacja bloków. Blok składa się z komórek i jego wymiar określany jest w parametrze wejściowym HOG-a. Gradienty jasności są czułe na zmiany oświetlenia na obrazie. Celem tej fazy algorytmu jest normalizacja oświetlania, aby uodpornić deskryptor na zmiany oświetlania. Zostanie tutaj opisana normalizacja za pomocą parametru L2. Najpierw zostaje wyliczona wartość L2 według wzoru3.4.

$$\|v\|_2 = \sqrt{\sum_{i=0}^{n} v(i)^2} \tag{3.4}$$

gdzie $\|v\|_2$ jest wartością L2, natomiast $v(i)$ jest i-tym elementem wektora v. L2 jest wartością normalizującą wektor v, który reprezentuje wartości deskryptora dla danego bloku. Wspomniany wektor składa się z wartości zawartych w kubełkach histogramu. Jak wiadomo z przedstawionego wyżej opisu, każdy histogram opisuje jedną komórkę, a blok zawiera n komórek. Przyjmując, że histogram reprezentujący komórkę posiada 9 wartości oraz rozmiar bloku wynosi 2x2 komórki, można wyznaczyć długość wektora v.

Normalizacja jest realizowana przy zastosowaniu Wzoru 3.5

$$v = \frac{v}{\|v\|_2} \tag{3.5}$$

gdzie $\|v\|_2$ jest wartością L2-norm dla wektora v. Działanie zgodne ze wzorem 3.5 oznacza, że każdy element wektora v jest dzielony przez wartość $\|v\|_2$.

Powyższy proces powtarza się dla wszystkich bloków. W wyniku działania algorytmu Histograms of Oriented Gradients wygenerowany zostaje deskryptor, czyli wektor reprezentujący wartości histogramów wszystkich komórek w obrazie.

Deskryptor HOG posiada kluczową zaletę w stosunku do innych metod ekstrakcji cech: działania na komórkach lokalnych pozwalają zmniejszyć efekt powstający przy ruchu pieszego. Z tego względu ta metoda nadaje się do detekcji ludzi na obrazach. W niniejszej pracy opisywana metoda posłużyła do wykrywania krawędzi, które informują o zmarszczkach. Dokładny sposób jej zaimplementowania w pracy opisuje sekcja 3.2

HOG - Zastosowanie w projekcie

W sekcji 3.2 został opisany algorytm HOG, w wyniku którego generowany jest deskryptor dla obrazu.

Omówiony wyżej deskryptor posłużył do wykrycia ilości zmarszczek w strefach zmarszczkowych. Jak wiadomo z opisu w sekcji 3.2, metoda HOG pozwala na wykrywanie krawędzi w danym obrazie. Jeśli w danym obszarze jest wiele krawędzi, to w wektorze deskryptora będzie znaczna ilość elementów tego wektora o dużej wartości, które będą identyfikowały krawędzie w obrazie. Jak wiadomo z sekcji 2.4 krawędzie mogą zostać zinterpretowane jako zmarszczki. W związku z powyższym zmodyfikowano sposób obliczenia współczynnika zmarszczek dla obrazu twarzy, opierając się o deskryptor wygenerowany przez algorytm HOG.

Wektor reprezentujący deskryptor jest generowany przez HOG-a na podstawie całego obrazu lub jego pewnego obszaru. Algorytm jest zaimplementowany w bibliotece OpenCV, która umożliwia ustawienie jego różnych parametrów. Do tych parametrów zaliczają się:

- rozmiar komórki w pikselach,
- rozmiar bloku w pikselach,
- liczba komórek w histogramie,

- rozmiar okna - wielkość analizowanego obrazu w pikselach.

W celu oszacowania ilości zmarszczek dla danego zdjęcia posłużono się następującym algorytmem:

W pierwszej fazie zostały wyznaczone strefy zmarszczkowe nr 1,3,4,5,6. (rysunek 2.13). Następnie dla każdej wyżej wymienionej strefy została wyznaczona cecha. Cecha ta jest reprezentowana przez zsumowany deskryptor HOG-a. Dla strefy 1 deskryptor jest opisany zmienną v_1, dla strefy 3 zmienną v_3 i tak dalej. Wskaźnik zmarszczek jest obliczany ze wzoru 3.6:

$$WF_{HOG} = \|v_1\|_1 + \|v_3\|_1 + \|v_4\|_1 + \|v_5\|_1 + \|v_6\|_1 \tag{3.6}$$

gdzie $\|v_1\|_1$ jest parametrem L1 dla wektora v_1. Analogicznie $\|v_3\|_1$ jest parametrem L1-norm dla wektora v_3.

Parametr L1 dla danego wektora v reprezentuje sumę jego elementów i jest wyznaczany ze wzoru 3.7:

$$\|v\|_1 = \sum_{i=1}^{n} |v(i)| \tag{3.7}$$

gdzie $v(i)$ jest i-tym elementem wektora.

Dla każdego zdjęcia ze zbioru podobnie jak w metodzie bazowej 2.3 wyznaczana jest para danych WF_{HOG} (wzór 3.6) oraz wiek. Następnie wygenerowany zbiór danych jest klasteryzowany za pomocą Fuzzy C-means (sekcja 2.6) - tak samo jak w metodzie bazowej. Wiek szacowany jest dokładnie tak samo jak w metodzie bazowej. Kolejna modyfikacja metody bazowej została opisana w sekcji 3.3

3.3 Metoda HOG oraz algorytm KNN

W pierwszej podsekcji 3.3.1 opisano sposób zastosowania algorytmu HOG-a oraz KNN w projekcie. W podsekcji 3.3.2 scharakteryzowano działanie algorytmu KNN.

3.3.1 Zastosowanie w projekcie

Kolejna modyfikacja polegała na zmianie sposobu wyznaczania wskaźnika zmarszczek. W sekcji 3.2 przedstawiono sposób wyznaczania wskaźnika zmarszczek, który polegał na sumowaniu deskryptorów z poszczególnych stref (wzór 3.6). W celu poprawy wyników uwzględniono deskryptory ze wszystkich stref z pominięciem strefy 2. Dla każdego zdjęcia ze zbioru wyznaczana jest para danych - wiek oraz wektor. $\overrightarrow{v_{KNN}}$ składający się z elementów według wzoru 3.8

$$\overrightarrow{v_{KNN}} = (\|v_1\|_1, \|v_3\|_1, \|v_4\|_1, \|v_5\|_1, \|v_6\|_1) \tag{3.8}$$

W wyniku analizy zbioru zdjęć treningowych zostaje wygenerowany zbiór zawierający wyżej wymienioną parę danych. Następnie w celu oszacowania wieku zostaje wykorzystany algorytm KNN (ang. k-Nearest Neighbor), który został opisany w sekcji 3.3.2.

3.3.2 Grupowanie KNN

Algorytm KNN to algorytm regresji nieparametrycznej używany w statystyce do prognozowania wartości pewnej zmiennej losowej. Został stworzony w roku 1970. KNN tłumaczone jest na język polski jako algorytm k- najbliższych sąsiadów. Algorytm KNN na wejściu otrzymuje zbiór danych nazywany zbiorem uczącym. Zbiór uczący zawiera dane zwane obserwacjami. Wspomniane obserwacje są parą danych (wzór 3.9).

$$O_i = (K_i, V_i) \tag{3.9}$$

gdzie O_i jest daną obserwacją, K_i - klasą, natomiast V_i - wektorem zmiennych objaśniających. Przykładowo taką jedną obserwację może tworzyć klasa określająca wiek danej osoby, a wektorem może być ilość zmarszczek.

Z kolei zbiór uczący będzie posiadał n powyższych obserwacji, na podstawie których można wywnioskować, do jakiej klasy będzie zaliczana obserwacja testowa. Obserwacja testowa to obserwacja, która posiada wektor zmiennych objaśniających i może zostać zaliczona do danej klasy za pomocą algorytmu KNN. Wracając do powyższego przykładu, obserwacja testowa będzie zawierała tylko wektor gradientów określający ilość zmarszczek. Natomiast algorytm KNN przypisze tę obserwację do klasy (wieku). Przypisywanie do danej klasy jest realizowane przez ocenę podobieństwa obserwacji testowej do zbioru uczącego. Ocena podobieństwa jest realizowana poprzez obliczanie odległości pomiędzy wektorami zmiennych objaśniających [11].

Przykładowymi miarami odległości są:

- miara Euklidesowa
- miara Manhattan
- miara Czebyszewa
- miara Minkowskiego

Odległość $D(\overrightarrow{n}, \overrightarrow{m})$ pomiędzy dwoma wektorami w mierze Euklidesowej jest opisana wzorem 3.10

$$D(\overrightarrow{n}, \overrightarrow{m}) = \sum_{i=1}^{c} (n(i) - m(i))^2 \tag{3.10}$$

Parametr $n(i)$ oznacza i-ty element wektora $\overrightarrow{n}$, $m(i)$ - i-ty element wektora $\overrightarrow{m}$, a c jest długością wektora $\overrightarrow{n}$ oraz $\overrightarrow{m}$.

W mierze Manhattan parametr $D(\overrightarrow{n}, \overrightarrow{m})$ oznacza odległość pomiędzy dwoma wektorami, która jest opisana Wzorem 3.11

$$D(\overrightarrow{n}, \overrightarrow{m}) = \sum_{i=1}^{c} |n(i) - m(i)| \tag{3.11}$$

Miara Czebyszewa opisuje odległość $D(\overrightarrow{n}, \overrightarrow{m})$ pomiędzy dwoma wektorami

zgodnie z Wzorem 3.11

$$D(\overrightarrow{n}, \overrightarrow{m}) = \max_{i=1:n}(|n(i) - m(i)|) \tag{3.12}$$

Odległość $D(\overrightarrow{n}, \overrightarrow{m})$ pomiędzy dwoma wektorami w mierze Minkowskiego jest opisana wzorem 3.13

$$D(\overrightarrow{n}, \overrightarrow{m}) = (\sum_{i=1}^{c} |n(i) - m(i)|^p)^{\frac{1}{p}} \tag{3.13}$$

Parametr p nazywany jest dystansem Minkowskiego.

W celu zmniejszenia błędów klasyfikacji w algorytmie KNN może zostać zastosowana standaryzacja lub normalizacja danych. Zastosowanie wyżej wymienionych technik pozwala na zmniejszenie dominacji wartości wektorów, których wartość jest znacznie większa lub mniejsza od ogólnej średniej. Przykładowo, przy założeniu, że wartością objaśniającą byłaby długość wyrażona w metrach i średnia tej wartości w całym zbiorze uczącym wyniosłaby 1 m, to dominującą wartością w tym zbiorze byłaby np. wartość 100 metrów. W przypadku badań w niniejszej pracy taka operacja nie była konieczna, ponieważ rozrzut wartości danych nie był duży.

Standaryzacja ma na celu obliczenie nowych wartości elementów wektorów ze zbioru uczącego. Standaryzacja jest zrealizowana poprzez wzór 3.14:

$$v_u(i) = \frac{v_u(i) - m(v_u)}{\sigma(v_u)} \tag{3.14}$$

gdzie v_u jest wektorem zmiennej objaśniającej ze zbioru uczącego, u - indeksem wektora zmiennej objaśniającej, i jest elementem wektora v_u, $m(v_u)$ jest średnią elementów wektora v_u. Natomiast $\sigma(v_u)$ jest odchyleniem standardowym elementów wektora v_u.

Normalizacja generuje wartości elementów wektorów ze zbioru uczącego,

tak aby powyższe wartości mieściły się w przedziale od 0 do 1 (Wzór 3.15).

$$v_u(i) = \frac{v_u(i) - \min(v_u)}{\max(v_u) - \min(v_u)} \tag{3.15}$$

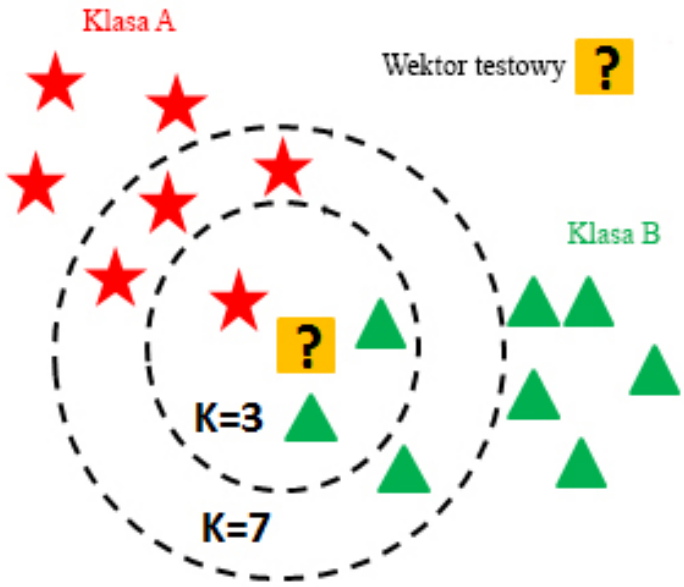
gdzie v_u jest wektorem zmiennej objaśniającej ze zbioru uczącego, u - indeksem wektora zmiennej objaśniającej, i jest elementem wektora v_u, $\max(v_u)$ oznacza maksymalną wartość spośród elementów wektora v_u. Natomiast $\min(v_u)$ minimalną wartość spośród elementów wektora v_u.

Powyżej zostały objaśnione terminy i idea algorytmu KNN. W kolejnym akapicie zostanie przedstawiony sam algorytm. Tak jak zostało opisane wyżej - algorytm otrzymuję zbiór uczący zawierający obserwację. Następnie zbiór uczący może być standaryzowany lub normalizowany. Ponadto algorytm otrzymuje parametr k. W celu wywnioskowania, do której klasy należy obserwacja testowa zawierająca wektor V_i, szukanych jest k najbliższych wektorów (wektorów sąsiadów) ze zbioru uczącego (według kryterium odległości). Obserwacja testowa zostaje przydzielona do klasy, która najczęściej występowała wśród k sąsiednich obserwacji [11]. Problem klasyfikacji algorytmem KNN jest pokazany na rysunku 3.4

Na rysunku 3.4 wyróżnione są wektory ze zbioru uczącego należące do klasy „A" oraz klasy „B". Ponadto na środku rysunku 3.4 znajduje się wektor, który ma zostać sklasyfikowany do jednej z wyżej wymienionych klas. Dla parametru „k" równego 3 wektor testowy zostanie przypisany do klasy „B", gdyż z trzech najbliższych wektorów dwa należą do klasy „B". W przypadku, gdy parametr „k" będzie równy 7 wektor testowy zostanie przypisany do klasy „A". Spośród siedmiu najbliższych sąsiadów, cztery wektory należą do klasy „A", natomiast trzy - do klasy „B".

Wybór parametru k jest zależny od rodzaju danych. Im większa wartość k, tym mniejszy wpływ na proces klasyfikacji ma szum, który określa błędne dane w zbiorze uczącym. Istnieją metody dobierające optymalną wartość k dla danego zbioru uczącego. Do jednej z nich należy optymalizacja hiperparametryczna.

Na końcu sekcji 3.3.1 został przedstawiony sposób, w jaki generowany jest



Rysunek 3.4: Przykład klasyfikacji KNN [23].

zbiór uczący. Szacowanie wieku metodą KNN w pierwszym kroku polega na stworzeniu wektoru, który został opisany wzorem 3.8. Następnym krokiem jest przydzielenie danego wektoru do klasy (wieku).

Rozdział 4

Badania

W niniejszym rozdziale zostaną przedstawione badania metody bazowej oraz jej modyfikacji. Sekcja 4.1 prezentuje środowisko pracy, a dokładniej mówiąc wszystkie niezbędne narzędzia i akcesoria, które pozwoliły skutecznie przeprowadzić badania. Ponadto zaprezentowana jest metodyka badań oraz statystyki z działania programu. Sekcja 4.2 ukazuje efekty wykrywania zmarszczek przez detektor Canny. Sekcja 4.3 zawiera porównanie metody bazowej oraz jej modyfikacji pod kątem skuteczności szacowania wieku.

4.1 Środowisko, metodyka badań oraz statystyki z działania algorytmów

W celu przeprowadzenia badań do niniejszej pracy magisterskiej wymagane było stworzenie programu, który wykrywa twarz oraz wyodrębnia cechy z twarzy. Wyżej wymieniony program został napisany w języku Java. Biblioteka, która pomogła wyodrębniać cechy z obrazu, to OpenCv. Sama ta biblioteka jest stworzona w języku C++, jednak istnieją jej modyfikacje, które zostały napisane w innych językach. Najczęściej spotykane są modyfikacje stworzone w języku Java oraz Python. Ponadto w Javie generowano dane dla metody bazowej oraz wszystkich jej modyfikacji. Dodatkowo to w tym języku zrealizowano szacowanie wieku we wszystkich metodach, z pominięciem metody, która stosuje algorytm KNN (sekcja 3.3).

Grupowanie danych algorytmem Fuzzy C-means zostało zrealizowane za pomocą programu Matlab. Również szacowanie wieku dla metody stosującej algorytm KNN przeprowadzono za pomocą wyżej wymienionego programu.

Dane z treningu były generowane do plików CSV oraz JSON. Ponadto większość kluczowych operacji była logowana do plików. Do tego celu posłużyła popularna biblioteka Log4j. Generacja plików JSON była zrealizowana przez bibliotekę GSON z pakietu com.google.code.gson, natomiast do generacji plików CSV nie została wykorzystana zewnętrzna biblioteka.

W pracy została odtworzona oryginalna praca autorów, która będzie dalej nazywana metodą bazową. W celu poprawienia wyników metody bazowej wprowadzono kilka modyfikacji, a w kolejnym kroku wyniki zostały porównane z pozostałymi metodami.

Do oceny skuteczności szacowania wieku za pomocą poszczególnych metod potrzebna jest baza treningowa oraz testowa. Baza testowa może pochodzić od części danych treningowych. Istnieje kilka metod rozdzielenia jednej bazy na bazę treningową oraz testową. Jedną z nich jest rozdzielenie bazy w oparciu o stosunek. Oznacza, to, że uzyskany zbiór treningowy będzie wynosił x procent np. 90%, a zbiór testowy odpowiednio $100\% - x$, np 10% [5].

Drugi ze sposobów to schemat CV-n (cross validation), zwany również walidacją krzyżową. Zakłada ona, że zbiór danych zostanie podzielony na n równych podzbiorów. Następnie wykonuje się n iteracji. W każdej iteracji jeden podzbiór zostaje zbiorem testowym, natomiast reszta podzbiorów tworzy zbiór treningowy. Każda iteracja daje wynik działania klasyfikatora; w przypadku niniejszej pracy jest to MAE (ang. mean absolute error), który określa średni błąd bezwzględny.

$$MAE = \frac{\sum_{i=1}^{n} |R_i - P_i|}{n} \tag{4.1}$$

gdzie R_i oznacza wartość rzeczywistą wieku dla i-tego zdjęcia, n oznacza liczbę zdjęć testowych. Natomiast P_i oznacza wartość szacowaną wieku. W wyniku n iteracji otrzymuje się uśredniony wynik działania klasyfikatora.[5].

Najczęściej dokonuje się tego przez wyliczenie średniej wartości n rezultatów uzyskanych po przeprowadzeniu n iteracji.

Metoda Leave-One-Out zakłada wykluczenie jednego obiektu, który reprezentuje zbiór testowy. Natomiast reszta obiektów zostaje zakwalifikowana do zbioru treningowego. Po przetestowaniu klasyfikatora powyższym obiektem otrzymuje się wynik skuteczności klasyfikacji. Proces ten jest powtarzany dla wszystkich obiektów. Metoda Leave-One-Out działa dokładnie tak samo jak metoda CV-n dla n równego liczbie obiektów w zbiorze danych.[5]

Dla każdej metody dokonano porównania 257 zdjęć, co stanowi 10% ogólnej liczby obrazów biorących udział w badaniach, natomiast pozostałe 90% obrazów brało udział w analizie danych, jako zbiór treningowy dla każdej metody szacowania wieku. Ponadto sprawdzono jakość klasyfikatora stosując metodę walidacji krzyżowej. Najczęściej w walidacji krzyżowej parametr n wynosi 5.

Baza zdjęć oraz statystyki z działania programu

Baza UTKface zawiera 23708 zdjęć. Nazwa każdego obrazu zawiera informację odnośnie wieku, płci i rasy fotografowanej osoby, podane w tej kolejności. Fotografie zebrane w bazie przedstawiają osoby w wieku od 1 do 116 lat.

Detektor twarzy był w stanie wyznaczyć obszar twarzy dla 14583 obrazów. Na pozostałych zdjęciach twarz nie została wykryta z różnych przyczyn. Część z tych zdjęć nie przedstawiała całej twarzy lub twarz była na nich częściowa przysłonięta. W niektórych przypadkach problem stanowiło złe oświetlenie sfotografowanej twarzy lub słaba jakość obrazu. Jednak spośród 14583 obrazów tylko 2574 brało udział w liczeniu zmarszczek, ponieważ jedynie w tej grupie algorytm poprawnie wykrył obszar nosa oraz oczu, które są niezbędne do dalszych obliczeń. Podobnie jak w przypadku braku identyfikacji twarzy, wśród przyczyn można wymienić złe oświetlenie twarzy, słabą jakość obrazu albo złe ustawienie twarzy względem obiektywu.

Średni czas generacji współczynnika zmarszczek dla obrazu po wykryciu twarzy wyniósł 0,467 sekundy dla metody bazowej. Dla metody, w której

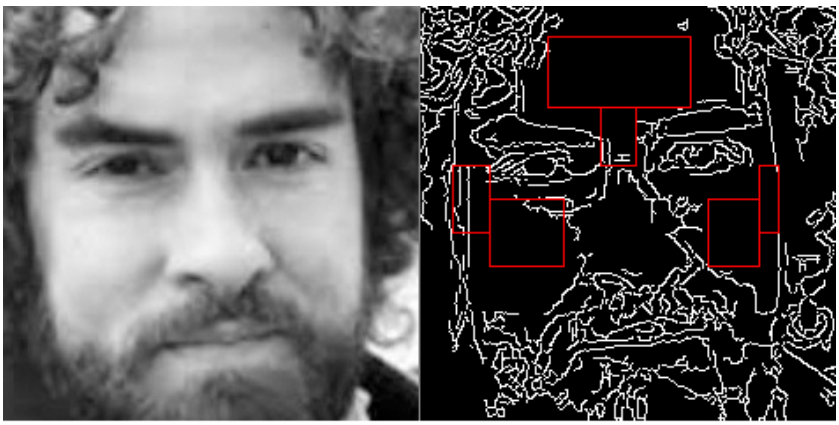
została odjęta strefa druga, generacja trwała 0,465 sekund, dla metody HOG 0,543 sekundy, natomiast dla metody HOG wraz z klasyfikacją KNN - 0,543 sekundy. Powyższe dane wykazują, że najszybciej wskaźnik zmarszczek był generowany przez algorytm metody po odjęciu jednej strefy.

Warto porównać również szybkość szacowania wieku na podstawie danego współczynnika zmarszczek. Średni czas szacowania wieku dla danego obrazu wyniósł 0,585 sekundy dla metody bazowej. Dla metody, w której została odjęta strefa druga, ten czas wyniósł 0,583 sekundy, a dla metody HOG 0,675 sekundy. Szacowanie wieku dla metody HOG wraz z klasyfikacją KNN przeprowadził program realizowany przez skrypt w Matlabie, więc nie porównano szybkości działania algorytmu zaimplementowanego w Matlabie do algorytmu zaimplementowanego w Javie. Reasumując - najszybciej szacował wiek algorytm zawierający odjętą strefę.

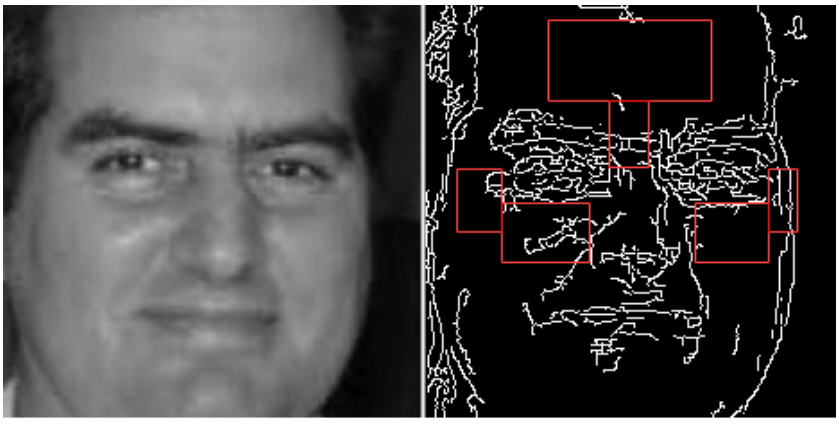
4.2 Wykrywanie zmarszczek

W tej sekcji zostaną przedstawione typowe problemy napotkane podczas detekcji zmarszczek metodą Canny oraz Histogram of Oriented Gradients. W wielu zdjęciach wystąpiły nieprawidłowe detekcje zmarszczek.

Krawędzie związane tylko ze zmarszczkami są widoczne na obrazie 4.1 Obraz 4.1 będzie dobrym punktem odniesienia do przypadków błędnych detekcji zmarszczek. Pierwszym przykładem jest wykrycie krawędzi związanych z brwiami. Na obrazie 4.2 można zauważyć, że ilość krawędzi w strefie 2 jest duża i nie identyfikują one zmarszczek. Z bazy zdjęć wyselekcjonowano 20 obrazów, na których zmarszczki zostały wykryte poprawnie, oraz 20 kolejnych obrazów, gdzie detekcja zmarszczek została zaburzona przez brwi nachodzące na strefę 2. Aby uzyskać możliwie miarodajne i uśrednione wyniki tego eksperymentu, wybrano fotografie osób w wieku od 24 do 38 lat, ponieważ w tym wieku zmarszczki na twarzy są zauważalne, ale nie za bardzo głębokie. Dla każdego obrazu wyliczono współczynnik zmarszczek ze strefy 2. Następnie dla obu grup wyliczono średnią oraz odchylenie standardowe współczynnika zmarszczek. W tabeli 4.1 przedstawiono wyniki powyższych badań dla detektora Cannego, natomiast w tabeli 4.2 przedstawiono wyniki



Rysunek 4.1: Przykład prawidłowej detekcji zmarszczek.
a) Obraz przed detekcją zmarszczek b) Obraz po detekcji zmarszczek



Rysunek 4.2: Przykład nieprawidłowej detekcji zmarszczek w strefie 2.
a) Obraz przed detekcją zmarszczek b) Obraz po detekcji zmarszczek

badań dla detekcji zmarszczek algorytmem HOG-a. W wyniku przeprowadze-

Tablica 4.1: Eksperyment 1: wykrywanie zmarszczek w strefie 2 dla detektora Canny.

	Obrazy ze zmarszczkami wykrytymi prawidłowo	Obrazy ze zmarszczkami wykrytymi nieprawidłowo
Średnia współczynnika zmarszczek	0,05	0,061
Odchylenie standardowe współczynnika zmarszczek	0,009	0,011

nia eksperymentu 1 dla detektora Canny uzyskano współczynnik zmarszczek równy 0,05 oraz odchylenie standardowe 0,009 dla obrazów ze zmarszczkami wykrytymi prawidłowo, natomiast brwi nachodzące na strefę 2 spowodowały zwiększenie współczynnika zmarszczek do 0,061, natomiast odchylenie standardowe wyniosło 0,011. Zauważona różnica nie jest jednak na tyle wysoka, by móc wyraźnie stwierdzić, że będzie miała istotny wpływ na szacowanie wieku. W przypadku HOG dla prawidłowo wykrytych zmarszczek średnia

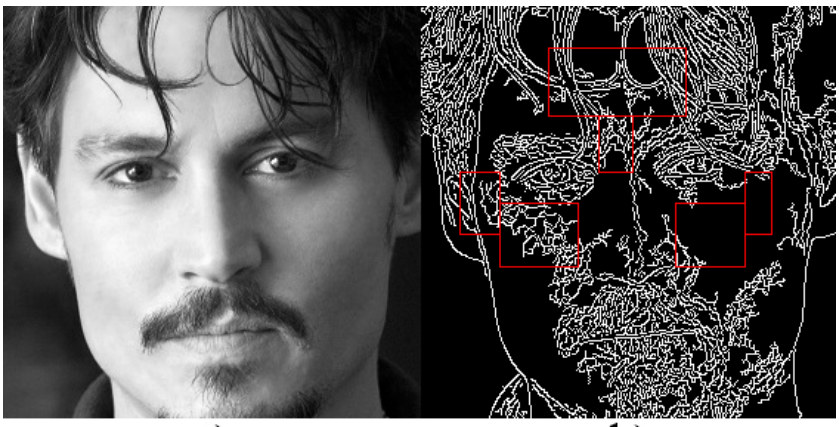
Tablica 4.2: Eksperyment 1: wykrywanie zmarszczek w strefie 2 dla detektora HOG-a.

	Obrazy ze zmarszczkami wykrytymi prawidłowo	Obrazy ze zmarszczkami wykrytymi nieprawidłowo
Średnia współczynnika zmarszczek	2,512	2,562
Odchylenie standardowe współczynnika zmarszczek	0,027	0,050

współczynnika zmarszczek wyniosła 2,512, natomiast odchylenie standardowe 0,027. Dla nieprawidłowo wykrytych zmarszczek średnia współczynnika zmarszczek zwiększyła się do 2,562, natomiast odchylenie standardowe wyniosło 0,050. Podobnie jak w przypadku detektora Cannego nie jest jasne, czy różnice mają wpływ na jakość szacowania wieku.

Kolejnym problemem okazała się grzywka nachodząca na obszar strefy 1. Na obrazie 4.3 można zauważyć wykryte krawędzie, które identyfikują

włosy. Podobnie jak powyżej, przeprowadzono porównanie, które ma na celu



Rysunek 4.3: Przykład nieprawidłowej detekcji zmarszczek w strefie 1. a) Obraz przed detekcją zmarszczek b) Obraz po detekcji zmarszczek

sprawdzenie wpływu nieprawidłowo wykrytych krawędzi na współczynnik zmarszczek. Do wspomnianego celu obliczono średni współczynnik zmarszczek oraz odchylenie standardowe ze strefy 1 dla osób w wieku od 24 do 38 lat. W tym eksperymencie wykorzystano 20 obrazów z poprzedniego eksperymentu, na których zmarszczki zostały wykryte poprawnie, oraz 20 kolejnych zdjęć, gdzie grzywka nachodząca na strefę 1 spowodowała nieprawidłowe wykrycie zmarszczek. Wyniki detekcji detektorem Cannego zaprezentowano w tabeli 4.3, natomiast wyniki dla algorytmu HOG-a pokazano w tabeli 4.4 Średnia współczynnika zmarszczek dla prawidłowo wykrytych zmarszczek,

Tablica 4.3: Eksperyment 2: wykrywanie zmarszczek w strefie 1 za pomocą detektora Canny.

	Obrazy ze zmarszczkami wykrytymi prawidłowo	Obrazy ze zmarszczkami wykrytymi nieprawidłowo
Średnia współczynnika zmarszczek	0,056	0,079
Odchylenie standardowe współczynnika zmarszczek	0,01	0,014

dla strefy 1 w eksperymencie nr 2 wyniosła 0,056, natomiast odchylenie standardowe było równe 0,01. Grzywka nachodząca na strefę 1 zwiększyła średni współczynnik zmarszczek do 0,079 oraz odchylenie standardowe do 0,014. Podobnie, jak we wnioskach po eksperymencie 1, nie można jednoznacznie stwierdzić, czy zwiększony współczynnik o 0,023 oraz odchylenie standardowe o 0,004 istotnie wpłynie na oszacowany wiek. W strefie 1 średni współczyn-

Tablica 4.4: Eksperyment 2: wykrywanie zmarszczek w strefie 1 za pomocą detektora Canny.

	Obrazy ze zmarszczkami wykrytymi prawidłowo	Obrazy ze zmarszczkami wykrytymi nieprawidłowo
Średnia współczynnika zmarszczek	2,569	2,617
Odchylenie standardowe współczynnika zmarszczek	0,041	0,056

nik zmarszczek dla obrazów zawierających prawidłowo wykryte zmarszczki wyniósł 2,569, natomiast odchylenie standardowe 0,041. Dla obrazów, gdzie detekcja dała nieprawidłowy wynik, średnia wartość współczynnika wyniosła 2,617, a odchylenie standardowe było równe 0,056. Podobnie jak w przypadku detekcji metodą Canny, nie jest możliwe jednoznaczne stwierdzenie, czy zauważona tendencja istotnie zafałszuje szacowany wiek.

4.3 Skuteczność poszczególnych metod szacowania wieku

W niniejszej sekcji zaprezentowano jakość szacowania wieku przez algorytm bazowy oraz jego modyfikację. W każdej metodzie zaprezentowano wyniki z testowania za pomocą schematu CV-5 oraz podziału na zbiór treningowy i testowy w stosunku 90% do 10%.

4.3.1 Metoda bazowa

Jako pierwsza została przetestowana metoda bazowa. Sprawdzono dokładność szacowania wieku zmieniając parametry algorytmu trenującego Fuzzy C-means:

- m - współczynnik rozmycia
- g - ilość grup
- ϵ - kryterium (wzór 2.44)

Ponadto warto wspomnieć, że detektor Canny używany przy detekcji zmarszczek posiadał niezmienne wartości progu $T_l = 10$ oraz $T_h = 100$ podczas prowadzenie badań. Przykładowy wynik testowania metody został przedstawiony w tabeli 4.5

Tablica 4.5: Przykładowy wynik testowania zawierający 15 wyników.

wiek prawdziwy	wiek wykryty	różnica
10	25	15
15	26	11
20	31	11
25	15	10
30	45	15
35	48	13
40	26	14
45	49	4
50	58	8
55	34	21
60	50	10
65	80	15
70	65	5
75	80	5
80	67	13

Wyniki z testowania 10% zdjęć

Badania rozpoczęto od przeprowadzenia testu z następującymi parametrami wejściowymi algorytmu FCM:

- m = 2,0,
- n = 1000,
- g = 10,
- ϵ=1e-5.

W kolejnych krokach badań zmieniano jeden z parametrów FCM, jednocześnie zachowując stałe wartości pozostałych parametrów. Tabela 4.6 przedstawia, jak zmieniała się wartość MAE w zależności od zmian parametru m. W kolejnej tabeli 4.7 przedstawiono wyniki przy zmianie parametru ϵ. W tabeli 4.8 przedstawiono jakość szacowania wieku przy modyfikacji parametru g.

Tablica 4.6: Wpływ parametru m na wartość wartość MAE w metodzie bazowej.

	m=2,0	m=3,0	m=5,0
MAE	12,79	11,25	11,68

Tablica 4.7: Wpływ parametru ϵ na wartość MAE w metodzie bazowej.

	$\epsilon = 1e-6$	$\epsilon = 1e-7$	$\epsilon = 1e-8$
MAE	11,18	11,3	11,34

Tablica 4.8: Wpływ parametru g na wartość MAE w metodzie bazowej.

	$g = 20$	$g = 50$	$g = 100$
MAE	10,99	10,76	10,98

Najlepszy wynik MAE dla modyfikacji parametru m osiągnięto dla m=3,0. Zwiększenie parametru m z 2,0 do 3,0 znacznie poprawiło jakość szacowania

wieku. Zwiększanie parametru ϵ nie miało istotnego wpływu na jakość szacowania wieku. Z kolei zmiana liczby grup z 20 do 50 nieznacznie polepszyła MAE. Ponadto widać, że również dalsze zwiększanie liczby grup nie wpływa na wynik MAE.

Metoda oryginalna uzyskała najlepszą jakość szacowania wieku dla następujących parametrów:

- m=2,0
- n=1000
- $\epsilon = 1e - 5$
- g=20

Dla tych parametrów MAE osiągnęło 10,76.

Wyniki z walidacji krzyżowej CV-5

Badania dla walidacji krzyżowej miały taki sam przebieg, jak w przypadku procesu testowania 10% zdjęć. Zmieniano parametr m (tabela 4.9), ϵ (tabela 4.10) oraz g (tabela 4.11). Zmiana parametru m z wartości 2,0

Tablica 4.9: Wpływ parametru m na wartość MAE w metodzie bazowej.

	m=2,0	m=3,0	m=5,0
MAE	8,64	8,64	9,07

Tablica 4.10: Wpływ parametru ϵ na wartość MAE w metodzie bazowej.

	$\epsilon = 1e - 6$	$\epsilon = 1e - 7$	$\epsilon = 1e - 8$
MAE	8,64	8,64	8,64

Tablica 4.11: Wpływ parametru g na wartość MAE w metodzie bazowej.

	$g = 20$	$g = 50$	$g = 100$
MAE	8,6	8,53	8,58

do 5,0 znacząco zwiększyła MAE. Modyfikacja parametru ϵ nie wpłynęła na wartość MAE, natomiast poprawa jakości szacowania wieku nastąpiło przy zwiększeniu grupy z 10 do 20. Najlepsza jakość szacowania wieku została osiągnięta dla następującej kombinacji parametrów wejściowych:

- m=2,0;
- g=20
- n=1000
- $\epsilon = 1e - 5$

Dla powyższej kombinacji parametrów FCM, MAE osiągnęło 8,58 roku. Testowanie za pomocą walidacji krzyżowej pokazało, że MAE ma znacznie lepszą wartość, niż w przypadku testowania 10% zdjęć.

4.3.2 Odjęcie strefy 2

Wyniki z testowania 10% zdjęć

W kolejnym kroku została przetestowana metoda, które nie uwzględnia zmarszczek ze strefy 2. Wykonano testy dla takich samych parametrów algorytmu Fuzzy C-means, jak w sekcji 4.3.1. Tabela 4.12 przedstawia, jak zmieniała się wartość MAE w zależności od zmian parametru m. W kolejnej tabeli 4.13 zawarto wyniki przy zmianie parametru ϵ. W tabeli 4.14 zaprezentowano jakość szacowania wieku przy modyfikacji parametru g.

Tablica 4.12: Wpływ parametru m na wartość wartość MAE w metodzie po odjęciu strefy 2.

	m=2,0	m=3,0	m=5,0
MAE	13,78	13,94	14,31

Zwiększanie parametru wejściowego m pogarszało wyniki szacowania wieku, natomiast ich poprawa nastąpiła przy zmniejszeniu ϵ do wartości $1e - 8$. Liczba grup nie miała większego wpływu na MAE. Dla parametrów FCM:

Tablica 4.13: Wpływ parametru ϵ na wartość MAE w metodzie po odjęciu strefy 2.

	$\epsilon = 1e-6$	$\epsilon = 1e-7$	$\epsilon = 1e-8$
MAE	13,75	13,97	12,42

Tablica 4.14: Wpływ parametru g na wartość MAE w metodzie po odjęciu strefy 2.

	$g = 20$	$g = 50$	$g = 100$
MAE	13,82	13,91	13,74

- m=2;
- g=10
- n=1000
- $\epsilon = 1e-8$

osiągnięto najlepszą wartość MAE równą 12,42 roku.

Wyniki walidacji krzyżowej CV-5

Podobnie jak w testowaniu za pomocą 10% zdjęć, zbadano wpływ zmiany parametrów na jakość szacowania wyników:

- m (tabela 4.15)
- ϵ (tabela 4.16)
- g (tabela 4.17)

Tablica 4.15: Wpływ parametru m na wartość MAE w metodzie po odjęciu strefy 2.

	m=2,0	m=3,0	m=5,0
MAE	10,74	11,03	11,23

Podobnie jak przy testowaniu 10% zdjęć, zwiększanie parametru wejściowego m pogarszało wyniki szacowania wieku. Zwiększenie ϵ z $1e-5$ do $1e-6$

Tablica 4.16: Wpływ parametru ϵ na wartość MAE w metodzie po odjęciu strefy 2.

	$\epsilon = 1e-6$	$\epsilon = 1e-7$	$\epsilon = 1e-8$
MAE	10,94	10,94	10,94

Tablica 4.17: Wpływ parametru g na wartość MAE w metodzie po odjęciu strefy 2.

	$g = 20$	$g = 50$	$g = 100$
MAE	10,9	11,01	11,03

dało minimalnie gorsze rezultaty. Dalsze zwiększanie tego parametru pozwoliło uzyskać identyczne MAE równe 10,94 roku. Liczba grup nie miała większego wpływu na MAE. Dla poniższych parametrów FCM:

- m=2;
- g=10
- n=1000
- $\epsilon = 1e-5$

osiągnięto najniższą wartość MAE równą 10,74 roku.

W porównaniu do metody bazowej szacowanie wieku szacowanie wieku w opisany wyżej sposób daje znacznie gorsze wyniki. Odjęcie jednej strefy powoduje utratę danych o zmarszczkach i zwiększa błąd szacowania wieku.

4.3.3 Metoda HOG

Kolejną przetestowaną modyfikacją była ta w której użyto algorytmu HOG. Podobnie jak w sekcjach 4.3.2 oraz 4.3.1 modyfikowano parametry algorytmu Fuzzy C-means, jednak dodatkowo dokonano zmiany parametru HOG. Parametrem poddanym modyfikacji był rozmiar komórki. Najpierw ustalono wymiary komórki na 3x3 pikseli, następnie na 6x6 pikseli, a na końcu 9x9 pikseli. Rozmiar bloku dla wszystkich modyfikacji rozmiaru komórek wyniósł 2x2 komórki. Liczba kubełków histogramu była stała i wynosiła 9 dla wszystkich pomiarów.

Wyniki z testowania 10% zdjęć

W pierwszym teście ustalono parametry Fuzzy C-means: m = 2,0, n = 100 oraz g = 10, n = 100 oraz ϵ = 1e-5. Wpływ parametru m oraz rozmiarów komórki na MAE jest przedstawiony w tabeli 4.18 Rozmiary komórki 3x3 oraz

Tablica 4.18: Wpływ parametru m oraz rozmiarów komórki na wartość MAE.

m=2,0	m=3,0	m=5,0	
16,65	16,92	11,68	komórka 3x3 piksele
16,48	16,78	11,3	komórka 6x6 piksele
11,99	16,89	11,32	komórka 9x9 piksele

6x6 pikseli przy zmianach parametru m wprowadzają niewielkie zmiany w jakość szacowania wieku. Zmiana rozmiaru komórki na 9x9 pikseli wprowadza duża zmianę wartości MAE. Wartość ta zmniejsza się drastycznie, do około 11,3 roku.

Następnie badano wpływ parametru ϵ (tabela 4.19).

Tablica 4.19: Wpływ parametru ϵ oraz rozmiarów komórki na wartość MAE.

$\epsilon = 1e-6$	$\epsilon = 1e-7$	$\epsilon = 1e-8$	
16,79	16,99	16,7	komórka 3x3 piksele
16,49	16,86	16,55	komórka 6x6 piksele
11,31	11,31	11,28	komórka 9x9 piksele

Podobnie jak przy badaniu wpływu parametru m (tabela 4.18), najlepszy wynik uzyskano dla komórki o rozmiarze 9x9 pikseli.

Analiza wpływu grup (tabela 4.20) na jakość szacowania wieku również daje rezultaty podobne do otrzymanych w wyniku badań wpływu parametru ϵ na MAE. (tabela 4.20).

Warto zauważyć, że dla komórki o rozmiarach 3x3 nastąpiła znaczna poprawa MAE przy zwiększeniu liczby grup z 20 do 50.

Reasumując wyniki badań dla testowania 10% zdjęć można powiedzieć, że najlepszy rezultat osiągnięto dla rozmiarów komórki 9x9. Zmiana parametrów FCM w niewielkim stopniu wpływały na jakość szacowania wieku, która oscylowała od wartości MAE 11,28 do 11,31 roku.

Tablica 4.20: Wpływ parametru g oraz rozmiarów komórki na wartość MAE.

$g = 20$	$g = 50$	$g = 100$	
16,59	15,48	15,49	komórka 3x3 piksele
16,78	16,17	16,55	komórka 6x6 piksele
11,28	11,3	11,28	komórka 9x9 piksele

Wyniki walidacji krzyżowej CV-5

Parametr m oraz rozmiar komórki wpływały na parametr MAE w sposób przedstawiony w tabeli 4.21

Tablica 4.21: Wpływ parametru m oraz rozmiarów komórki na wartość MAE.

m=2,0	m=3,0	m=5,0	
9,81	9,78	9,9	komórka 3x3 piksele
9,93	9,9	9,85	komórka 6x6 piksele
9,95	9,92	9,83	komórka 9x9 piksele

Badanie wpływu parametru m metodą CV-5 ujawnia niewielki rozrzut wyniku MAE. Wartości tych wyników zmieniają się o około 0,1 roku. Warto podkreślić, że w przypadku tego testu wartość MAE jest znacznie mniejsza niż przy testowaniu za pomocą 10% zbioru.

Wpływ zmiany parametru ϵ przedstawiono w tabeli 4.22.

Tablica 4.22: Wpływ parametru ϵ oraz rozmiarów komórki na wartość MAE.

$\epsilon = 1e-6$	$\epsilon = 1e-7$	$\epsilon = 1e-8$	
9,81	9,81	9,81	komórka 3x3 piksele
9,93	9,93	9,93	komórka 6x6 piksele
9,95	9,95	9,95	komórka 9x9 piksele

Podobnie jak przy badaniu wpływu parametru m (tabela 4.18), rozrzut danych jest niewielki. Warto podkreślić, że najlepszy wynik uzyskano dla najmniejszej komórki, o wymiarach 3x3 piksela. Ponadto zmiana parametru ϵ dla każdego rozmiaru komórek nie miała żadnego wpływu na wartość MAE.

Na końcu przetestowano wpływ ilości grup w parametrze FCM. Wyniki zaprezentowano w tabeli 4.23

Tablica 4.23: Wpływ parametru g oraz rozmiarów komórki na wartość MAE.

$g = 20$	$g = 50$	$g = 100$	
9,81	9,85	11,49	komórka 3x3 piksele
9,94	10,08	11,55	komórka 6x6 piksele
9,94	11,3	11,38	komórka 9x9 piksele

Przy testowaniu wpływu ilości grup widoczna jest tendencja pogarszania się wyników dla każdego rozmiaru komórki wraz ze wzrostem ilości grup.

Wyniki testowania 10% zdjęć oraz metodą CV pokazują, że najlepszy wynik MAE został osiągnięty dla komórki 3x3 piksela oraz dla następujących parametrów FCM:

- m=3,0;
- g=10;
- n=1000;
- $\epsilon = 1e - 5$

W porównaniu do poprzedniej modyfikacji polegającej na odjęciu strefy 2 następuje znaczna zmniejszenie wartości MAE. Jednak nie osiągnięto tak dobrej jakości szacowania wieku, jak w metodzie bazowej.

4.3.4 Metoda HOG + KNN

Ostatnią testowaną metodą była ta, która zawiera algorytm HOG, a do klasyfikacji i szacowania używa algorytmu KNN. Zmieniano parametr k dla algorytmu KNN, natomiast dla algorytmu HOG - rozmiary komórki. Parametr k był równy 1, 3 oraz 5. Rozmiary komórki miały takie same wartości jak przy badaniach w sekcji 4.3.3. Liczba kubełków histogramu była stała i wynosiła 9. Podobnie jak w sekcji 4.3.3 ustalono rozmiary bloku 2x2 komórki.

Wyniki z testowania 10% zdjęć

W tabeli 4.24 podsumowano jakość szacowania wieku przy zmianach parametru k oraz dla różnych rozmiarów komórek. Dla komórek o rozmiarach

Tablica 4.24: Wpływ parametru k oraz rozmiarów komórki na wartość MAE.

$k = 1$	$k = 3$	$k = 5$	
11,76	11,75	11,49	komórka 3x3 piksele
10,87	10,34	10,08	komórka 6x6 piksele
10,14	10,08	9,98	komórka 9x9 piksele

3x3 oraz 9x9 pikseli zwiększanie parametru k poprawiało nieznacznie wartość MAE. Najlepsza jakość klasyfikacji została osiągnięta dla komórki o rozmiarach 9x9 piksela i parametru k=5.

Wyniki walidacji krzyżowej CV-5

Tabela 4.25 zawiera wartości MAE osiągnięte w procesie walidacji krzyżowej CV-5 dla opisywanych modyfikacji.

Tablica 4.25: Wpływ parametru k oraz rozmiarów komórki na wartość MAE.

$k = 1$	$k = 3$	$k = 5$	
10,77	10,79	10,54	komórka 3x3 piksele
10,06	9,95	10	komórka 6x6 piksele
9,66	9,6	9,64	komórka 9x9 piksele

Tak, jak w poprzednich testach metodą CV-5, można wywnioskować, daje ona bardziej miarodajne wyniki szacowania wieku niż analiza 10% obrazów. Błąd MAE osiąga najmniejsze wartości dla komórki o rozmiarach 9x9 pikseli. Najlepsza uzyskany wynik MAE wynosi 9,6 i została osiągnięta dla parametru k = 3. Co istotne, metoda opierająca się na połączeniu algorytmów HOG i KNN daje błąd MAE o najmniejszych wartościach spośród wszystkich modyfikacji metody bazowej.

Rozdział 5

Podsumowanie

TODO

Bibliografia

[1] T. F. Cootes A. Lanitis, Ch. J. Taylor. *Toward automatic simulation of aging effects on face images*. IEEE Transactions of Pattern Analysis and Machine Intelligence, 2002.

[2] Abid K. Alexander Mordvintsev. Face Detection using Haar Cascades . `https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_objdetect/py_face_detection/py_face_detection.html`, 2019. [Online; Dostęp: 28.09.19].

[3] Munish Kumar Ashu Kumar, Amandeep Kaur. *Face detection techniques: a review*. Springer, 2018.

[4] BADGERATI. https://computersciencesource.wordpress.com/2010/09/03/computer-vision-the-integral-image/, 2010. [Online; Dostęp: 28.09.19].

[5] Pawel Cichosz. *Data Mining Algorithms: Explained Using R*. Wiley, 2014.

[6] Coutaz J Crowley JL. *Vision for man-machine interaction*. Robotics and Autonomous Systems, 1997.

[7] Navneet Dalal, Bill Triggs. *Histograms of Oriented Gradients for Human Detection*. Conference on Computer Vision Pattern Recognition, 2005.

[8] M.M. Dehshibi, A. Bastanfard. *A new algorithm for age recognition from facial images*. Signal Processing, 2010.

[9] Brian Everitt. *Cluster analysis*. Wiley, 2011.

[10] Erik Fritts. https://www.videomaker.com/article/c03/18848-tell-your-story-more-effectively-with-the-correct -application-of-hard-and-soft-light, 2017. [Online; Dostęp: 07.10.19].

[11] Alvador Garcia, Joaquin Derrac, Jose Cano, Francisco Herrera. *Prototype Selection for Nearest Neighbor Classification: Taxonomy and Empirical Study*. IEEE, 2012.

[12] Yun Fu Thomas S. Huang Guodong Guo, Guowang Mu. *Human Age Estimation Using Bio-inspired Features*. IEEE Conference on Computer Vision and Pattern Recognition, 2009.

[13] Noor A. Ibraheem, Mokhtar M. Hasan, Rafiqul Z. Khan, Pramod K. Mishr. *Understanding Color Models: A Review*. ARPN Journal of Science and Technology, 2012.

[14] Scott Kelby. *Sekrety mistrza fotografii cyfrowej*. Helion, 2014.

[15] Vladimir Khryashchev, Alexander Ganin, Olga Stepanova, Anton Lebedev. *Age Estimation from Face Images: Challenging Problem for Audience Measurement Systems*. Yaroslavl State University, 2014.

[16] Wan-Chi Siu Kwok-Wai Wong, Kin-Man Lam. *Age Estimation from Face Images: Challenging Problem for Audience Measurement Systems*. Centre for Multimedia Signal Processing, Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hung Hom, Hong Kong, 2000.

[17] Dominik Ślęzak, Jakub Wróblewski. http://edu.pjwstk.edu.pl/wyklady/adn/scb/wyklad 2006. [Online; Dostęp: 26.10.19].

[18] Xiaoju Ma, Bo Li, Ying Zhang, Ming Yan. *The Canny Edge Detection and Its Improvement*. Springer, 2012.

[19] Satya Mallick. https://www.learnopencv.com/histogram-of-oriented-gradients/, 2016. [Online; Dostęp: 21.10.19].

[20] Robert K. McConnell. Method of and apparatus for pattern recognition, 1982. Patent nr US4567610.

[21] Henrik Skov Midtiby. Example: Rgb color mixing, 2019. [Online; Dostęp: 28.09.19].

[22] R. Chellappa N. Ramanathan. *Face verification across age progression.* IEEE Transactions on Image Processing, 2006.

[23] Avinash Navlani. https://www.datacamp.com/community/tutorials/k-nearest-neighbor-classification-scikit-learn, 2018. [Online; Dostęp: 22.10.19].

[24] Michael Jones Paul Viola. *Rapid Object Detection using a Boosted Cascade of Simple Features.* Accepted conference on computer vision and pattern recognition, 2001.

[25] Matti Pietikäinen, Abdenour Hadid, Guoying Zhao, Timo Ahonen. *Computer Vision Using Local Binary Patterns.* Springer, 2011.

[26] Rituparna Sahaa Ranjan Janaa, Debaleena Dattaa. *Age Estimation from Face Image using Wrinkle Features.* RCC Institute of Information Technology, 2014.

[27] Qaim Mehdi Rizvi. *A Review on Face Detection Methods.* Qassim University, 2011.

[28] Peter M. Roth, Helmut Grabner, Christian Leistner, Horst Bischof, Martin Winter. *Interactive earning a person detectpr: fewer clicks – less frustration.* Researchgate, 2008.

[29] Przemyslaw Spurek. http://analizaobrazu.x25.pl/articles/11, 2018. [Online; Dostęp: 19.10.19].

[30] Richard Szeliski. *Computer Vision: Algorithms and Applications.* Springer, 2011.

[31] Piotr Wilkowski. *Wykorzystanie algorytmu detekcji i lokalizacji w zadaniu chwytania*. Politechnika Warszawska, Wydział Elektroniki I Technik Informacyjnych, 2009.

[32] K. Smith-Miles X. Geng, Z. Zhou. *Automatic age estimation based on facial aging patterns*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007.

[33] N. da V. Lobo Y. H. Kwon. *Age classification from facial images. Computer Vision and Image Understanding*. 1999.

Dodatki

Dokumentacja techniczna

Program został zrealizowany zarówno w programie Java oraz Matlab.

Funkcje Java

[*dane_trenigowe*] = generateDataFromImagesAuto()

Funkcja generuje dane treningowe według jednego z algorytmów

Dane wejściowe:

propertiesLoader	PropertiesLoader	Klasa PropertiesLoader zawierająca ścieżkę do obrazów treningowych oraz konfigurację algorytmów generujących wskaźnik zmarszczek

Dane wyjściowe:

dane_trenigowe	File	Plik zawierający dane wygenerowanych wskaźników zmarszczek

[*wiek*] = recognizeAge()

Funkcja szacująca wiek na podstawie danych wygenerowanych z algorytmu FCM oraz danego obrazu.

Dane wejściowe:

path	String	Ścieżka do obrazu, z którego szacowany jest wiek
age2centers	String	Ścieżka do centroidów wygenerowanych z algorytmu FCM

Dane wyjściowe:

wiek	int	Szacowany wiek z danego obrazu

Funkcje Matlaba

[*age*2*centers*] = clusterToFuzzyCMeans()

Funkcja generujące centroidy z wykorzystaniem algorytmu Fuzzy C-means na podstawie danych z treningu (wygenerowane za pomocą funkcji generateDataFromImagesAuto())

Dane wejściowe:

dane_treningowe	File	Dane z treningu wygenerowane za pomocą funkcji generateDataFromImagesAuto()

Dane wyjściowe:

age2centers	File	Centroidy niezbędne do szacowania wieku w funkcji recognizeAge()

[*wiek_szacowany*] = knnHog59()

Funkcja trenująca i szacująca wiek dla metody HOG + KNN.

Dane wejściowe:

KnnHog59	File	Plik z danymi treningowymi. Dane wygenerowane za pomocą algorytmu HOG, w którym komórka miała rozmiary 5x5 piksela
TrainKnnHog59	File	Plik z danymi testowymi

Dane wyjściowe:

wiek_szacowany	File	Plik zawierający wiek szacowany dla danych testowych dla parametrów k=1,3,5 algorytmu KNN

[*wiek_szacowany*] = knnHog79()

Funkcja trenująca i szacująca wiek dla metody HOG + KNN.

Dane wejściowe:

KnnHog79	File	Plik z danymi treningowymi. Dane wygenerowane za pomocą algorytmu HOG, w którym komórka miała rozmiary 7x7 piksela
TrainKnnHog79	File	Plik z danymi testowymi

Dane wyjściowe:

wiek_szacowany	File	Plik zawierający wiek szacowany dla danych testowych dla parametrów k=1,3,5 algorytmu KNN

[*wiek_szacowany*] = knnHog99()

Funkcja trenująca i szacująca wiek dla metody HOG + KNN.

Dane wejściowe:

KnnHog79	File	Plik z danymi treningowymi. Dane wygenerowane za pomocą algorytmu HOG, w którym komórka miała rozmiary 9x9 piksela
TrainKnnHog99	File	Plik z danymi testowymi

Dane wyjściowe:

wiek_szacowany	File	Plik zawierający wiek szacowany dla danych testowych dla parametrów k=1,3,5 algorytmu KNN

Spis skrótów i symboli

FCM - *Fuzzy C-means* - metoda klasteryzacji miękkiej (rozmytego).

HOG - *Histograms of Oriented Gradients* - metoda generacji deskryptora obrazu

KNN - *k-Nearest Neighbors* - algorytm regresji lub klasyfikacji.

MAE - *Mean Absolute Error* - średni błąd bezwzględny

BIF - *Biologically Inspired Features* - detektor cech o charakterze biologicznym

LBP - *Local Binary Patterns* - detektor cech (lokalnych wzorców binarnych)

FPR - *False Positive Rate* - współczynnik fałszywych klasyfikacji, jako obiekt.

TP - *True Positive* - poprawna klasyfikacja, jako obiekt

TN - *True Negative* - poprawna klasyfikacja, jako nie - obiekt

FP - *False Positive* - błędna klasyfikacja jako obiekt

FN - *False Negative* - błędna klasyfikacja jako nie - obiekt

INRIA - *Institut National de Recherche en Informatique et en Automatique* - francuski instytut zajmujący się badaniami w zakresie informatyki i automatyki

Zawartość dołączonej płyty

Do pracy dołączona jest płyta CD z następującą zawartością:

- praca w formacie `pdf`,
- źródła programów .m i główna klasa generująca wskaźnik zmarszczek w formacie .java
- wersja elektroniczna pracy:
  - wersja kompletna
  - wersja bez spisu treści, bibliografii, tabel, załączników oraz rysunków
- wyniki działania programów

Spis rysunków

1.1 Przykład rozpoznawania osób na obrazie [28]. 2

2.1 Faza 1 algorytmu 7
2.2 Faza 2 algorytmu 7
2.3 Przykład twarzy oświetlonej ostrym (twarz po lewej) oraz miękkim światłem [10]. 9
2.4 Różne techniki wykrywania twarzy [3]. 10
2.5 Mieszanie kanałów RGB [21]. 13
2.6 Kolor R=153 G=217 B=234. 13
2.7 Filtr Haara a) krawędziowy b) liniowy c) szachownica [2] . . . 15
2.8 Filtr Haara nałożony na krawędź twarzy [2] 15
2.9 Tabela jasności poszczególnych pikseli: a) przed zastosowaniem całkowania b) po zastosowaniu całkowania [4] 16
2.10 Sumowanie okna [4] 16
2.11 Kaskada klasyfikatorów [31]. 19
2.12 Wykryty nos oraz oczy. 20
2.13 Strefy zmarszczkowe widoczne w białych prostokątach. 21
2.14 Wyznaczenie strefy znajdującej się na czole. 21
2.15 Pomocnicze współrzędne do wyliczania stref zmarszczkowych. 22
2.16 Punkty wyznaczające położenie stref. 23
2.17 a) Obraz z nieznaczną ilością szumów b) Obraz ze znaczną ilością szumów [29]. 27
2.18 a) Oryginalny obraz b) Obraz z wykrytymi krawędziami. . . . 30

3.1 Gradient jasności identyfikujący granicę pomiędzy głową a tłem [19]. 39
3.2 Komórki o rozmiarze 8x8 pikseli na zdjęciu [19]. 40
3.3 Komórki o rozmiarze 8x8 pikseli na zdjęciu [19]. 41
3.4 Przykład klasyfikacji KNN [23]. 49

4.1 Przykład prawidłowej detekcji zmarszczek.
a) Obraz przed detekcją zmarszczek b) Obraz po detekcji zmarszczek 55
4.2 Przykład nieprawidłowej detekcji zmarszczek w strefie 2.
a) Obraz przed detekcją zmarszczek b) Obraz po detekcji zmarszczek 55
4.3 Przykład nieprawidłowej detekcji zmarszczek w strefie 1.
a) Obraz przed detekcją zmarszczek b) Obraz po detekcji zmarszczek 57

Spis tablic

4.1 Eksperyment 1: wykrywanie zmarszczek w strefie 2 dla detektora Canny. 56
4.2 Eksperyment 1: wykrywanie zmarszczek w strefie 2 dla detektora HOG-a. 56
4.3 Eksperyment 2: wykrywanie zmarszczek w strefie 1 za pomocą detektora Canny. 57
4.4 Eksperyment 2: wykrywanie zmarszczek w strefie 1 za pomocą detektora Canny. 58
4.5 Przykładowy wynik testowania zawierający 15 wyników. . . . 59
4.6 Wpływ parametru m na wartość wartość MAE w metodzie bazowej. 60
4.7 Wpływ parametru ϵ na wartość MAE w metodzie bazowej. . . 60
4.8 Wpływ parametru g na wartość MAE w metodzie bazowej. . . 60
4.9 Wpływ parametru m na wartość MAE w metodzie bazowej. . 61
4.10 Wpływ parametru ϵ na wartość MAE w metodzie bazowej. . . 61
4.11 Wpływ parametru g na wartość MAE w metodzie bazowej. . . 61
4.12 Wpływ parametru m na wartość wartość MAE w metodzie po odjęciu strefy 2. 62
4.13 Wpływ parametru ϵ na wartość MAE w metodzie po odjęciu strefy 2. 63
4.14 Wpływ parametru g na wartość MAE w metodzie po odjęciu strefy 2. 63
4.15 Wpływ parametru m na wartość MAE w metodzie po odjęciu strefy 2. 63

4.16 Wpływ parametru ϵ na wartość MAE w metodzie po odjęciu strefy 2. 64
4.17 Wpływ parametru g na wartość MAE w metodzie po odjęciu strefy 2. 64
4.18 Wpływ parametru m oraz rozmiarów komórki na wartość MAE. 65
4.19 Wpływ parametru ϵ oraz rozmiarów komórki na wartość MAE. 65
4.20 Wpływ parametru g oraz rozmiarów komórki na wartość MAE. 66
4.21 Wpływ parametru m oraz rozmiarów komórki na wartość MAE. 66
4.22 Wpływ parametru ϵ oraz rozmiarów komórki na wartość MAE. 66
4.23 Wpływ parametru g oraz rozmiarów komórki na wartość MAE. 67
4.24 Wpływ parametru k oraz rozmiarów komórki na wartość MAE. 68
4.25 Wpływ parametru k oraz rozmiarów komórki na wartość MAE. 68