

Data przekazania opiekunowi: 26.01.2017

Konspekt pracy magisterskiej

Inż. Dorian Krefft

Tytuł pracy dyplomowej: Licznik uśmiechów

SPIS TREŚCI

Spis treści	2
1. Motywacja i kontekst pracy.....	3
2. Cel pracy.....	3
3. Planowany zakres pracy.....	3
4. Źródła informacji	4
5. Podstawy teoretyczne, metodologie, standardy	4
6. Proponowane środki implementacyjne i narzędziowe oraz ich omówienie.....	5
6.1. Baza danych	5
6.2. Aplikacja desktopowa	5
6.3. Aplikacja webowa	6
6.4. Narzędzie wersjonowania kodu	6
6.5. Narzędzie do tworzenia backlogu.....	6
7. Proponowany spis treści.....	7
8. Plan prac i harmonogram	7
9. Potencjalne zastosowania pracy	8
10. Analiza ryzyka.....	8

1. MOTYWACJA I KONTEKST PRACY

Tematyka pracy dotyczy problemu rozpoznawania emocji u użytkownika systemu informatycznego, który może być rozwiązany przy pomocy różnych sposobów i polega na badaniu reakcji użytkownika na konkretną sytuację (np. tempo ruchów myszki czy kliknięć). Tematem pracy jest liczenie uśmiechów użytkownika, w związku z czym zdecydowano skupić się na analizie obrazu z kamery podłączonej do komputera.

Koncepcja rozpoznawania emocji użytkownika na podstawie obrazu z zainstalowanej kamery w jego systemie nie jest nowa – na rynku istnieje dużo bibliotek oferujących mechanizmy rozpoznawania twarzy i emocji. Problem wykrywania uśmiechu w czasie rzeczywistym wymaga jednak szybkiego działania algorytmu, aby aplikacja działała płynnie i stabilnie.

Motywacją do pojęcia tego tematu jest zainteresowanie tematyką algorytmów rozpoznawania twarzy oraz chęć rozwinięcia swojej wiedzy w tym temacie. W związku z taką motywacją należy także przeprowadzić rozpoznanie istniejących sposobów wykrywania uśmiechu w czasie rzeczywistym, a następnie porównanie ich ze sobą i wybranie najlepiej nadającego się do rozwiązania problemu.

2. CEL PRACY

Celem niniejszej pracy dyplomowej jest utworzenie aplikacji zliczającej liczbę uśmiechów w ciągu dnia, miesiąca oraz od początku jej instalacji. Aplikacja powinna w czasie rzeczywistym analizować obraz z kamery podłączonej do stacji, na której została zainstalowana, a następnie w wyniku analizy wykryć twarz i określić, czy użytkownik się uśmiechnął. Ponadto, różne instancje aplikacji powinny posiadać wspólną bazę danych, co umożliwiłoby zliczanie wszystkich uśmiechów, jakie zostały wykryte.

Produkt powstały podczas tworzenia pracy dyplomowej ma spełnić rolę przede wszystkim zapewnienia rozrywki oraz prezentować możliwości nowoczesnych algorytmów do wykrywania twarzy. Kolejną ważną potrzebą jest zapewnienie rozwoju i zachęcenie do uśmiechania się dzieci z autyzmem – dzieci takie często mają problemy z prawidłowym oddawaniem emocji.

3. PLANOWANY ZAKRES PRACY

Podstawowym zakresem pracy jest realizacja programu zliczającego liczbę uśmiechów w zadany okresie czasu, w czasie rzeczywistym, a także zapisywanie jej w bazie danych, celem złączenia statystyk z kilku instancji aplikacji. Program taki powinien być możliwy do uruchomienia na komputerze, który posiada interfejs graficzny i dostęp do kamery.

Aplikacja powinna być w dużym stopniu konfigurowalna. Podstawowym kryterium konfigurowalności jest możliwość wprowadzenia nazwy komputera, z którego zliczane będą

uśmiechy oraz mechanizm pozwalający na zamieszczenie krótkiego tekstu, widocznego przez cały czas na ekranie podczas pracy programu.

Dodatkowym wymogiem stawianym przez opiekuna jest występowanie na ekranie z obrazem z kamery elementów skupiających uwagę osoby korzystającej z aplikacji. Elementy takie powinny zachęcać użytkownika do uśmiechania się – po wykryciu uśmiechu aplikacja powinna uruchomić proste animacje.

W związku z powyższymi wymaganiami podstawowym zakresem pracy jest przeprowadzenie rozpoznania algorytmów i bibliotek oferujących rozpoznawanie uśmiechu na obrazie z kamery podłączonej do stacji komputerowej. Po rozpoznaniu należy następnie wybrać bibliotekę sprawdzającą się najlepiej do postawionego zadania, a następnie wykorzystać ją w trakcie implementacji aplikacji.

Opcjonalnym zakresem pracy jest utworzenie aplikacji webowej, która oferuje dostęp do mechanizmów wykrywania uśmiechu w czasie rzeczywistym. Obraz powinien być pobierany z komputera klienta, a następnie przesyłany na serwer, który w odpowiednio krótkim czasie zajmie się wykryciem uśmiechu, a następnie zwróci informację, czy udało się go wykryć. Aplikacja webowa może też służyć za źródło zbieranych danych do przyszłych prac dyplomowych o podobnej tematyce – po uprzednim wyrażeniu zgody przez użytkownika, obraz z kamery z wykrytym uśmiechem zostałby zapisany w bazie danych.

Weryfikacja programu odbędzie się przy pomocy opiekuna oraz osób, które wyrażą chęć pomocy przy testowaniu.

Po zakończeniu etapu implementacji i weryfikacji należy także zająć się wdrożeniem aplikacji w miejscach wskazanych przez opiekuna.

4. ŹRÓDŁA INFORMACJI

Głównym źródłem informacji wykorzystywanym w pracy będą dokumentacje bibliotek oraz opisów algorytmów przez nich stosowanych. W rozwiązywaniu problemów implementacyjnych pomocne okażą się serwisy zrzeszające programistów (np. *StackOverflow*), oparte na zasadzie pytanie-odpowiedź.

Dodatkowym źródłem informacji może być kontakt ze specjalistą w dziedzinie rozpoznawania emocji – opiekunem projektu, dr inż. Agnieszką Landowską.

Pomocna może okazać się także praca dyplomowa Pawła Blumy (*Rozpoznawanie emocji na podstawie obrazu z kamery*), w której zawiera się szczegółowy opis algorytmów służących do rozpoznawania emocji na podstawie obrazu ludzkiej twarzy. W szczególności zawiera ona informacje o odczuwaniu radości, której głównym sygnałem jest uśmiech.

5. PODSTAWY TEORETYCZNE, METODOLOGIE, STANDARDY

Nie dotyczy.

6. PROPONOWANE ŚRODKI IMPLEMENTACYJNE I NARZĘDZIOWE ORAZ ICH OMÓWIENIE

Jednym z kryteriów doboru narzędzi jest ich dostępność, uniwersalność, nowoczesność oraz darmowość. Autor pracy stara się dobrać narzędzia w taki sposób, by powstały produkt nie był oparty o przestarzałe technologie, co ma umożliwić ewentualną rozbudowę produktu w przyszłości.

6.1. Baza danych

Dane zbierane przez aplikację nie są skomplikowane i nie wymagają złożonych mechanizmów, które oferują rozbudowane systemy zarządzania bazą danych. W związku z takim spostrzeżeniem, zdecydowano się zaimplementować warstwę danych przy pomocy bazy danych *no-sql* (not only SQL), *MongoDB*.

Jest to system zapewniający zaawansowane mechanizmy zapisywania danych w bazie danych w postaci dokumentów zbudowanych przy pomocy formatu *JSON*. Takie podejście umożliwia aplikacjom łatwiejsze przetwarzanie danych, zwłaszcza że format ten zyskuje rosnącą popularność ze względu na niewielki narzut. Brak wbudowanych mechanizmów tworzenia relacji pomiędzy różnymi dokumentami nie jest problemem w przypadku zakładanego zakresu projektu – dane przechowywane w bazie można podzielić na proste obiekty, przechowujące podstawowe informacje o dacie, nazwie komputera oraz ew. ramkę z obrazu kamery, na której wykryto uśmiech. Taka ilość informacji sprawia, że mechanizmy relacyjności i transakcyjności nie są potrzebne i wydają się być zbędnym narzutem.

Dodatkowym atutem przemawiającym za wykorzystaniem *MongoDB* jest fakt, że autor pracy posiada podstawową jego znajomość, co pozwala na uniknięcie czasu spędzonego na nauce narzędzia.

6.2. Aplikacja desktopowa

Aplikacja desktopowa powinna posiadać prosty i przejrzysty interfejs. Ponieważ autor pracy posiada największe doświadczenie w programowaniu w *Java 8*, zdecydowano się na wybranie tego języka do implementacji, aby uniknąć narzutu konieczności zapoznania się z frameworkami do pisania aplikacji desktopowych w innych językach.

Kolejnym atutem wybrania takiego języka jest możliwość uruchamiania aplikacji na systemach operacyjnych wspierających *Java JRE*, a zatem na większości najpopularniejszych systemów.

Jak wynika z wstępnego rozpoznania, język ten umożliwia wykorzystanie szybkich i stabilnych bibliotek do wykrywania emocji z obrazów – większość z nich nie jest jednak napisana w *Javie*, ale w *C*, dzięki czemu problem mniejszej wydajności programów napisanych przy jej użyciu nie powinien okazać się uciążliwy w pracy z aplikacją.

Podczas pracy z aplikacją desktopową należy rozważyć wydzielenie usług zapewniających wykrywanie obrazu, w celu ich współdzielenia z aplikacją webową. Dzięki temu znacząco zredukuję się czas potrzebny na jej wytworzenie.

6.3. Aplikacja webowa

Aplikacja webowa powinna zostać napisana w prostej architekturze klient-serwer i umożliwiać podstawowe mechanizmy wykrywania uśmiechu. Dodatkową możliwością przez nią oferowaną może być zbieranie (za zgodą użytkowników) obrazów, na których został wykryty uśmiech, celem ich wykorzystania w późniejszych pracach dyplomowych.

Stronę klienta zdecydowano się zrealizować przy użyciu technologii znanej już w stopniu bardzo dobrym autorowi – *AngularJS*. Jest to framework napisany przy użyciu języka *Javascript*, oferujący zaawansowane mechanizmy do tworzenia aplikacji typu *SPA* (Single Page Application). Kolejnym atutem wykorzystania *AngularJS* jest fakt, że oferuje on gotowe moduły służące do komunikacji z serwerem i pobierania obrazu z kamery użytkownika - jest bardzo popularną technologią, do której istnieje wiele poradników i gotowych komponentów.

Stronę serwera zdecydowano się napisać przy użyciu technologii *Java EE*. Skompilowaną aplikację można uruchomić na serwerze *WildFly 10*, który jest kontenerem aplikacji, oferującym wiele zaawansowanych i wbudowanych mechanizmów.

Do komunikacji klienta z serwerem zdecydowano się wykorzystać endpointy REST-owe, które będą umożliwiały przesyłanie żądań do usług wykrywających uśmiech. Dodatkowo, przy użyciu narzędzia *Swagger*, wszystkie endpointy będą na bieżąco dokumentowane oraz możliwe do wywołania przy użyciu prostego interfejsu z poziomu przeglądarki. Znacząco pomoże to w przeprowadzaniu prostych testów aplikacji.

Do tworzenia wersji release aplikacji webowej zostaną wykorzystane technologie *NodeJS* oraz *Grunt* lub *Gulp*, w zależności od tego, które z nich okaże się bardziej przydatne.

6.4. Narzędzie wersjonowania kodu

Do wersjonowania kodu aplikacji zdecydowano się wykorzystać repozytorium *Git*, stworzone w serwisie *GitHub*, ze względu na fakt, że jest to obecnie najbardziej popularne narzędzie wersjonujące. Repozytorium stworzone w serwisie oznaczone jest jako prywatne i zawierać będzie dokumentację projektu oraz cały kod.

6.5. Narzędzie do tworzenia backlogu

Produkt będzie powstawał w procesie iteracyjnym i przyrostowym. Do tworzenia zadań oraz błędów zdecydowano się wykorzystać system *YouTrack*, który oferuje proste mechanizmy do zarządzania przebiegiem prac w projekcie. Bardziej skomplikowany system nie wydaje się być potrzebny.

7. PROPONOWANY SPIS TREŚCI

1. Streszczenie i Abstract
2. Wstęp
3. Rozpoznawanie twarzy i emocji
Dział powinien zawierać informacje o tematyce rozpoznawania twarzy i emocji. Podanie algorytmów realizujących ten cel.
4. Problem i koncepcja rozwiązania
Zawiera podstawowe informacje o planowanym sposobie rozwiązania problemu rozpoznawania uśmiechu oraz implementacji aplikacji.
5. Przegląd bibliotek do rozpoznawania emocji
W tym rozdziale znajdzie się lista rozpoznanych bibliotek do rozpoznawania emocji wraz z wyszczególnieniem ich wad oraz zalet w kontekście zastosowania w projekcie.
6. Wybrana biblioteka do rozpoznawania emocji
Zawiera informację o wybranej bibliotece do rozpoznawania emocji wraz z podaniem powodów, dla której zdecydowano się wykorzystać właśnie ją.
7. Implementacja rozwiązania
Dział dotyczy informacji na temat zaimplementowanego rozwiązania problemu – ogólne opisanie mechanizmów, które zostały zrealizowane w ramach pracy dyplomowej. Dział powinien zawierać także informacje o sposobach uruchamiania aplikacji (webowej, desktopowej) oraz sposobie łączenia się z bazą danych. Dodatkowo dział pełni rolę dokumentacji do sposobu konfigurowania programów.
8. Ewaluacja rozwiązania
Dział obejmuje proces wdrożenia aplikacji w ośrodku dla dzieci z autyzmem. Dodatkowo, powinna tu być zawarta opinia osób korzystających z aplikacji oraz opiekuna pracy.
9. Podsumowanie
Krótkie podsumowanie pracy, tego co zostało zrealizowane oraz tego, w jaki sposób produkt spełnia oczekiwania stawiane na początku projektu.

8. PLAN PRAC I HARMONOGRAM

Plan prac zakłada rozpoznanie bibliotek do rozpoznawania uśmiechu ludzkiej twarzy, na podstawie obrazu z kamery. Następnie należy wybrać bibliotekę najlepiej sprawdzającą się w zastosowaniu oraz zaimplementować mechanizm liczenia uśmiechów w aplikacji desktopowej. Należy także ocenić postęp prac i zdecydować, czy zostanie implementowana także aplikacja w wersji webowej – harmonogram zakłada, że w przypadku braku niespodziewanych opóźnień, taka wersja aplikacji powstanie.

1. Początek semestru zimowego:
 - określenie zakresu pracy
 - ustalenie wymagań z opiekunem
2. Koniec semestru zimowego:
 - przeprowadzenie rozpoznania bibliotek oferujących rozpoznawanie emocji
 - porównanie bibliotek i wybór odpowiedniej do zastosowania w pracy
3. Początek semestru letniego:
 - rozpoczęcie prac implementacyjnych aplikacji desktopowej
 - wykorzystanie wybranej biblioteki i implementacja mechanizmów wykrywania uśmiechu
 - decyzja o realizacji aplikacji webowej
4. Koniec semestru letniego:
 - zakończenie prac nad implementacją aplikacji desktopowej
 - zakończenie prac nad ew. implementacją aplikacji webowej
 - regularna walidacja i weryfikacja rozwiązania
5. Okres wakacyjny:
 - ewaluacja rozwiązania
6. Początek Września:
 - obrona pracy

9. POTENCJALNE ZASTOSOWANIA PRACY

Ze względu na możliwość zbierania obrazów z wykrytym uśmiechem, praca ma potencjał zostać wykorzystana w przyszłych projektach o tematyce wykrywania emocji na twarzy człowieka.

Dodatkowo, zgodnie z celem pracy, ma ona posłużyć dzieciom z autyzmem do zachęcenia ich do wyrażania emocji.

10. ANALIZA RYZYKA

Zakres projektu wydaje się być realizowalny w wyznaczonym terminie.

Głównym ryzykiem jest problem z wydajnością aplikacji, który sprawi, że praca z nią będzie przebiegać mało sprawnie. Może się tak stać przy wyborze nieoptymalnych algorytmów, które nie poradzą sobie z przetwarzaniem obrazu w czasie rzeczywistym. Aby temu zapobiec, należy wybrać najbardziej wydajną bibliotekę w trakcie rozpoznawania, a także na bieżąco testować wydajność aplikacji. Przydatnym sposobem do walki z tym problemem może się okazać także odpowiedni dobór liczby klatek na sekundę, które będzie analizował algorytm – powinna to być jednak na tyle duża liczba klatek, by potrafiła poradzić sobie z w miarę szybkimi zmianami wyrazów twarzy.

Innym ryzykiem, które może zagrozić terminowości projektu jest możliwość wystąpienia problemów z samym rozpoznawaniem twarzy. Aby aplikacja była przydatna, powinna radzić

sobie z wykrywaniem uśmiechu w możliwie jak największym kącie obrotu twarzy względem kamery. Aby sobie poradzić z tym problemem, należy zdefiniować jasne wymagania odnośnie kątów widzenia twarzy, aby móc testować aplikację na bieżąco.