

**WYDZIAŁ ELEKTROTECHNIKI, AUTOMATYKI,  
INFORMATYKI I INŻYNIERII BIOMEDYCZNEJ**

Praca dyplomowa magisterska

*Projekt systemu sterowania oparty na algorytmie przetwarzania obrazów z elementów i mimiki twarzy*

*<Project of steering system based on algorithm for images of face and mimcry processing >*

Autor: *Jakub Zieliński*

Kierunek studiów: Informatyka i Elektronika Medyczna

Opiekun pracy: *dr inż. Krzysztof Lalik*

Kraków, 2015

*Oświadczam, świadomy(-a) odpowiedzialności karnej za poświadczenie nieprawdy, że niniejszą pracę dyplomową wykonałem(-am) osobiście i samodzielnie i że nie korzystałem(-am) ze źródeł innych niż wymienione w pracy.*

*…………………………………………*

# Spis treści

[Spis treści 3](#_Toc422169782)

[1. Wstęp 4](#_Toc422169783)

[1.1 Przedstawienie problemu 5](#_Toc422169784)

[2. Istniejące rozwiązania 5](#_Toc422169785)

[2.1 Techniki przetwarzania obrazów twarzy 5](#_Toc422169786)

[2.1.1 Wykrywanie twarzy na obrazach ze statycznym tłem sceny 6](#_Toc422169787)

[2.1.1.1 Wykrywanie twarzy na podstawie koloru 6](#_Toc422169788)

[2.1.1.2 Wykrywanie twarzy na podstawie ruchu (odejmowanie tła) 6](#_Toc422169789)

[2.1.2 Wykrywanie twarzy na obrazach z niekontrolowanym tłem 7](#_Toc422169790)

[2.1.2.1 Wykrywanie twarzy z wykorzystaniem modelu 7](#_Toc422169791)

[2.1.2.2 Wykrywanie twarzy na podstawie konturów 8](#_Toc422169792)

[2.2 Dostępne rozwiązania 8](#_Toc422169793)

[3. Wstęp teoretyczny 11](#_Toc422169794)

[3.1 Wykrywanie obiektów przy pomocy algorytmu Viola-Jones i AdaBoost 11](#_Toc422169795)

[3.1.1 Cechy (ang. features) 11](#_Toc422169796)

[3.1.2 Obraz integracyjny (ang. Integral Image) 13](#_Toc422169797)

[3.1.3 AdaBoost 14](#_Toc422169798)

[3.2 Śledzenie pozycji głowy 19](#_Toc422169799)

[3.2.1 Active Appearance Model 20](#_Toc422169800)

[3.2.1.1 Active Shape Model 20](#_Toc422169801)

[3.2.1.2 Statyczny Model Tekstury 22](#_Toc422169802)

[3.2.2 Pose from Orthography and Scaling with Iteration 24](#_Toc422169803)

[3.2.2 Scaled Orthographic Projection and Perspective Projection 26](#_Toc422169804)

[3.2.4 Geometria SOP 26](#_Toc422169805)

[3.3 Podsumowanie 30](#_Toc422169806)

# Wstęp

Najbardziej popularnym sposobem interakcji człowieka z komputerem od wielu lat jest – i prawdopodobnie na długo jeszcze pozostanie – system oparty o interfejs graficzny, manipulacje przy użyciu myszy oraz wprowadzanie danych przy pomocy klawiatury. Dla specyficznych zastosowań powstają dedykowane kontrolery, pozwalające na wprowadzanie danych skojarzonych z wybranymi aplikacjami w sposób najbardziej ergonomiczny i intuicyjny. W przypadku rozrywki multimedialnej wykorzystywane są joysticki, pady, oraz kontrolery ruchu. Obecnie powstają również systemy wykorzystujące inne drogi komunikacji, oparte na pozostałych zmysłach, a nawet łączące w sposób równoległy wiele sposobów komunikacji człowieka z maszyną. Podejście to określane jest mianem interfejsów multimodalnych (multimodal interfaces). Obserwacja sposobu, w jaki komunikują się ludzie w kontakcie bezpośrednim, pozwala stwierdzić, że duży udział w przekazywanej informacji ma część niewerbalna – mowa ciała, a w szczególności mimika twarzy. Przeniesienie niewerbalnej warstwy komunikacji do interfejsu człowiek – maszyna niesie ze sobą wiele korzyści. W przypadku osób dotkniętych chorobą lub kalectwem, odbierającym im sprawność rąk, dłoni lub co gorsza większych partii ciała (paraliż) ruchomość głowy, mimika pozostają często jedynym sposobem komunikacji ze światem zewnętrznym. Ponadto w wielu przypadkach, podczas wykonywania czynności angażujących wiele partii ciała, wymagających skupienia (operacja chirurgiczna, prowadzenie pojazdu), wykorzystanie kolejnego portu komunikacji z urządzeniami jest bardzo ważnym aspektem pozwalającym na wykonywanie większej liczby czynności, lub wykonywanie ich w sposób bardziej naturalny, mniej absorbujący, co może przełożyć się na większą precyzję, bezpieczeństwo i optymalizację działań, a nawet ich automatyzację jeżeli część decyzyjną pozostawimy po stronie techniki, sami dostarczając jedynie bodźce, które system może interpretować.

## Przedstawienie problemu

Celem pracy magisterskiej jest opracowanie i implementacja systemu śledzącego ruchy głowy użytkownika komputera za pomocą kamery internetowej. Ponadto system powinien rozpoznawać mrugnięcia użytkownika. Docelowym przeznaczeniem aplikacji ma być sterowanie kursorem myszy przy pomocy ruchów głowy, oraz obsługa kliknięć realizowanych przez mrugnięcia.

Istotnym założeniem pracy jest opracowanie aplikacji działającej w czasie rzeczywistym, nie wymagającej od użytkownika ingerencji w proces konfiguracji, rozpoznawania twarzy, oraz manipulacji przy konfiguracji sprzętowej.

Stworzenie systemu spełniającego wszystkie przedstawione wymagania nie jest łatwym zadaniem, biorąc pod uwagę złożoność operacji przekształceń obrazów, różnorodności warunków (np. oświetlenie) oraz cech osobniczych dotyczących parametrów twarzy. Świadczy o tym również fakt, iż nie istnieje aplikacja komercyjna wykorzystywana na szeroka skalę, korzystająca z zaledwie jednej kamery. Autor uważa, że opracowanie prostego, skutecznego systemu realizującego podstawowe funkcje wpłynie pozytywnie na popularyzację tego typu systemów przede wszystkim wśród osób najbardziej tego potrzebujących. Jednakże uniwersalizm, oraz zastosowanie kamery internetowej, sprawiają, iż jakość interpretacji gestów będzie niższa w porównaniu z systemami komercyjnymi.

Rynek oferuje wiele rozwiązań pozwalających na rozpoznawanie mimiki twarzy, gestów, komunikację głosową. Jednak produkty komercyjne charakteryzują się oczywiście wysoka ceną, a większość rozwiązań wymaga dedykowanych urządzeń pozwalających na śledzenie wybranych parametrów. Ponadto aby zapewnić optymalne działanie użytkownik jest zmuszony przebrnąć przez obszerną konfigurację, dostosowującą parametry do aktualnych warunków, lub personalizujących komunikację.

# Istniejące rozwiązania

Obecnie istnieje wiele rozwiązań pozwalających na śledzenie i rozpoznawanie głowy, twarzy oraz innych elementów ciała człowieka. Jednak autor nie dotarł do informacji na temat systemów oferujących kompleksową obsługę interfejsu użytkownika. Dlatego też w niniejszym rozdziale uwagę poświęcono systemom pozwalającym na rozpoznawanie i śledzenie.

## Techniki przetwarzania obrazów twarzy

Dziedzina przetwarzania cyfrowych obrazów wciąż zyskuje na popularności, pojawiają się nowe zastosowania oraz techniki rozwiązania danego problemu. Nie inaczej jest w przypadku rozpoznawania twarzy na obrazie. Można wyróżnić co najmniej kilka sposobów podejścia do zagadnienia, a każde z nich posiada zalety specyficzne dla różnych zastosowań.

## Wykrywanie twarzy na obrazach ze statycznym tłem sceny

Techniki wykorzystujące kontrolowane warunki otoczenia. Podstawowym założeniem jest statyczne tło, jednolitego koloru, odcinającego się od obserwowanego obiektu. Ponadto ważnym elementem jest równomierne oświetlenie.

### Wykrywanie twarzy na podstawie koloru

Technika bazująca na rozpoznawaniu koloru światła odbitego przez skórę twarzy człowieka.

Do uzyskania zadowalających efektów potrzebna jest znajomość parametrów kamery, spektrum światła generowanego przez źródło, oraz model skóry oferujący uśrednione wartości długości fali odbitej przez skórę koloru w przybliżeniu odpowiadającemu każdej z ras. [Skin colour detection under changing lighting conditions]



Rysunek Po lewej: Widmo światła odbitego przez skórę człowieka odpowiedni dla: rasy kaukaskiej, mongolskiej oraz afrykańskiej. Po prawej: Widmo światła odbitego przez skórę rasy kaukaskiej o różnej jasności.

Algorytm wykorzystujący powyższe parametry jest szybki (prosty algorytm wykorzystujący binaryzację) i niewrażliwy na rotację obiektu. Jego wadą natomiast jest wrażliwość na zmieniające się warunki otoczenia oraz konieczność uwzględnienia parametrów sprzętowych. [Skin colour detection under changing lighting conditions]

### Wykrywanie twarzy na podstawie ruchu (odejmowanie tła)

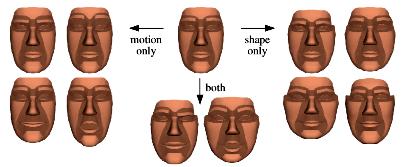
Podobnie jak rozwiązanie w punkcie 2.1.1.2, również w tym przypadku podstawą do rozpoznania twarzy jest model koloru skóry. Jest on jednak wykorzystywany w fazie inicjalizacji. Algorytm wykorzystuje założenie, że tło jest nieruchome. Dzięki temu znacznie łatwiej wyodrębnić poruszające się obiekty w obrębie obserwowanej sceny. Uwzględniając nierównomierny rozkład oświetlenia i korzystający z mechanizmu predykcyjnego pozwalającego w pewnym stopniu przewidzieć kierunek przemieszczania się obiektu otrzymano zadowalające wyniki, nawet w przypadku śledzenia kilku obiektów jednocześnie. [Tracking Human Faces in Real-Time]

Zaletą tej metody jest szybkość działania, możliwość śledzenia wielu obiektów jednocześnie przy zachowaniu ich odrębności. Wadą natomiast jest ograniczenie dotyczące statycznego tła sceny oraz wrażliwość na skokowe zmiany oświetlenia, lub kolorystki tła. [Tracking Human Faces in Real-Time]

## Wykrywanie twarzy na obrazach z niekontrolowanym tłem

Rozwiązania tego typu bazują na bardziej szczegółowych cechach obiektu, takich jak wzajemne położenie struktur, krawędzie[Real-Time Face Detection Using Edge-Orientation Matching], rozpoznawanie struktur na podstawie uczenia maszynowego, lub klasyfikatorów kaskadowych (*ang. weak classifier cascades*). Ich złożoność jest o wiele większa niż wymienonych w punkcie 2.1.1, jednak ilość informacji, które są w stanie dostarczyć pozwala na budowę użytecznych aplikacji. [http://www.cs.rutgers.edu/~decarlo/facetrack.html] [https://facedetection.com/techniques/]

### Wykrywanie twarzy z wykorzystaniem modelu

Model w rozumieniu przetwarzania obrazów twarzy, jest zbiorem punktów reprezentujących uśrednione parametry opisujące: wygląd, kształt, ruch obiektu. [http://www.cs.rutgers.edu/~decarlo/facetrack.html]

Rysunek Model kształtu i ruchów twarzy

Model reprezentowany przez zbiór punktów, można dostosować do aktualnie obserwowanego obiektu poprzez modyfikację parametrów kształtu, lub przemieszczenia.

Takie rozwiązanie, choć znacznie bardziej wymagające obliczeniowo, oferuję informacje nie tylko o położeniu, ale również może posłużyć do wyznaczenia położenia w 3D, co ma niebagatelne znaczenie w praktycznym zastosowaniu. Ponadto pozwala na określenie położenia i zniekształceń (ruchy ust, brwi) elementów twarzy. Dzięki czemu otrzymujemy kolejne punkty umożliwiające rozbudowanie interfejsu.

Tworzenie modelu odbywa się poprzez nanoszenie punktów na obrazy w miejscach odpowiadających określonym strukturom. W celu uzyskania najlepszych efektów tworzy się model w oparciu o konkretnego użytkownika. Trzeba jednak zauważyć, że jest to żmudna procedura wykonywana ręcznie co zwiększa jej podatność na błędy. Rozbudowany opis powstawania modelu zostanie opisany w dalszej części pracy.

### Wykrywanie twarzy na podstawie konturów

Rozwiązanie jest oparte na sławnym algorytmie Viola Jones [VIOLA JONES] pozwalającym na rozpoznawaniu cech. Jedna ze względu na mnogość dostępnych cech (16000) wyszukiwanie każdej z nich na obrazie jest nie efektywne. Dlatego też ograniczenie ilości szczegółów na obrazie, pozwala na redukcję wyszukiwanych cech co wpływa pozytywnie na efektywność algorytmu.

Śledzenie konturów owalu twarzy, oraz kilku cech charakterystycznych w jego obrębie skutkuje otrzymaniem szybkiego algorytmu, o ograniczonej funkcjonalności.[Simple and Fast Face Detection System Based on Edges]



Rysunek (a) Obraz twarzy (b) Wyodrębnione krawędzie

## Dostępne rozwiązania

Na rynku istnieje wiele rozwiązań, oferujących gotowe aplikacje lub biblioteki zawierające zbiór funkcji niezbędnych w wykrywaniu interesujących cech. Wybrane rozwiązania wraz z krótkim opisem znajdują się w tabeli poniżej:

Tabela Porównanie istniejących systemów do analizy obrazów twarzy

|  |  |  |  |
| --- | --- | --- | --- |
| Nazwa | Opis | Platforma/technologie | Analiza w Czasie rzeczywistym |
| Lambda Labs – Face recognition | API pozwalające na wykrywanie twarzy, oczu oraz płci na zdjęciach | iOS, Google Glass | brak |
| FaceR | API oferujące wykrywanie struktur twarzy (obszar twarzy, oczy, nos usta) | Java, Ruby, PHP | brak |
| Sky Biometry | API pozwalające na wykrywanie twarzy, oczu, okularów(ciemnych lub jasnych) | Chmura | brak |
| Faceplusplus | Rozbudowane API oferujące wykrywanie oraz śledzenie twarzy. Oferuje również analizę uśmiechu, rozpoznawanie płci, wieku, rasy oraz rozpoznawanie twarzy. | Chmura, wszystkie platformy | tak |
| Eyedea recognition | RESTowe API oferujące wykrywanie twarzy na zdjęciach, wraz z segmentacją na poszczególne struktury. | Web | nie |
| Visage Face Track | System do śledzenia twarzy w 3D, udostępniający informację o przemieszczeniach i rotacji w formie SDK | C++ | tak |
| Faceware Analyzer | Producent podaje informacje, iż jest to najszybszy system śledzenia twarzy dostępny na rynku. Oferuje śledzenie twarzy, oczu (kierunek patrzenia), | PC | tak |

Warto zaznaczyć, że powyższe rozwiązania skupiają się głównie na rozpoznawaniu twarzy na obrazach. Istnieje wiele innych projektów rozwijanych pod kątem przetwarzania obrazów w czasie rzeczywistym, jednak ze względu na ich stan lub akademicką formę nie zostały ujęte w zestawieniu.

Istotnym faktem jest również udział wykorzystanych technologii w obecnych projektach. W większości przypadków implementowane jest wykrywanie twarzy z wykorzystaniem modelu. Pomimo dużego narzutu obliczeniowego, jest to najczęściej wybierana ścieżka rozwoju, ze względu na możliwości jakie oferuje.

# Wstęp teoretyczny

## 3.1 Wykrywanie obiektów przy pomocy algorytmu Viola-Jones i AdaBoost

Rozwiązanie opisane przez Paula Viola i Michaela Jones’a [**Rapid Object Detection using a Boosted Cascade of Simple Features**] przedstawia podejście wykorzystujące uczenie maszynowe. Pozwala ono na bardzo szybkie przetwarzanie obrazów, z zachowaniem wysokiej skuteczności rozpoznawania obiektów. Cały proces powstawania kaskad klasyfikatorów można podzielić na trzy etapy:

1. **Przedstawienie obrazu w postaci integralnej(ang. Integral image) – pozwala na bardzo szybkie przetwarzanie cech obrazów**
2. **Algorytm uczący, bazujący na AdaBoost - selekcja cech**
3. **Tworzenie bardziej złożonych klasyfikatorów w postaci kaskad.**

## 3.1.1 Cechy (ang. features)

Cechami w kontekście rozpoznawania obrazów twarzy nazywamy struktury, które pozwalają zidentyfikować twarz na obrazie. Twarz charakteryzuję się owalem, regionami oczu, ust, uszu, oraz bardziej szczegółowymi cechami jak tęczówka oka lub szpara ust. Tego typu struktury, na obrazie zostają przybliżone przy pomocy cech przypominających funkcje bazowe Haar’a. W pracy [adnotacja jak powyżej do Viola-Jones]wykorzystano trzy rodzaje cech.

* Cecha dwóch prostokątów (ang. two-rectangle feature) – różnica pomiędzy sumą pikseli wewnątrz dwóch prostokątnych regionów obrazu, gdzie, oba regiony posiadają taki sam rozmiar i kształt, oraz przylegają do siebie w pionie lub w poziomie
* Cecha trzech prostokątów (ang. three-rectangle feature) – suma pikseli dwóch zewnętrznych prostokątów odjęta od sumy pikseli regionu centralnego
* Cecha czterech prostokątów (ang. four-rectangle feature) – różnica sum regionów leżących na przekątnej.



Rysunek Typy cech poszukiwane na obrazie. Suma pikseli leżących wewnątrz białego prostokątu jest odejmowana od sumy pikseli wewnątrz prostokątu zacieniowanego. Cecha dwóch prostokątów (A) oraz (B), trzech prostokątów (C), oraz czterech prostokątów (D)

Zbiór wybranych cech wykorzystany w pracy [dalej viola jones] przedstawia rys. 5



Rysunek Cechy zastosowane w algorytmie Viola-Jone's

Aby uzmysłowić sobie jak naprawdę cechy mogą pomóc w rozpoznawaniu twarzy, należy przyrównać je do jej elementów. Rys. 6 prezentuje cechy naniesione na odpowiadające im specyficzne części twarzy. Można zaobserwować, że dobór cech jest dosyć intuicyjny, jeżeli analizowany obraz przedstawiony jest w skali szarości. Obszar oczu (oczodoły) jest ciemniejszy niż fragment twarzy pod oczami, stąd można go przybliżyć cechą dwóch prostokątów. Z kolei grzbiet nosa jest wyraźnie jaśniejszy od jego ścian, dlatego cecha trzech prostokątów pozwala rozpoznać ten fragment twarzy. Warto również zauważyć, że jakość obrazu na rys. 6 nie jest przypadkowa. Do analizy opartej na cechach, wcale nie jest potrzebny obraz o wysokiej rozdzielczości, ponieważ



Rysunek Cechy naniesione na odpowiadające im części twarzy.

## 3.1.2 Obraz integracyjny (ang. Integral Image)

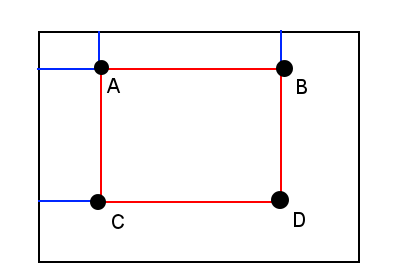
Integral Image to forma obrazu, której istotą istnienia jest możliwość bardzo szybkiego obliczenia wartości obrazu w danym punkcie. Piksel o współrzędnych (x, y) przechowuje wartość równą sumie wartości pikseli w fragmencie obrazu, którego lewy górny róg wyznaczany jest przez punkt (0, 0) natomiast prawy dolny róg znajduje się w punkcie (x, y).

Tak jak wspomniano powyżej, wartością punktu (x, y) jest suma wartości tablicy powyżej i na lewo od punktu (x, y), co przedstawia równanie(1).

Co więcej, integral image może zostać obliczony w trackie jednej iteracji po obrazie, korzystając z faktu, że jego wartość w punkcie (x, y) to:

Dzięki temu, obliczanie wartości dowolnego prostokątnego fragmentu obrazu, może być zrealizowane w stałym skończonym czasie z zaledwie czterokrotnym odwołaniem do obrazu. Korzystając z notacji, przedstawionym na rys. 7 suma prostokąta rozpiętego na punktach A, B, C oraz D to:

Gdzie:



Rysunek Sposób oznaczenia i obliczania sumy wartości pikseli obszaru

Dzięki zastosowaniu takiego formatu obrazu, wykorzystanie cech do rozpoznawania kształtu staje się rozwiązaniem bardzo wydajnym i skutecznym. Pomimo prymitywnej i ograniczonej budowy cech opartych na funkcjach Haara.

## 3.1.3 AdaBoost

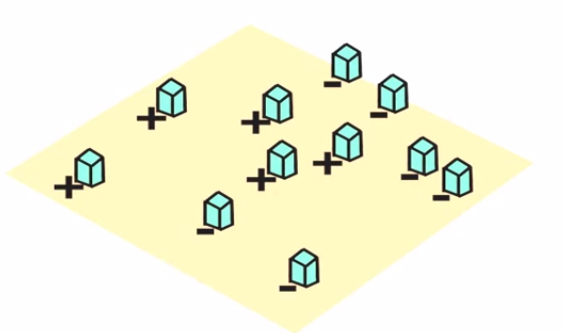
Boosting to nazwa dla metod służących zwiększaniu skuteczności dowolnego algorytmu uczenia. Ideą algorytmu AdaBoost jest tworzenie mocnego i złożonego (kaskadowego) klasyfikatora ze słabych i prostych klasyfikatorów.

Trzeba zdać sobie sprawę, że do dyspozycji mamy zbiór ponad 160000 cech. To niejednokrotnie więcej, niż liczba pikseli obrazu. Zadaniem boostingu, jest znalezienie tylko tych cech, które faktycznie przyczyniają się do zlokalizowania obiektu na obrazie, a następnie zbudowaniu z nich klasyfikatora, który skutecznie pozwoli odfiltrować obrazy, na których szukany element faktycznie się znajduje.

Gdzie: F(x) – silny klasyfikator, f(x) – słaby klasyfikator, α – waga

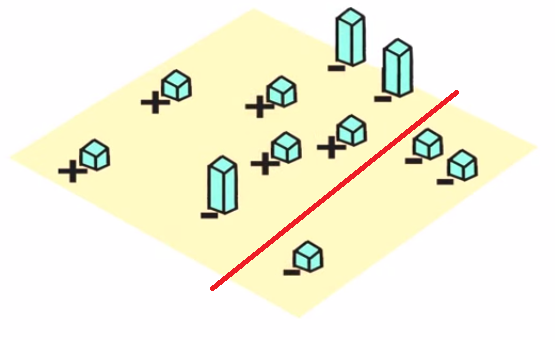
Klasyfikator jest rozważany, jako warty włączenia do kaskady, jeżeli jego skuteczność jest większa niż losowe oznaczanie obrazów zbioru uczącego jako zawierających szukany element, lub nie. Ponieważ jest to decyzja binarna, błąd nie może osiągać 50% względem zbioru uczącego.

Na początku, algorytm przypisuje równe wagi dla wszystkich przykładów ze zbioru uczącego, co przedstawia rys. 8.



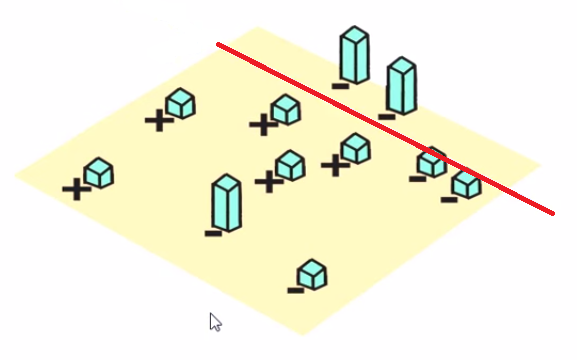
Rysunek Wagi przypisane do posczególnych przykładów zbioru uczącego. Niech "+" oznacza obraz zawierający twarz, "-" nie

Następnie wybrany klasyfikator jest stosowany dla wszystkich przykładów zbioru. Załóżmy że, dla wszystkich przykładów na lewo od czerwonej linii na rys. 9, klasyfikator uznał, że poszukiwana struktura znajduje się na obrazie. Jak widać, w zbiorze znajdują się również przykłady negatywne. Wówczas dla nich zwiększana jest waga. Ma to na celu, skupieniu większej uwagi kolejnych klasyfikatorów na obrazach które błędnie zostały oznaczone jako pozytywne.

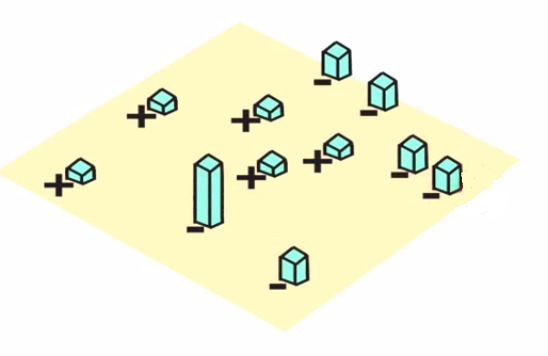


Rysunek Wynik pracy słabego klasyfikatora

W kolejnych iteracjach algorytm jest powtarzany, dla kolejnych klasyfikatorów. Zakładając, że inny klasyfikator skorygował część błędnych wskazań swojego poprzednika, ale popełnił swoiste błędy, co zostało zobrazowane na rys.10 ponownie zmienią się wagi dla poszczególnych przykładów rys.11 Przypomnieć w tym miejscu należy, że każdy z klasyfikatorów analizuje obrazu pod kątem indywidualnych cech.

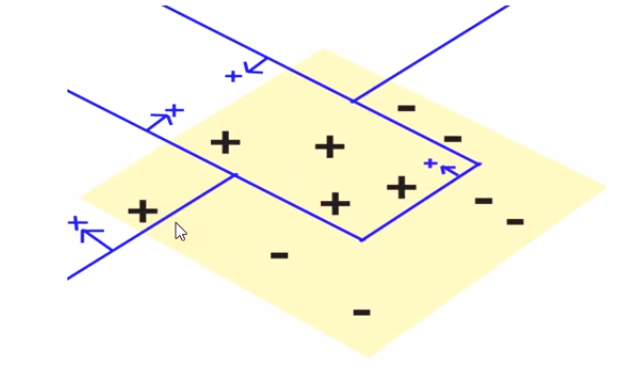


Rysunek Wynik dla kolejnego klasyfikatora



Rysunek Modyfikacja wag na podstawie wyników pracy dwóch klasyfikatorów

Ostatecznie, w wyniku analizy przeprowadzonej pod kątem wszystkich dostępnych cech, określony zostaje zbiór obrazów pozytywnych, oraz zbiór klasyfikatorów które przyczyniły się do wyodrębnienia takiego zbioru.



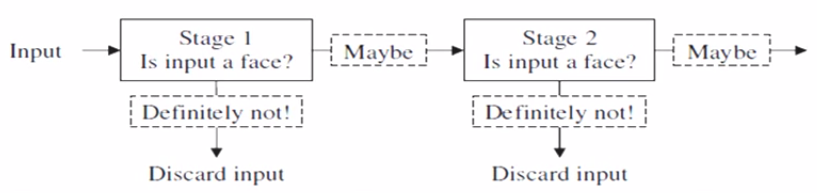
Rysunek Graficzna reprezentacja kombinacji klasyfikatorów wyselekcjonowanych podczas kolejnych iteracji algorytmu.

Wynik działania algorytmu można przedstawić przy pomocy równania:

()

Pomimo, że poszczególne klasyfikatory prawdopodobnie nie byłby w stanie poprawnie określić zbioru obrazów pozytywnych, ich liniowa kombinacja, przedstawiona równaniem(5) sprawia, że poprawnie interpretują zbiór. Co jednak równie ważne, ilość cech, kluczowych dla danej struktury poszukiwanej na obrazie zostaje drastycznie zredukowana. W [viola jones] wyselekcjonowano zbiór niezbędnych do wykrycia twarzy w płaszczyźnie czołowej. Ma to ogromny wpływ na wydajność algorytmu.

Ponieważ podstawowym założeniem algorytmu opisanego w [viola - jones] jest skanowanie detektorem tego samego obrazu, za każdym razem z inną wielkością okna, należy zwrócić uwagę, na fakt, że nawet przy wielu twarzach (lub innych poszukiwanych strukturach) duża część obrazu, podlegałaby zbędnej analizie, skoro wiadomo, że w pewnych obszarach twarz nie zostanie odnaleziona. Dlatego też, algorytm powinien skupiać się na fragmentach obrazu, które mają przesłanki o tym, że twarz faktycznie może się tam znajdować. Dlatego też, kaskadowość klasyfikatorów ma tak duże znaczenie. Każdy etap kaskady, zawiera silny klasyfikator(ogół słabych klasyfikatorów pogrupowany jest w kolejnych etapach). Zadaniem każdego z etapów jest określenie, czy w analizowanym fragmencie obrazu może znajdować się twarz. Jeżeli nie, algorytm natychmiast porzuca daną porcję obrazu, nie wykonując na niej analizy przy pomocy kolejnych poziomów kaskady. Taki mechanizm przerywania poszczególnych iteracji w możliwie najwcześniejszej fazie, wpływa pozytywnie na szybkość działania algorytmu.



Rysunek Uproszczony schemat działania algorytmu Viola-Jones

## 3.2 Śledzenie pozycji głowy

Wiele systemów rozpoznawania twarzy, czy też po prostu wchodzących w interakcję z z ciałem człowieka, narzuca wiele ograniczeń dotyczących pozycji. Często pojawiają się założenia, że twarz umiejscowiona jest na wprost kamery, a linia wzroku jest prostopadła do matrycy kamery. Z doświadczenia jednak łatwo stwierdzić, że utrzymywanie głowy w jednej pozycji jest trudne, niewygodne i przedewszystkim nienaturalne. Na ruch głowy wpływa wiele czynników, na które użytkownik może nie mieć wpływu. Wszelkie ruchy mięśni (oddychanie, ruchy związane z przemieszczaniem punktu skupienia wzroku) spotęgowane przez odległość twarzy od kamery, przekładają się na znaczne różnice w rejestrowanym położeniu.

Dlatego też, pozycja głowy jest ważną dziedziną, w której badania prowadzone są w ramach badań nad interakcją człowieka z komputerem (ang human – computer interfaces, HCI). Metody śledzenia pozycji głowy, można podzielić na dwie grupy:

* Śledzące cechy charakterystyczne twarzy
* Bazujące na modelu głowy

Model wykorzystany w jednej z metod pozwala na estymację pozycji głowy dzięki istnieniu zależności 2D do 3D między cechami, i za pomocą tych zależności i ich relacji wyznaczana jest pozycja.

Grupa metod oparta na cechach charakterystycznych twarzy zakłada, że istnieje relacja pomiędzy pozycją 3D głowy a własnościami obrazu twarzy. Zależności te, zostają wyznaczone w etapie treningowym, w którym sieć neuronowa zostaje wytrenowana na podstawie dużych zbiorów danych, zawierających zdjęcia twarzy w znanej pozycji.

Oba te rozwiązania oferują ciekawe możliwości, dlatego warto poddać je dyskusji i ocenie, co ma miejsce w kolejnym punkcie pracy.

## 3.2.1 Active Appearance Model

Aktywny Model wyglądu [Fully Automatic Real-Time 3D Object Tracking using Active Contour and Appearance Models] (AAM) to model statystyczny informujący o kształcie i teksturze. Algorytm ten pozwala stwierdzić gdzie i jak model jest zlokalizowany na obrazie.

Dlatego też, AAM wykorzystywany jest wi wielu dziedzinach skojarzonych z obrazami, od segmentacji obrazów medycznych aż po śledzenie głowy. Dzięki temu, że jest to algorytm oparty na danych – brak konieczności doboru parametrów, które określają działanie. Zamiast tego, algorytm automatycznie dostraja się do danych podczas inicjalizacji. Jest to wygodne z punktu widzenia użytkownika, jednak może to powodować wadliwe działanie. Przyczyną tego jest fakt, że algorytm bardzo silnie bazuje na zbiorze uczącym. Oznacza to, że w zastosowaniu wobec obrazów nie pochodzących ze zbioru uczącego jego skuteczność wyraźnie spada. Można temu zapobiec, dołączając indywidualne dane użytkownika do zbioru, jednak wówczas atut braku konieczności doboru parametrów jest niwelowany przez żmudny proces tworzenia danych do zbioru treningowego. Oczywiście jest możliwość automatyzacji nanoszenia danych na specyficzne obrazy, jednak są to zagadnienia wykraczające poza zakres tej pracy. Aby zrozumieć AAM warto poświęcić parę słów poszczególnym segmentom.

## Active Shape Model

Aktywny model kształtu [active shape models their training and application] to część AAM odpowiedzialna za model kształtu. Powstaje on poprzez kombinacje zmian kształtu. Do treningu algorytmu, wymagany jest zbiór obrazów z adnotacjami. Adnotacjami w tym wypadku są punkty w kluczowych, charakterystycznych miejscach twarzy, które pozwolą na wydobycie głównych cech. Bardzo ważnym założeniem jest fakt, że na wszystkich obrazach, adnotacje naniesione są w dokładnie tej samej kolejności. Oznacza to, że manualne nanoszenie landmarków na obrazy, jesteśmy narażenie na błędy, wynikające z bardzo dużego prawdopodobieństwa pomyłki kolejności lub lokalizacji punktów.



Rysunek Przykład obrazu z naniesionymi adnotacjami

Przykład zaprezentowany na rys.14 posiada 76 punktów charakterystycznych (ang. landmarks). Zdjęcie na rys.14 pochodzi ze zbioru danych MUCT, który oferuje niespełna cztery tysiące zdjęć wraz z adnotacjami. Na obrazach zaznaczone zostały: owal twarzy, kontur ust, nosa, brwi co pozwala na samodzielną pracę i eksperymenty w dziedzinie AAM.

Z wyżej wspomnianego zbioru, można wygenerować model zmian kształtu. Ponieważ adnotacje opisują kształt obiektu, w pierwszej kolejności należy dopasować wszystkie zbiory punktów do układu współrzędnych. Można tego dokonać przy pomocy analizy Prokrusta (postać z mitologii greckiej, zbójnik który, swe ofiary kładł na łóżku. Jeżeli jej ciało było za długie i wystawało poza ramę łóżka, obcinał je, jeżeli zaś za krótkie - naciągał). Jest to forma analizy kształtu, która przeprowadza dopasowanie poprzez translację, rotację i skalowanie obiektów. Zbiór uczący zapisany jest w formie wektorów zawierających współrzędne charakterystycznych punktów. Dane dotyczące rozmiaru i położenia obiektów zostały wyeliminowane z wektora dzięki zastosowaniu analizy Prokrusta.

Matematyczny zapis kształtu s, w postaci wektora 2n wymiarowego.

Zbiór danych uczących jest podawany normalizacji, a następnie wyznaczane się jego statyczne

parametry przy użyciu analizy głównych składowych (PCA). Do analizy głównych składowych

konieczne jest wyliczenie wartości oraz wektorów własnych macierzy kowariancji *C*, gdzie przez *N*

oznacza się ilość obrazów w zbiorze trenującym.

()

PCA to liniowa transformacja, która przekształca do nowego układu współrzędnych, używane są wyłącznie wektory własne odpowiadające największym wartościom własnym. Ilość wektorów zastosowanych zależy od zróżnicowania danych wejściowych. Jest to metoda służąca do redukcji wymiarowości zbioru danych.

## Statyczny Model Tekstury

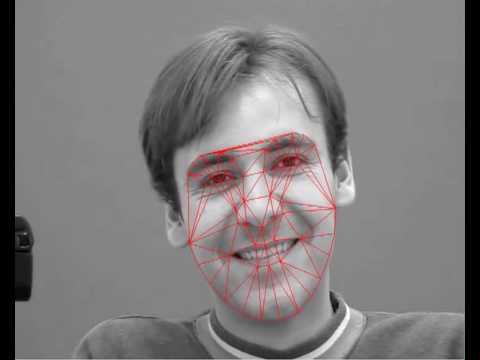
Tekstura powstaje na podstawie zbioru uczącego z naniesioną na obrazy siatką 2d na twarzy. Siatka powstaje dzięki zastosowaniu metody triangulacji Delone [MASTERING OPENCV]na zbiorze punktów charakterystycznych.

Triangulacja Delone maksymalizuje minimalne kąty w trójkątach ponieważ podczas w czasie triangulacji często powstają cienkie trójkąty o bardzo ostrym kącie między bokami.



Rysunek przykład triangulacji

W przypadku modelu tekstury twarzy wygląda to tak, jak pokazano na rys. 16

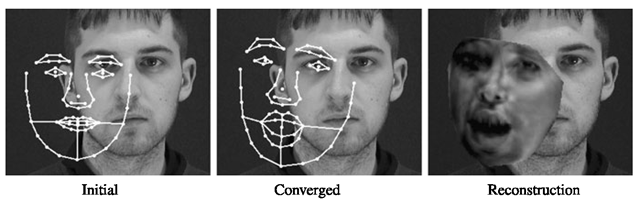


Rysunek Twarz z naniesioną siatką 2D

Otrzymany model tekstury opisuje różnice wyglądu między obrazami wejściowymi. Jednak aby istniała możliwość utworzenia takiego modelu, fragmenty twarzy na obrazach zbioru uczącego powinny się pokrywać. Można to osiągnąć stosując deformację kształtów poszczególnych obrazów twarzy na kształt główny. Transformacja zbioru uczącego do wspólnego kształtu jest wyznaczana stosując mapowanie afiniczne poszczególnych trójkątów siatki twarzy.

AAM jest często wykorzystywanym podejściem w określaniu pozycji 3D głowy. Idea opera się na stworzeniu modelu 2D twarzy, poprzez trening przygotowanymi obrazami, z naniesionymi punktami, tworzącymi siatkę 2D w obrębie obrysu twarzy. Na podstawie odkształceń 2D można odkreślić przybliżoną pozycje 3D.

Metoda jest skuteczna, w przypadku gdy twarz, którą chcemy wykryć znajduje się w zbiorze treningowym. Najlepszym rozwiązaniem byłoby, gdyby cały zbiór uczący składał się jedynie ze z obrazów ściśle związanych ze strukturą, która później jest śledzona. Niestety, jeżeli model tworzony jest dla szerokiej grupy osób (wiele odbiegających od siebie cech), dochodzi do niepoprawnego dopasowania cech i dosyć szybkiego rozsynchronizowania. Ponadto, konieczność nanoszenia cech ręcznie na obraz, sprawia że jest to rozwiązanie pracochłonne i nie generyczne.



Rysunek Błąd dopasowania cech do obrazu.

W przypadku tej pracy, autorowi nie udało się uzyskać zadawalających wyników z wykorzystanie samego algorytmu AAM, na podstawie własnego zbioru uczącego, jak i również z wykorzystaniem dostępnym open source’owo baz.

## 3.2.2 Pose from Orthography and Scaling with Iteration

Zadaniem algorytmu jest obliczenie macierzy rotacji oraz wektora translacji obiektu. Macierz rotacji R obiektu, składa się z wierszy, które zawierają współrzędne wektorów jednostkowych i, j, k układu współrzędnych kamery, wyrażonych w układzie współrzędnych obiektu(). Zadaniem macierzy R jest transformacja współrzędnych wektora obiektu np. do współrzędnych zdefiniowanych w układzie współrzędnych kamery. Wynik działania: czyli iloczyn pierwszego wiersza macierzy R i wektora współrzędnych obiektu, to rzutowanie wektora na wektor jednostkowy układu współrzędnych kamery.



Rysunek Rzutowanie punktów obiektu

Macierz rotacji wygląda następująco:

Gdzie to współrzędne wektora i w układzie współrzędnych obiektu.

Aby obliczyć rotację potrzeba wyliczyć jedynie i oraz j w układzie współrzędnych obiektu. Wektor k można wówczas otrzymać z zależności: i x j. Translacja wektora T, to wektor , pomiędzy środkiem rzutu O i punktem . Jeżeli punkt jest widocznym na obrazie punktem charakterystycznym, to translacja wektora T jest równoległa do wektora , i jest równa:

Aby określić pozycję obiektu musimy jednak określić jedynie współrzędna , co wraz z da nam jednoznacznie zdefiniowaną pozycję.

## Scaled Orthographic Projection and Perspective Projection

Skalowany perspektywiczne rzutu (SOP) jest przybliżeniem prawdziwego rzutu. W przybliżeniu tym zakładamy, że Punkty obrazu *Mi*  znajdują się w tej samej odległości od kamery, a różnica rozmiaru obiektu względem dystansu obiektu od kamery jest pomijalna. Oznacza to, że punkty obiektu znajdują się na tyle daleko od kamery, że można pominąć różnicę głębi. Można to zaobserwować na rys. 18 gdzie odległość od punktu . W rzucie perspektywicznym obraz punktu obiektu, to punkt na płaszczyźnie G obrazu, który ma współrzędne:

Natomiast punkt mi rzutu, można uzyskać dzięki zależności:

Współczynnik , jest miarą skalowania rzutu. Punkt posiada taki sam obraz jak punkt o współrzędnych , w SOP. Współrzędne punktu mogą zostać zapisane również w postaci:

## 3.2.4 Geometria SOP

Geometria rzutu perspektywicznego jest przedstawiona na rys. 18. Zazwyczaj perspektywa punktu mi jest miejsce przecięcia się prostej na której leży pnkt oraz płaszczyzny G. W SOP jest to płaszczyzna K, równoległa do płaszczyzny G. Płaszczyzna K znajduje się w odległości od środka projekcji. Punkt jest rzutowany na płaszczyznę K w punkcie , poprzez rzutowanie prostokątne. Następnie Punkt Pi jest rzutowany na płaszczyznę G w punkcie poprzez rzut perspektywiczny. Wektor jest równoległy do wektora i przeskalowany przez wartość współczynnika s równanie 8. Określa on po prostu proporcjonalność pomiędzy dwoma wektorami.

Rozważyć należy równania, które charakteryzują prawdziwą perspektywę i odnoszą się do nieznanych wierszy macierzy rotacji oraz nieznaną współrzędną wektora translacji, w kontekście znanych współrzędnych wektora w układzie współrzędnych obiektu, oraz znanych współrzędnych, punktów i . Rozwiązanie tych równań dostarcza informacji o położeniu obiektów.

Gdzie to:

Gdzie

Na podstawie rys. 18 rozważmy dwa punkty obiektu – i oraz płaszczyznę K równoległą do płaszczyzny obiektu. Na prostej, na której leży punkt przecinającej płaszczyznę K w punkcie oraz rzut na płaszczyznę K w - wektor jest sumą trzech wektorów:

Wektor oraz jego obraz m0mi są proporcjonalne w stosunku . Dwa wektory oraz Cm są również proporcjonalne w dwóch jednakowych trójkątach oraz w stosunku równym do współrzędnej z dwóch pozostałych wektorów tworzących trójkąt . Stosunek ten wynosi . Suma tych trzech wektorów może być wyrażona jako:

Następnie obliczamy Iloczyn skalarny równiania 13 i wektora jednostkowego i układu współrzędnych kamery. Iloczyn skalarny równy jest zero. Iloczyn to współrzędna x, wektora natomiast iloczyn to współrzędna iksowa.

Rozważmy punkty , rzut punktu czyli na płaszyźnie K, oraz jego obraz , którego współrzędne oznaczamy . Wektor jest sumą dwóch wektorów i . Pierwszy z nich, , i jego obraz są proporcjonalne w stosunku . Dlatego:

Iloczyn skalarny tego wektora, oraz wektora jednostkowego *i* oraz iloczyn skalarny wektora oraz  *i* są równe zero, natomiast *i*  jest współrzędną x równą – wektora . Stąd otrzymujemy:

Oraz analogicznie:

Porównując te równania z równaniami 9,10 zauważamy, że współrzędne punktu mogą być wyrażone jako:

,

Równania 9 i 10 można również zapisać jako

Gdzie:

Podstawowym założeniem metody jest, że wartości równania 16,17 dostarczają liniowych równań, w których jedynymi niewiadomymi są współrzędne I, J. Gdy I, J zostaną obliczone, i oraz j są uzyskiwane poprzez normalizację I,J. również jest uzyskiwane na podstawię normalizacji wspomnianych wektorów. Jak już wcześniej wspominano ta część algorytmu, rozwiązująca liniowe równania nazywana jest POS. W praktyce POS znajduje pozycję obiektu w oparciu o stałe wartości .  
Wynikiem działania POS jest jedynie przybliżona pozycja obiektu, jeżeli wartości nie są dokładne. Jednak gdy nieznane wartości i oraz j zostaną obliczone, nowe, bardziej dokładne wartości mogą zostać obliczone ponownie, przy użyciu nowych, bardziej precyzyjnych danych. Stąd nazwa – iteracyjny algorytm POSIT (POS with Iterations). Ta część algorytmu sprawia, że wartości i, j oraz zbliżają się do właściwiej pozycji obiektu.

W trakcie inicjalizacji . Z tego założenia wynika, że a i są sobie równe. Obrazek 19 opisuje tą konfigurację wektorów pomiędzy dwoma punktami.



Rysunek 19 Konfiguracja wektorów w czasie inicjalizacji

W POSIT opisanym powyżej, należy rozwiązać równania 16, 17. Można je zapisać w skróconej formie:

Gdzie:

,

Stosując równanie 16 dla n punktów , , …. oraz ich obrazów mi tworzymy zbiór nieznanych wektorów I, J.

Gdzie A, to macierz współrzędnych punktu , x’ jest wektorem i-tej współrzędnej a y’ . Ponieważ podstawą do wyznaczenia pozycji, jest uzyskanie trzech punktów nie współpłaszczyznowych, macierz a posiada wymiar 3x3,

Gdzie B jest macierzą odwrotną macierzy A.

Znając geometryczny rozkład punktów cech , macierz B można uzyskać poprzez operację lub poprzez dekompozycję macierzy A przez Rozkład Według Wartości Osobliwych (ang. Single Value Decomposition SVD).

Gdy znamy wartości I oraz J, wektory jednostkowe i, j są otrzymywane przez normalizację I oraz J, jak wspominano wcześniej. Trzy elementy pierwszego wiersza macierzy, to współrzędne wektora i, trzy elementy kolejnego wiersza to to współrzędne wektora j, oraz trzy elementy trzeciego wiersza to współrzędne wektora k, przy czym otrzymywane są one z wykorzystaniem wektorów i, j jak wspominano wcześniej. Gdy posiadamy powyższe dane, można uzyskać wektor translacji. Jest to wektor pomiędzy środkiem rzutu, a punktem a początkiem układu współrzędnych obiektu.

Gdy posiadamy macierze rotacji i translacji, możemy określić położenie obiektu, następnie może ono zostać doprecyzowane w kolejnych iteracjach.

## Podsumowanie

Podsumowując ten długi i obszerny rozdział, autor chciałby przypomnieć podstawę działania, i zadania, które mają spełniać przedstawione algorytmy.

Twarz na obrazie zostaje wykryta dzięki kaskadom klasyfikatorów, przeszukujących obraz w poszukiwaniu cech charakterystycznych. Powstają one w procesie uczenia, w oparciu o duże zestawy obrazów wzorcowych zawierających, oraz nie zawierających szukany element.

Aktywny Model Kształtu (AAM) potrafi śledzić i rozpoznawać kształty w oparciu o modele kształtu i tekstury – które powstają podobnie w przypadku algorytmu Viola – Jones, w oparciu o zbiór danych. Dzięki temu nie ma potrzeby konfiguracji, a algorytm sam dopasuje się do obrazu podczas inicjalizacji. Pomimo silnej zależności działania, od zbioru uczącego, algorytm posiada szeroki wachlarz zastosowań – od obrazowania medycznego do rozpoznawania włącznie. Do budowy modelu wykorzystywana jest metoda składowych głównych (PCA).

POSIT to algorytm który pozwala na estymację pozycji głowy w 3D, poprzez określenie macierzy rotacji i wektora translacji. Aby uzyskać pozycję konieczne jest wyznaczenie co najmniej czterech punktów, które nie leżą na jednej płaszczyźnie i należą do powierzchni obiektu. Działanie algorytmu składa się na dwie części – wstępnej estymacji pozycji (POS) oraz drugiej w której dochodzi do iteracyjnej poprawki wyniku.

# Opis własnego rozwiązania

## 4.1 Detekcja twarzy i oczu

 Pierwszym krokiem, oraz podstawą działania algorytmu, jest wykrycie na obrazie twarzy, a następnie oczu. W tym celu wykorzystano metodę opisaną w pracy {VIOLA JONES}, o której wspomniano również w rozdziale 3.1. Jest to najpopularniejsza metoda, wykorzystywana do wykrywania twarzy. Jej zaletami są szybkość działania, oraz skuteczność, chociaż oczywiście nie jest pozbawiona wad. Jak wspomniano w rozdziale 3.1 algorytm wykorzystuje klasyfikatory, w oparciu o które, następuje detekcja. W pierwszej kolejności wykrywany jest obszar twarzy na obrazie. Dzięki temu, z całego obrazu wyodrębniona zostaje część, która może zawierać inne cechy, istotne dla działania programu. Ma to duże znaczenie dla wydajności i skuteczności działania algorytmu. Dzięki zawężeniu obszaru, kolejne klasyfikatory – w tym wypadku poszukujące oczu na obrazie, nie musze przeszukiwać wszystkich pikseli obrazu, a jedynie obszar wyodrębniony przez pierwszy stopień kwalifikacji. Jednak oznaczenie obszaru twarzy na obrazie, nie jest jednoznaczne z tym, że faktycznie twarz się na ni znajduje. Wpływ oświetlenia jest na tyle duży, że może dawać fałszywe wyniki. Przykład niekorzystnego oświetlenia, które na pewno zaburzy ocenę przedstawiono na rys. 20.

Rysunek 20 Boczne oświetlenie twarzy, nnegatywnie wpływa na skuteczność klasyfikatorów

Dlatego też, oprócz tego, w wyodrębnionym obszarze obrazu poszukiwane są oczy. Jak już wspomniano, takie rozwiązanie wpływa pozytywnie na szybkość działania i pozwala wyeliminować błędy, które mogłyby powodować wykrycie poszukiwanych struktur rozsianych po obrazie, niekoniecznie mających związek z twarzą. Aby usprawnić działanie, i zaoszczędzić czas, zdecydowano się na przeszukiwanie tylko górnej połowy obrazu. Ma to związek ściśle z budową ludzkiej twarzy. Oczy muszę znajdować się powyżej nosa i ust (oczywiście w neutralnej, pionowej pozycji głowy). Dzięki temu, wiemy, że przeszukiwanie całego fragmentu obrazu jest zbędne. Jeżeli w górnej połowie obrazu, zostało wykryte oboje oczu, będących na tym samym poziomie, można stwierdzić że, twarz faktycznie została wykryta. Ponadto twarz została podzielona na lewą i prawą stronę, co będzie wykorzystane w dalszej części pracy.



Rysunek 21 Przykład działania algorytmu, z niejednolitym tłem

### Przebieg algorytmu

1. Przeszukanie pobranego obrazu pod kątem wykrycia twarzy, przy pomocy kaskadowych klasyfikatorów, efektem jest znalezienie wszystkich potencjalnych twarzy na obrazie. Odrzucone zostają wszystkie, poza największą czyli potencjalnie osoby siedzącej najbliżej kamery.
2. Wyodrębniony obszar w punkcie 1 zostaje poddany analizie przy użyciu kolejnego klasyfikatora, tym razem wyszukującego obszar oka. Aby ograniczyć zapotrzebowanie na zasoby, przeszukuje górną połowę wyodrębnionego obrazu.
3. Analiza wzajemnego położenia oczu i twarzy, ocena czy na obrazie faktycznie znajduje się twarz, podzielenie twarzy na część lewą i prawą.



Schemat Algorytm wykrywania twarzy i nosa.