

Samolokalizacja bezzałogowego statku powietrznego uwzględniająca zmienną orientację kamery

Teresa Zielińska

Politechnika Warszawska, Instytut Techniki Lotniczej i Mechaniki Stosowanej, ul. Nowowiejska 24, 00-665 Warszawa

Tomasz Pogorzelski

Sieć Badawcza Łukasiewicz – Instytut Lotnictwa, Aleja Krakowska 110/114, 02-256 Warszawa

Streszczenie: W niniejszej publikacji zaproponowano metodę lokalizacji wizyjnej z wykorzystaniem obrazów z symulowanej kamery oraz mapy georeferencyjnej. Model BSP oraz symulacja lotu zostały wykonane w pakiecie MATLAB Simulink, który przesyłał dane dotyczące orientacji BSP do opisywanego programu. Wizualizacja obrazu z kamery została wykonana w czasie rzeczywistym przy pomocy oprogramowania FlightGear, której obraz również był przechwytywany przez program NW. Metoda ta realizowana jest przez dwa procesy w dwu modułach: Global Positioning Component oraz Motion Positioning Component. Pierwszy z nich porównuje obraz z symulowanej kamery z ortofotomapą. Drugi wyznacza pozycję na podstawie oceny przemieszczenia punktów charakterystycznych w obrazie w stosunku do ostatnio znanej lokalizacji. Wynik działania obu modułów ilustrowany jest w oknie graficznym aplikacji NW, co pozwala na wizualne porównanie uzyskanych wyników. W przypadku globalnej metody lokalizacji wymagana jest dodatkowa korekcja orientacji kamery w celu wyznaczenia pozycji w przestrzeni 2D. W tym celu wykorzystano dane dotyczące bieżącej orientacji kamery wyrażone w kwaternionach. Pozwoliło to na wprowadzenie poprawki pozycji, co znacząco poprawiło dokładność wyniku uzyskiwanego w module GPC mimo znacznych przechyleń BSP podczas symulowanego lotu.

Słowa kluczowe: lokalizacja wizyjna, orientacja kamery, redundancja GNSS, porównywanie obrazów

1. Wprowadzenie

Nawigacja Bezzałogowych Statków Powietrznych (BSP) wykorzystuje do samolokalizacji głównie systemy satelitarne GNSS (ang. *Global Navigation Satellite Systems*). W sytuacji utraty łączności lub braku dostępu usługi GNSS sprawne pilotowanie statku bezzałogowego jest niemożliwe. Powstaje więc potrzeba zapewnienia bezpieczeństwa lotów i nieprzerwanej nawigacji przez wprowadzenie redundancji informacyjnej. Potrzeba ta wynika również z konieczności zmniejszenia

podatności systemu nawigacji satelitarnej na zakłócenia czy oszustwa (ang. *jamming/spoofing*). Ze względu na oczekiwaną uniwersalność rozwiązania, preferowane są systemy nawigacji, których oprzyrządowanie jest lekkie i zajmuje niewiele miejsca a wykorzystujące je metody nawigacji nie potrzebują dodatkowej zewnętrznej infrastruktury.

Jednym z możliwych rozwiązań zapewnienia ciągłej nawigacji BSP, bez konieczności dodatkowej infrastruktury zewnętrznej, jest system lokalizacji wykorzystujący analizę obrazu pozyskiwanego z kamery pokładowej [1]. Takie systemy często wyposażone są w stabilizator elektromechaniczny, pozwalający na uzyskanie stabilnego obrazu z kamery w celu zniwelowania zniekształceń w obrazie. Autorzy artykułu [2] wykorzystali techniki głębokiego uczenia maszynowego. Do treningu sztucznej sieci neuronowej zastosowano wcześniej przygotowane zdjęcia lotnicze. Wytrenowana sieć identyfikowała charakterystyczne cechy w obrazie uzyskanym na żywo z kamery pokładowej zamontowanej na gimbalu.

Z kolei w publikacji [3] przedstawiono system, który nie wymaga stosowania stabilizatora. Składa się on z trzech głównych komponentów. Pierwszy komponent stosuje metodę VIO

Autor korespondujący:

Teresa Zielińska, teresaz@meil.pw.edu.pl

Artykuł recenzowany

nadesłany 10.10.2022 r., przyjęty do druku 14.11.2022 r.



Zezwala się na korzystanie z artykułu na warunkach licencji Creative Commons Uznanie autorstwa 3.0

(ang. *Visual Inertial Odometry*) do nawigacji względnej. Wykorzystuje ona metodę odometrii wizyjnej o nazwie VINS-Mono, która integruje urządzenie MIMU (ang. *Micro Inertial Measurement Unit*) i kamerę monokularową, aby oszacować stany ruchu kamery (np. pozycję, położenie i prędkość) i zrekonstruować pozycję 3D śledzonych cech. Drugi komponent porównuje obrazy z kamery pokładowej z ortofotomapami. Z obrazu lotniczego najpierw wyodrębniane są cechy mapy w trybie off-line za pomocą algorytmu SIFT (ang. *Scale-Invariant Feature Transform*) z biblioteki OpenCV. Następnie obraz z kamery jest obracany i skalowany w celu uzyskania podobnej orientacji i skali, jaką ma mapa z repozytorium off-line. Do tych transformacji wykorzystywany jest zwrot UAV oraz wysokość wyznaczona z zastosowaniem metody VIO. Charakterystyczne punkty są następnie porównywane z ortofotomapą. Do lokalizacji obiektu wykorzystywany jest algorytm PNP (Perspektywa-N-Punkt), który polega na oszacowaniu pozycji kamery, bazując na zestawie punktów 3D oraz odpowiadającym im rzutom 2D na obrazie. Ostatnim komponentem jest graf dokonujący fuzji danych (łączy on względne wyniki nawigacji z VIO z pozycjami geodezyjnymi pochodzącymi z ortofotomapy). Przeprowadzono walidację proponowanej metody. Kamera wraz z MIMU zostały sztywno przymocowane do korpusu UAV. Wykonano dwa loty w różnych środowiskach o nieznanymi współrzędnymi początkowych. Wyniki pokazują, że proponowana metoda może zapewnić dokładność lokalizacji z błędem pozycji mniejszym niż 4 m w locie na wysokości 100 m i błędem mniejszym niż 9 m w locie na wysokości 300 m.

W publikacji [4] opisano system łączący tradycyjne techniki widzenia komputerowego z sieciami głębokiego uczenia do porównania obrazu satelitarnego (referencyjnego) z obrazem pochodzącym z UAV (docelowego) na potrzeby lokalizacji. Parametrami wejściowymi są obraz z kamery, ortofotomapa

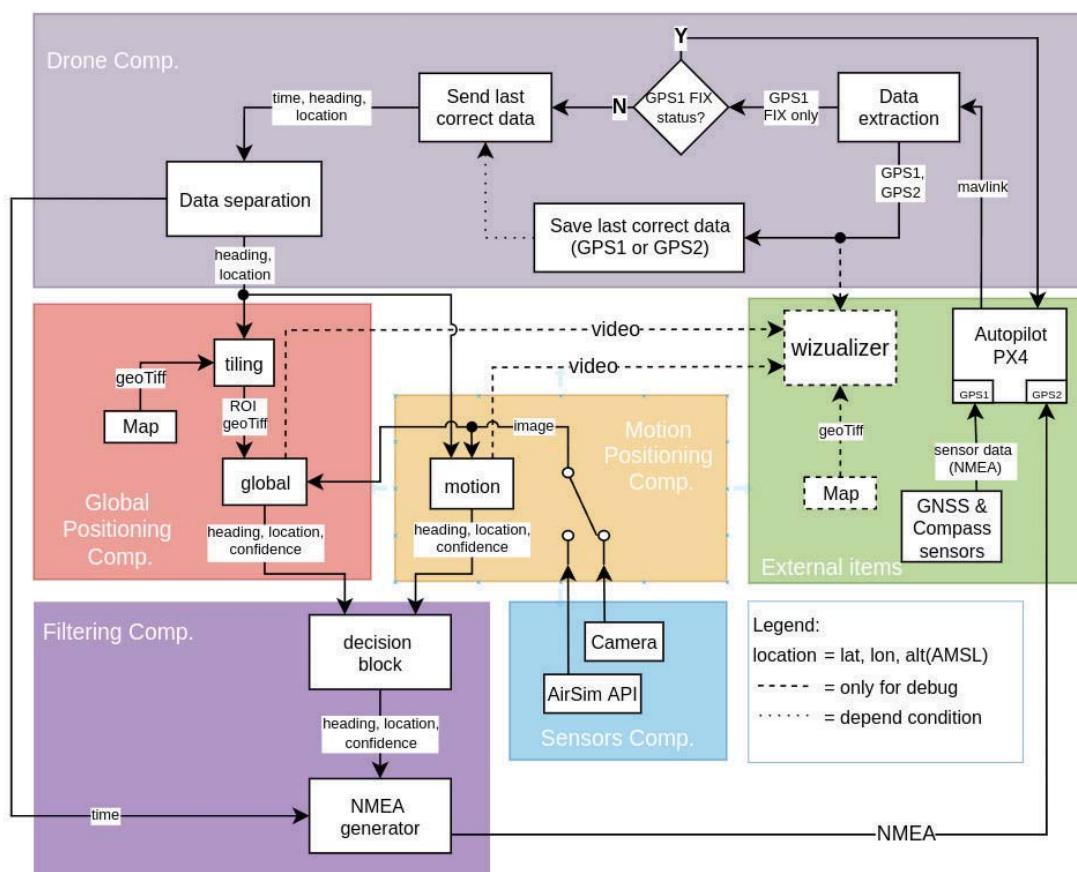
wraz z georeferencją oraz wstępną pozycją UAV. Moduł kalibracji odpowiada za wyodrębnienie obszaru ROI (ang. *Region Of Interest*) oraz wstępne wyznaczenie lokalizacji na podstawie poniższych równań. Przy pomocy równań (1) i (2) obliczane są współrzędne pikseli pix_x , pix_y określających położenie UAV, przy znanych współrzędnych geograficznych UAV: lon , lat , współrzędnych brzegowych obrazu: lon_n , lat_e , lat_w , lon_n oraz znanymi wymiarze mapy: $width_{min}$, $width_{max}$, $height_{min}$, $height_{max}$. Wzory (3) i (4) służą do obliczenia współrzędnych geograficznych: lat , lon na podstawie współrzędnych na obrazie w pikselach: pix_h , pix_w . Proces odnajdywania punktów charakterystycznych jest realizowany tak, aby przetwarzanie klatek wideo odbywało się w czasie rzeczywistym. Porównano wynik działania algorytmów ORB, SURF i SIFT wyszukujących cechy charakterystyczne na obrazach. Stwierdzono, że metoda ORB jest szybsza od pozostałych, jednak charakteryzuje się mniejszą dokładnością dopasowania porównywanych dwu obrazów w przypadku zmiany skali jednego z nich.

$$pix_x = \frac{(width_{max} - width_{min})(lon - lon_w)}{(lat_s - lat_w)} \tag{1}$$

$$pix_y = \frac{(height_{max} - height_{min})(lat - lat_n)}{(lon_s - lon_w)} \tag{2}$$

$$lat = \frac{lat_n + (lat_n - lat_s)(pix_h - height_{min})}{(height_{max} - height_{min})} \tag{3}$$

$$lon = \frac{lon_w + (lon_e - lon_w)(pix_w - width_{min})}{(width_{max} - width_{min})} \tag{4}$$



Rys. 1. Schemat koncepcyjny systemu NW (Źródło: opracowanie własne)
 Fig. 1. Conceptual diagram of the NW system (Source: own study)

Mniejsza dokładność dopasowania obrazów o różnych skalach powodowała spadek dokładności pozycjonowania UAV podczas zmiany jego wysokości. W celu eliminacji tego problemu zaimplementowano funkcję skalowania mapy na podstawie wysokości lotu. W kolejnym kroku wykorzystano wytrenowaną sieć UNet do semantycznej segmentacji obiektów. Sieć ta została wybrana ze względu na wysoką wydajność obliczeniową. Następnym krokiem była ocena podobieństwa wykrytych kształtów na obrazie z kamery pokładowej do tych wykrytych na mapie. Dopasowane kształty są stosowane do obliczania macierzy homografii [5] w celu zwiększenia dokładności pozycjonowania UAV.

2. Opis metody nawigacji

Schemat pokazany na Rys. 1. przedstawia, opracowaną w ramach prowadzonych badań, koncepcję systemu nawigacji wizyjnej (NW). Podczas prawidłowego funkcjonowania lokalizacji satelitarnej autopilot PixHawk korzysta z jej danych do określenia swojego położenia geograficznego. Moduł Drone Component w sposób ciągły pobiera dane z autopilota PixHawk za pomocą protokołu MAVLink (ang. *Micro Air Vehicle Link* – protokół do komunikacji stosowany w pojazdach bezzałogowych). Następnie sprawdzany jest status danych podających lokalizację. Dzieje się to w bloku decyzyjnym oznaczonym na schemacie FIX status. W przypadku pozytywnego statusu (dane kompletne) dane z sensorów lokalizacyjnych zapisywane są w pamięci. W przypadku negatywnego statusu, dane przechowywane w pamięci przekazywane są do modułu GPC (ang. *Global Positioning Component*) oraz MPC (ang. *Motion Positioning Component*), natomiast znacznik czasu, w którym te dane zostały zarejestrowane przekazywany jest do Filtering component. Global Positioning Component wyznacza pozycję na podstawie porównania obrazu ze znaną mapą (przechowywaną w pamięci), natomiast Motion Positioning Component wyznacza pozycję na podstawie oceny przemieszczenia punktów charakterystycznych w obrazie w stosunku do ostatnio znanej lokalizacji. Pierwszy z nich wykorzystuje metody klasycznej analizy obrazu zaimplementowane w OpenCV oraz metody oparte na uczeniu maszynowym. Charakteryzuje się długim czasem przetwarzania obrazów oraz niewielkim błędem systematycznym wyniku. Drugi komponent (MPC) wykorzystuje algorytm Optical Flow, co przekłada się na krótszy czas realizacji obliczeń oraz większy błąd systematyczny. W celu uzyskania małego błędu systematycznego oraz dużej szybkości przetwarzania danych oba komponenty pracują równolegle. Wyznaczona w module Global Positioning Component lokalizacja przekazywana jest do Filtering component oraz Motion Positioning Component w celu aktualizacji pozycji odniesienia. Oba moduły korzystając z funkcji OpenCV, wyznaczają współczynnik ufności uzyskanego wyniku (ang. *confidence*). W kolejnym kroku moduł Filtering component, uwzględniając obliczone współczynniki ufności, określa pozycję oraz w odniesieniu do ostatniego znanego czasu, generuje pakiet danych zgodny ze standardem NMEA, który trafia do autopilota. Wyznaczona na podstawie danych wizyjnych lokalizacja jest również deponowana w module Drone Component w celu ponownego wykorzystania w przypadku braku dostępnych danych z systemów satelitarnych (dane z GNSS & Compass sensors).

3. Ortofotomapa

Składnikiem Global Positioning Component jest funkcja odpowiedzialna za wygenerowanie wycinka ortofotomapy w otoczeniu lokalizacji BSP z użyciem danych dotyczących całego

obszaru przechowywanego w pamięci. Początkowo dane dotyczące mapy stanowiły jeden plik o rozszerzeniu *.geotiff zawierający wybrany obszar o wysokiej rozdzielczości. W metadanych danego pliku znajdowały się również informacje na temat lokalizacji narożników mapy oraz stosowanego układu współrzędnych. Jednak ze względu na ograniczenia pamięci w przypadku otwierania dużych plików, zastosowano metodę podziału mapy znaną jako Slippy Map [14]. Jest to metoda stosowana powszechnie w mapach internetowych typu Google Maps [15] czy OpenStreetMap [16]. Pozwala ona na podzielenie ortofotomapy dużego obszaru na małe kafelki a następnie uporządkowanie ich w odpowiednich katalogach o nazwach związanych z rozdzielczością oraz podkatalogach o nazwach związanych z lokalizacją wzdłuż równoleżników. Nazwy plików graficznych *.png o rozdzielczości 256 × 256 pikseli skojarzone są z lokalizacją wzdłuż południków. Na podstawie wysokości BSP, pobranej przez Drone Component, możliwe jest obliczenie GSD (ang. *Ground Sampling Distance*), co pozwala na dobranie właściwej rozdzielczości mapy. Pobrana lokalizacja geograficzna determinuje wybór odpowiednich katalogów oraz plików zawierających ortofotomapy w otoczeniu danej lokalizacji. Tak uporządkowany system pozwala na wczytanie minimalnej ilości danych, co przekłada się na mniejsze obciążenie pamięci oraz na szybkość działania.

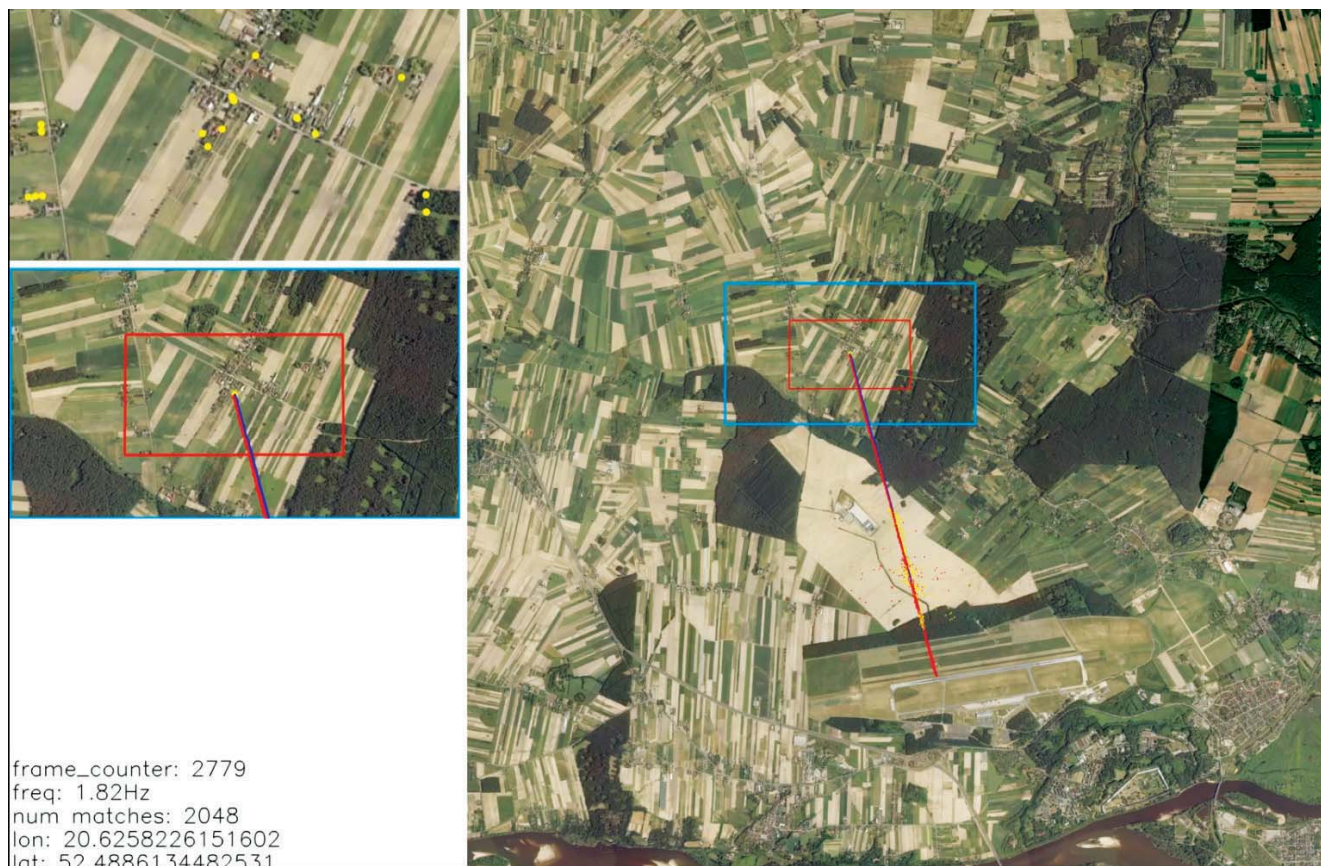
4. Badania symulacyjne

W celu pozyskania filmów, obrazujących nagrania z kamery wideo zbierane w trakcie lotu BSP, wykorzystano otwarty źródłowy symulator FlightGear [17] Wybór ten został podyktowany możliwością integracji z oprogramowaniem MATLAB Simulink, w którym możliwe jest zaimplementowanie różnych modeli latających, co opisano między innymi w publikacji [6]. W przyszłości pozwoli to na testowanie opracowywanej nawigacji na różnych platformach powietrznych – zarówno załogowych, jak i bezzałogowych.

Poniżej przedstawiono wizualizację funkcjonowania systemu dla przelotu z kamerą, której oś optyczna skierowana jest prostopadle do płaszczyzny ziemi. Dla tego przypadku, wspomniany wcześniej komponent GPC prawidłowo określa pozycję BSP, ponieważ obraz z kamery stanowi prostokątny wycinek mapy, a geometryczny środek tego prostokąta – lokalizację BSP w rzucie na tę mapę. Wizualizacja została podzielona na cztery części: obraz w prawym górnym rogu – widok z kamery z zaznaczonymi punktami charakterystycznymi algorytmu Optical Flow, poniżej – wycinek mapy z zaznaczonym rzutem pola widzenia kamery oraz trajektoriami lotu BSP, lewy dolny róg – parametry wykorzystane do obliczeń wraz z wyznaczoną lokalizacją geograficzną, obraz po prawej – cała wczytana mapa z zaznaczonym wycinkiem obszaru wokół BSP, polem widzenia kamery oraz wyznaczonymi trajektoriami lotu.

Trudniejszym zadaniem jest porównanie obrazów uzyskanych pod różnymi kątami widzenia. Kąt widzenia nie tylko zmienia geometrię obrazu, lecz również wpływa na nierównomierności w rozkładzie rozdzielczości w danym widoku. W pracy skupiono się na geometrycznym wyznaczeniu lokalizacji podczas wychylenia BSP względem poziomu (rys. 2). Ma to na celu uniknięcie potrzeby stosowania stabilizowanej mechanicznie głowicy optycznej, co znacznie zmniejsza masę i objętość systemu, oraz zapewnia większą niezawodność dzięki minimalizacji elementów ruchomych. Na potrzeby przedstawionego zadania wygenerowano symulację przelotu BSP ze zmiennymi wychyleniami do 30° względem płaszczyzny ziemi. Na rys. 3 pokazano wizualizację działania systemu bez korekcji kąta wychylenia.

Wyznaczona lokalizacja przez Global Positioning Component znacząco odbiega od prawidłowej, wyznaczonej przez GNSS.



Rys. 2. Wizualizacja funkcjonowania systemu; niebieski – rzeczywista trajektoria lotu BSP, żółty – trajektoria wyznaczona przez moduł MPC, czerwony – trajektoria oraz rzut pola widzenia kamery na mapę 2D – wynik algorytmu GPC (Źródło: opracowanie własne)

Fig. 2. Visualization of the system functioning; blue – the actual UAV flight trajectory, yellow – the trajectory determined by the MPC module, red – the trajectory and the projection of the camera's field of view onto the 2D map – the result of the GPC algorithm (Source: own study)

Spowodowane jest to brakiem kompensacji kąta wychylenia dla wyznaczonej pozycji.

5. Szacowanie pozycji BSP (kamery)

Na podstawie dopasowanych punktów źródłowych na obrazie wideo do odpowiadających im na mapie, zostaje wyznaczona macierz homografii H [5], która zawiera pozycję i orientację kamery. Do jej wyznaczenia zastosowano metodę RANSAC (ang. *RANdom SAmple Consensus*), która wymaga co najmniej

czterech, niewspółliniowych punktów charakterystycznych. Za pomocą macierzy H rzutujemy punkty płaszczyzny kamery na płaszczyznę mapy oraz przy uwzględnieniu wymiarów obrazu wideo wyznaczane są punkty czworokąta, który reprezentuje pole widzenia kamery na płaszczyźnie mapy. Jeśli oś optyczna kamery jest skierowana prostopadle w kierunku ziemi, rzutowane pole widzenia jest prostokątem o długościach boków w proporcji adekwatnej do wymiarów sensora kamery. Położenie centroidu L (punktu określającego geometryczny środek figury) wyznaczonego prostokąta odpowiada rzutowi prostopadłemu punktu lokalizacji BSP na mapę 2D.



Rys. 3. Symulacja funkcjonowania systemu w przypadku zmiennych wychyleń BSP – pitch, roll, yaw [7] w zakresie od 0° do 30° względem układu NED, którego początek związany jest z lokalizacją BSP; GPC – czerwony, GNSS – niebieski (Źródło: Opracowanie własne)

Fig. 3. Simulation of the system operation in the case of variable BSP deflections – pitch, roll, yaw [7] in the range from 0° to 30° in relation to the NED system, the beginning of which is related to the BSP location; GPC – red, GNSS – blue (Source: Own study)



Rys. 4. Wizualizacja wyznaczania lokalizacji za pomocą centroidu L prostokąta stanowiącego rzut pola widzenia kamery na płaszczyznę mapy, gdy oś optyczna kamery jest skierowana prostopadle w kierunku ziemi (Źródło: Opracowanie własne)

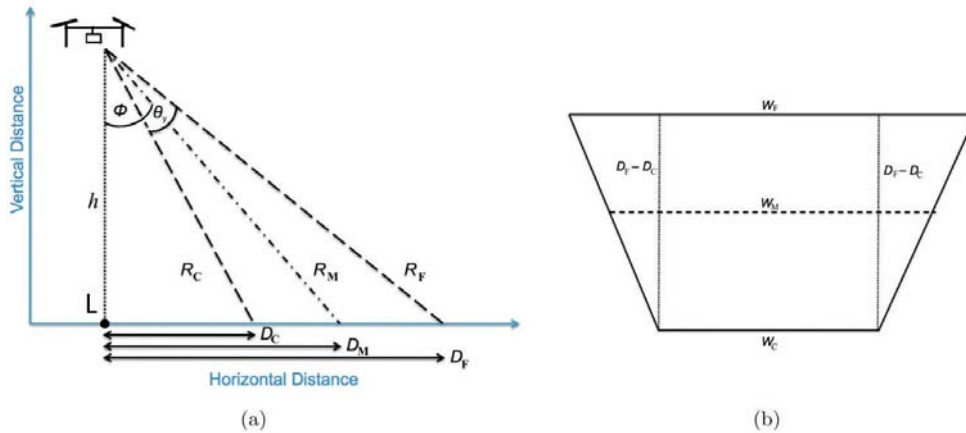
Fig. 4. Visualization of determining the location using the centroid L of a rectangle constituting the projection of the camera's field of view onto the map plane, when the optical axis of the camera is directed perpendicularly towards the ground (Source: Own study)

W przypadku zmiany kąta Φ osi optycznej kamery względem ziemi występuje zniekształcenie obrazu – kamera obejmuje szerszy obszar ziemi w kierunku, w którym nastąpiła zmiana kąta. Przy znacznym wychyleniu płaszczyzny kamery względem płaszczyzny ziemi, pole widzenia kamery nie obejmuje obszaru na którym znajduje punkt lokalizacyjny L. Powoduje to konieczność wprowadzenia korekcji lokalizacji w zależności od kąta Φ .

Przy transformacji współrzędnych uwzględnia się trzy układy współrzędne o oznaczeniach:

- NED – (ang. *Nord East Down*), układ o nieziennej orientacji względem mapy, o początku związanym z poruszającym się w przestrzeni BSP. W tym układzie oś X skierowana jest na północ, oś Y na wschód, a oś Z w dół;
- BSP – układ związany ze statkiem powietrznym o początku pokrywającym się z początkiem układu NED lecz o orientacji zmieniającej się względem NED zgodnie z obrotami roll, pitch oraz yaw statku powietrznego BSP;
- MAP – układ, którego orientacja oraz pozycja jest na stałe związana z mapą.

Do testowania opracowanej metody wykorzystano model platformy BSP sprzężony z kamerą, opracowany z wykorzystaniem pakietu MATLAB Simulink. Jednym z parametrów wyjściowych modelu jest orientacja kamery względem układu



Rys. 5. Pole widzenia kamery; D_c , D_m oraz D_f to najbliższa, środkowa oraz najdalsza odległość w polu widzenia kamery, natomiast h to wysokość BSP nad ziemią; R_c , R_m oraz R_f to odległości między dolną, środkową a górną częścią pola widzenia kamery (a). Rzut pola widzenia kamery na płaszczyznę mapy. Krawędź W_c znajduje się bliżej kamery (b) (Źródło [8])

Fig. 5. Camera field of view; D_c , D_m and D_f are the closest, middle and farthest distance in the field of view of the camera, while h is the UAV height above the ground; R_c , R_m , and R_f are the distances between the lower, middle, and upper parts of the camera's field of view (a). Projection of the camera's field of view onto the map plane. The edge of the toilet is closer to the camera (b) (Source [8])

MAP wyrażona kwaternionem ${}^{MAP}q$. Jest on reprezentowany przez cztery elementy wskazane w (5), gdzie q_0 , q_1 , q_2 , q_3 są liczbami rzeczywistymi, natomiast i , j i k są wzajemnie ortogonalnymi wektorami jednostkowymi osi układu MAP.

$$q = q_0 + iq_1 + jq_2 + kq_3 \quad (5)$$

Każdy pojedynczy obrót wokół pewnego wektora opisany jest przez kwaternion jednostkowy, natomiast podwójny obrót – przez pomnożenie kwaternionów jednostkowych. Kwaternion sprzężony q^* reprezentuje obrót przeciwny do danego co jest reprezentowane przez zmianę znaku części wektorowej kwaternionu: $q^* = q_0 - iq_1 - jq_2 - kq_3$. Aby wyznaczyć kwaternion ${}^{NED}p'$ obrotu kamery zamocowanej na BSP, względem układu NED należy pomnożyć: kwaternion orientacji ${}^{BSP}q$ kamery w układzie BSP, wektor ${}^{NED}P_{camera}$ stanowiący oś obrotu układu BSP względem NED oraz kwaternion sprzężony ${}^{BSP}q^*$ orientacji kamery.

$${}^{NED}p' = {}^{BSP}q {}^{NED}P_{camera} {}^{BSP}q^* \quad (6)$$

Osie układu współrzędnych nieobrótowej kamery pokrywają się z osiami układów NED i BSP, natomiast wektor orientacji ${}^{BSP}P_{camera}$ kamery w układzie BSP wyrażony jest jako:

$${}^{BSP}P_{camera} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad (7)$$

Wynikiem działania (6) jest kwaternion ${}^{NED}p'$:

$${}^{NED}p' = \begin{bmatrix} 0 \\ {}^{NED}x_p \\ {}^{NED}y_p \\ {}^{NED}z_p \end{bmatrix} \quad (8)$$

z którego współrzędne wektora ${}^{NED}x_p$, ${}^{NED}y_p$, ${}^{NED}z_p$ wskazują kierunek oraz zwrot osi optycznej kamery w układzie NED. Rzutowane są one na płaszczyznę mapy 2D w postaci współrzędnych ${}^{MAP}x_m$ oraz ${}^{MAP}y_m$, przy uwzględnieniu wysokości h :

$${}^{MAP}x_m = \frac{{}^{NED}x_p}{{}^{NED}z_p} h \quad (9)$$

$${}^{MAP}y_m = \frac{{}^{NED}y_p}{{}^{NED}z_p} h \quad (10)$$

Wartości współrzędnych ${}^{MAP}x_m$ oraz ${}^{MAP}y_m$, stanowiące punkt przecięcia się osi optycznej kamery z płaszczyzną mapy 2D, odejmowane są od współrzędnych wyznaczonego przez komponent GPC centroida L. Pozwala to na uwzględnienie orientacji kamery podczas wyznaczania rzeczywistego położenia BSP.

Na rysunku 6 porównano trajektorie wyznaczonej lokalizacji za pomocą komponentu GPC, bez uwzględnienia orientacji kamery, oraz z jej uwzględnieniem w stosunku do rzeczywistej trajektorii wyznaczonej przez moduł GNSS. Dla przeprowadzonej symulacji wykazano znaczącą poprawę dokładności pozycjonowania. Dalsze prace będą polegały na



Rys. 6. Wizualizacja obliczonej lokalizacji przez komponent GPC bez korekcji – kolor czerwony, GPC z uwzględnieniem korekcji – kolor cyjan, w porównaniu do rzeczywistej trajektorii lotu – kolor niebieski, w przypadku zmiennych wychyleń do 30° względem płaszczyzny ziemi (Źródło: Opracowanie własne)

Fig. 6. Visualization of the calculated location by the GPC component without correction – red, GPC with correction – cyan, compared to the actual flight trajectory – blue, in the case of variable deflections up to 30° with respect to the ground plane (Source: Own study)

wyznaczeniu maksymalnych wychyleń BSP przy zachowaniu zadowalających wyników.

6. Wnioski

Poprzednie prace [1, 9] dotyczyły opisu metody nawigacji wizyjnej przy założeniu, że oś optyczna kamery jest zawsze prostopadła do płaszczyzny terenu. Sytuacja taka dotyczy układu wykorzystującego gimbal (stabilizator elektromechaniczny). Stabilizuje on orientację kamery ale znacząco zwiększa stopień skomplikowania konfiguracji sprzętowej. W pracy skupiono się na geometrycznym wyznaczeniu lokalizacji BSP z uwzględnieniem zmian orientacji kamery BSP względem układu NED, a tym samym względem układu odniesienia mapy. Pokazano, że podczas odchylenia osi optycznej kamery od osi prostopadłej do płaszczyzny ziemi, lokalizacja wyznaczona przez algorytm GPC zawiera błąd, który może zostać skompensowany przez uwzględnienie zmiennej orientacji kamery. W tym celu wykorzystano kwaternion orientacji kamery pochodzący z modelu zaimplementowanego w programie MATLAB Simulink. Uwzględnienie orientacji kamery znacząco poprawiło dokładność wyznaczania lokalizacji (rys. 6). Przyszłe prace będą skupiały się na wykorzystaniu danych z żyroskopu autopilota do stabilizacji obrazu. Pozwoli to na przeprowadzenie testów na rzeczywistej platformie nieposiadającej stabilizatora. Planowane są również prace związane z wykorzystaniem kamer zdarzeniowych do wykrywania punktów charakterystycznych wymaganych jako dane w module MPC. Pozwoli to osiągnąć nie tylko wystarczającą do samonawigacji dokładność wyznaczania pozycji geograficznej, ale i odpowiednią częstotliwość generowania wyniku w celu zapewnienia całkowitej redundancji modułu GNSS.

Bibliografia

1. Pogorzelski T., Zielińska T., *Vision Based Navigation Securing the UAV Mission Reliability*, [In:] Automation 2022: New Solutions and Technologies for Automation, Robotics and Measurement Techniques, AISC, Vol. 1427, 2022, 251–263, DOI: 10.1007/978-3-031-03502-9_26.
2. Mughal M.H., Khokhar M.J., Shahzad M., *Assisting UAV Localization via Deep Contextual Image Matching*,

- “IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing”, Vol. 14, 2021, 2445–2457, DOI: 10.1109/JSTARS.2021.3054832.
3. Mao J., Zhang L., He X., Qu H., Hu X., *Precise Visual-Inertial Localization for UAV with the Aid of A 2D Georeferenced Map*, arXiv, 2021, DOI: 10.48550/arXiv.2107.05851.
4. Nassar A., Amer K., ElHakim R., ElHelw M., *A Deep CNN-Based Framework For Enhanced Aerial Imagery Registration with Applications to UAV Geolocalization*, [In:] IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2018, DOI: 10.1109/CVPRW.2018.00201.
5. Ivashechkin M., Barath D., Matas J., *USACv20: robust essential, fundamental and homography matrix estimation*, arXiv, 2021, DOI: 10.48550/arXiv.2104.05044.
6. Welcer M., Szczepański C., Krawczyk M., *The Impact of Sensor Errors on Flight Stability*, „Aerospace”, Vol. 9, No. 3, 2022, DOI: 10.3390/aerospace9030169.
7. Kowalik R., Łusiak T., Novak A., *A Mathematical Model for Controlling a Quadrotor UAV*, „Transactions on Aerospace Research”, Vol. 2021, No. 3, 2021, 58–70, DOI: 10.2478/tar-2021-0017.
8. Burke C., Rashman M., Wich S., Symons A., Theron C., Longmore S., *Optimising observing strategies for monitoring animals using drone-mounted thermal infrared cameras*, „International Journal of Remote Sensing”, Vol. 40, No. 2, 2019, 439–467, DOI: 10.1080/01431161.2018.1558372.
9. Pogorzelski T., Majek K., *Onboard to Satellite Image Matching for Relocalization of the UAVs*, [In:] Proceedings of the 3rd Polish Conference on Artificial Intelligence, 2022, 25–27.
10. Babinec A., Apeltauer J., *On accuracy of position estimation from aerial imagery captured by low-flying UAVs*, „International Journal of Transportation Science and Technology”, Vol. 5, No. 3, 2016, 152–166, DOI: 10.1016/j.ijst.2017.02.002.
11. Fellner A., Koniczka R., *Rotorcraft in the performance based navigation international civil aviation organization implementation*, „Transactions on Aerospace Research”, Vol. 2019, No. 1, 2019, 53–64, DOI: 10.2478/TAR-2019-0005.
12. Rose D., *Rotation Quaternions, and How to Use Them*, 2015. [On-line] <https://danceswithcode.net/engineeringnotes/quaternions/quaternions.html>.
13. Craig J.J., *Introduction to Robotics*, Pearson Education Limited, 2014.

Inne źródła

14. *Slippy map tilenames*, [On-line] https://wiki.openstreetmap.org/wiki/Slippy_map_tilenames.
15. Google, *Google Maps, Alphabet*, [On-line] www.google.pl/maps/.
16. Coast S., *Open Street Map*, OpenStreetMap Foundation, [On-line] www.openstreetmap.org.
17. *FlightGear*, [On-line], <https://www.flightgear.org/>.

Self-Localization of the Unmanned Aerial Vehicle Taking Into Account the Variable Orientation of the Camera

Abstract: This publication proposes a visual localization method using images from a simulated camera and a georeferenced map. The UAV model and flight simulation were made in the MATLAB Simulink package, which sent UAV orientation data to the described program. The visualization of the camera image was performed in real time using the FlightGear software, the image of which was also captured by the NW program. This method is performed by two processes in two modules: Global Positioning Component and Motion Positioning Component. The first one compares the image from the simulated camera with the orthophotomap. The second determines the position based on the assessment of the displacement of characteristic points in the image in relation to the last known location. The result of the operation of both modules is illustrated in the graphic window of the NW application, which allows for a visual comparison of the obtained results. With the global method of location, additional camera orientation correction is required to determine the position in 2D space. For this purpose, data on the current camera orientation expressed in quaternions were used. This allowed for the introduction of a position correction, which significantly improved the accuracy of the result obtained in the GPC module despite significant UAV tilts during the simulated flight.

Keywords: vision localization, camera orientation, GNSS redundancy, image comparison

Prof. dr hab. inż. Teresa Zielińska

teresaz@meil.pw.edu.pl
ORCID: 0000-0002-0475-0875

Jest profesorem na Wydziale Mechanicznym Energetyki i Lotnictwa (MEiL) Politechniki Warszawskiej (PW). Na Politechnice pracuje od 1987 r. Sprawowała tu liczne funkcje: prodziekana ds. ogólnych MEiL, zastępcy dyrektora ds. naukowych Instytutu Techniki Lotniczej i Mechaniki Stosowanej (ITLiMS), dyrektora ITLiMS. Obecnie pełni funkcję zastępcy dyrektora ds. naukowych ITLiMS. Była członkiem Rady Naukowej Przemysłowego Instytutu Automatyki i Pomiarów PIAP. Jest członkiem Komitetu Automatyki i Robotyki Polskiej Akademii Nauk oraz członkiem Rady Narodowego Centrum Nauki. Była zatrudniona w Loughborough University of Technology w Wielkiej Brytanii (1992) oraz w Nanyang Technological University w Singapurze (1998–2001). Od 2008 r. opiekuje się studiami drugiego stopnia EMARO (European Master on Advanced Robotics) oraz JEMARO (Japan-European Master on Advanced Robotics). Jej zainteresowania badawcze koncentrują się na zagadnieniach związanych z maszynami kroczącymi, robotami humanoidalnymi, planowaniem ruchu robotów oraz współpracą człowiek-robot. Jest autorką lub współautorką ponad 300 publikacji z tego zakresu. Uzyskała dwa patenty.



mgr inż. Tomasz Pogorzelski

tomasz.pogorzelski@ilot.lukasiewicz.gov.pl
ORCID: 0000-0002-8299-0727

Absolwent Wydziału Mechatroniki i Lotnictwa Wojskowej Akademii Technicznej. W 2013 r. uzyskał stopień magistra inżyniera w specjalności mechatronika. W 2014 r. podjął pracę w Engineering Design Center - Instytut Lotnictwa. Obecnie zatrudniony jako samodzielny inżynier ds. oprogramowania w Dziale Teledetekcji, Centrum Technologii Bezzałogowych w Sieci Badawczej Łukasiewicz – Instytut Lotnictwa (SBŁ-ILOT). Czynny pilot bezzałogowych statków powietrznych (BSP) w zakresie VLOS oraz BVLOS. W latach 2018–2019 uczestniczył w projekcie integracji systemów UTM zarządzających ruchem BSP dla Federal Aviation Administration (FAA) w stanie Nevada. Główne zainteresowania obejmują zastosowanie technik przetwarzania obrazu w robotyce. W 2020 r. rozpoczął prace nad doktoratem wdrożeniowym na Wydziale Mechaniki Energetyki i Lotnictwa Politechniki Warszawskiej oraz SBŁ-ILOT dotyczącym nawigacji wizyjnej dla BSP z wykorzystaniem technik uczenia maszynowego.



