

WYDZIAŁ INFORMATYKI, ELEKTRONIKI I TELEKOMUNIKACJI
KATEDRA INFORMATYKI



*System wykrywania anomalii u kierującego samochodem z
wykorzystaniem urządzeń mobilnych*

Autorzy: Jakub Janusz
Maciej Makówka
Krzysztof Węgrzyński

Prowadzący: dr. inż. Piotr Nawrocki (Systemy mobilne)
dr. hab inż. Bartłomiej Śnieżyński (IWiUM)

Kraków 2018

1. Wstęp

1.1. Motywacja

Pomimo dużego postępu technologicznego na polskich drogach nadal co roku giną tysiące osób [1]. Istnieje zatem potrzeba zwiększenia bezpieczeństwa kierujących pojazdami oraz ich pasażerów czy przechodniów. Istnieją już systemy badania stanu kierowcy oraz pojazdu, jednakże są one zintegrowane z pojazdem oraz ich montaż wiąże się z dodatkowymi kosztami. Normą w dzisiejszych czasach jest posiadanie urządzenia mobilnego (np. smartfony). Czujniki zamontowane w tych urządzeniach mogą służyć do profilowania sposobu zmiany ich położenia. Daje to możliwość rozpoznawania negatywnych wzorców poruszania się związanych z nieprawidłowym sposobem kierowania samochodem. Jest zatem nisza rynkowa, która stanowi motywację tego projektu.

1.2. Cel projektu

Celem projektu jest utworzenie aplikacji na urządzenia mobilne wykorzystujące system Android. Aplikacja ta powinna wykrywać oraz powiadamiać o anomaliach związanych ze stanem kierującego pojazdem lub stanem pojazdu.

1.3. Założenia

- Aplikacja powinna działać bez ingerencji prowadzącego poza jej uruchomieniem
- Głównym źródłem danych jest typowe urządzenie mobilne:
 - Akcelerometr
 - GPS
 - Kamera
 - Ruch GSM (rozmowy, SMS)
 - Monitorowanie maili oraz komunikatorów
- Dozwolone jest użycie danych o natężeniu ruchu
- Wykrycie anomalii powinno być sygnalizowane, możliwości zawierają:
 - Wiadomość dźwiękowa
 - Poczta elektroniczna
 - Wyświetlenie ostrzeżenia
 - SMS
 - Zawiadomienie służb

1.4. Anomalie

Jako anomalię definiujemy:

- Anomalny stan pojazdu, objawami mogą być:
 - Szarpnięcia
 - Wstrząsy
 - Brak płynności jazdy
- Anomalny stan osoby kierującej pojazdem, sygnalizowany przez następujące czynniki:
 - Brak akcji
 - Zmęczenie
 - Zmiana pozy
 - Opadanie głowy
 - Wyraz twarzy
 - Stan oczu
 - Zасыpianie

2. Przegląd podobnych rozwiązań

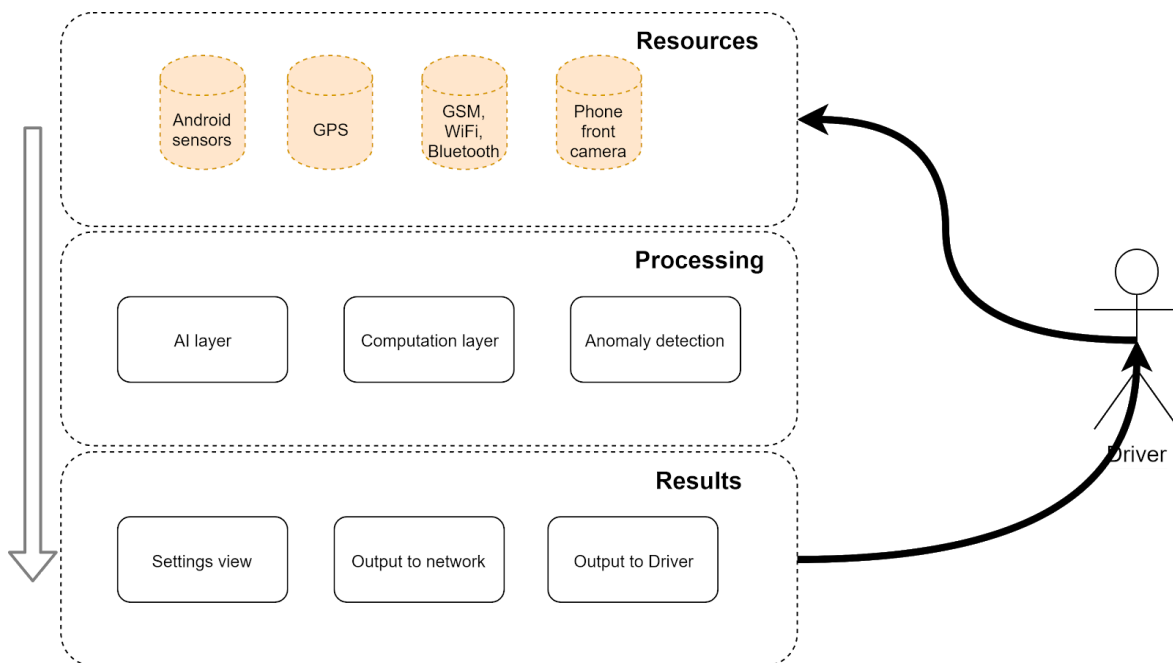
Częścią zadania projektowego jest zrobienie przeglądu literatury pod względem istniejących już rozwiązań. W celu wykonania tego zadania, opracowane zostały publikacje naukowe traktujące o tej tematyce. W niniejszej sekcji zostaną one wymienione oraz krótko opisane. Wykonany został również przegląd obecnie dostępnych źródeł danych o ruchu drogowym. Uzyskane w ten sposób informacje przedstawiają się następująco:

- Effect model of urban traffic congestion on driver's lane-changing behavior [2]:
 - Badany jest wpływ ruchu na akcję serca kierowcy oraz skłonności do ryzykownych manewrów (agresywna zmiana pasów)
 - Wyróżnione cechy kierowcy które mają wpływ na kierowanie:
 - wiek
 - płeć
 - doświadczenie w kierowaniu (w latach)
 - Na wejściu modelu jest:
 - akcja serca kierowcy w odniesieniu do zagęszczenie ruchu
 - stan kierowcy (kamera, GPS) w odniesieniu do zagęszczenie ruchu
- Pattern Analysis of Driver's "Pressure-State-Response" in Traffic Congestion [3]:
 - Mierzony jest stan kierowcy na podstawie stanu jego oczu, brane pod uwagę są:

- częstość mrugania
 - szybkość poruszania się oka
 - średni czas skupienia na obiekcie
 - inne
- Driver behavior profiling: An investigation with different smartphone sensors and machine learning [4]:
 - Wykorzystane sensorów telefonów z androidem do profilowania zachowania kierowcy
 - Potencjalne zastosowania takiego profilowania:
 - wykrywanie anomalii (zapobieganie sytuacjom krytycznym)
 - zmniejszenie zużycia paliwa
 - w branży ubezpieczeń komunikacyjnych
 - Przykładową anomalią może być zbyt ostre/agresywne wykonanie zakrętu
 - Istniejące już rozwiązania:
 - <https://www.aviva.co.uk/car-insurance/drive/> - personalny użytek, smartfony
 - <https://greenroad.com/> - zastosowanie we flotach
 - <https://www.ingenie.com/> - wykorzystywane jest "czarne pudełko" które zbiera dane
 - <https://www.progressive.com/auto/discounts/snapshot/>
 - <https://www.seeingmachines.com/industry-applications/automotive/> - rozwiązanie zintegrowane z pojazdem
 - <https://www.microsoft.com/en-us/research/project/nericell/> - windows phone, np. używa mikrofonu do wykrywania trąbienia
 - <http://vehicularlab.uni.lu/projects/sensefleet/>
- Driving Style Recognition Using a Smartphone as a Sensor Platform [5]:
 - Autorzy proponują system klasyfikujący styl jazdy kierowcy jako agresywny lub typowy (nieagresywny).
 - Już sama wiedza kierowcy o monitorowaniu jego jazdy przez urządzenie wpływa pozytywnie na bezpieczeństwo jazdy.
 - Proponowane rozwiązanie działa w całym zakresie na urządzeniu mobilnym. Nie są używane zewnętrzne dane oraz zasoby obliczeniowe.
 - Dźwiękowe powiadomienie o agresywnej jeździe. Udostępnione są dane z czujników na podstawie których ogłoszono alarm oraz te z okresu przed takim wykryciem.
 - Na podstawie danych z żyroskopu, akcelerometru oraz kątów Eulera orientacji urządzenia wykrywane są fragmenty sygnału które odpowiadają za wykonanie manewru. Następnie te fragmenty są porównywane z wcześniej przygotowanymi wzorcami w celu ich klasyfikacji jako agresywne lub nie.

- WreckWatch: Automatic Traffic Accident Detection and Notification with Smartphones [6]:
 - Proponowany jest system wykrywający wypadki drogowe oraz automatycznie powiadamiający odpowiednie służby.
 - Zapobieganie fałszywym alarmom jest kluczowym wyzwaniem w tego typu systemach.
 - System może przekazywać również podstawowe informacje na temat sceny zdarzenia do służb (ratownikom), np. dane z kamery urządzenia oraz mikrofonu.
- Safe Driving Using Mobile Phones [7]:
 - Analiza skuteczności wykorzystania urządzeń mobilnych do asystowania kierowcy w celu zwiększenia bezpieczeństwa.
 - Rozważana jest czułość przyrządów pomiarowych czy wpływ położenia urządzenia w pojeździe na rozpoznawanie manewrów.
 - Autorzy sugerują mierzenie stanu pojazdu (np. wykrywanie wczesnych zmian biegów a automatycznych skrzyniach biegów), analizę wzorców zachowania kierowcy oraz analizę stanu drogi (stan nawierzchni np. dziury).
- Survey of smartphone-based sensing in vehicles for intelligent transportation system applications [8]:
 - Przegląd literatury traktującej o zastosowaniu urządzeń mobilnych w inteligentnych systemach transportowych (ITS).

3. Opis rozwiązania oraz architektury



Rys. 1. Schemat architektury systemu

W środowisku przewidziano miejsce na użytkownika oraz system. Użytkownik jest jednocześnie źródłem danych dla systemu oraz odbiorcą przetworzonych informacji. System składa się z trzech warstw:

- zasobów - obejmuje zbiór sensorów, modeli danych oraz pamięci dyskowej udostępniony przez system operacyjny aplikacji do monitorowania anomalii. W skład tego zbioru, oprócz przestrzeni dyskowej, wchodzi m.in. żyroskop, akcelerometr, dane z GPS, ruch GSM, kamera frontowa.
- przetwarzającej - zakres funkcjonalności tej warstwy obejmuje wykorzystanie danych zbieranych w warstwie zasobów, w celu wykonania określonych obliczeń. Dane dostarczane jako *input* mają charakter *real-time*, co stanowi znaczące wyzwanie. W związku z tym nie wyklucza się oddelegowanie części zadań tej warstwy (które nie muszą być przetworzone natychmiast) zewnętrznemu serwerowi zdolnego wykonywać złożone operacje. Mogłoby to być uzasadnione gdyby wykorzystanie energii na obliczenia przewyższało wykorzystanie jej na komunikację. Obliczenia wykonywane na danych przetwarzane byłyby częściowo równolegle przez moduły odpowiedzialne za sztuczną inteligencję (np. uczenie maszynowe; detekcja oczu), wykrywanie anomalii (np. nietypowe odczyty z akcelerometru) czy moduły matematyczne (parsowanie wchodzących danych).
- rezultatów - odpowiada za udostępnienie warunków środowiskowych (np. ustawień aplikacji) oraz wytworzonych rezultatów, w określonej formie, klientowi. Takie wyniki mogą być zarówno eksportowane do sieci (w celu późniejszej analizy przez klienta bądź administratora), na ekran smartphone'a do wglądu graficznego, jak również w postaci określonych sygnałów ostrzegawczych (w czasie rzeczywistym) w celu ostrzeżenia klienta o wykrytej sytuacji nietypowej.

3.1. Stos technologiczny

Projekt tworzony jest w IDE Android Studio 3. Implementacja oparta jest o język programowania Java w wersji 8, dodatkowo do budowy aplikacji wykorzystano następujący zbiór narzędzi, technologii i bibliotek:

- Android Camera API
- Android Sensors
- Gradle
- Git
- LIBSVM
- MPAndroidChart
- WEKA

3.2. Wykorzystane zasoby

W implementacji systemu zdecydowano się wykorzystać podstawowe źródła danych z urządzeń mobilnych, na których podstawie jest się w stanie stwierdzić zaistnienie, bądź nie, anomalii u kierującego samochodem. Finalnie wykorzystano:

- Przednią kamerę urządzenia - przy użyciu Android Camera API obraz otrzymany z kamery poddawany jest przetworzeniu polegającemu na wyszukaniu na nim twarzy kierującego. W przypadku znalezienia twarzy aplikacja przystępuje do kolejnego kroku, jakim jest wykrycie oczu, a następnie badanie w czasie rzeczywistym czy są one otwarte czy zamknięte. Długotrwałe zamknięcie oczu przez kierującego pojazdem wskazuje na jego skrajne zmęczenie, czego efektem może być zaśnięcie. W takim przypadku aplikacja generuje niejednostajny sygnał dźwiękowy o zmiennym natężeniu celem pobudzenia kierującego. Natomiast brak wykrytej twarzy może świadczyć o zmianie pozy kierującego, co może oznaczać dezorientację oraz zwiększenie czasu reakcji, lub o skrajnym zmęczeniu powodujących opadanie głowy, czego efektem może być zaśnięcie.
- Akcelerometr, magnetometr i żyroskop - na podstawie danych ze wszystkich trzech sensorów w czasie rzeczywistym wyliczane jest przyspieszenie działające na urządzenie, które skorelowane jest z kierunkiem ruchu pojazdu ("prawdziwe przyspieszenie"). Pozwala to na dowolne ułożenie urządzenia względem kierunku jazdy bez zaburzenia pracy detektora anomalii. Otrzymane w ten sposób "prawdziwe przyspieszenie" aplikacja prezentuje na wykresie, który pokazuje jego zmienność w czasie wzdłuż każdej z trzech osi. Dokładny opis metody wyznaczania "prawdziwego przyspieszenia" znajduje się w [9].
- Ruch GSM - w czasie działania aplikacji stale monitorowany jest ruch GSM pod kątem odebranych wiadomości (SMS/MMS) oraz połączeń telefonicznych. W ramach połączeń rozróżniane są połączenia przychodzące, wychodzące oraz nieodebrane i odrzucone. W przypadku połączeń nawiązanych (przychodzących i wychodzących) aplikacja wskazuje czy któreś z nich jest także w trakcie trwania.

3.3. Detekcja manewrów

Manewrem potocznie można nazwać agresywną zmianę pasa ruchu, gwałtowny skręt, gwałtowne hamowanie, itd. W kontekście projektu manewr reprezentowany jest jako macierz o wymiarach $4 \times n$, gdzie n stanowi liczbę pomiarów tworzącą dany manewr. Każda kolumna macierzy reprezentuje kolejno:

- Indeks pomiaru w obrębie manewru tj. jedno z:
 - Liczba naturalna
 - Znacznik czasowy pomiaru

- Przesunięcie urządzenia reprezentowane przez przyspieszenie na 3 osiach (X, Y, Z) jako liczba wymierna:
Opisaną tu macierz można rozumieć również jako zestaw 4 n-elementowych wektorów. Detekcja manewrów odbywa się w dwóch głównych krokach:
- Utworzenie modelu idealnego manewru na podstawie przygotowanych wcześniej danych treningowych, do regresji używana jest maszyna wektorów nośnych (dokładnie jej wariant SVR - Support Vector Regression);
- Klasyfikacja strumienia danych w czasie rzeczywistym, binarne klasyfikowanie okna czasowego zawierającego manewr jako jeden z wcześniej przygotowanych manewrów lub nie.

3.4. Zastosowana regresja - uczenie z nadzorem

Regresja w ramach tego projektu została wykorzystana do utworzenia modelu "idealnego" manewru. Konkretnie wykorzystany został algorytm SVR będącym wariantem metody maszyny wektorów wspierających, użyto biblioteki LIBSVM [9] oraz nakładki do tej biblioteki oferowaną przez WEKA [10].

Każda oś wykonywanych pomiarów tj. X, Y lub Z jest poddawana algorytmowi regresji osobno. Wynika to z ograniczenia metody SVR - nie umożliwia ona regresji wielowymiarowej.

Dla każdej rozważanej osi, dane uczące zawierają 2 atrybuty:

- Indeks pomiaru w manewrze
- Liczbę wymierną stanowiącą pomiar przyspieszenia w kierunku reprezentowanym przez daną oś

Tworzony podmodel dla indeksu pomiaru, zwraca nam przewidywaną wartość przyspieszenia w danym kierunku, która jak najlepiej odzwierciedla dany manewr. Trzy takie podmodele (dla każdej osi) tworzą razem właściwy model manewru. Model ten następnie jest używany w procesie klasyfikacji, sprawdzającym czy bieżąca próbka danych może być sklasyfikowana binarnie jako rozważany manewr. Każdy manewr jest rozważany osobno (każdy ma swój klasyfikator binarny).

3.5. Klasyfikacja manewrów w czasie rzeczywistym

Do klasyfikacji pomiarów, pobieranych w czasie rzeczywistym, jako należących do jednej z klas zdefiniowanych przez dane treningowe wykonywane jest porównanie tychże danych z utworzonymi wcześniej modelami regresji. Porównanie to składa się na 2 główne etapy:

- Obliczenie jak bardzo różnią się dane bieżące oraz model. Realizowane przez zastosowanie metody DTW (Dynamic Time Warping) do obliczenia odległości pomiędzy dwoma sygnałami. Sygnał jest tutaj rozumiany jako funkcja, która dla danego indeksu pomiaru zwraca nam amplitudę przyspieszenia dla danej

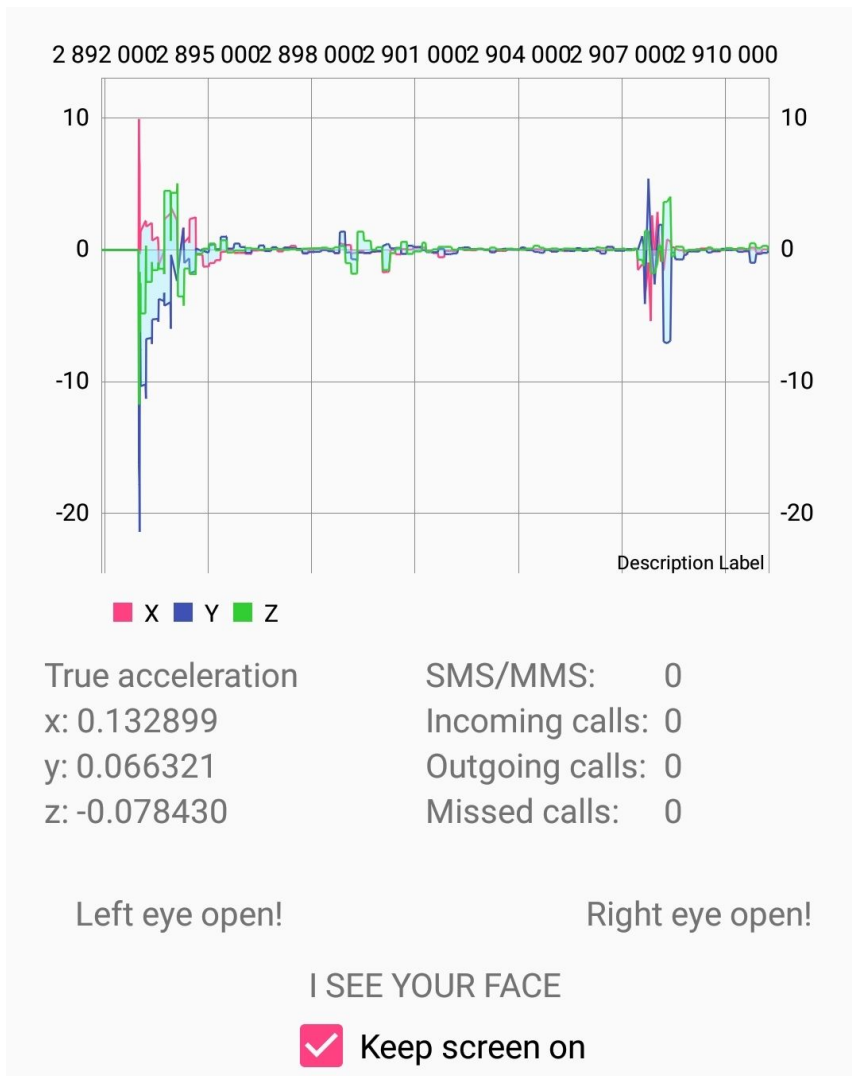
osi. Takie porównanie daje w rezultacie trzy liczby, oznaczające jak bardzo pomiar różni się od modelu dla każdej z osi.

- Suma liczb uzyskanych w poprzednim punkcie jest następnie porównywana z progiem ustanowionym heurystycznie. Próg ten jest zsumowaną średnią wartością modułów amplitud pomiarów dla każdej osi, przemnożoną przez czynnik czułości ustalony arbitralnie (obecnie 0.7). Jeżeli liczba stanowiąca podobieństwo próbki danych bieżących oraz modelu regresyjnego, jest wystarczająco niska (poniżej progu) to kierowcy raportowana jest anomalia, oznaczająca wykonanie niebezpiecznego manewru. Dodatkowo próg ten jest modyfikowany, w zależności od innych mierzonych czynników, które mogą istotnie wpłynąć na bezpieczeństwo jazdy. Czynniki te to m. in.:
 - Detekcja twarzy - jeżeli to tylko możliwe to twarz powinna być widoczna dla przedniej kamery urządzenia. W przypadku braku detekcji twarzy można uznać, że poza kierowcy mogła ulec zmianie w niebezpieczny sposób np. obrócił się on od głównego kierunku jazdy (dezorientacja) lub jego głowa opada z powodu zasypiania (zmęczenie).
 - Detekcja stanu oczu - zamknięte lub otwarte. Podobnie jak w punkcie a), zamknięte oczy powodują, że próg jest mniej restrykcyjny. Może to być interpretowane jako symptom zasypiania.
 - Dane z ruchu GSM - rozmowa telefoniczna, a także wysyłanie i odbieranie wiadomości SMS podczas jazdy zmniejsza bezpieczeństwo poprzez odwracanie uwagi kierującego.

4. Ewaluacja rozwiązania

Po zakończeniu implementacji systemu przystąpiono do jego ewaluacji. Całość rozwiązania stanowi jeden widok aplikacji mobilnej na system Android (w wersji minimum 7.0), który został przedstawiony na Rys. 2. Na samej jego górze znajduje się wykres prezentujący graficznie wartość prawdziwego przyspieszenia we wszystkich trzech osiach (x, y, z) w zależności od czasu w oknie 500 ostatnich odczytów. Poniżej wykresu, po lewej stronie znajdują się aktualne wartości true acceleration, aktualizowane po dowolnym odczycie z jednego z użytych sensorów (akcelerometru, żyroskopu, magnetometru). Po prawej stronie natomiast znajdują się ilości odebranych wiadomości SMS i MMS, a także połączeń przychodzących, wychodzących i nieodebranych (w tym także odrzuconych). Wartości te przybierają kolor czerwony jeśli połączenie danego typu jest w trakcie. Poniżej znajdują się kontrolki odpowiedzialne za wyświetlanie informacji na temat wykrycia twarzy kierowcy. Środkowa daje informację czy aplikacja wykryła twarz prowadzącego pojazd, natomiast pozostałe dwie, lewa i prawa, mówią użytkownikowi czy aplikacja wykryła odpowiednio lewe i prawe oko otwarte bądź zamknięte. Na samym dole znajduje się pole wyboru. W zależności od jego wartości ekran urządzenia będzie

ciągłe aktywny, bądź też zablokuje się automatycznie zgodnie z ustawieniami systemu operacyjnego.



Rys. 2. Widok stworzonej aplikacji.

Walidacja działania systemu przeprowadzona została dwójako. Pierwszym ze sposobów są automatyczne testy integracyjne algorytmu wykrywającego anomalie, które pokrywają większość typowych przypadków. Drugim ze sposobów są przeprowadzone testy empiryczne. Były one możliwe dzięki notyfikacji dźwiękowej w momencie wykrycia anomalii. Korzystając z tej funkcjonalności można było stwierdzić, że aplikacja w czasie działania zachowuje się zgodnie z jej planowanym zastosowaniem i wykrywa anomalie w sposób zadowalający.

5. Podsumowanie

Biorąc pod uwagę otrzymane rezultaty, aplikacja dobrze sobie radzi z postawionym jej zadaniem. Zaimplementowane metody uczenia maszynowego doskonale znajdują zastosowanie w postawionym problemie. Co więcej, dane

zbierane i analizowane przez aplikację wydają się wystarczające do wykrywania głównych anomalii w czasie kierowania samochodem.

Projekt posiada kilka możliwych ścieżek rozwoju. Jedną z nich jest dodanie monitorowania danych sieciowych celem filtrowania ich pod kątem używania w czasie jazdy komunikatorów internetowych, a także poczty elektronicznej. Detekcja używania komunikatorów możliwa jest także poprzez monitorowanie powiadomień push systemu Android. Kolejną ścieżką rozwoju może być również poszerzenie detekcji przeciążeń celem klasyfikacji ekstremalnych wartości jako zdarzenia drogowe i alarmowanie odpowiednich służb. Ostatnią z proponowanych ścieżek rozwoju jest analiza dźwiękowa wnętrza pojazdu celem detekcji np. zbyt głośnego słuchania muzyki lub prowadzenia długotrwałych rozmów, co może prowadzić do rozkojarzenia kierującego samochodem.

W obliczu osiągniętych wyników projekt uznaje się za spełniający postawione wymagania i realizujący je w pełni.

6. Bibliografia

- [1] <http://statystyka.policja.pl/st/ruch-drogowy/76562.Wypadki-drogowe-raport-y-roczne.html>
- [2] Qi, W., Wen, H., Wu, Y., & Qin, L. (2017). Effect model of urban traffic congestion on driver's lane-changing behavior. *Advances in Mechanical Engineering*, 9(9), 1687814017724087.
- [3] Weiwei Qi, Yulong Pei, Mo Song, and Yiming Bie, "Pattern Analysis of Driver's "Pressure-State-Response" in Traffic Congestion," *Discrete Dynamics in Nature and Society*, vol. 2013, Article ID 853845, 11 pages, 2013. doi:10.1155/2013/853845
- [4] Ferreira, J., Carvalho, E., Ferreira, B. V., de Souza, C., Suhara, Y., Pentland, A., & Pessin, G. (2017). Driver behavior profiling: An investigation with different smartphone sensors and machine learning. *PLoS ONE*, 12(4), e0174959. <http://doi.org/10.1371/journal.pone.0174959>
- [5] D. A. Johnson and M. M. Trivedi, "Driving style recognition using a smartphone as a sensor platform," 2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC), Washington, DC, 2011, pp. 1609-1615. doi: 10.1109/ITSC.2011.6083078
- [6] Jules White, Chris Thompson, Hamilton Turner, Brian Dougherty, and Douglas C. Schmidt. 2011. WreckWatch: Automatic Traffic Accident Detection and Notification with Smartphones. *Mob. Netw. Appl.* 16, 3 (June 2011), 285-303. DOI=<http://dx.doi.org/10.1007/s11036-011-0304-8>
- [7] M. Fazeen, B. Gozick, R. Dantu, M. Bhukhiya and M. C. González, "Safe Driving Using Mobile Phones," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 3, pp. 1462-1468, Sept. 2012. doi: 10.1109/TITS.2012.2187640

- [8] J. Engelbrecht, M. J. Booysen, G. J. van Rooyen and F. J. Bruwer, "Survey of smartphone-based sensing in vehicles for intelligent transportation system applications," in IET Intelligent Transport Systems, vol. 9, no. 10, pp. 924-935, 12 2015. doi: 10.1049/iet-its.2014.0248
- [9] <https://stackoverflow.com/questions/15158769/android-acceleration-direction>
- [10] C.-C. Chang and C.-J. Lin. LIBSVM : a library for support vector machines. ACM Transactions on Intelligent Systems and Technology, 2:27:1--27:27, 2011. pdf, ps.gz, ACM digital lib.
- [11] <https://www.cs.waikato.ac.nz/ml/weka/>