**1. Wprowadzenie**

**Witam Państwa**

Nazywam się Michał Kapiczyński i zaprezentuję dzisiaj Państwu efekty mojej pracy dyplomowej o temacie:

"Algorytm wyznaczania i wizualizacji obszaru przeszukanego wraz z implementacją przy wykorzystaniu systemu OpenStreetMap".

Praca powstała pod kierownictwem dr. Pawła Wnuka.

Temat ten jest bezpośrednio powiązany z bezzałogowymi statkami powietrznymi, powszechnie znanymi jako drony. Tytułowy algorytm ma za zadanie wyznaczanie obszaru przeszukanego właśnie przez te pojazdy.

**2. Cel pracy**

Moja praca skupia się na analizie i realizacji rozwiązania, wspomagającego wykorzystanie bezzałogowych statków latających.

Celem pracy było opracowanie algorytmu umożliwiającego wizualizację obszaru przeszukanego, rozumianego jako reprezentacja terenu zarejestrowanego przez kamerę zamontowaną na dronie, na podstawie zebranych danych geolokalizacyjnych pojazdów i znajomości parametrów kamer do nich zamontowanych oraz zaprojektowanie i implementacja systemu informatycznego realizującego tę funkcję.

Na prezentacji mogą Państwo zobaczyć koncepcyjną strukturę systemu.

Dron na podstawie sygnału GPS określa swoje położenia, a następnie przesyła niezbędne dane do serwera. Te je analizuje, przeprowadza obliczenia związane z wyznaczeniem obszaru przeszukanego i rozsyła odpowiednie dane do aplikacji klienckich.

**3. Wstęp teoretyczny**

Pierwszym analizowanym elementem systemu był geolokalizator mający za zadanie zbieranie danych geolokalizacyjnych pojazdów. Jak się okazało geolokalizacja może odbywać się w stosunku do różnych układów odniesienia. Najczęściej do układu związanego z geoidą ziemską bądź elipsoidą.

W skrócie:

Geoida to teoretyczna powierzchnia ekwipotencjalna, pokrywająca się w przybliżeniu z powierzchnią oceanów przy pełnej równowadze znajdujących się w nich mas wody.

Elipsoida ziemska jest zaś spłaszczoną elipsoidą obrotową, której powierzchnia jest najbardziej zbliżona do powierzchni geoidy na całej Ziemi.

Elipsoida jest zatem przybliżeniem geoidy, łatwiejszym do wykorzystywania w obliczeniach.

Istnieje wiele modeli geoidy i elipsoidy Ziemskiej jednak w pracy omówiono dwie najpopularniejsze czyli geoidę EGM96 oraz elipsoidę WGS84.

Znany nam wszystkim dobrze system nawigacji satelitarnej określa położenie obiektów w stosunku do elipsoidy ziemskiej. To znaczy pomiar z jego wykorzystaniem daje nam długość i szerokość geograficzną określone w stosunku do elipsoidy oraz wysokość elipsoidalną.

Jednak poza aktualnym położeniem dronów do analizy tego co kamera na nich zamontowana zarejestrowała, a czego nie potrzebny był jakiś model powierzchni ziemskiej.

Tu z pomocą przyszło mi pojęcie numerycznych modeli terenu oraz misja o dumnie brzmiącej nazwie Shuttle Radar Topography Mission (w skrócie SRTM), przeprowadzona przez Narodową Agencję Aeronautyki i   
Przestrzeni Kosmicznej Stanów Zjednoczonych NASA, Niemiecką Agencję Kosmiczną DRL oraz Włoską Agencję Kosmiczną ASI w 2000 roku. Misja ta miała na celu zebranie najbardziej kompleksowego i dokładnego numerycznego modelu Ziemi.

Po zakończeniu misji, numeryczne modele, będące jej efektem zostały umieszczone w Internecie i są bezpłatnie dostępne dla każdego w formie plików reprezentujących rastrowy, numeryczny model terenu.

Pliki z misji SRTM ze względu na bezpłatność i łatwą dostępność zostały wykorzystane przeze mnie w pracy. Zatem mogę skromnie mówić, iż współpracowałem z NASA.

Współrzędne poziomie, czyli długość i szerokość w modelu SRTM są również podane w odniesieniu do elipsoidy WGS84, jednak wysokości są już podane w stosunku do geoidy EGM96.

W pracy dokonano analizy tego problemu i zaproponowano rozwiązanie w postaci wyznaczenia średniej wartości undulacji, czyli różnicy między geoidą i elipsoidą, na terenie na którym projektowany system miał działać, lub jeszcze lepiej dokonania dokładnych pomiarów i stworzenia całej tablicy wartości undulacji dla różnych położeń.

**4. Algorytm**

Przejdę teraz do omówienia sposobu realizacji algorytmu wyznaczania obszaru przeszukanego.

Algorytm za dane wejściowe przyjmował położenie geograficzne drona pobrane z użyciem systemu nawigacji satelitarnej GPS, dane modelujące powierzchnie ziemską z misji SRTM oraz kąt widzenia kamery na nim zamontowanej. Przy czym w pracy przyjęto, iż kamera jest skierowana pionowo w dół oraz że rejestruje obraz o powierzchni kołowej.

Docelowy algorytm został podzielony na trzy części:

* algorytm wyznaczania otoczki obszaru przeszukanego,
* algorytm wyznaczania obszarów wewnątrz otoczki niezarejestrowanych przez kamerę oraz
* algorytm łączenia pojedynczych obszarów w całość.

Pierwsze dwie części algorytmu zostały rozwiązane z wykorzystaniem heurystyki, a trzeci algorytm wykorzystał naukową teorię kształtów alfa.

Teraz postaram się Państwu przybliżyć sposób rozwiązania każdego z tych algorytmów.

**I algorytm wyznaczania otoczki obszaru przeszukanego:**

Pokazany rysunek przedstawia koncepcję działania algorytmu wyznaczania otoczki obszaru przeszukanego.

Punkt **D** odpowiada położeniu drona. Wysokość **H**  jest wysokością statku nad ziemią. Znając położenie i wysokość drona nad ziemią oraz kąt widzenia kamery **α** rozpoczęto wyznaczanie kolejnych okręgów reprezentujących potencjalną otoczkę zarejestrowanego obszaru. Wyznaczanie okręgów rozpoczęto od wysokości drona pomniejszonej o skok **dh,** którego wartość na drodze eksperymentalnej dobrano na 2 m. Znając wartość **dh** oraz kąt **α** widzenia kamery wyznaczono wartość promienia **r** zgodnie ze wzorem:

W dalszym wyznaczono punkty na okręgu o promieniu **r** iśrodku w punkcie odpowiadającym położeniu drona. Liczba wyznaczonych punktów, dobrana na drodze eksperymentalnej, to 360 punktów rozmieszczonych na okręgu co 1o.

Dla każdego z dobranych punktów dokonano następnie porównania jego wysokości (równej wysokości drona pomniejszonej o aktualne **dh**) z wysokością modelu Ziemi w danym punkcie. Jeśli wysokość punktu na okręgu była mniejsza lub równa wysokości modelu punkt zostawał uznany za punkt otoczki obszaru przeszukanego, a wartość stopnia na okręgu, odpowiadająca temu punktowi zostawała usunięta z dalszych rozważań. Następnie wartość **dh** zostawała zwiększona, a algorytm ten był powtarzany, aż do momentu, gdy dla każdej z 360 wartości stopni na okręgu znaleziono odpowiadający punkt modelu.

Tutaj widzimy przykładowy wynik działania algorytmu. Fragmenty okręgów o mniejszym promieniu reprezentują tereny o większej wysokości, a te o większym promieniu o mniejszej. W ten sposób można łatwo odczytać gdzie wystąpił spadek, a gdzie wzniesienie terenu i którą część terenu kamera zarejestrowała, a która została zasłonięta.

**II część algorytmu**

Druga część algorytmu jest rozwinięciem algorytmu podstawowego. Po wyznaczeniu otoczki obszaru przeszukanego dla kąta widzenia kamery **α**, powtórzono tę samą operację dla kąta kamery zmniejszonego o określoną stałą wartość **β**, jednocześnie zapamiętując wyznaczoną otoczkę dla poprzedniej wartości kąta kamery. Iterację tę powtarzano, aż do osiągnięcia kąta zerowego. W efekcie po każdej iteracji otrzymywano dwie otoczki: jedną dla kąta widzenia kamery **γ** (kąt kamery z poprzedniej iteracji) oraz drugą dla kąta **γ - β.**

Dla obu otoczek porównano wysokości odpowiednich punktów, leżących na odpowiadających promieniach, czyli leżących na tej samej współrzędnej kątowej okręgu. Jeśli wysokość punktu **B** (leżącego na otoczce "γ - β") jest większa od wysokości punktu **A** (leżącego na otoczce " γ ") oznacza to, że między tymi dwoma punktami występuje wzniesienie. Wówczas zaczynając od wysokości punktu **B** powtórzono operację wykorzystaną przy wyznaczaniu otoczki, czyli stopniowe zmniejszanie wysokości o stały skok **dh**, aż do wysokości punktu **A**. Dla każdego "schodka" dokonano porównania wysokości punktu na promieniu wodzącym kamery o kącie **γ - β** z wysokością modelu punktu o tej samej długości i szerokości geograficznej. Jeśli wysokość punktu na promieniu wodzącym była większa od wysokości punktu modelowego zostawał on uznany za niewidoczny.

Po osiągnięciu wysokości punktu **A** z listy niewidocznych punktów wybrano punkt pierwszy i ostatni, czyli dwa punkty definiujące linię, będącą reprezentacją niewidocznego obszaru. Operacja ta była powtarzana dla każdego z 360 kątów, dla których wyznaczano punkty otoczki.

W ten sposób otrzymano listę dziur w postaci linii, które sumarycznie reprezentują powierzchnię obszarów niezarejestrowanych przez kamerę.

Odpowiednie pogrupowanie punktów reprezentujących sąsiednie linie pozwoliło na wyznaczenie zbiorów punktów reprezentujących poszczególne dziury. Następnie dla każdego zbioru wyznaczono wielokąt, zawierający wszystkie punkty zbioru zgodnie z algorytmem wyznaczania otoczki wklęsłej, który zostanie za moment omówiony wraz z trzecią częścią algorytmu.

**III część algorytmu**

W zaprojektowanym systemie istnieje podział obszaru przeszukanego na obszar ostatnio przeszukany i obszar dotychczas przeszukany. Obszar ostatnio przeszukany to obszar zarejestrowany przez kamerę dla ostatniego znanego położenia drona, podczas gdy obszar dotychczas przeszukany jest sumą wszystkich obszarów zarejestrowanych do tej pory, nie wliczając w to obszaru ostatnio przeszukanego.

Każda zmiana położenia drona skutkuje wyznaczeniem nowego obszaru przeszukanego, który zgodnie z omówionym wcześniej algorytmem składa się z listy punktów, uszeregowanych według współrzędnych biegunowych, tworzących otoczkę wyznaczonego obszaru oraz listy punktów definiujących dziury w tymże obszarze. Aby być w stanie prawidłowo wizualizować obszar przeszukany nie tylko dla pojedynczych położeń drona, ale dla całych śladów potrzebna jest metoda na łączenie tych obszarów w jeden.

Zadanie na pierwszy rzut oka mogłoby się wydawać trywialne, jednak po dokładniejszej analizie okazuje się, że wcale takie nie jest. Dlaczego? Ponieważ pojedyncze obszary są często wielokątami wklęsłymi i ich połączenie w jeden wymaga algorytmu wyznaczającego otoczkę wklęsła dla danego zbioru punktów.

Z matematycznego punktu widzenia dla dowolnego zestawu punktów P nie istnieje jednoznaczny wielokąt wklęsły zawierający wszystkie te punkty. Zazwyczaj istnieje wiele takich wielokątów. Problem ten został zobrazowany na rysunku widocznym na ekranie.

W celu rozwiązania tego problemu w pracy wykorzystano naukową koncepcję kształtu α.

**Kształt α** dowolnego zbioru punktów jest subgrafem triangulacji Delaunay tych punktów, takim że dwa punkty stanowią jego krawędź jeśli istnieje pusta kula o promieniu 1/α stykająca się z tymi dwoma punktami.

**Triangulacja Delaunay** zbioru punktów P jest takim podziałem obszaru wyznaczonego przez ten zbiór na trójkąty, że żaden z punktów tego zbioru nie znajduje się we wnętrzu któregokolwiek z okręgów opisanych na trójkątach powstałych podczas triangulacji.

Kształt alfa jest generalizacją otoczki wypukłej, to znaczy każda otoczka wypukła jest kształtem alfa, ale nie każdy kształt alfa jest otoczką wypukłą.

W praktyce w celu połączenia dwóch obszarów w jeden wykonano takie operacje:

1. Dla zbiorów reprezentujących obszar dotychczas i ostatnio przeszukany przeprowadzono densyfikację punktów, czyli takie zwiększenie gęstości ich rozmieszczenia, aby dowolne dwa, kolejne punkty zbioru były od siebie oddalone o wartość nie większą niż wartość parametru λ, którego wartość została dobrana eksperymentalnie.
2. Następnie dokonano połączenia zbiorów wejściowych w sumaryczny zbiór punktów P.
3. Dla zbioru punktów P przeprowadzono triangulację Delaunay.
4. Dokonano wyznaczenia grafu, reprezentuącego poszukiwaną otoczkę poprzez połączenie wszystkich trójkątów, dla ktorych promień okręgu na nich opisanego był mniejszy od dobranej wartości parametru **α**.

Punkt trzeci oraz czwarty powyższego algorytmu zostały w pracy zrealizowane z wykorzystaniem biblioteki JTS oraz implementacji otoczki α-wklęsłej autorstwa Eric'a Grosso.

Dobranie odpowiedniej wartości parametru **α** pozwoliło na uzyskanie zadowalających rezultatów. Należy jednak mieć na uwadze, jak zostało to wcześniej wspomniane, że dla dowolnego zbioru punktów nie istnieje jednoznaczna reprezentacja otoczki wklęsłej, a koncepcja otoczki α-wklęsłej jest jedynie przybliżeniem poszukiwanego wielokąta.

Na zakończenie rozważań dotyczących algorytmu zostanie przedstawiony przykład zastosowania opisanego w tym rozdziale algorytmu dla zbioru punktów P przedstawionego na ekranie.

5. System

W ramach pracy poza opracowaniem omówionego algorytmu, został również zaprojektowany i zaimplementowany system informatyczny składający się z:

* aplikacji mobilnej - DronTracker, pełniącej rolę geolokalizatora do potencjalnego zamontowania na dronie,
* aplikacji serwerowej - DronSerwer, odpowiedzialnej za przeprowadzanie skomplikowanych obliczeń i komunikację z pozostałymi elementami systemu oraz
* głównej mobilnej aplikacji klienckiej - DronVision, służącej do wizualizacji wyznaczonego obszaru przeszukanego.

Aplikacje mobilne zostały stworzone na platformę Android z wykorzystaniem narzędzia SDK, a część serwerowa została stworzona jako aplikacja w języku JAVA, z wykorzystaniem platformy Java Enterprise Edition, oraz uruchomiona na serwerze aplikacyjnym Wildfly.

W celu implementacji funkcjonalności związanej z mapami w aplikacji DronVision wykorzystano dane z systemu OpenStreetMap oraz bibliotekę Osmdroid, zapewniającą funckje związane z rysowaniem i edycją map.

**6. Filmik**

W tym miejscu chciałbym zaprezentować Państwu filmik demonstrujący działanie aplikacji DronVision.

**7. Podsumowanie**