

DOI: 10.5604/20830157.1148048

URZĄDZENIA MOBILNE JAKO REJESTRATORY PRZYSPIESZENIA

Jakub Smolka, Maria Skublewska-Paszowska, Marcin Badurowicz, Edyta Łukasik

Politechnika Lubelska, Wydział Elektrotechniki i Informatyki, Instytut Informatyki

Streszczenie. Artykuł przedstawia możliwości zastosowania urządzeń mobilnych jako rejestratorów przyspieszenia. Wykonane badania dotyczą użycia urządzeń działających na trzech platformach mobilnych: Android, iOS oraz Windows Phone. W tym celu stworzono trzy odrębne aplikacje mobilne, przy pomocy których mierzone jest przyspieszenie w trzech kierunkach (X, Y, Z). Przed wykonywaniem pomiarów, urządzenia są kalibrowane. Dodatkowo aplikacja dla systemu Android umożliwia przeliczenie wartości z układu współrzędnych urządzenia do układu współrzędnych świata. Wybrane wyniki są przedstawione w postaci wykresów.

Słowa kluczowe: urządzenie mobilne, rejestrator przyspieszenia

MOBILE DEVICES AS ACCELERATION RECORDERS

Abstract. The paper investigates possibility of using mobile devices as acceleration recorders. Presented research focuses on mobile devices running three mobile platforms: Android, iOS and Windows Phone. Three mobile applications are created in order to measure the acceleration in three directions (X,Y,Z). Prior to data recording all devices are calibrated. Additionally the Android application allows for converting the results from device's to world's coordinate system. The results are presented in graphs.

Keywords: mobile device, acceleration recorders

Wstęp

We współczesnych smartfonach i tabletach wbudowanych jest wiele elektronicznych czujników. Można podzielić je na trzy kategorie: czujniki ruchu, czujniki położenia oraz czujniki środowiska. Do pierwszej kategorii należą m.in. akcelerometry czy żyroskopy, do drugiej m.in. magnetometry, natomiast do trzeciej barometry, termometry i fotometry [9]. Czujniki umożliwiają np. pomiar przyspieszenia, którego doznaje urządzenie, pola magnetycznego w jakim urządzenie się znajduje czy prędkości obrotowej urządzenia. Dzięki wspomnianym czujnikom możliwe staje się określenie ustawienia urządzenia w przestrzeni czy też tego jak się porusza. Integracja zestawu czujników w smartfonach/tabletach, które de facto są wydajnymi przenośnymi komputerami daje szereg nowych możliwości. Powszechność urządzeń mobilnych powoduje, że stanowią one kuszące narzędzie do zbierania różnego rodzaju informacji. Użytkownicy tych urządzeń mogą być zarówno użytkownikami jak i źródłem informacji. Schemat taki jest już stosowany, np. urządzenia mobilne wyposażone w moduł GPS oraz moduł WiFi zbierają dane na temat położenia punktów dostępowych sieci bezprzewodowej, które są następnie wykorzystywane do określania położenia bez dostępu do systemu GPS. Podobny model można zastosować w przypadku innego rodzaju danych np. jakości dróg. Badanie jakości dróg nie jest tematem nowym. Jednak takie badania są prowadzone w sposób planowy i z wykorzystaniem specjalnego sprzętu [2, 7, 11]. Opracowanie metody oceny jakości dróg prowadzonej przez użytkowników dróg umożliwiłoby szybkie zbudowanie bazy danych na temat problematycznych odcinków. Pozwoliłoby również na opracowanie map dla systemów nawigacji, które planowałyby trasę prowadzącą po drogach o dobrej jakości.

Aby wymienione (i inne) zastosowania były możliwe programiści tworzący oprogramowanie przeznaczone dla smartfonów czy tabletów muszą mieć dostęp do sensorów urządzenia. Dostęp ten jest zapewniony poprzez użycie odpowiednich interfejsów (ang. application programming interfaces) znajdujących się w systemie operacyjnym urządzenia mobilnego.

Artykuł ten przedstawia możliwości użycia akcelerometrów w trzech popularnych systemach mobilnych. Przedstawiono również wyniki zebrane za pomocą trzech własnych aplikacji przygotowanych dla platform Android, iOS oraz Windows Phone. Wszystkie aplikacje umożliwiają pomiar i rejestrowanie przyspieszenia z jakiego doznawały urządzenia testowe wzdłuż trzech osi (X, Y oraz Z). Zarejestrowane wyniki zostały następnie przeanalizowane, pod kątem ich wykorzystania w badaniach ruchu człowieka lub pojazdu.

1. Możliwości sensorów w urządzeniach mobilnych

Jak już wspomniano systemy operacyjne dla urządzeń mobilnych udostępniają aplikacjom, za pośrednictwem odpowiedniego API, szereg różnych czujników. Oprócz podziału przytoczonego powyżej (na czujniki ruchu, położenia i otoczenia) można wyróżnić kolejno: np. czujniki sprzętowe oraz czujniki programowe. Czujniki sprzętowe dostępne poprzez API umożliwiają bezpośredni dostęp do odczytów pochodzących z elementów pomiarowych faktycznie znajdujących się na płycie głównej urządzenia mobilnego. Warto zaznaczyć, że oprócz odczytywania „surowych” wartości zwracanych przez czujniki sprzętowe często też możliwe jest też uzyskanie wartości uwzględniających np. wyniki kalibracji. Czujniki programowe udostępniają wyniki obliczeń przeprowadzonych na podstawie wartości uzyskanych za pomocą czujników sprzętowych. Przykładem może być tutaj czujnik położenia, którego wskazania często uzyskiwane są na podstawie wskazań akcelerometru oraz magnetometru [9].

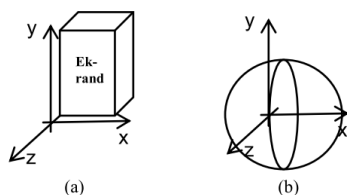
Tabela 1. Czujniki dostępne za pośrednictwem API poszczególnych systemów operacyjnych [1, 3-9]

Czujnik	Android	iOS	Windows Phone
Grawitacji	Tak	Tak	Tak
Orientacji	Tak	Tak	Tak
Oświetlenia	Tak	Nie	Nie
Pola magnetycznego	Tak	Tak	Tak
Prędkość rotacji (żyroskop)	Tak	Tak	Tak
Przyspieszenia	Tak	Tak	Tak
Przyspieszenia liniowego	Tak	Tak	Tak
Rodzaju ruchu	Nie	Tak	Nie
Temperatury	Tak	Nie	Nie
Wilgotności względnej	Tak	Nie	Nie
Wykrywanie / liczenie kroków	Tak	Tak	Nie
Zbliżeniowy	Tak	Tak*	Nie
Znaczącego przemieszczenia	Tak	Nie	Nie

* brak pomiaru odległości (tylko 2 stany: blisko, daleko)

Jak wynika z tabeli 1 wszystkie systemy operacyjne oferują podobne możliwości obsługi i wykorzystania czujników wbudowanych w urządzenia mobilne. Cechą wyróżniającą system Android jest obsługa sensorów otoczenia (termometr, barometr, higrometr) a także możliwość wyzwalania działań znaczącym przemieszczeniem. System iOS wyróżnia się z kolei ciekawym programowym czujnikiem rozpoznającym rodzaj ruchu – tzn. pozwalającym na określenie czy użytkownik porusza się samochodem czy też na przykład biegnie.

Wszystkie systemy operacyjne oferują dostęp do czujników odczytujących wartość całkowitego przyspieszenia doznawanego przez urządzenie (tzn. sumy zewnętrznego przyspieszenia i przyspieszenia ziemskiego), a także czujniki pozwalające na osobne odczytanie przyspieszenia spowodowanego przez siły zewnętrzne i siłę grawitacji. Stosowane jest różne nazewnictwo wspomnianych sensorów – czujnik przyspieszenia liniowego (Android / Windows Phone) lub czujnikiem przyspieszenia użytkownika (iOS). Jednostki, w których prezentowane są wyniki też różnią się między platformami – są to albo wielokrotności przyspieszenia ziemskiego g (iOS/ Windows Phone) lub m/s^2 (Android). Również sposób dostępu do czujników jest różny. W przypadku platform Android i iOS użycie sensorów nie wymaga żadnych uprawnień i dostęp do nich ma każda aplikacja. W przypadku systemu Windows Phone aplikacja musi, w trakcie instalacji, uzyskać zgodę użytkownika.



Rys. 1. Układy współrzędnych stosowane przy pomiarach w urządzeniach mobilnych: (a) współrzędne urządzenia, (b) współrzędne świata

Ważnym zagadnieniem jest układ współrzędnych, w którym mierzone są przyspieszenia. We wszystkich urządzeniach pomiary wykonywane są w układzie współrzędnych związanym z urządzeniem. Układ ten przedstawiony jest na rysunku 1a. W niektórych zastosowaniach przydatna jest możliwość przeliczenia pomiarów do układu współrzędnych związanym ze światem zewnętrznym. Układ taki przedstawiono na rysunku 1b – Oś Y wskazuje północ, oś X (w przybliżeniu) wschód, natomiast oś Z jest skierowana w górę (od powierzchni Ziemi).

Wszystkie omawiane platformy oferują możliwość przeliczenia pomiarów przyspieszenia z układu współrzędnych urządzenia do układu współrzędnych świata. Na podstawie wskazań akcelerometru i magnetometru może być obliczona macierz rotacji, która pozwala na dokonanie wspomnianego przeliczenia. Warto zaznaczyć, że wszystkie platformy oferują gotowe metody obliczania macierzy rotacji, wektora rotacji (kwaternionu jednostkowego) oraz kątów Eulera (ang. yaw, pitch, roll), które stanowią różne sposoby reprezentacji ustawienia urządzenia w przestrzeni.

2. Akcelerometr w aplikacjach mobilnych

Problemem, który występuje podczas rejestrowania przyspieszenia za pomocą urządzeń mobilnych jest dokładność akcelerometrów. Tabela 2 przedstawia wartości uzyskane na wybranych urządzeniach testowych. Urządzenia leżały nieruchomo. Sprawdzone dwa różne ustawienia urządzeń. W pierwszym ustawieniu oś Z układu związanego z urządzeniem była skierowana ku górze (urządzenie leżało ekranem do góry), w drugim ustawieniu skierowana do góry była oś Y.

Tabela 2. Odczyty akcelerometrów w nieporuszających się urządzeniach testowych

urządzenie	ustawienie	przyspieszenie	a_x	a_y	a_z
			[m/s^2]	[m/s^2]	[m/s^2]
Nexus 7	Z-góra	Liniowe	0,0	0,0	0,2-0,3
Nexus 7	Y-góra	Liniowe	-0,1-0	0,1	-0,1-0
Nexus 4	Z-góra	Liniowe	0,0	0,0	0,0
Nexus 4	Y-góra	Liniowe	0,0	0,0	0,0
Nexus 7	Z-góra	Całkowite	0,1	0-0,1	10,1
Nexus 7	Y-góra	Całkowite	-0,1-0	9,9	0-0,1
Nexus 4	Z-góra	Całkowite	-0,1-0	0-0,1	9,8
Nexus 4	Y-góra	Całkowite	-0,1	9,8	0,1-0,2

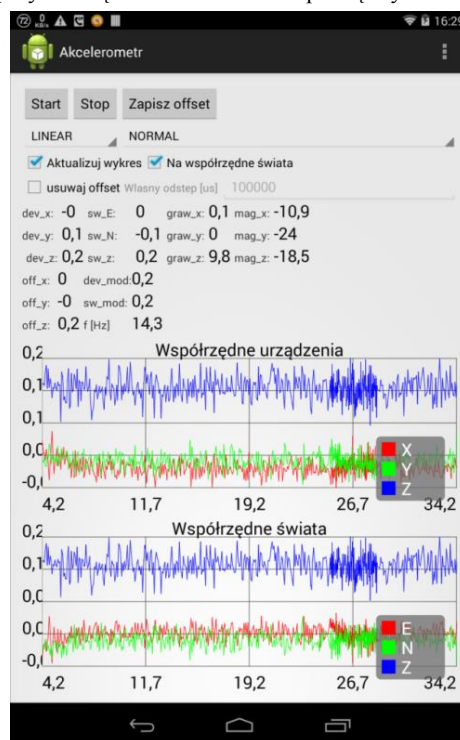
Jak widać odczyty przyspieszenia nieruchomego smartfonu czy tabletu nie zawsze wynoszą 0. Zależą one nie tylko od samego urządzenia (modelu urządzenia, modelu wbudowanego

akcelerometru, czy też konkretnego egzemplarza) lecz również od jego ustawienia. Z przeprowadzonych prób wynika, że w urządzeniach testowych wahania odczytów w nieruchomym urządzeniu nie przekraczają $0,1 m/s^2$. Warto też zaznaczyć, że moduł odczytanego przyspieszenia może różnić się od rzeczywistej wartości nawet o $0,3m/s^2$

Rozwiązaniem tego problemu jest przeprowadzenie prostej kalibracji. Polega ona na zarejestrowaniu serii odczytów podczas gdy urządzenie leży nieruchomo i obliczenie średnich przesunięć jakimi charakteryzują się pomiary wzdłuż osi X, Y oraz Z. Podczas wykonywania właściwych pomiarów przesunięcia są odejmowane. Warto zaznaczyć, że ponieważ przesunięcia odczytów zależą od ustawienia smartfonu/tabletu to po wykonaniu kalibracji urządzenie nie powinno go zmieniać.

2.1. Aplikacja mobilna na platformę Android

Aplikacja przeznaczona dla systemu Android korzysta ze standardowego frameworku sensorów, który jest częścią pakietu android.hardware [9]. Ze względu na to, że wykorzystano dwa rodzaje akcelerometrów (rejestrujący przyspieszenie liniowe oraz całkowite) aplikacja wymaga systemu Android w wersji min. 2.3 (we wcześniejszych wersjach czujnik przyspieszenia liniowego nie był dostępny [9]). Oprócz wyboru rodzaju mierzonego przyspieszenia możliwy jest wybór częstotliwości wykonywanych pomiarów (dostępne są częstotliwości próbkowania: 14,3Hz, 50,1Hz, 200,3Hz, a także opcja ustawienia własnego odstepu między rejestracją kolejnych wartości). Aplikacja umożliwia również przeprowadzenie kalibracji akcelerometru i oferuje opcję automatycznego usuwania zmierzonych przesunięć. Dane pochodzące z pomiarów prezentowane są na wykresach a także zapisywane w pliku CSV. Bieżące wartości wyświetlane są w etykietach tekstowych. Jak widać na rys. 2 są to nie tylko wartości przyspieszeń ale również przyspieszenie wynikające wyłącznie z siły grawitacji oraz wartości indukcji magnetycznej w μT . Są one niezbędne do przeliczenia przyspieszenia z układu współrzędnych urządzenia do układu współrzędnych świata.



Rys. 2. Aplikacja rejestrująca przyspieszenia przeznaczona dla systemu Android

2.2. Aplikacja mobilna na platformę iOS

W aplikacji na smartfon działający na platformie iOS, użyto frameworku Core Motion [1]. Zapewnia on dostęp do danych pochodzących m. in. z akcelerometru, żyroskopu czy

magnetometru (bardziej szczegółowy wykaz przedstawiono w tabeli 1).

Stworzona na potrzeby artykułu aplikacja umożliwia rejestrację przyspieszenia wzdłuż osi X, Y i Z w układzie współrzędnych urządzenia. Rejestracja odbywa się z częstotliwością 100 Hz. Dane dla poszczególnych osi są wyświetlane w etykietach tekstu. Przed każdą rejestracją wskazana jest kalibracja urządzenia. Przycisk *Reset* umożliwia wyzerowanie wyników. Po przeprowadzonej rejestracji, uzyskane dane są zapisywane do pliku CSV. Plik ten można wysłać korzystając z wbudowanej aplikacji pocztowej.

2.3. Aplikacja mobilna na platformę Windows Phone

Aplikacja przeznaczona dla platformy Windows Phone składa się z dwóch głównych elementów: wskazania aktualnych wartości i listy archiwalnych pomiarów.

Menu dolne aplikacji jest wysuwane, aby zwiększyć przestrzeń dostępną dla danych. Aplikacja wykorzystuje framework Windows Phone Silverlight [10] w wersji 8.0 i pobiera dane z akcelerometru za pomocą klasy `Accelerometer` pochodzącej z przestrzeni nazw `Microsoft.Devices.Sensors` [5]. Przed każdą rejestracją można przeprowadzić kalibrację urządzenia. Dostępne w menu są opcje takie jak „start/stop” uruchamiające i zatrzymujące serię pomiarów, „clear” powodująca wyczyszczenie listy wyników oraz opcja „upload” służąca do wysyłania zapisanych w pamięci danych na zdalny serwer HTTP. Aplikacja posiada dwie częstotliwości pomiarów – standardowo wykonywane są pomiary co każde 20 milisekund (50 Hz), ale możliwe jest uaktywnienie trybu „high frequency”, co zwiększa częstotliwość do 100 Hz (pomiar co każde 10 milisekund).

3. Przedstawienie przykładowych wyników pomiarów z wykorzystaniem aplikacji

Ze względu na fakt, że docelowo jedną z rozpatrywanych przez autorów hipotez jest wykorzystanie stworzonych aplikacji do pomiaru jakości dróg, aplikacje testowano podczas jazdy seryjnym, niemodyfikowanym, samochodem Peugeot 1007, na kilku odcinkach o różnej nawierzchni. Testy prowadzono, aby zbadać słuszność koncepcji i zmierzyć, czy możliwe będzie zaobserwowanie drgań samochodu w zestawach danych. Pierwszy odcinek testowy składał się z asfaltowych płyt betonowych tworzących jeden z parkingów studenckich na Politechnice Lubelskiej. Smartfony umieszczone były poziomo, przez co nierówności podłoża powinny zostać zaobserwowane w osi Z.

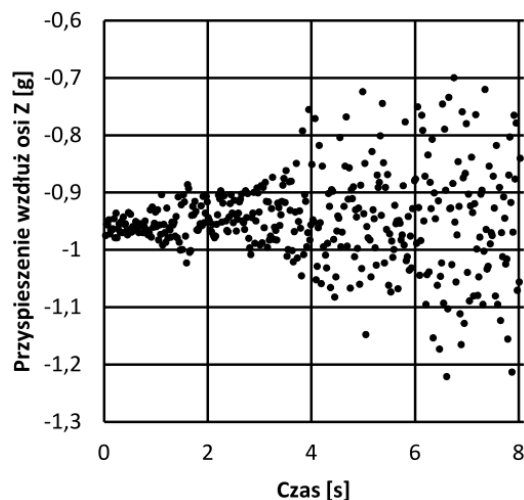
Wykres prezentowany na rys. 3 zawiera pomiary wykonywane przez około 8 sekund badania – tj. 394 próbki. Dane zostały zarejestrowane za pomocą telefonu Nokia Lumia 920 działającego pod kontrolą systemu Windows Phone. W początkowej fazie rejestrowane drgania mają mniejszą amplitudę niż w kolejnej fazie ruchu, co związane jest z przyspieszeniem samochodu od 0 do około 30 km/h. Warto zwrócić uwagę, że zarejestrowane drgania są bardzo regularne i odpowiadają kolejnym „dziurom”, w które wpada samochód, a drgania przekazywane są do karoserii i kabiny.

Wykres z rys. 4 przedstawia próbki zarejestrowane pomiędzy 32-gą do 39-tą sekundą przejazdu testowego po ul. Koncertowej w Lublinie. Wyraźnie tutaj widoczna jest wprawdzie zmiana przyspieszenia w osi Y będąca odzwierciedleniem zmiany kierunku samochodu i wykonania skrętu. Wartości przyspieszeń rosną, przez co, zgodnie z rysunkiem 1a, urządzenie rejestruje przyspieszenie w swoją własną prawą stronę. Następnie, widoczne jest pojawienie się ujemnego przyspieszenia w osi X, która odpowiada kierunkowi „na wprost” samochodu, co może być łatwo przetłumaczone na zmniejszenie prędkości poruszania się pojazdu.

Wykresy przedstawione na rys. 5 oraz 6 zawierają dane zarejestrowane podczas dwóch kolejnych przejazdów testowych.

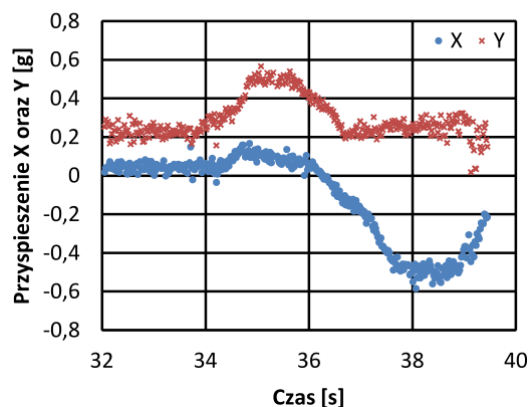
Próby przeprowadzono za pomocą dwóch urządzeń – działającego pod kontrolą systemu Android smartfonu Nexus 4 oraz sterowanego przez system iOS telefonu iPhone 5. Przejazdy miały miejsce na wspomnianym wcześniej parkingu z płytami asfaltowymi oraz na nowo wyremontowanym odcinku ulicy Narutowicza w Lublinie.

Pomiary na wyboistej nawierzchni (Lumia 920)



Rys. 3. Pomiary przyspieszenia w osi Z na parkingu

Pomiary na gładkiej nawierzchni (Lumia 920)



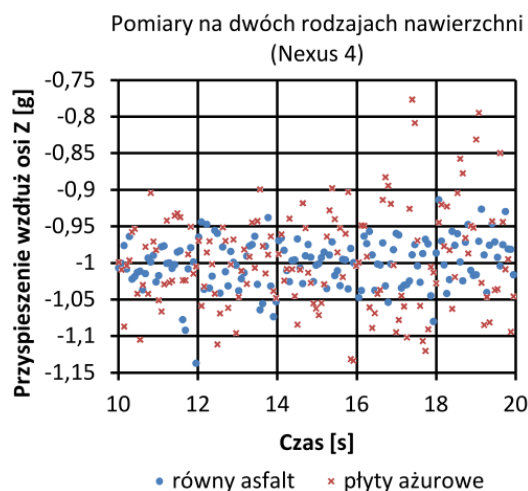
Rys. 4. Przyspieszenia w osiach X i Y na ulicy

Różnica pomiędzy dwoma typami nawierzchni jest wyraźnie widoczna. W tabeli 1 zamieszczono wskaźniki (średnia, wariancja rozrzut przyspieszenia w osi Z) obliczone na podstawie pomiarów. Jak widać wartość średnia przyspieszenia w osi Z nie zależy od rodzaju nawierzchni (ujemne wartości przyspieszenia wynikają z tego, że telefony leżały ekranem do góry). Różnice między poszczególnymi urządzeniami mogą wynikać z zastosowania w nich różnych akcelerometrów oraz z drobnych różnic w ustawieniu samych urządzeń. Wartości pozostałych wskaźników różnią się w zależności od rodzaju nawierzchni. Znaczące różnice widać w przypadku wariancji oraz rozrzutu. Zarówno wariancja jak i rozrzut rosną w przypadku przejazdu po płytach asfaltowych (jedynym wyjątkiem jest rozrzut w przypadku urządzenia Nexus 4). Warto zauważyć, że różnice w wartościach wskaźników są widoczne pomimo wykorzystania różnych urządzeń oraz różnych częstotliwości próbkowania (Nexus 4 – 14,9 Hz, iPhone5 – 100Hz, Lumia 920 – 10Hz).

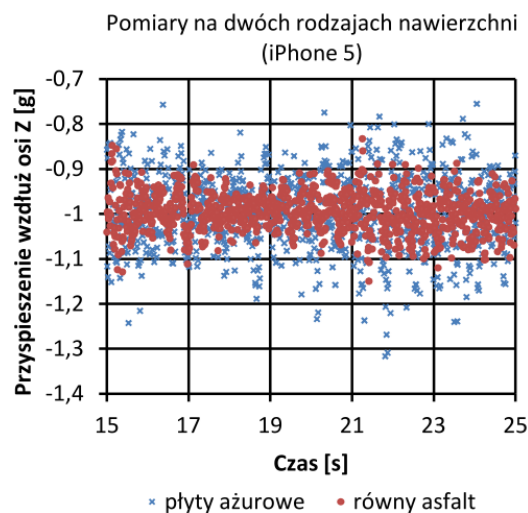
Tabela 3. Wskaźniki obliczone na podstawie pomiarów przyspieszenia w osi Z

Urządzenie	Nexus 4	iPhone 5	Lumia 920
Średnia (gładka droga) [g]	-1,0003	-0,9958	-0,9329
Średnia (płyty) [g]	-1,0118	-0,9949	-0,9493
Wariancja (gładka droga)	0,00113	0,00200	0,00076

Wariancja (plyty)	0,00164	0,00699	0,00924
Rozrzut (gładka droga) [g]	0,224	0,317	0,146
Rozrzut (plyty) [g]	0,212	0,561	0,842



Rys. 5. Porównanie przyspieszeń w osi Z podczas testów na różnych nawierzchniach (urządzenie testowe – Nexus 4)



Rys. 6. Porównanie przyspieszeń w osi Z podczas testów na różnych nawierzchniach (urządzenie testowe – iPhone 5)

4. Podsumowanie

W artykule przedstawiono własne aplikacje mobilne stworzone dla popularnych platform mobilnych. Zadaniem aplikacji było rejestrowanie przyspieszenia w osiach X, Y i Z. Wykonane pomiary posłużyły do potwierdzenia przydatności urządzeń mobilnych w badaniu jakości nawierzchni. Okazuje się, że wszystkie platformy oferują podobne możliwości w zakresie rejestrowania przyspieszenia. Najistotniejsze różnice dotyczą dodatkowych możliwości pomiarowych (inne typy czujników) oraz wspomagania programisty w analizowaniu rejestrowanych sygnałów.

Ważnym zagadnieniem jest też dokładność pomiarów, którą należy brać pod uwagę podczas analizy. Okazuje się, że urządzenie znajdujące się w spoczynku może rejestrować przyspieszenie liniowe wynoszące nawet $0,3 \text{ m/s}^2$

Literatura

- [1] Core Motion Framework Reference, iOS Developer Library, https://developer.apple.com/library/ios/documentation/coremotion/reference/coremotion_reference/_index.html
- [2] Ground Penetrating Radar used for new road surface quality control, <http://trafficsafe.org/index.php/ground-penetrating-radar-used-new-road-surface-quality-control-8437/> [2014.12.29]

- [3] iPhone Proximity Sensor, Stack Overflow, <http://stackoverflow.com/questions/165539/iphone-proximity-sensor> [2014.8.28]
- [4] Is there a light sensor on the iPhone, Stack Overflow, <http://stackoverflow.com/questions/6309643/is-there-a-light-sensor-on-the-iphone-and-if-so-how-can-i-read-the-data> [2014.8.28]
- [5] Microsoft.Devices.Sensors Namespace, Developer Network, <http://msdn.microsoft.com/en-us/library/microsoft.devices.sensors.aspx>
- [6] Proximity sensor implementation, Windows Phone | Dev Center https://dev.windowsphone.com/en-US/OEM/docs/Driver_Components/Proximity_sensor_implementation [2014.8.28]
- [7] Rappaport C., Holbrook D., Adams Ch., Busuico D., Doughty J., Road surface quality measurement using inexpensive radar, Geoscience and Remote Sensing Symposium (IGARSS), 2010 IEEE International, 4318-4321.
- [8] Sensor coordinate system, Windows Phone | Dev Center, https://dev.windowsphone.com/en-US/OEM/docs/Driver_Components/Sensor_coordinate_system [2014.8.28]
- [9] Sensors overview, Android Developers, http://developer.android.com/guide/topics/sensors/sensors_overview.html [2014.8.28]
- [10] Windows Phone Silverlight 8.1 apps, Developer Network, <http://msdn.microsoft.com/en-us/library/dn642082%28vs-vs.105%29.aspx>
- [11] Wykorzystanie laserów do pomiaru jakości i stanu nawierzchni dróg, http://cordis.europa.eu/news/rcn/35636_pl.html [2014.12.29]

Dr inż. Jakub Smolka

e-mail: jakub.smolka@pollub.pl

Pracownik naukowo-dydaktyczny w Instytucie Informatyki Wydziału Elektrotechniki i Informatyki Politechniki Lubelskiej. Tytuł magistra uzyskał na Politechnice Lubelskiej natomiast stopień doktora na Politechnice Śląskiej. Jego działalność naukowa związana jest z przetwarzaniem obrazów cyfrowych w szczególności z ich segmentacją oraz kompresją, a także analizą ruchu.

Uczestnik projektu Kwalifikacje dla rynku pracy - Politechnika Lubelska przyjazna dla pracodawcy współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego

Dr inż. Maria Skublewska-Paszowska

e-mail: maria.paszowska@pollub.pl

Pracownik naukowo-dydaktyczny pracujący w Instytucie Informatyki Wydziału Elektrotechniki i Informatyki Politechniki Lubelskiej. Tytuł magistra uzyskała na Politechnice Lubelskiej. Stopień doktora otrzymała na Politechnice Śląskiej. Działalność naukowa związana jest z: metodami akwizycji ruchu, analizą danych ruchu 3D, transformacjami falkowymi, miarami jakości oceny wizualnej obrazów i zastosowania ich w adaptacyjnej kompresji obrazów. Uczestnik projektu Kwalifikacje dla rynku pracy - Politechnika Lubelska przyjazna dla pracodawcy współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego

Mgr inż. Marcin Badurowicz

e-mail: m.badurowicz@pollub.pl

Od 2010 roku pracuje na Politechnice Lubelskiej, na stanowisku asystenta w Instytucie Informatyki. W ramach pracy naukowej zajmuje się tematyką inżynierii aplikacji mobilnych, w szczególności przeznaczonych na smartfony.

Uczestnik projektu Kwalifikacje dla rynku pracy - Politechnika Lubelska przyjazna dla pracodawcy współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego

Dr Edyta Łukasik

e-mail: e.lukasik@pollub.pl

Ukończyła studia matematyczne na UMCS w Lublinie. Tytuł doktora uzyskała na Wydziale Matematyki, Fizyki i Informatyki UMCS w Lublinie w roku 2007. Tytuł rozprawy doktorskiej: Metody iteracyjne dla nieliniowych regularnie osobliwych układów równań.

Od 1998 roku pracownik naukowy Politechniki Lubelskiej. W latach 1998-2007 zatrudniona na stanowisku asystenta, a od maja 2007 na stanowisku adiunkta w Instytucie Informatyki PL. Obszar zainteresowań naukowych to przede wszystkim języki programowania i algorytmizacja, struktury danych, metody numeryczne i metody optymalizacji, aplikacje mobilne.

otrzymano/received: 23.10.2014

przyjęto do druku/accepted: 30.12.2014