

4EF-DI

Szymon Starzak

nr albumu: 127261

POLITECHNIKA RZESZOWSKA
im. Ignacego Łukasiewicza

Wydział Elektrotechniki i Informatyki

Projekty



*Sprawozdanie z projektu: „Mobilny program: Obliczanie
objętości bali drzew”*

1. Założenia projektu

- Celem projektu było stworzenie aplikacji na urządzenie mobilne, która na podstawie wykonanych 3 zdjęć ma obliczyć objętość bali drzew znajdujących się na wagonie.

2. Przebieg planowanej realizacji projektu.

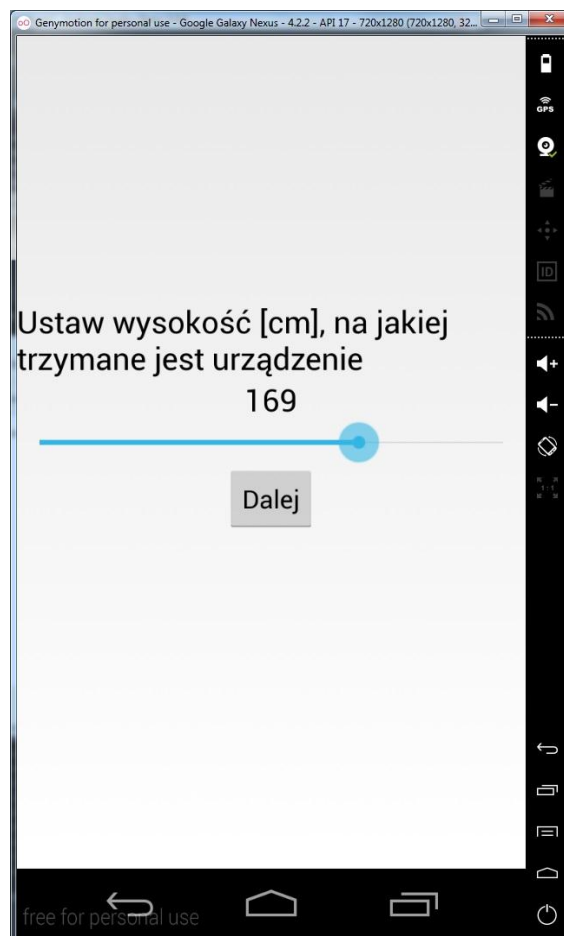
- Odczyt rozdzielczości aparatu.
- Ustalenie wysokości na jakiej znajduje się urządzenie względem podłoża wagonu.
- Pomiar odległości między urządzeniem mierniczym (np. telefonem, tabletem) a boczną stroną wagonu. W zależności od położenia urządzenia odczytywane są wartości kątów nachylenia urządzenia:
 - a) dla położenia *screenOrientation="portrait"* odczytywana jest wartość *Pitch* sensora *Sensor.TYPE_ORIENTATION*
 - b) dla położenia *screenOrientation="landscape"* odczytywana jest wartość *Roll* sensora *Sensor.TYPE_ORIENTATION*
- Pomiar długości wagonu. Tym razem wykorzystany zostaje różnica kątów między wartościami *Azimuth* sensora *Sensor.TYPE_ORIENTATION*
- Wykonanie zdjęcia frontowej strony wagonu wbudowaną kamerą urządzenia.
 - a) Służyć do tego ma dostarczana wraz ze środowiskiem klasa *Camera*, wywołana jako nowa aktywność *MediaStore.ACTION_IMAGE_CAPTURE*. Zaraz po wykonaniu zdjęcia w metodzie rezultatu aktywności (*onActivityResult()*) ma nastąpić zapisanie zdjęcia w z góry określonej lokalizacji. Następnie otwierana jest nowa aktywność która na podstawie zrobionego zdjęcia wykorzystuje klasy biblioteki open-cv w celu wyodrębnienia bali drzew od tła.
 - b) Wykorzystanie klas biblioteki open-cv do wyodrębnienia bali drzew od tła podczas wykonywania zdjęcia.
- Na podstawie informacji o rozdzielczości zdjęcia, odległości z jakiej wykonane zostało zdjęcie, oraz ilości wyodrębnionych pikseli obliczane jest pole powierzchni w [m²]
- Czynność wykonywane dla frontowej strony wagonu powtarzamy dla tylnej strony wagonu.

- Ostatnim etapem w obliczaniu objętości jest powiązanie otrzymanych informacji i obliczenie objętości jako średnia pól powierzchni frontowej i tylnej stron wagonu, zwielokrotniona o długość bali.

3. Przebieg realizacji projektu: etapy pomiaru, modele matematyczne,

- Ustalenie wysokości urządzenia mierniczego względem podłoża wagonu. Ta wartość potrzebna jest do późniejszego obliczenia dystansów i może być zmieniana nawet kilka razy.

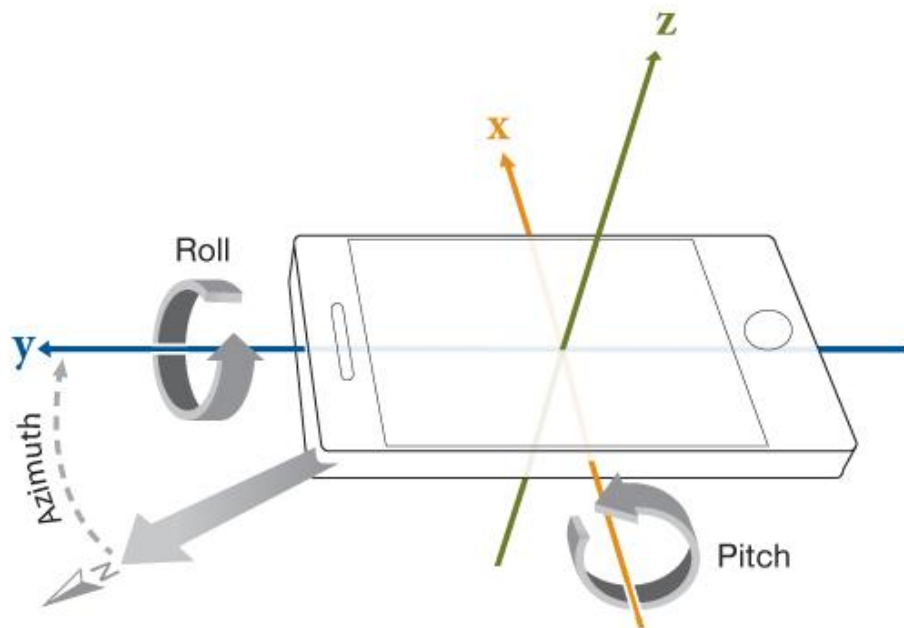
a) Wysokość podaje użytkownik



- Pomiar odległości między urządzeniem miernicznym (np. telefonem, tabletem) a boczną stroną wagonu. W zależności od położenia urządzenia odczytywane są wartości kątów nachylenia urządzenia:

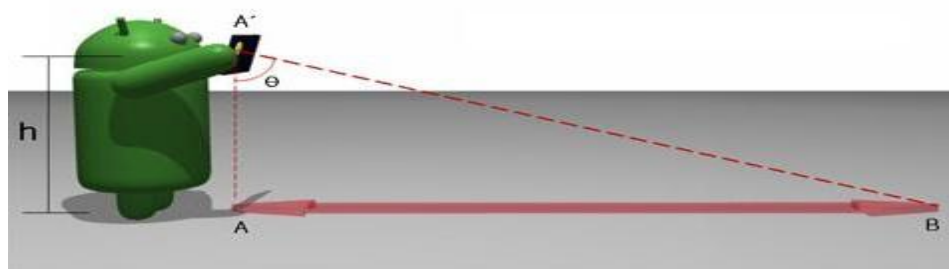
a) dla położenia *screenOrientation="portrait"* odczytywana jest wartość *Pitch* sensora *Sensor.TYPE_ORIENTATION*

- b) dla położenia *screenOrientation="landscape"* odczytywana jest wartość *Roll* sensora *Sensor.TYPE_ORIENTATION*



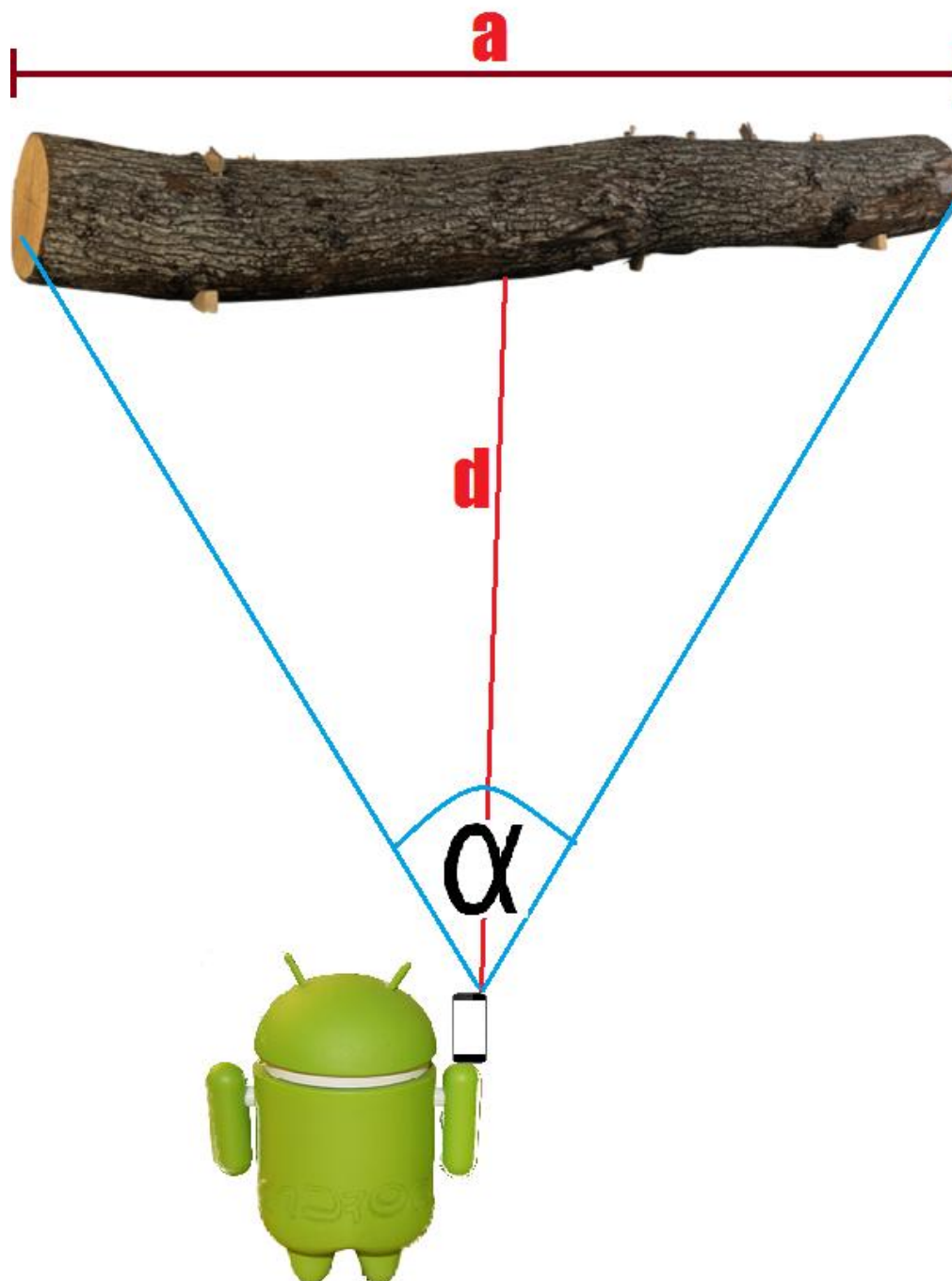
Sensory położenia urządzenia

- c) Wartość odcinka $|AB|$ otrzymujemy z prostej zależności matematycznej:

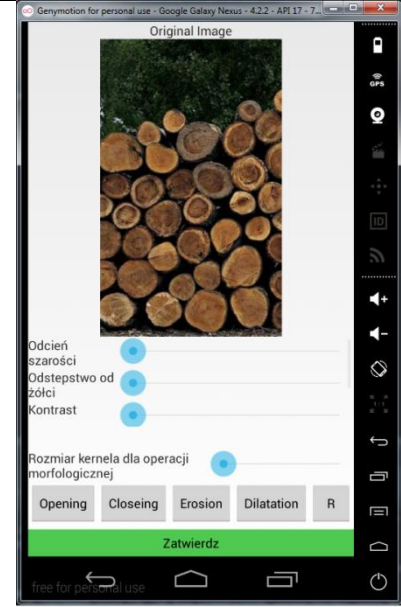
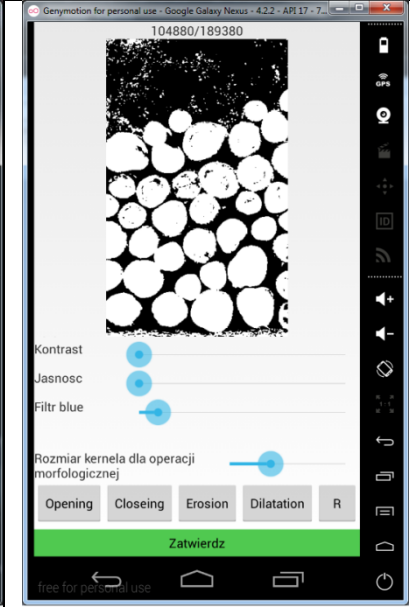
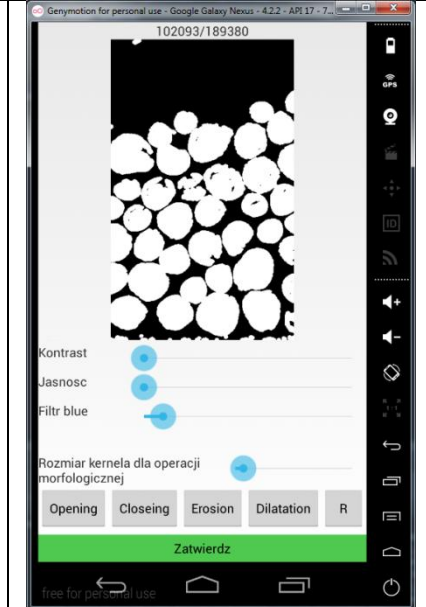
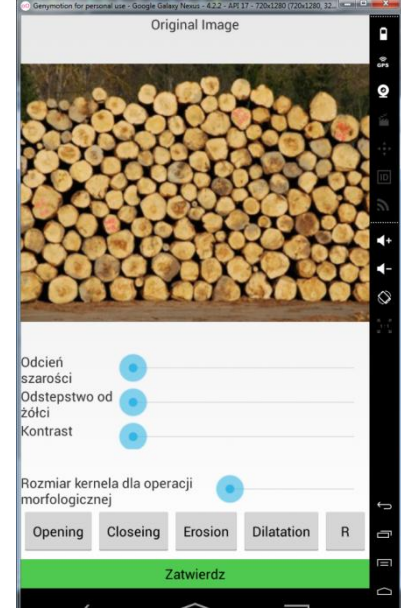
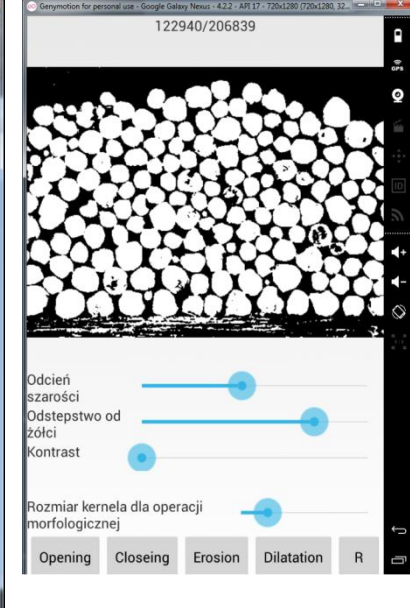
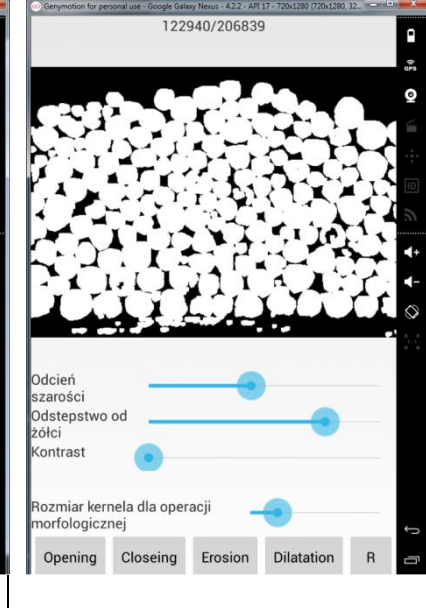


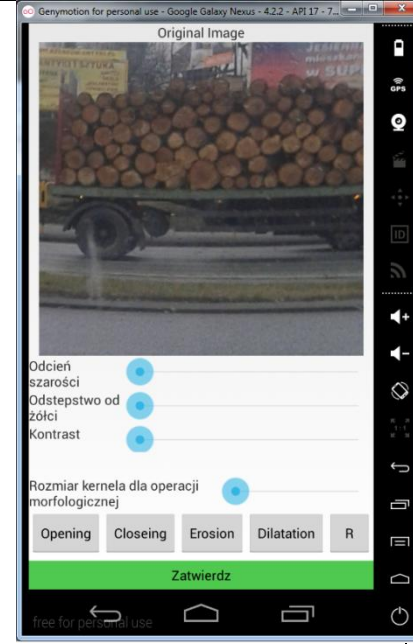
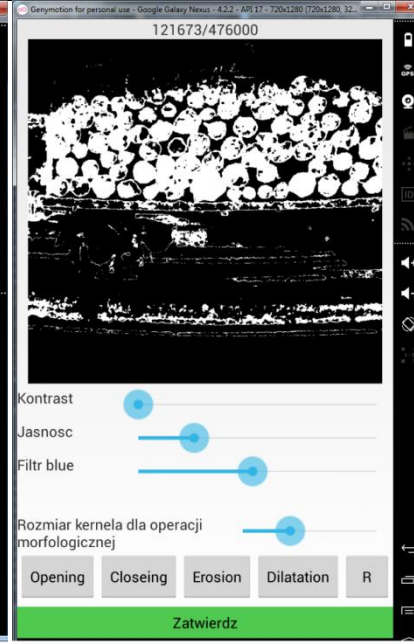
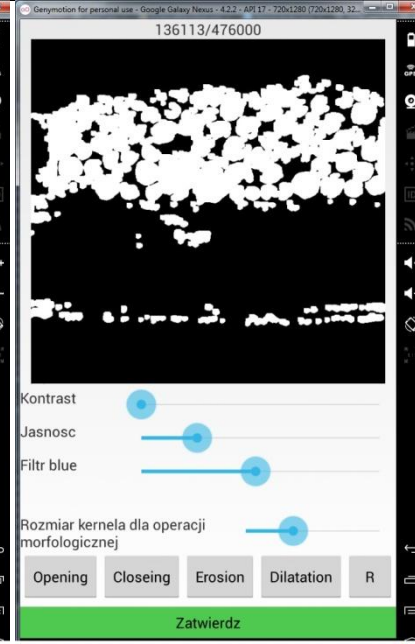
$$|AB| = h * \tan \theta$$

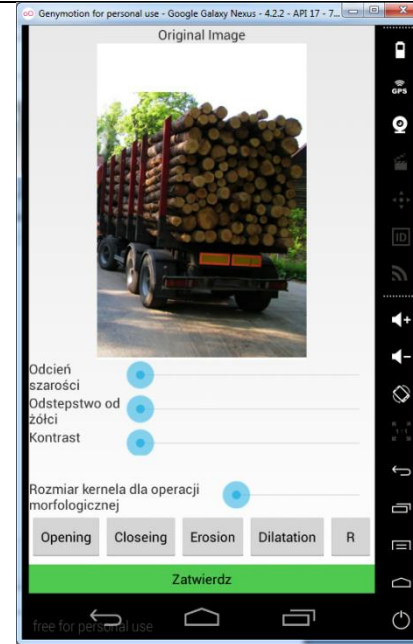
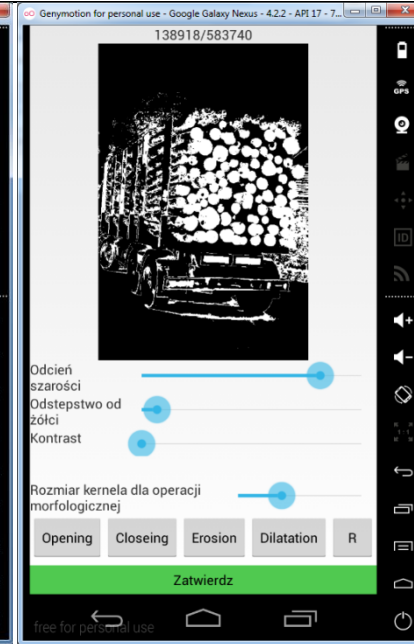
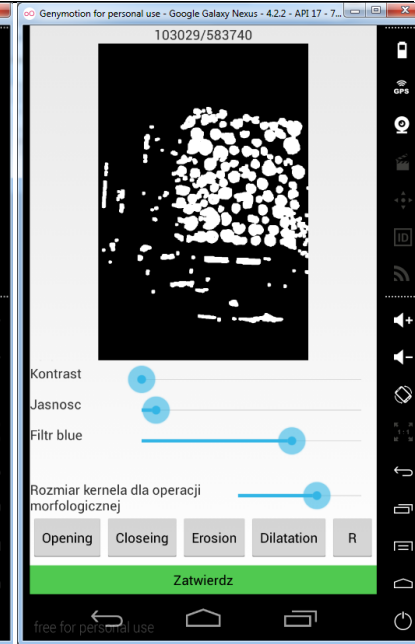
- Pomiar długości wagonu. Tym razem wykorzystany zostaje różnica kątów między wartościami *Azimuth* sensora *Sensor.TYPE_ORIENTATION*



- Wartość długości wagonu otrzymujemy z równania: $a = 2 * d * \tan \frac{\alpha}{2}$
- Obliczanie pola powierzchni słoików drzew na podstawie przeprowadzonych operacji morfologicznych na zbinaryzowanym obrazie. Poniżej pokazano kilka przykładów wyznaczania konturów słoików, oraz wypełniania ich.

		
<p>Obraz wczytany do aplikacji</p>	<p>Obraz zbinaryzowany względem odcieni koloru żółtego</p>	<p>Obraz po przeprowadzeniu dostępnych operacji morfologicznych</p>
		
<p>Obraz wczytany do aplikacji</p>	<p>Obraz zbinaryzowany względem odcieni koloru żółtego</p>	<p>Obraz po przeprowadzeniu dostępnych operacji morfologicznych</p>

		
<p>Obraz wczytany do aplikacji. Zdjęcie wykonane zza plastikowej szyby.</p>	<p>Obraz zbinaryzowany względem odcieni koloru żółtego</p>	<p>Obraz po przeprowadzeniu dostępnych operacji morfologicznych</p>

		
<p>Obraz wczytany do aplikacji</p>	<p>Obraz zbinaryzowany względem odcieni koloru żółtego</p>	<p>Obraz po przeprowadzeniu dostępnych operacji morfologicznych</p>

Obraz wczytany do aplikacji	Obraz zbinaryzowany względem odcieni koloru żółtego	Obraz po przeprowadzeniu dostępnych operacji morfologicznych

W momencie, gdy użytkownik wykona wszystkie pomiary, z poziomu Menu głównego możliwy jest odczyt wszystkich zmierzonych oraz obliczonych parametrów. Przedstawia to poniższy rysunek:



4. Technologie

- Środowisko programistyczne Android Studio (Beta) 0.8.6
- Język programowania: Java
- Framework SugarORM - framework dla bazy danych przechowującej wartości poszczególnych pomiarów.
- Otwarta biblioteka open-cv dla języka Java

5. Opis utworzonych klas Java

- CameraPreview - klasa rozszerzająca interfejs SurfaceView. To na niej wyświetlany jest podgląd z kamery, oraz crosshair służący do pomiarów.
- Measurement - klasa rozszerzająca SugarRecord. Pełni rolę pojedynczej tabeli w bazie danych, w której przechowywane są wartości pomiarów.
- MySensors - klasa pomocnicza, rejestrująca sensory. Dzięki swoim metodom, umożliwia przejrzysty odczyt wartości sensorów.
- ObjectLength - klasa obliczająca długość obiektu na podstawie otrzymanych danych w konstruktorze. Jej główna metoda to `getLength()` zwracająca wartość długości w [cm].
- ObjectDistance - klasa obliczająca dystans na podstawie wysokości i kąta nachylenia.
- IdentificationAngleHelper - prosta klasa zwracająca wartość kąta nachylenia w zależności od orientacji telefonu.

6. Aktywności

- MainActivity - główne okno programu. Z niego dostępne są wszystkie pomiary, oraz na nim odczytywana jest wartość objętości.
- GetDistanceActivity - aktywność miernicza. Mierzy i zapisuje w bazie wartość odległości od obiektu.
- GetHeightActivity - Jej jedynym celem jest pobranie od użytkownika wysokości na jakiej znajduje się urządzenie.
- GetTreeLengthActivity - aktywność miernicza. Mierzy i zapisuje w bazie wartość długości od obiektu.
- HowToMeasureActivity - aktywność informacyjna. Zawiera zestaw instrukcji jak wykonać pomiar.

- DendrochronologyVolume - obliczanie pola powierzchni słoików na podstawie obróbki zdjęć

7. Wnioski

- Celem projektu była realizacja problemu mierzenia objętości bali drzew. Ze względu na problemy występujące z dostępnym fizycznym urządzeniem, w projekcie zabrakło własnej implementacji modułu dotyczącego zapisu obrazu z kamery. Zdjęcie należy wykonać dostępną aplikacją "Aparat" lub przesłać z komputera dowolny obraz na urządzenie mobilne, a następnie wczytać je do aplikacji. Trudnym zadaniem okazał się dobór uniwersalnych parametrów filtrujących dla każdego możliwego zdjęcia. Dlatego zaimplementowano odpowiednie suwaki, dzięki którym, użytkownik sam dobiera parametry filtrujące. Czynność tą może wykonywać dowolną ilość razy, a każda zmiana powoduje odświeżenie obrabianego obrazu. Operacje morfologiczne przeprowadzane są na podstawie wprowadzonego rozmiaru dla kernela o kształcie elipsy. Dlatego mniejsze obiekty od ustawionego rozmiaru kernela, lub obiekty o innym kształcie, w przypadku operacji erozji, lub otwarcia są usuwane.