

Dokładność pomiaru kątów z wykorzystaniem czujników inercyjnych w trójwymiarowym układzie współrzędnych

Mateusz Miziołek

Politechnika Lubelska, Instytut Informatyki, Nadbystrzycka 36B, 20-618 Lublin, Polska

Streszczenie. Artykuł dotyczy dokładności pomiaru kątów z wykorzystaniem czujników inercyjnych w trójwymiarowym układzie współrzędnych. Jako system odniesienia wykorzystano system akwizycji ruchu firmy Vicon. W artykule opisano przebieg badania, aplikację, która posłużyła do zbierania danych ze smartfona. W artykule zawarto także metody, wzory oraz algorytm, których użyto, aby porównać uzyskane dane.

Słowa kluczowe: czujniki inercyjne; smartfon; akcelerometr; magnetometr; vicon nexus

Adres e-mail: mateusz.miziolek94@gmail.com

Angle measurement accuracy assessment using inertial sensors in three-dimensional coordinate system

Mateusz Miziołek

Institute of Computer Science, Lublin University of Technology, Nadbystrzycka 36B, 20-618 Lublin, Poland

Abstract. This article discusses angle measurement accuracy assessment using inertial sensors in three-dimensional coordinate system. Vicon's acquisition system was used as a reference system. The article describes conduct of the study and application that was used for collecting data from the smartphone. The article also contains methods, formulas and algorithms that were used to compare obtained data.

Keywords: inertial sensors; smartphone; accelerometer; magnetometer; vicon nexus

E-mail address: mateusz.miziolek94@gmail.com

1. Wstęp

Sprzedaż telefonów komórkowych w roku 2018 zanotowała spadek wynoszący 3% [1]. Mimo to, badania firmy IDC dowodzą, że niedługo można się spodziewać wzrostu sprzedaży na poziomie około 2,8% [2]. Dzisiejszy świat miniaturyzacji dotyka każdej dziedziny życia. Miniaturyzacja nie ominęła również medycyny. Jest obecna w e-rehabilitacji. Istnieją aplikacje wspomagające użytkownika w powrocie do pełnej sprawności po złamaniu ręki lub nogi [3]. Takie aplikacje wykorzystują czujniki inercyjne telefonów. Zalet takiego rozwiązania jest mnóstwo. Jednym z nich jest niewątpliwie ogromna wygoda. Sposobów wykorzystania czujników jest więcej, Zależą tylko od pomysłu i potrzeb użytkownika. Można tworzyć zróżnicowane aplikacje. Jedną z nich może być aplikacja do rozpoznawania rodzaju ruchów na podstawie odczytów sensorów urządzenia mobilnego.

W niniejszym artykule przedstawiono badanie dokładności pomiaru kątów za pomocą czujników inercyjnych. Jako system odniesienia obrano system akwizycji ruchu firmy Vicon. Jest to system stacjonarny. Przeznaczony do pracy w laboratorium. Dzięki wykorzystaniu czujników inercyjnych w telefonie komórkowym można przenieść małą część laboratorium i obliczać kąty w trójwymiarowej przestrzeni w razie takiej potrzeby.

2. Przegląd istniejących zastosowaniach czujników w smartfonie

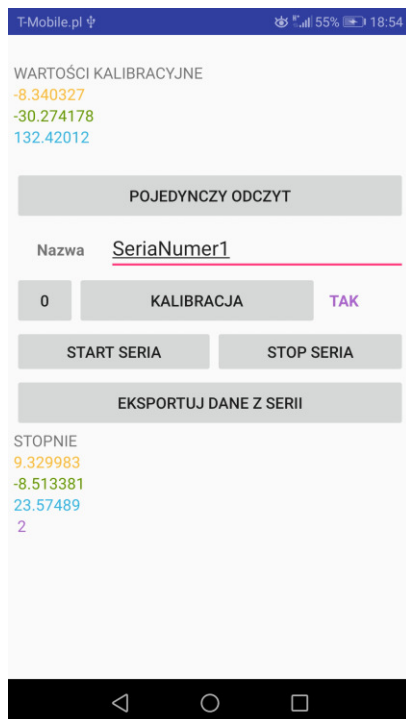
Dzisiejszy standardowy telefon komórkowy to nie tylko aparat do komunikacji, ale też ogromne źródło wielu danych. Smartfon w obecnych czasach posiada bardzo dużo czujników, w tym żyroskop, akcelerometr, czujnik światła, itp. Profesjonalne wykorzystanie takich czujników odkrywa zachowanie użytkownika smartfona. Najlepszym przykładem są aplikacje do joggingu. Telefony komórkowe mogą zastąpić również inne urządzenia takie jak kompas lub GPS. Dzisiaj osoba, która trafia do nowego miasta, nie musi już zaopatrywać się w mapę lub chociażby pytać o drogę. Wystarczy jej odpowiednia aplikacja. Tak samo odpowiednie oprogramowanie w praktycznie takim samym stopniu zastępuje nawigację samochodową. Do zastosowania czujników można podejść bardziej naukowo. Aplikacja łącząca działanie kilku czujników może rozpoznać styl jazdy samochodem i sklasyfikować kierowcę. Do tego typu badań wykorzystać można żyroskop oraz akcelerometr. Dzięki żyroskopowi określane jest położenie katowe. Natomiast atutem akcelerometru jest udostępnienie danych określających przyspieszenie liniowe. Na podstawie takich danych można określić, jak szybko auto jedzie, jak gwałtownie przyspiesza oraz hamuje. Zinterpretowanie takich danych może dać odpowiedź, jakim jest się kierowcą [4,5].

3. Aplikacja do pomiarów

Aplikacja mobilna została opracowana dla systemu operacyjnego Android. Korzysta z dwóch czujników - akcelerometru oraz magnetometru [6]. Łączy ona wyniki ich działania za pomocą metody *getRotationMatrix()* [7]. Wynikiem działania tej metody jest macierz obrotu. Dzięki metodzie *getOrientation()*, na podstawie macierzy obrotu, można obliczyć szukane kąty obrotu [8]. Są to:

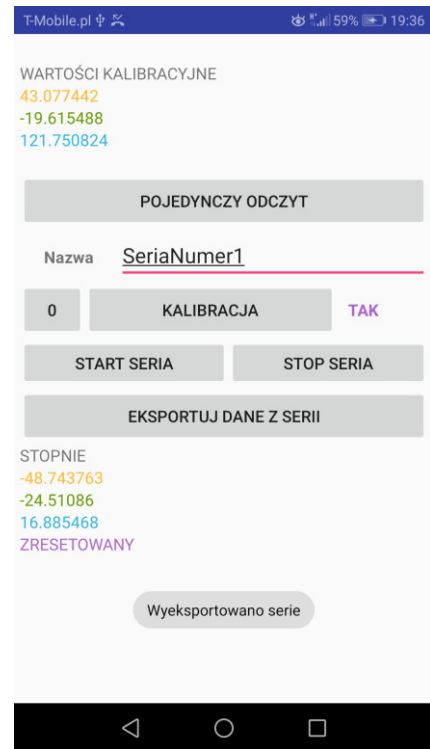
- azymut (ang. Azimuth), czyli kąt obrotu wokół osi $-z$. Wartość ta reprezentuje kąt pomiędzy osią y urządzenia i biegunem północnym magnesu.
- skok (ang. Pitch), czyli kąt obrotu wokół osi x . Wartość ta reprezentuje kąt pomiędzy płaszczyzną równoległą do ekranu urządzenia, a płaszczyzną równoległą do ziemi.
- rolka (ang. Roll), czyli kąt obrotu wokół osi y . Wartość ta reprezentuje kąt pomiędzy płaszczyzną prostopadłą do ekranu urządzenia, a płaszczyzną prostopadłą do podłoża.

Aplikacja posiada możliwość odczytu pojedynczego, jak i serii odczytów. Są one eksportowane do pliku tekstowego. Istnieje również możliwość kalibracji, czyli zmiany punktu odniesienia. Punkt odniesienia określa ustawienie punktu telefonu w przestrzeni. Natomiast punkt zerowy to punkt, gdy wszystkie kąty obrotu wynoszą zero stopni. Na rysunkach 1 oraz 2 przedstawiono interfejs aplikacji. Kolor jasno pomarańczowy określa kąt obrotu wokół osi y , kolor zielony określa kąt obrotu wokół osi x , a kolor jasnoniebieski określa obrót wokół osi z .



Rys. 1. Interfejs aplikacji

liki, jakie można uzyskać za pomocą tej aplikacji, to plik z pojedynczym zapisem, kalibracją oraz serią zapisów. Kilka rekordów zaprezentowano na przykładzie 1.



Rys. 2. Interfejs aplikacji po zebraniu serii danych

Przykład 1. Seria danych

```
1;18;-4.7431893;-40.2664;20.557674;||;-1.4575933;-40.2664;-48.070034;
2;42;-1.7075988;-42.43523;31.480942;||;1.5779971;-42.43523;-37.146767;
3;59;-4.544641;-42.18836;26.97633;||;-1.2590452;-42.18836;-41.65138;
4;76;-8.535869;-42.960033;20.504326;||;-5.2502728;-42.960033;-48.123383;
5;93;-1.3106618;-40.74677;31.838531;||;1.9749341;-40.74677;-36.789177;
6;109;-1.2624013;-41.606133;33.46056;||;2.0231946;-41.606133;-35.16715;
7;126;-2.963713;-43.51383;32.199604;||;0.32188302;-43.51383;-36.428104;
8;148;-4.324525;-40.69429;24.223255;||;-1.0389287;-40.69429;-44.404453;
```

W roli separatora występuje średnik, natomiast jako dodatkowy separator wykorzystano „||”. Pierwsza kolumna to indeks, czyli numer zapisu. Druga wartość to czas w milisekundach. Kolejne trzy wartości to obliczane kąty. Od lewej są to kąty obrotu wokół osi y , x oraz z . Po dodatkowym separatorze znajdują się trzy wartości skalibrowane.

4. Badanie

W celu zbadania dokładności kątów uzyskiwanych za pomocą czujników inercyjnych i systemu akwizycji ruchu w trójwymiarowym układzie współrzędnych przeprowadzono badania. W celach komunikacyjnych skonstruowano sieć lokalną, której zadaniem było umożliwienie komunikacji między telefonem a komputerem, na którym pracował system Vicon. Sam telefon został umieszczony na statywie, a na jego krańcach umieszczono markery w taki sposób, aby utworzyły bryłę imitującą telefon. Całość zaprezentowano na rysunku 3.



Rys. 3. Statyw z telefonem

Badanie przeprowadzono zgodnie z opracowanym wcześniej scenariuszem. Zakładał on badanie każdej osi oddzielnie. Zatem jest podzielony na kilka części. Punktem początkowym był punkt w, którym wartości kątów obrotowych wynosiły 0 stopni.

Dla obrotu wokół osi y określono następujące czynności:

1. Ustawienie telefonu w postaci leżącej tak, aby wartość Roll wynosiła 0 stopni.
2. Przekręcenie telefonu w prawą stronę zgodnie z ruchem wskazówek zegara według osi y, aż do zmiany położenia wartości Roll na wartość 180 stopni.
3. Wykonanie punktu 2 w odwrotnej kolejności. Z kąta o wartości 180 stopni do 0 stopni.
4. Wyeksportowanie pliku S1RCZ1.txt.
5. Obrót telefonu tak, aby zamienić jego krawędź górną z dolną. Ponowne umieszczenie na statywie.
6. Wykonanie punktów od 1 do 3 Różnicą jest to, że wartość Roll zamiast 180 stopni będzie wynosić - 180 stopni
7. Wyeksportowanie pliku „S1RCZ2.txt”.

Wynikiem były serie danych o nazwach S1RCZ1 oraz S1RCZ2.

1. Dla obrotu wokół osi x określono następujące czynności:
2. Ustawienie telefonu w postaci leżącej tak, aby wartość Pitch wynosiła 0 stopni.
3. Obrót telefonu w stronę „do siebie”. Przekręcając aż do momentu początku (punktu początkowego).
4. Odczekanie sekundy.
5. Obrót telefonu w stronę „od siebie”. Przekręcając, aż do momentu początku (punktu początkowego).

6. Wyeksportowanie plik „S1P.txt”.

Wynikiem była seria danych o nazwie S1P.

Dla obrotu wokół osi z określono następujące czynności:

1. Ustawienie telefonu w postaci leżącej tak, aby wartość Azimuth wynosiła 0 stopni.
2. Patrząc od góry. Przekręcanie telefonu w prawą stronę zgodnie z ruchem wskazówek zegara, aż do momentu początku (punktu początkowego).
3. Odczekanie sekundę.
4. Patrząc od góry. Przekręcanie telefonu w lewą stronę przeciwnie do ruchu wskazówek zegara, aż do momentu wyjścia.
5. Wyeksportowanie plik „S1Y.txt”.

Wynikiem była seria danych o nazwie S1Y.

4.1 Obliczanie wartości kątowych na podstawie nagrań z systemu akwizycji ruchu

Dane uzyskane z systemu Vicon to pozycje markerów w trójwymiarowym układzie współrzędnych. Należało je przeliczyć na kąty obrotowe, aby można było je porównać do kątów zebranych przez smartfon. W tym celu wykorzystano wzory i metody z artykułu Planning Algorithms profesora Stevena M. LaValle [9]. Na początku należało obliczyć macierz obrotu z trzech markerów, które były przymocowane do ekranu. Jako, że operowano w trójwymiarowym układzie współrzędnych macierz wynikowa ma rozmiar trzy na trzy. Co widać we wzorze 1.

$$\text{macierz} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{32} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \quad (1)$$

Na podstawie powyższej macierzy obliczono kąty rotacji wokół osi X (kąt α), Y (kąt β) oraz Z (kąt γ). Wzory dane są równaniami 2,3,4.

$$a = a \tan 2(r_{21}, r_{11}) \quad (2)$$

$$\beta = a \tan 2(-r_{31}, \sqrt{r_{32}^2 + r_{33}^2}) \quad (3)$$

$$\gamma = a \tan 2(r_{32}, r_{33}) \quad (4)$$

4.2 Wyznaczanie opóźnień

Kolejnym krokiem było określenie opóźnienia między zebranymi seriami danych. Niestety długości nagrań danych przez system Vicon oraz telefon komórkowy różnią się. Czas w danych pochodzących z urządzenia mobilnego zapisany jest w milisekundach, natomiast w systemie Vicon są to klatki. Jedna klatka to 10 milisekund. Aby wyrównać dane wykorzystano interpolację liniową. Następnie do obliczenia opóźnienia wykorzystywano korelacje. Przykładowy algorytm dla osi Y zaprezentowano na listingu 2. W tym skrypcie zmienna tablicowa OY zawiera serię danych, która

zawiera kąty rotacji pochodzące z systemu Vicon. Zmienna tablicowa OYm przechowuje dane pochodzące z telefonu. Zmienna tablicowa Frame zawiera czasy z nagrania z systemu Vicon. Natomiast t_org odpowiada czasom zarejestrowanym przez telefon. Skrypt jest kompatybilny ze środowiskiem Matlab.

Przykład 2. Skrypt wyliczający opóźnienie

```
len_vicon = size(OY,1);
len_mobile = size(OYm,1);
t = t_org/10;

%Wyrównanie przedziałów danych-POCZĄTEK
for i=len_vicon:-1:1
    if Frame(i) > t(len_mobile)
        ;
    else
        Frame = Frame(1:i);
        OY = OY(1:i);
        break;
    end
end
len_vicon = i ;
%Wyrównanie przedziałów danych-KONIEC

%Przepisanie i interpolacja danych-POCZĄTEK
OYminter = zeros(len_vicon,1);
for i=2:len_vicon
    for j=1:(len_mobile-1)
        if t(j) == Frame(i)
            OYminter(i) = OYm(j);
            break;
        elseif t(j+1) == Frame(i)
            OYminter(i) = OYm(j+1);
            break;
        elseif t(j) < Frame(i) && Frame(i) < t(j+1)
            OYminter(i) = OYm(j) +
            ((OYm(j+1)-OYm(j))/(t(j+1) - t(j))) * (Frame(i)-t(j));
            break;
        end
    end
end
end
%Przepisanie i interpolacja danych-KONIEC

%obliczanie opóźnienia-POCZĄTEK
shift = zeros(2);
[r,lag]=xcorr(OY,OYminter);
[max_r, max_r_index] = max(r);
shift(1,2) = -1*lag(max_r_index);
shift(2,1) = lag(max_r_index);

%obliczanie opóźnienia-KONIEC
```

4.3 Przesunięcie rozkładu oraz obliczenie średniego błędu

Najważniejszymi wynikami skryptu zamieszczonego w przykładzie 2 jest seria danych OYminter oraz przesunięcie. OYminter to seria danych z kątami obrotowymi wokół osi Y, która powstała z serii danych OYm po użyciu na niej algorytmu interpolacji liniowej. Przesunięcie należało wykorzystać, aby dopasować serie danych ze smartfona do serii danych z systemu Vicon. Dla tak przygotowanych serii danych obliczano średnią różnicę kątów. Jako, że są to kąty, a ich zasięg waha się od -180 stopni do 180 wykorzystano funkcję atan2 [10]. Równanie 5 przedstawia sposób obliczenia średniej wartości kąta.

$$a \tan 2\left(\sum_{j=1}^n \sin \alpha_j, \sum_{j=1}^n \cos \alpha_j\right) \quad (5)$$

Kąt α to różnica kątów między poszczególnymi próbkami w seriach z systemu Vicon oraz Smartfona.

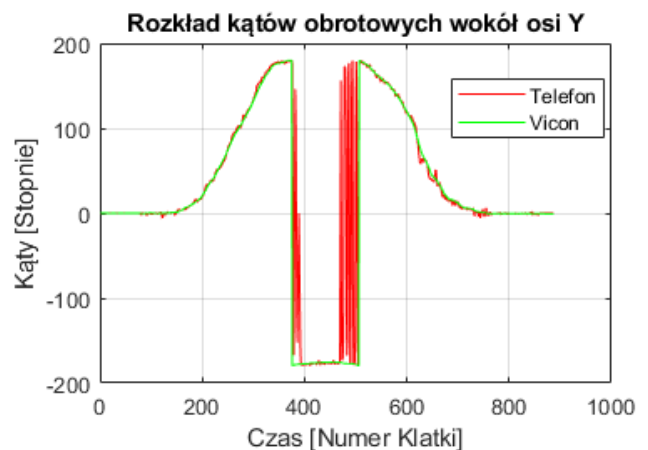
5. Wyniki

Badanie przeprowadzono zgodnie ze scenariuszem zamieszczonym w punkcie 4. Dla każdej części wykonano dwa podejścia. W każdym z nich na nowo ustawiano stanowisko badawcze. W każdym podejściu wykonano trzy powtórzenia scenariusza. Pierwsza część badała kąty obrotu wokół osi Y.

Tabela 1. Wyniki dla pierwszej części scenariusza

Numer Podejścia	Numer Próby	Średnia Różnica [°]
1	1	0,0499
	2	-0,2262
	3	-0,2759
2	1	-0,2856
	2	-0,1362
	3	0,0422

Rozkład kątów obrotu wokół osi Y próby trzeciej w podejściu pierwszym zaprezentowano na rysunku 4.



Rys. 4. Rozkład kątów rotacyjnych OY - podejście 1 – próba 3

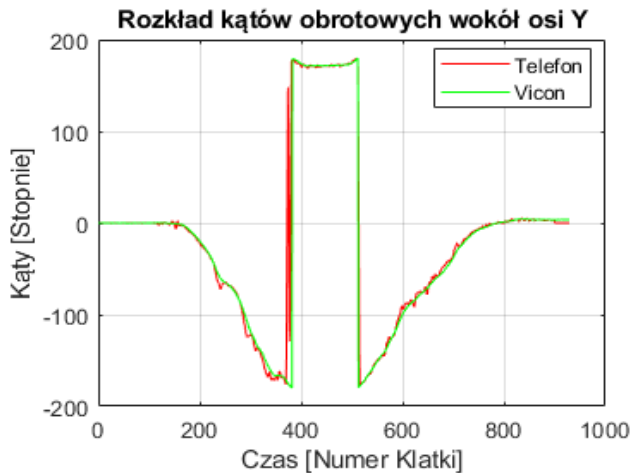
Analizując wyniki zamieszczone w tabeli 1 stwierdzono, że średnia różnica kątów waha się od -0,2856 do 0,2759 stopnia. Najbliżej zera jest wynik ostatniej próby podejścia drugiego i wynosi on 0,0422 stopnia. Jest to najlepszy wynik. W tym przypadku stwierdzono, że dokładność kątów jest bardzo wysoka.

Wyniki dla drugiej części scenariusza zamieszczono w tabeli 2. One również przedstawiają wyniki dla obrotu wokół osi Y.

Tabela 2. Wyniki dla drugiej części scenariusza

Numer Podejścia	Numer Próby	Średnia Różnica [°]
1	1	0,0396
	2	0,2897
	3	-0,2432
2	1	0,0803
	2	0,0997
	3	brak danych

Rozkład kątów obrotu wokół osi Y dla podejścia pierwszego przy pierwszej próbie zamieszczono na rysunku 5.



Rys. 5. Rozkład kątów rotacyjnych OY - podejście 1 – próba 3

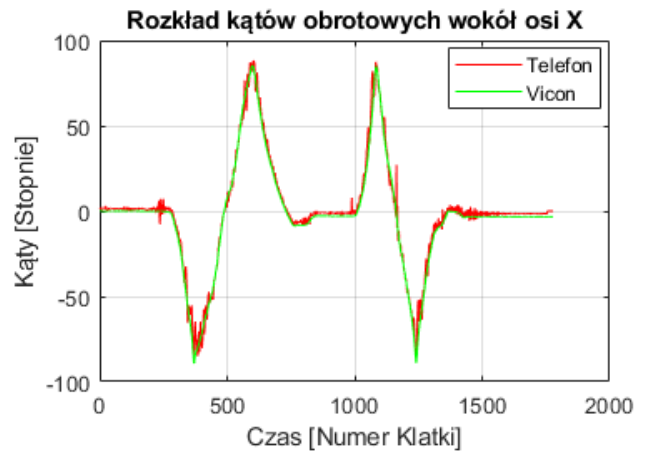
Analizując tabelę 2 w której zamieszczono wyniki dla drugiej części scenariusza stwierdzono, że zakres średniej dokładności waha się od -0,2432 do 0,2897 stopnia. Najbliższe zera jest wynik próby pierwszej w podejściu o tym samym numerze. Sama dokładność w tym przypadku wynosi 0,0396 stopnia. Dokładność kątów w tej części scenariuszu oceniono na bardzo dobrą.

Wyniki dla trzeciej części scenariusza zamieszczono w tabeli 3. Trzecia część przedstawia wyniki dla obrotu wokół osi X.

Tabela 3. Wyniki dla trzeciej części scenariusza

Numer Podejścia	Numer Próby	Średnia Różnica [°]
1	1	0,1981
	2	brak danych
	3	1,174
2	1	1,677
	2	-1,285
	3	-0,848

Rozkład kątów obrotu dla osi X w podejściu drugim dla próby pierwszej zamieszczono na rysunku 6.



Rys. 6. Rozkład kątów rotacyjnych OX - podejście 2 – próba 1.

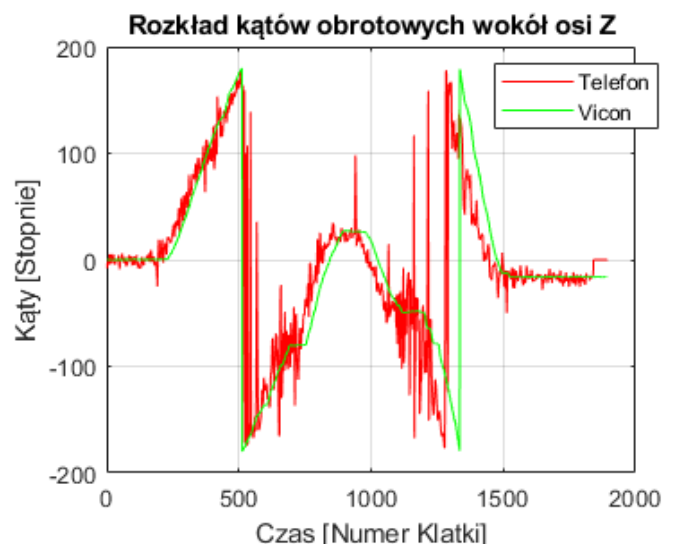
Analizując wyniki zamieszczone w tabeli 3 stwierdzono, że średnia różnica kątów waha się od -1,285 do 1,677 stopni. Najlepszym wynikiem jest wynik uzyskany w podejściu numer jeden w pierwszej próbie. Wynosi on 0,1981 stopnia. Dokładność w trzech przypadkach jest gorsza o ok 1 stopień niż w poprzednich częściach scenariusza. Mimo to dokładność jest dobra.

Wyniki dla czwartej części scenariusza zamieszczono w tabeli 4. Czwarta część przedstawia wyniki dla obrotu wokół osi Z.

Tabela 4. Wyniki dla drugiej części scenariusza

Numer Podejścia	Numer Próby	Średnia Różnica [°]
1	1	2,9883
	2	2,0259
	3	-0,7042
2	1	-2,6156
	2	-1,4688
	3	2,4699

Rozkład kątów obrotu dla osi Z w podejściu drugim dla próby pierwszej zamieszczono na rysunku 7.



Rys. 7. Rozkład kątów rotacyjnych OZ - podejście 2 – próba 1

Analizując wyniki, które zamieszczono w tabeli 4. Średni wynik różnicy kątów waha się od -2,6156 do 2,9883 stopni. Najlepszy wynik też jest gorszy o ok 0,6 stopnia w stosunku do poprzednich wyników. Wynosi on -0,7042 stopnia. Badając wykres zamieszczony na rysunku 7 stwierdzono, że jest poszarpany. Wyniki jak i rozkład prawdopodobnie spowodowany jest tym, że badanie było przeprowadzone po zbyt małym łuku.

6. Wnioski

Analizując wszystkie wyniki widać, iż wahają się one od -2,6156 do 2,9883 stopni. Natomiast najlepszy wynik to 0,0396 stopnia. Wpływ na wyniki z pewnością miała częstotliwość próbkowania telefonu komórkowego, która wahała się od 51 do 67. Początkowe wyniki ze smartfona miały charakter stochastyczny. Natomiast Vicon pracował z częstotliwością 100 próbek na sekundę. Jego wyniki były deterministyczne, czyli stałe. Jedna próbka co 10 milisekund. Drugim istotnym czynnikiem, który miał miejsce był czynnik ludzki. Telefon obracano ręcznie za pomocą uchwytu na statywie. Mimo iż nogi statywu były wsparte całość lekko zmieniła swoje położenie. Promień był jedynie kilkucentymetrowy, co z pewnością powiększyło wahania wartości kątów. Lepsze wyniki można było uzyskać wydłużając promień obrotu. Dlatego, aby wyniki były bardziej wiarygodne wykonywano badania kilkakrotnie. Średnia dokładność wyniosła około jednego stopnia. Oceniono ten wynik jako bardzo dobry.

Literatura

- [1] S. Srivastava, Global Smartphone Market Declined YoY For Second Successive Quarter, 2018.
- [2] A. Scarsella, M. Chau, R. Reith, M. Shirer, Smartphones Hit Pivotal Stage as Worldwide Shipment Volumes Decline 0.5% in 2017, But Return to Growth is Expected, According to IDC, 2018.
- [3] <https://play.google.com/store/apps/details?id=pl.ortomedSPORT.mobile> (dostęp marzec 2018).
- [4] M. Van, M. Sujitha, M. Trivedi, Driver Classification and Driving Style Recognition using Inertial Sensors, IEEE Intelligent Vehicles Symposium (IV) Australia, 2013.
- [5] D. Johnson, M. Trivedi, Driving Style Recognition Using a Smartphone as a Sensor Platform, International IEEE Conference on Intelligent Transportation Systems Washington, DC, USA., 2011.
- [6] <https://developer.android.com/reference/android/hardware/SensorManager> [03.2018].
- [7] [https://developer.android.com/reference/android/hardware/SensorManager#getRotationMatrix\(float\[\],%20float\[\],%20float\[\],%20float\[\]\)](https://developer.android.com/reference/android/hardware/SensorManager#getRotationMatrix(float[],%20float[],%20float[],%20float[])) [03.2018].
- [8] [https://developer.android.com/reference/android/hardware/SensorManager.html#getOrientation\(float\[\],%20float\[\]\)](https://developer.android.com/reference/android/hardware/SensorManager.html#getOrientation(float[],%20float[])) [03.2018].
- [9] S. LaValle, Planning Algorithms, Cambridge University Press 2006.
- [10] https://en.wikipedia.org/wiki/Mean_of_circular_quantities [03.2018].