



# Inteligentny podsystem naprowadzania na cel objektu BSP

**Autor:**

**Opiekunowie:**

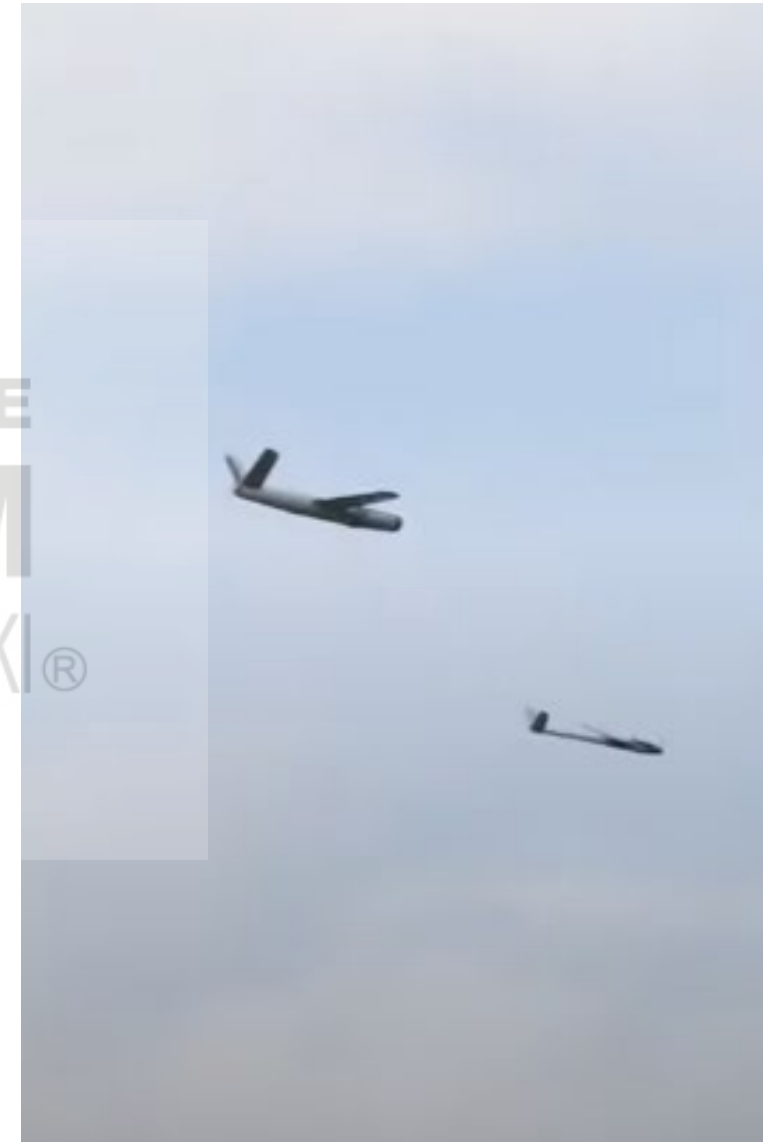
inż. Maciej Kawka, Wydział Cybernetyki, Wojskowa Akademia Techniczna  
dr. inż. Paweł Skokowski, Wydział Elektroniki, Wojskowa Akademia Techniczna  
dr. inż. Dariusz Pierzchała, Wydział Cybernetyki, Wojskowa Akademia Techniczna

## SPIS TREŚCI

1. Misja naprowadzanego BSP, wymagania na podsystem
2. Architektura podsystemu oraz platformy sprzętowej
3. Algorytm fazy pościgu
4. Algorytm fazy przejściowej
5. Algorytm fazy przechwycenia
6. Testy w symulacji i warunkach rzeczywistych
7. Podsumowanie i wnioski
8. Bibliografia

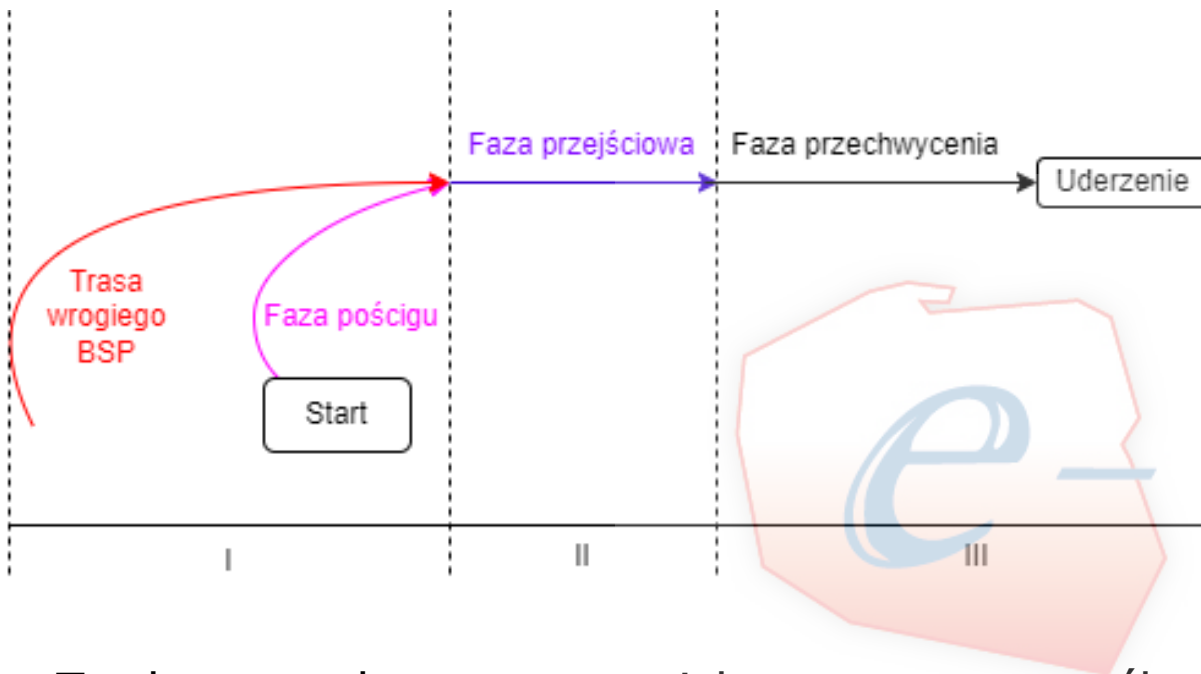
## CEL PROJEKTU

Celem projektu jest opracowanie podsystemu przeciwdziałania Bezzałogowym Systemom Powietrznym, który umożliwi autonomiczne przechwytywanie wrogich jednostek BSP za pomocą sojuszniczego bezzałogowca, co umożliwi wykorzystanie efektorów bliskiego kontaktu, takich jak efekторы radiowe lub/ oraz efektorów kinetycznych, do neutralizacji wrogich jednostek bezzałogowych pod nazwą „Inteligentny Podsystem Naprowadzania na Cel Obiektu BSP”.



# 1. Misja naprowadzanego BSP, wymagania na podsystem

## PLAN MISJI

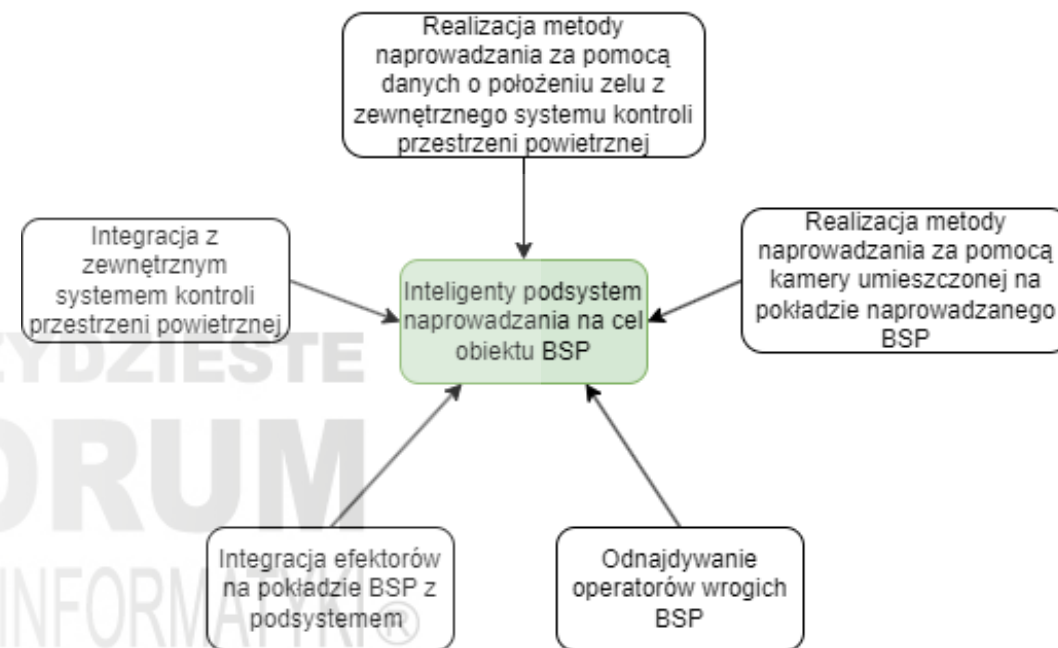


Tytułowy podsystem zaprojektowano w sposób umożliwiający realizację misji przechwycenia i kontroli wrogich jednostek BSP. Na misję składają się następujące etapy:

1. faza pościgu,
2. faza przejściowa,
3. faza przechwycenia.

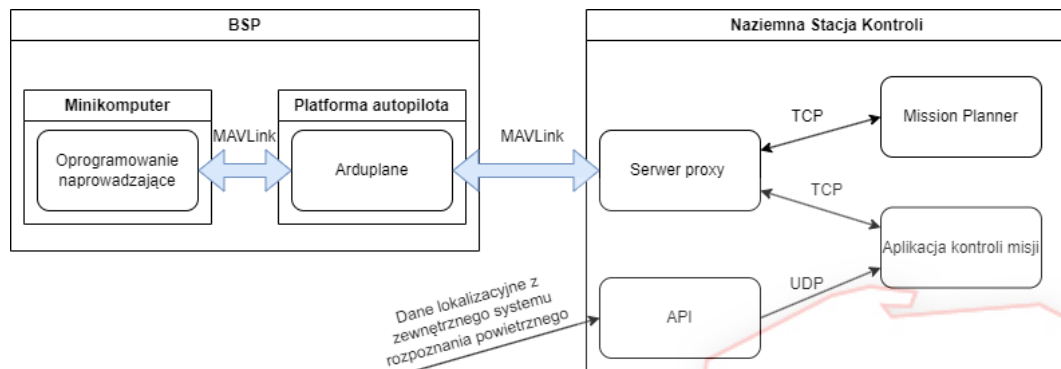
Etapy działania algorytmu oraz przejść między nimi zilustrowane są na powyższym rysunku.

## WYMAGANIA NA PODSYSTEM



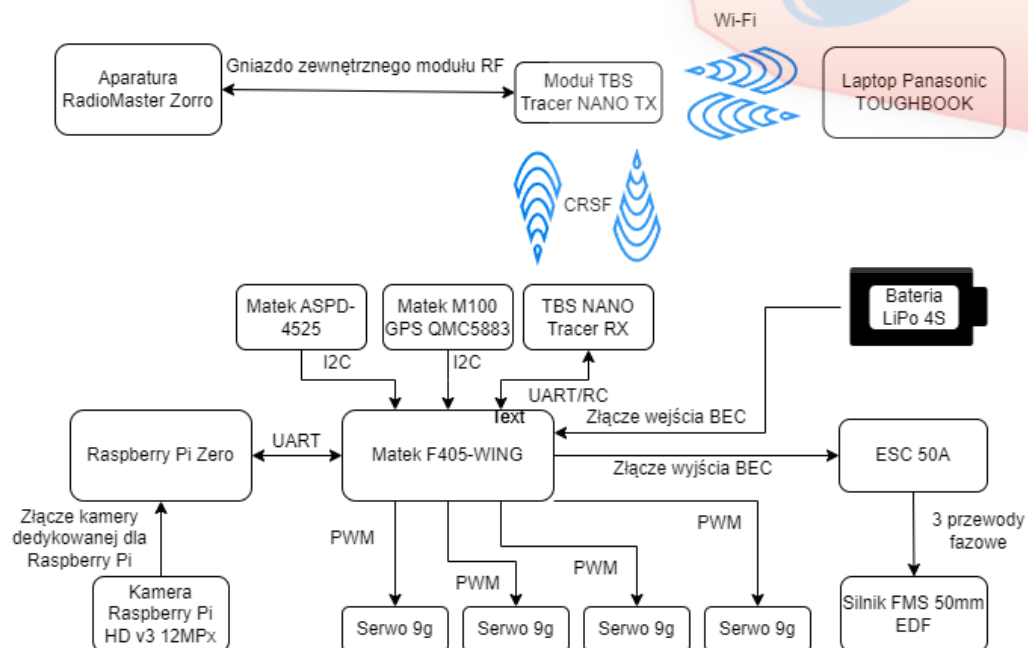
Podsystem został zaprojektowany tak aby spełniał wymagania przedstawione na powyższym rysunku, które obejmują integrację z zewnętrznymi systemami kontroli przestrzeni powietrznej, zastosowanie trzyetapowego algorytmu naprowadzania na cel oraz wykorzystanie efektorów radiowych i kinetycznych do przejęcia kontroli nad wrogim BSP, co ma na celu bezpieczne wyprowadzenie niechcianego bezzałogowca ze strefy kontrolowanej oraz jego bezpieczną neutralizację.

### SCHEMAT ARCHITEKTURY ROZPROSZONEJ PODSYSTEMU



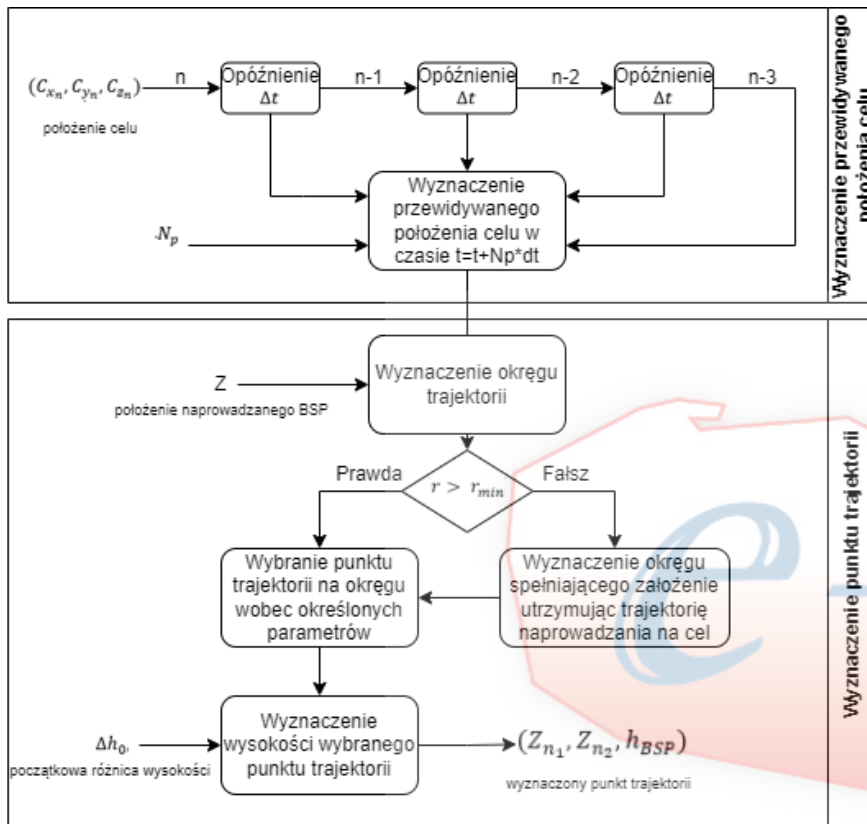
W tytułowym podsystemie wykorzystano architekturę rozproszoną będącą połączeniem wariantu architektury opartej na usługach oraz architektury agentowej. Na schemacie po lewej stronie przedstawiona jest logiczna architektura projektowanego podsystemu. Można na niej wyróżnić kluczowe elementy podsystemu takie jak oprogramowanie autopilota, aplikacja kontroli misji, API oraz sposoby komunikacji między nimi. Taka architektura umożliwia również implementację systemu na różnych platformach bezzałogowych. Jedynymi wymaganiami dla platformy sprzętowej jest logiczne rozdzielanie jednostki autopilota i mikrokontrolera, odpowiedzialnego za realizację misji, oraz dowolna kamera monoskopowa. Dzięki temu tytułowy podsystem umożliwia neutralizację wrogich dronów za pomocą dowolnego bezzałogowca.

### SCHEMAT PLATFORMY SPRZĘTOWEJ

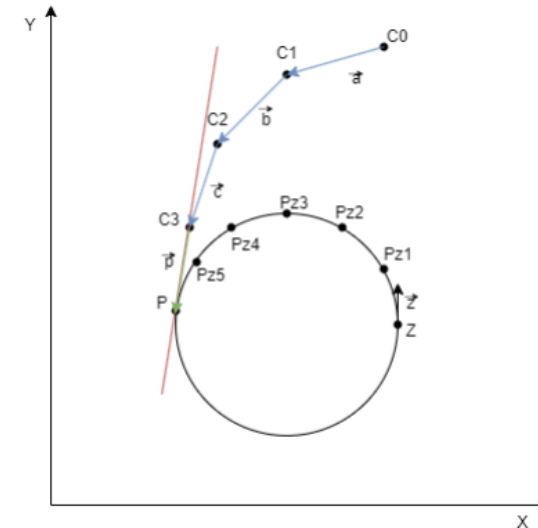


Prototypowa platforma sprzętowa wykorzystana do implementacji podsystemu naprowadzania została opracowana zgodnie ze schematem zamieszczonym po lewej stronie. Schemat przedstawia platformę sprzętową, na której został zaimplementowany podsystem. Zawiera ona komponenty niezbędne do sterowania płotowcem jak i komponenty systemu. Komponenty służące do sterowania to serwomechanizmy oraz silnik. Dodatkowo wykorzystane są czujniki takie jak IMU w autopilocie (Matek F405-WING) oraz GPS. Do komunikacji wykorzystany został system firmy TBS pod nazwą „Tracer” umożliwiający komunikację na odległości do 15 km. Elementami na których zaimplementowany został podsystem, są mikrokomputer Raspberry Pi Zero z podłączoną do niego kamerą, laptop Panasonic oraz sprzętowa platforma autopilota. Poprzez zaprojektowany system łączności możliwa jest komunikacja pomiędzy każdym z 3 urządzeń.

### 3. Algorytm fazy pościgu



### OKREŚLENIE ZMIANY AZYMUTU I ELEWACJI NAPROWADZANEGO BSP



### PRZEWIDYWANIE KOLEJNEGO POŁOŻENIA CELU

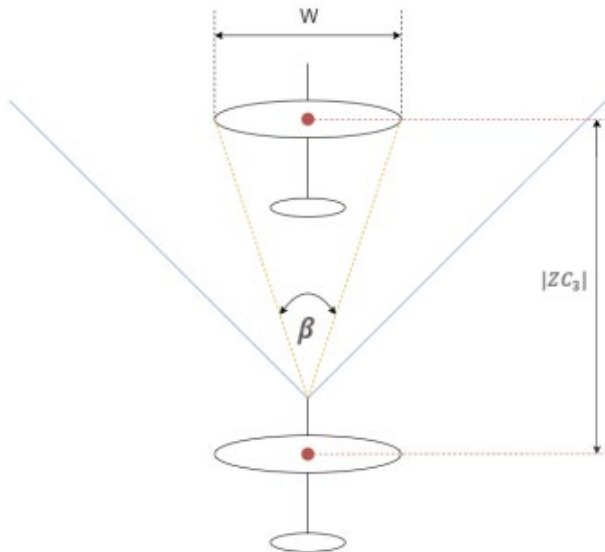
Algorytm fazy pościgu zakłada dostęp do informacji na temat położenia wrogiego bezzałogowca z zewnętrznego systemu kontroli przestrzeni powietrznej. Dane te są następnie przekształcane na jednostki metryczne za pomocą operacji projekcji kartograficznej, a następnie zapisywane w przyjętej formie, na którym operuje algorytm.

Pierwszym etapem działania algorytmu jest przewidywanie przyszłych położenia wrogiego BSP na podstawie danych z czasów o indeksach  $t$ ,  $t-1$ ,  $t-2$  oraz  $t-3$ . Wartości bezwzględne chwil czasowych są ustalane na podstawie częstotliwości odbierania danych od zewnętrznego systemu kontroli przestrzeni powietrznej. Do wykonania przewidywania wykorzystywany jest standard symulacji DIS (Distributed Interactive Simulation) oraz zawarta w nim formuła modelowania ruchu celu manewrującego. Formuła ta została dostosowana tak, aby na podstawie czterech położenia celu wyznaczyć przewidywane położenie na  $N_p$  interwałów czasowych w przód. Wysoka dokładność przewidywań jest możliwa dzięki uwzględnieniu prędkości, przyspieszenia i zrywu w trzech kierunkach.

Kolejnym etapem jest wyznaczenie punktu trajektorii celu na podstawie wygenerowanych i zebranych informacji. Dobór kolejnych punktów trasy został zwizualizowany na powyższym rysunku. Jeżeli punkt  $Z$  jest obecnym położeniem naprowadzanego BSP, wektor  $\vec{z}$  jest jego wektorem ruchu, punkty  $C_0, C_1, C_2$  i  $C_3$  są punktami położenia celu w czterech ostatnich chwilach, włącznie z obecną, a punkt  $P$  jest przewidywanym położeniem celu, to kolejny punkt, na który powinien być naprowadzony bezzałogowiec, wyznaczany jest na podstawie okręgu wykreślonego w taki sposób, aby był on styczny do prostej wyznaczonej przez obecne położenie celu oraz przewidywane jego położenie, oraz tak, aby przechodził przez punkt  $Z$ .

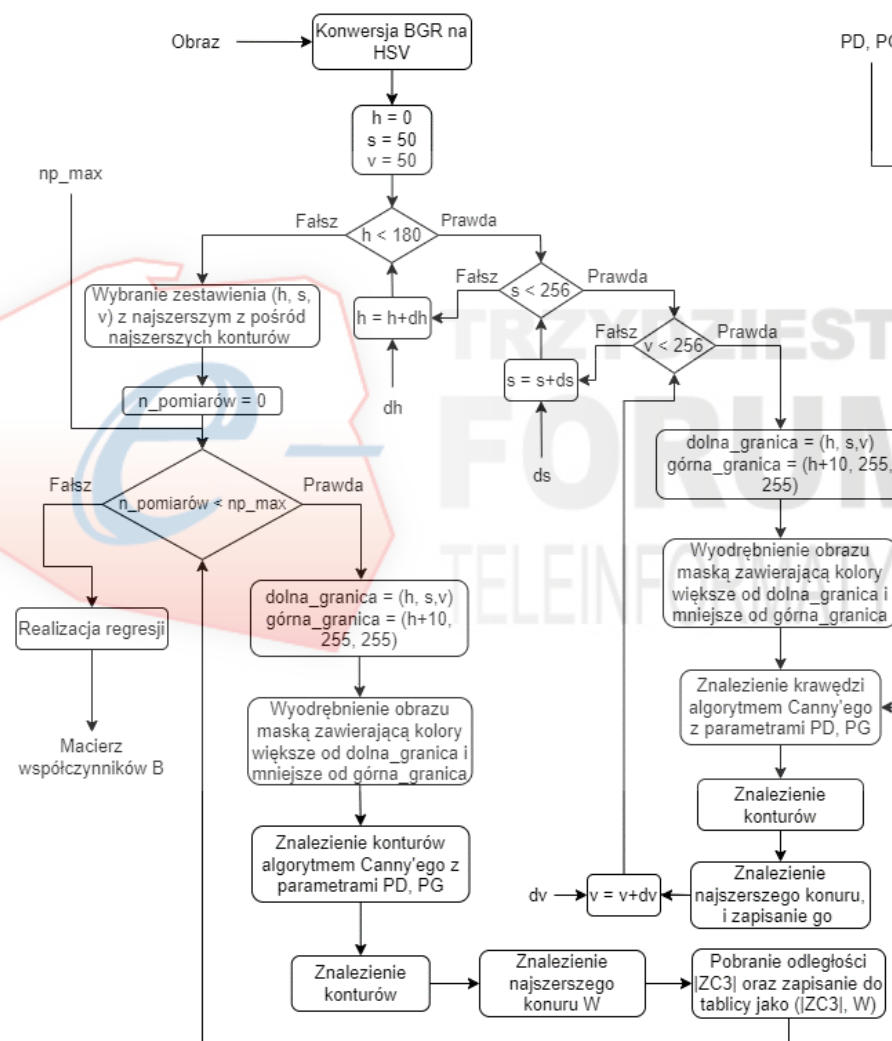
Następnie na tym okręgu wyznaczane są punkty z zadaną rozdzielczością, po czym wybierany jest jeden z nich, ten który jest najbliżej punktu  $Z$ . Algorytm kontroluje, czy wyznaczany okrąg ma odpowiednio duży promień, tak aby nie był mniejszy niż ustalony promień minimalny  $r_{min}$ . Jeżeli okrąg ten byłby mniejszy, algorytm sprawdza, czy doszło do przechwycenia, czyli momentu, w którym pomiędzy obiektami jest najbliższa dopuszczalna odległość, a kąt pomiędzy wektorami ruchu obu obiektów jest mniejszy niż 5 stopni, co oznacza, że naprowadzany bezzałogowiec jest bezpośrednio za celem i może przejść do kolejnego etapu, którym jest faza przejściowa. Jeżeli natomiast warunek nie został spełniony, algorytm wyznaczy kolejne punkty i będzie to powtarzał aż do spełnienia warunku. W przypadku algorytmu fazy pościgu, prędkość ustawiona jest na maksymalną możliwą dla naprowadzanego bezzałogowca.

## KONCEPCJA ODWRÓCONEJ KAMERY STEREO



Algorytm fazy przejściowej dokonuje kalibracji, aby w fazie przechwycenia używać jedynie danych z kamery, bez stałego połączenia z operatorem i bez informacji o położeniu celu z zewnętrznego systemu kontroli przestrzeni powietrznej. Wykorzystuje metodę odwróconej kamery stereo, która umożliwia widzenie przestrzenne za pomocą kamery monoskopowej. Polega ona na mierzeniu szerokości skrzydeł celu (samolotu) i zestawieniu ich z rzeczywistymi odległościami pochodzącymi z zewnętrznego systemu. Dzięki temu, za pomocą regresji liniowej, można wyznaczyć zależność odległości pomiędzy obiektami i szerokości skrzydeł, co pozwoli w kolejnej fazie na określenie odległości na podstawie szerokości skrzydeł na obrazie kamery. Umożliwia to wykorzystanie tanich bezzałogowców, które nie kożytają z zaawansowanych głowic naprowadzających wyposażonych w systemy radarowe bądź lidary.

## ALGORYTM FAZY PRZEJŚCIOWEJ



Algorytm fazy przejściowej dostosowuje prędkość BSP i kalibruje kamerę, aby przejść z fazy pościgu do fazy przechwycenia. Algorytm zaczyna działać po przechwyceniu, czyli w momencie, w którym naprowadzany bezzałogowiec zbliżył się do celu na wyznaczoną minimalną odległość, przy kącie pomiędzy wektorami ruchu obu obiektów wynoszącym +/- 5 stopni. Wtedy sterowanie jest zmieniane na takie, w którym algorytm ustala azymut i elewację bezpośrednio na położenie celu, korzystając z danych z zewnętrznego systemu kontroli przestrzeni powietrznej. Prędkość BSP w algorytmie dobierana jest według proporcji prędkości celu do minimalnej systemowo ograniczonej odległości pomiędzy obiektami. Kalibracja kamery obejmuje identyfikację celu na obrazie, wykorzystując algorytm detekcji krawędzi Canny'ego. W kolejnych iteracjach algorytm testuje różne maski w przestrzeni barw HSV i odczytuje na nich szerokość najszerzego kształtu. Następnie, po dokonaniu wszystkich ustalonych iteracji, wybierana jest maska, na której wykryto najszerzy z najszerzych obiektów. Najszerzy kształt uzyskany w wyniku nałożenia takiej maski oraz uruchomienia algorytmu Canny'ego będzie interpretowany jako skrzydła celu. Po określeniu wymaganej maski następuje kalibracja kamery, jak zostało to już wcześniej opisane na obecnym slajdzie. Testy wykazały wysoką skuteczność regresji liniowej. Ponadto algorytm detekcji krawędzi okazał się znacznie skuteczniejszy od algorytmów sztucznej inteligencji, szczególnie w przypadku wykorzystania kamer z małą rozdzielczością.

## 5. Algorytm fazy przechwycenia

W ostatnim etapie misji wykorzystywany jest algorytm fazy przechwycenia, który ma na celu podążanie za wrogą jednostką, używając jedynie danych z kamery, niezależnie od zewnętrznego systemu kontroli przestrzeni powietrznej. To pozwala na wykorzystanie efektorów walki elektronicznej bez wpływu na skuteczność naprowadzania, ponieważ podczas jego działania nie jest potrzebne łącze z operatorem. Takie działanie umożliwia wyprowadzenie celu ze strefy kontrolowanej oraz jego bezpieczną neutralizację za pomocą sfalszowanego sygnału GPS bądź przejęcia łącza. Algorytm pozwala również na podążanie w sposób niezauważony za śledzoną jednostką do czasu jej powrotu do miejsca startu, a co za tym idzie, odnalezienia jej operatorów.

Algorytm fazy przechwycenia, podobnie jak algorytm fazy przejściowej, wykorzystuje algorytm detekcji krawędzi Canny'ego, używając informacji o wcześniej dobranej masce w przestrzeni barw HSV. Algorytm dostosowuje azymut oraz elewację, aby utrzymać wrogiego bezzałogowca na środku obrazu z kamery. Prędkość dobierana jest w taki sam sposób jak w algorytmie fazy przejściowej, z tą różnicą, że odległość odczytywana jest z szerokości obiektu na obrazie z kamery za pomocą zależności określonej w poprzedniej fazie. Do określenia azymutu i elewacji, aby utrzymać cel na środku obrazu z kamery, wykorzystywane są następujące formuły:

Dla kierunku poziomego:

$$\varepsilon_{poz} = \arctg\left(\frac{e_{poz}}{h}\right)$$

Dla kierunku pionowego:

$$\varepsilon_{pion} = \arctg\left(\frac{e_{pion}}{h}\right)$$

gdzie,

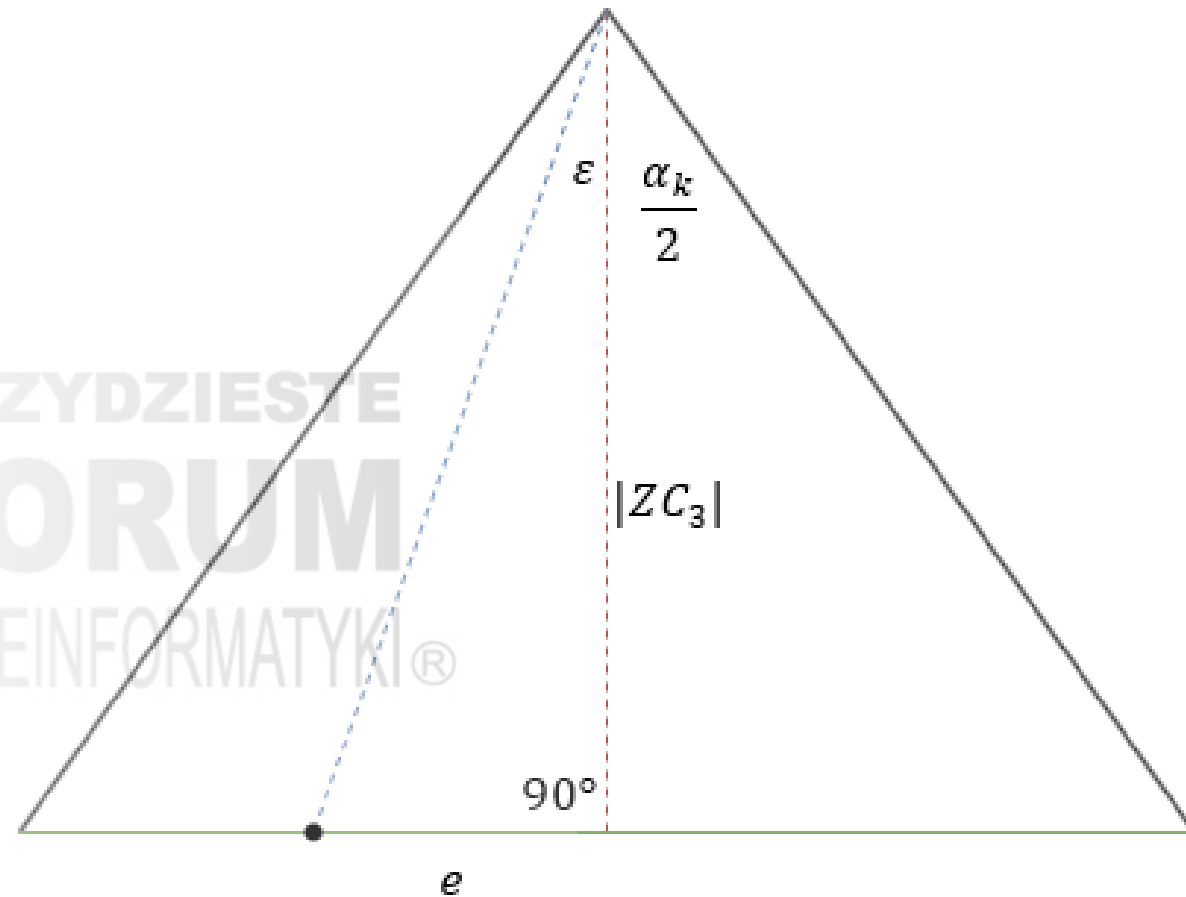
$\varepsilon_{poz}$  – wyrażona w stopniach wartość, o którą powinno naprowadzane BSP zmienić kąt w płaszczyźnie poziomej (azymut);

$\varepsilon_{pion}$  – wyrażona w stopniach wartość, o którą powinno naprowadzane BSP zmienić kąt w płaszczyźnie pionowej (elewacja);

$e_{poz}$  – odległość wyrażona w pikselach celu na kamerze w płaszczyźnie poziomej od środka obrazu;

$e_{pion}$  – odległość wyrażona w pikselach celu na kamerze w płaszczyźnie pionowej od środka obrazu;

$h$  - odległość pomiędzy obiektami oszacowana na podstawie szerokości celu na obrazie po wykorzystaniu zależności z regresji wyznaczonej w etapie II.



## 6. Testy w symulacji i warunkach rzeczywistych

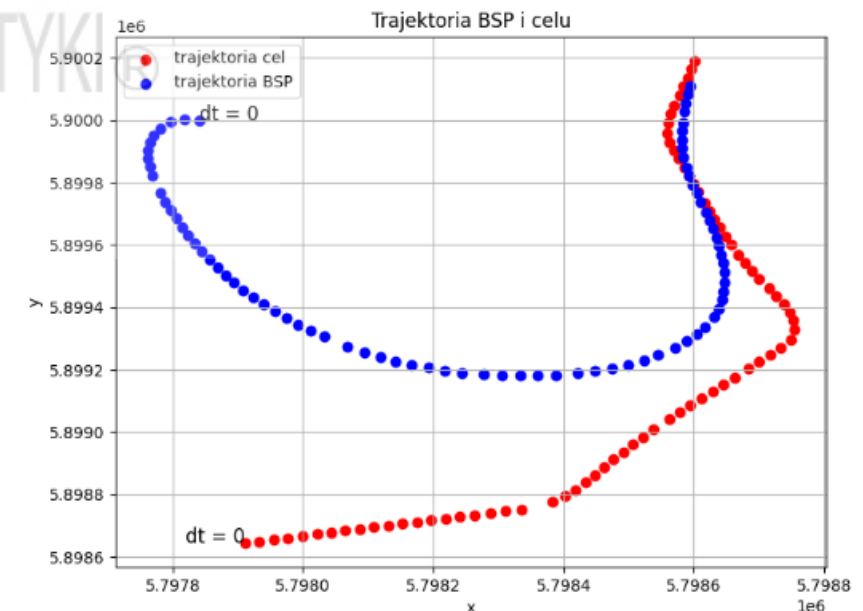
Podsystem został sprawdzony zarówno w symulacji jak i w warunkach rzeczywistych. Do wykonania testów symulacyjnych wykorzystano dwie równolegle uruchomione instancje symulatora „Xplane 11” połączone do 2 instancji autopilota „Arduplane”. W symulacji samolot będący zagrożeniem leciał po stałej trasie, na której wykonywał wiele manewrów, podczas których stale zmieniał azymut, elewację oraz prędkość. Drugi samolot sterowany był za pomocą tytułowego podsystemu.

Po pozytywnych próbach symulacyjnych wykonano próby w warunkach zbliżonych do rzeczywistych. Samolotem imitującym zagrożenie był szybki motoszybowiec, zwany hotlinerem, który aby imitował cel silnie manewrujący był sterowany ręcznie przez doświadczonego operatora. Naprowadzonym bezzałogowcem natomiast był samolot przedstawiony na poniższym rysunku. Jest to samolot specjalnie zaprojektowany i wybudowany na potrzeby projektu. Jest on zdolny do wykonania pełnej akrobacji oraz do osiągania prędkości 180 km/h co pozwala na skuteczne przechwytywanie celu, a tym samym na skuteczne przetestowanie



Testy w warunkach zbliżonych do rzeczywistych wykazały że podsystem jest tak skuteczny jak w symulacji. Na rysunku powyżej widać zdjęcie z kamery naprowadzanego BSP, tuż po wykonaniu kalibracji. Na ilustracji można zaobserwować że algorytm poprawnie zidentyfikował cel, o czym świadczy czerwona obwódka. Ponadto można zauważyć że zdjęcie jest bardzo słabej jakości co dodatkowo potwierdza skuteczność algorytmu nawet w trudnych warunkach oświetleniowych. Pokazują to przewagę wykorzystania algorytmu detekcji krawędzi Canny'ego nad wykorzystaniem sztucznej inteligencji, która aby identyfikować tego typu cele w różnych warunkach, przy wykorzystaniu różnego sprzętu mogła by okazać się zawodna, szczególnie w przypadku wykorzystania zbyt ubogich danych treningowych.

Na poniższym rysunku przedstawione zostały trajektorie obu obiektów podczas jednego z testów wykonanych w warunkach rzeczywistych. Niebieskim kolorem oznaczono trajektorię naprowadzanego BSP, natomiast czerwonym bezzałogowca imitującego zagrożenie. Jak potwierdza rysunek algorytm fazy pościgu w poprawny sposób doprowadził do przechwycenia, po czym algorytm fazy przejściowej poprawnie skalibrował kamerę, co pozwoliło na utrzymanie przechwycenia w fazie przechwycenia. Poniższe rezultaty są jednymi z wielu wykonanych, które potwierdzają skuteczność podsystemu. Zaprojektowane metody naprowadzania wykazały wyższą skuteczność w porównaniu z konwencjonalnymi metodami takimi jak nawigacja proporcjonalna.





## 7. Podsumowanie i wnioski

- **Innowacyjność rozwiązania:** Przedstawione rozwiązania, w tym wieloetapowy algorytm naprowadzania na cel, wyróżniają się elastycznością w działaniu, możliwością precyzyjnego pozycjonowania obiektów latających względem siebie oraz możliwością wykorzystania w różnych platformach BSP. Algorytmy wykorzystują zaawansowane techniki takie jak odwrócona kamera stereo, detekcja krawędzie oraz techniki nawigacyjne, co zwiększa ich uniwersalność i efektywność działania.
- **Skuteczność w testach symulacyjnych i rzeczywistych:** Wyniki testów symulacyjnych przeprowadzonych za pomocą oprogramowania „X-Plane 11” oraz testów w warunkach rzeczywistych, przeprowadzonych za pomocą bezzałogowych płatowców potwierdzają wysoką skuteczność podsystemu.
- **Precyzyjne rozpoznawanie i przechwytywanie celów:** Zaprojektowane algorytmy w poprawny sposób identyfikują cele na obrazie z kamery, nawet w przypadku obrazu o bardzo niskiej jakości, umożliwiając tym samym skuteczne śledzenie oraz przechwytywanie celów w różnych warunkach oświetleniowych i pogodowych.
- **Szerokie możliwości zastosowania i aktualność względem obecnej sytuacji bezpieczeństwa w kraju:** Nowoczesne systemy obronne, aby były skuteczne, muszą być odporne na zagrożenia, którymi coraz częściej stają się bezzałogowce. Przedstawione w pracy rozwiązanie pokazuje ogromny potencjał implementacji zaawansowanych algorytmów naprowadzających, które umożliwiają skuteczne przechwytywanie i neutralizację wrogich obiektów bezzałogowych na terenie obiektów kontrolowanych. System po integracji z zewnętrznymi systemami kontroli przestrzeni powietrznej oferuje duże możliwości ochrony przed tego typu zagrożeniami. Dzięki temu możliwe jest zapewnienie bezpieczeństwa krytycznych infrastruktur takich jak: porty, lotniska, gazociągi oraz granice, umożliwiając jednocześnie skuteczną reakcję na zagrożenia w czasie rzeczywistym, co podkreśla aktualność i niezbędność rozwijania takich technologii w nowoczesnych systemach obronnych.
- **Perspektywy rozwoju:** Podsystem ten posiada duży potencjał do dalszego rozwoju, szczególnie w kontekście integracji z nowymi technologiami i systemami obronnymi. Możliwość adaptacji algorytmów do różnych platform BSP oraz dalsze udoskonalanie technik naprowadzania i neutralizacji zagrożeń zwiększa jego wartość i zastosowanie w przyszłych operacjach obronnych oraz badawczych.



- **Osiągnięcia:** Projekt zdobył pierwsze miejsce, w konkursie o nagrodę Ministra Obrony Narodowej na realizację projektu BSP, BSL lub BSM do zastosowań związanych z obronnością i bezpieczeństwem państwa, w kategorii bojowej jako środek przeciwdziałania Bezzałogowym Systemom Powietrznym.

Dziękuję za uwagę 😊

- [1] Al-Qudsi YS., El-Bayoumi GM: *Intercept Algorithm for Maneuvering Targets Based on Differential Geometry and Lyapunov Theory*, INCAS BULLETIN, 10, 175-192, 01.03.2018.
- [2] Rahman S., Robertson, D.A., *Radar micro-Doppler signatures of drones and birds at K-band and W-band*. Sci Rep 8, 17396, 2018
- [3] Jana S., Tony LA., Bhise AA., Varun VP., Ghose D.: *Interception of an aerial maneuvering target using monocular vision*, Robotica, 40, 4535-4554, 03.08.2022.
- [4] Razooqi O.D., Ali A.H.: *Drones neutralized by utilize electromagnetic pulse (EMP) system*. Proceedings of the 2022 5th International Conference on Engineering Technology and its Applications (IICETA), 31.05.2022
- [5] Gan Z. F. T., Feng K., Khouli F.: *Development and Optimization of Flight Dynamics, Control Laws and Avionics System for UAV with a Multi-scale Optimized Blended Wing Body Configuration*. CASI AERO 2019, Laval (Quebec), 14.06.2019
- [6] <https://ardupilot.org/dev/docs/companion-computer-nvidia-tx2.html> [dostęp: 30.03.2024]
- [7] Grzyb M., Stefański K.: *Naprowadzanie bomby kierowanej z wykorzystaniem specjalnego algorytmu dla identyfikacji i śledzenia celu*. Pomiar Automatyka Robotyka, 15 (2), 668-676, 06.04.2011
- [8] Looker JR.: *Minimum Paths to Interception of a Moving Target When Constrained by Turning Radius*, Defence Science and Technology Organization Air Operations Division, Edinburgh (Australia), 01.12.2008.
- [9] Li K., Su W., Chen L.: *Performance analysis of three-dimensional differential geometric guidance law against low-speed maneuvering targets*. Astrodynamics, 2, 233-247, 21.06.2018
- [10] <https://open-dis.github.io/dis-tutorial/DeadReckoningStateUpdate> [dostęp: 22.01.2024]
- [11] Zhang X., Kusrini K., *Autonomous long-range drone detection system for critical infrastructure safety*, Multimedia Tools and Applications, 80(9), 23723-23743, 12.02.2021
- [12] Hearn D., Baker M.P., Carithers W.: *Computer Graphics with OpenGL*. Wydawnictwo Prentice Hall, USA, 4th Edition, 2010.
- [13] O'Riordan A., Newe T., Toal D.J.F., Dooly G.: *Stereo Vision Sensing: Review of existing systems*. Proceedings of the International Conference on Sensing Technology, University of Limerick, 04.12.2024
- [14] Maulud D., Abdulazeez A.M.: *A Review on Linear Regression Comprehensive in Machine Learning*. Journal of Applied Science and Technology Trends, 1(4), 140-147, 31.12.2020
- [15] Gan Z. F. T., Feng K., Khouli F.: *Development and Optimization of Flight Dynamics, Control Laws and Avionics System for UAV with a Multi-scale Optimized Blended Wing Body Configuration*. CASI AERO 2019, Laval (Quebec), 14.06.2019
- [16] <https://ardupilot.org/dev/docs/sitl-with-xplane.html> [dostęp: 30.03.2024]