

ALGORYTM STEROWANIA LOTEM MIKROSAMOLOTU W KANIONIE ULIC Z WYKORZYSTANIEM SYSTEMU WIZYJNEGO

Cezary KOWNACKI*

*Faculty of Mechanical Engineering, Białystok Technical University, ul. Wiejska 45 C, 15-351 Białystok

cezarkw@poczta.onet.pl

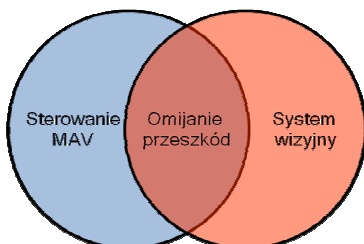
Streszczenie: Artykuł przedstawia algorytm autonomicznego sterowania lotem mikrosamolotu w kanionach ulic z wykorzystaniem obrazu z kamery jako nośnika informacji o otoczeniu mikrosamolotu. Struktura algorytmu i proces przetwarzania informacji video została tak skonstruowana, aby możliwa była jego realizacja z wykorzystaniem dostępnych urządzeń tj. mikrokamer, zaawansowanego autopilota oraz procesorów sygnałowych DSP. Na podstawie opracowanego modelu algorytmu przygotowano i wykonano badania symulacyjne. Uzyskane wyniki potwierdzają skuteczność opracowanego algorytmu w autonomicznym sterowaniu lotem mikrosamolotu w kanionach ulic. Jednakże przeprowadzone symulacje nie wykorzystywały rzeczywistego obrazu video, a jedynie uproszczony model perspektywicznego obrazu mapy ulicy. Dlatego też należy przeprowadzić dalsze badania uwzględniające wszystkie etapy przetwarzania obrazu.

1. WPROWADZENIE

Lot mikrosamolotu w nieznanym i niepewnym terenie wymaga znacznie większej autonomii sterowania aniżeli w przypadku lotu na dużych wysokościach, gdzie ryzyko napotkania przeszkody jest znikome. Poruszanie się wśród niezidentyfikowanych przeszkód wymaga specjalizowanych technik percepcji w celu odtworzenia stanu otoczenia. Zastosowanie czujników laserowych, ultradźwiękowych, radarowych pozwala określić pozycje przeszkód, jednakże jest to pomiar punktowy i ograniczony przestrzennie, co zubaża informację o otoczeniu mikrosamolotu. Sterowanie oparte

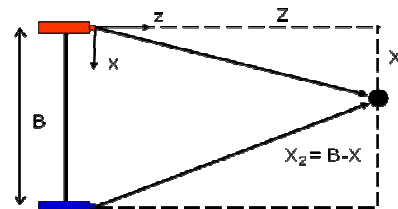
o wyżej wymienione sensory pozwala na bezkolizyjny lot, ale cel misji jest nadal kontrolowany przez system nadrzędny, jakim jest człowiek.

Stąd też prowadzi się badania na wizyjnych systemach sterowania lotem autonomicznych mikrosamolotów. Obraz przestrzeni wokół mikrosamolotu zawiera, bowiem informację nie tylko o potencjalnych przeszkodach, ale również o możliwych celach misji, a także pozwala na określenie przestrzennej orientacji mikrosamolotu. Techniki rozpoznawania obrazów pozwalają wyodrębnić na rejestrowanym obrazie obiekty o ściśle określonych celach. Obraz jest wówczas wykorzystywany nie tylko do autonomii sterowania lotem, ale również, jako źródło informacji o patrolowanym terenie.



Rys. 1. Rola systemu wizyjnego w sterowaniu lotem MAV

Nad wykorzystaniem systemów wizyjnych w algorytmach autonomicznego sterowania lotem MAV prowadzi się wiele badań (Anderson, 2002; Dong i inni, 2005; Frew i inni, 2004a, b, c; Frew, 2004; Frew i Langelan, 2005; Rathinam i inni, 2005; Saunder i inni, 2005). Przykładem systemu wizyjnego wykorzystywanego w MAV jest układ dwóch równoległych kamer (Rys. 2).



Rys. 2. Układ dwóch równoległych kamer wykorzystywany do określania pozycji przeszkody

Efektom tych badań są skuteczne, ale dość złożone algorytmy autonomicznego sterowania. Opracowane w warunkach laboratoryjnych z wykorzystaniem wydajnych komputerów osobistych prezentują one znaczne możliwości oraz potwierdzają, że przyszłe autonomiczne MAV będą oparte właśnie na systemach wizyjnych. Jednakże złożoność tych algorytmów znacznie przewyższa możliwości sprzętu pokładowego obecnych MAV. Jeżeli ograniczy się funkcjonalność systemu wizyjnego do ściśle określonego zadania możliwe będzie opracowanie specjalizowanego algorytmu, którego implementacja będzie możliwa w dostępnych energooszczędnych mikroprocesorach i procesorach DSP. Jednym z zadań, które może być zrealizowane z wykorzystaniem systemu wizyjnego jest lot w kanionie ulic.

Autor przygotował koncepcję autonomicznego sterowania lotem MAV w kanionach ulic, która zakłada, że na obrazie kamery pokładowej mikrosamolotu zawsze

można wyodrębnić linie proste, które ograniczają obszar bezpiecznego lotu. Kontrolując położenie tych linii na obrazie można sterować lotem MAV w taki sposób, aby uniknął on kolizji z budynkiem. Wyodrębnione linie proste mogą reprezentować krawędzie powierzchni drogi, krawędzie chodników dla pieszych, dolne krawędzie budynków. Najważniejszymi cechami tych linii jest to, że większość z nich schodzi się w jednym punkcie oraz ich długość pozwala na skuteczną eliminację tych linii, które na przykład reprezentują obiekty na i wokół drogi (Rys. 3).



Rys. 3. Najdłuższe linie proste wyodrębnione na obrazie tworzą bezpieczny obszar lotu

Jeśli mikrosamolot będzie sterowany tak, aby linie proste były położone w określonej odległości od punktu kontrolnego na obrazie to mikrosamolot będzie mógł w sposób autonomiczny kontynuować lot w kanionie ulicy.

2. AUTONOMICZNY SYSTEM STEROWANIA

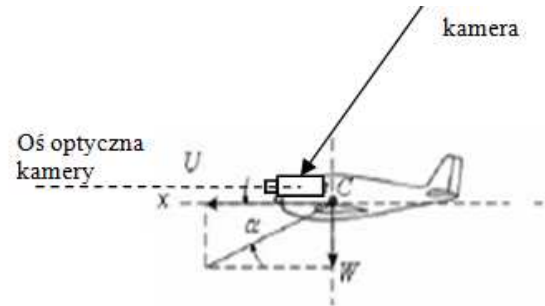
Podstawą opracowanej koncepcji algorytmu autonomicznego sterowania lotem mikrosamolotu w kanionie ulicy jest miniaturowa kamera sztywno zamocowana do płatowca MAV (Rys. 5), której oś optyczna jest równoległa do osi wzdłużnej mikrosamolotu.



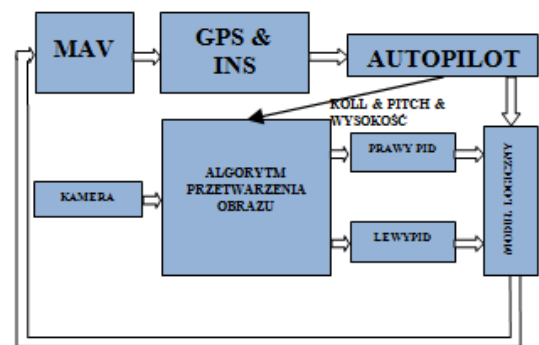
Rys. 4. Miniaturowa kamera LC-S762 z przetwornikiem SONY o czułości 0,05 lux, rozdzielczości 420TVL, trybie pracy dzień/noc

Montaż kamery na pokładzie mikrosamolotu w taki sposób, aby oś optyczna kamery była równoległa do osi wzdłużnej mikrosamolotu (Rys. 5) pozwoli uzyskać efekt, że punkt centralny obrazu będzie położony na wprost mikrosamolotu. Znając pochylenie mikrosamolotu i jego wysokość na powierzchni terenu będzie możliwe zlokalizowanie tego punktu w przestrzeni. Dodatkowo posiadając dane o polu i o kątach widzenia kamery możliwe będzie oszacowanie położenia innych punktów na obrazie.

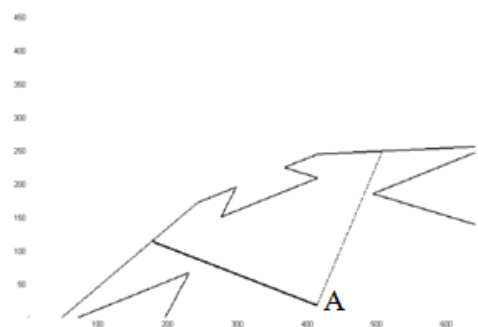
W opracowanym algorytmie zastosowano dwa regulatory PID sterujące osobno dystansami po obu stronach mikrosamolotu. Lewy regulator będzie kontrolował manewr skrętu w prawo, natomiast prawy w lewo. Ostatecznie o kierunku manewru skrętu zadecyduje moduł logiczny, którego zadaniem jest nadzorowanie aktualnych celów lotu w zależności od stanu otoczenia estymowanego na podstawie obrazu z kamery. Schemat opracowanej koncepcji sterowania lotem mikrosamolotu w kanionach ulic jest przedstawiony na Rys. 6.



Rys. 5. Sposób montażu kamery na pokładzie mikrosamolotu



Rys. 6. Koncepcja algorytmu autonomicznego sterowania lotem MAV w kanionach ulic



Rys. 7. Położenie punktu kontrolnego „A” – obraz symulacji MATLAB – SIMUINK (linie na obrazie tworzą bezpieczny obszar lotu, w którym powinien znajdować się punkt „A”)

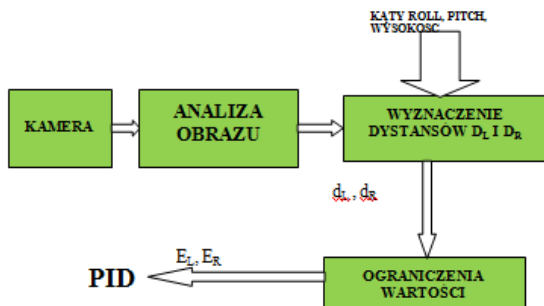
Idea opracowanej koncepcji opiera się na odpowiednim przetwarzaniu obrazu w celu uzyskania informacji o położeniu linii prostych ograniczających bezpieczny obszar lotu (Rys. 7 i 16). Położenie linii prostych będzie określone względem ustalonego punktu kontrolnego „A” położonego na rejestrowanym obrazie. Jego lokalizacja będzie ściśle zależna od kąta przechylenia mikrosamolotu, aby niwelo-

wać jego wpływ na pomiar położenia linii (Rys. 7). Zaś jego odległość od środka obrazu jest stała i równa połowie wysokości obrazu.

Położenie linii prostych ograniczających bezpieczny obszar lotu będzie określone względem punktu „A” w odpowiednio oznaczonych kierunkach. Dysponując danymi z autopilota o orientacji i położeniu mikrosamolotu oraz danymi optyki zastosowanej kamery możliwe będzie oszacowanie przestrzennej lokalizacji mikrosamolotu względem obiektów reprezentowanych przez linie na obrazie. Szczegółowy algorytm przetwarzania obrazu zaprezentowany będzie w dalszych rozdziałach.

Dane o przestrzennej pozycji (odległości) mikrosamolotu względem obiektów reprezentowanych przez linie (krawędzie drogi, chodników dla pieszych itd.) będą sygnałami sterującymi lotem mikrosamolotu. Różnica pomiędzy przyjętą bezpieczną odległością, a odległością oszacowaną na podstawie obrazu jest uchybem minimalizowanym przez standardowe regulatory PID, które z kolei sterują kątem przechylenia i promieniem skrętu mikrosamolotu. Ponieważ lot w kanionie ulicy powinien odbywać się również z zachowaniem nadrzędnego celu, którymi są punkty przelotowe, moduł logiczny przełącza sterowanie mikrosamolotem pomiędzy regulatorami PID a układem autopilota realizującego lot wzdłuż zaplanowanej trasy. Innymi słowy moduł logiczny będzie decydował o aktualnym priorytecie zadań sterowania tj. lotu według wyznaczonej trasy i bezkolizyjnego lotu w kanionach ulic.

Algorytm przetwarzania obrazu, którego efektem końcowym są sygnały odległości mikrosamolotu od obiektów ograniczających bezpieczny obszar lotu jest przedstawiony na Rys. 8.

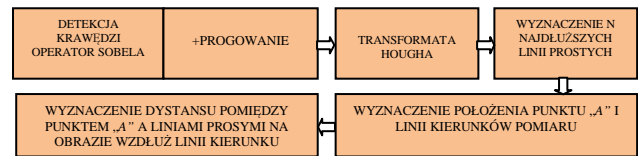


Rys. 8. Algorytm przetwarzania obrazu w opracowanej koncepcji autonomicznego sterowania lotem MAV w kanionach ulic, D_L, D_R – położenie obiektów względem mikrosamolotu w kierunku poprzecznym do bieżącego kursu, E_L, E_R – sygnały uchybu, różnica pomiędzy wartościami d_L, d_R a bezpiecznym dystansem D_{safe}

Kamera rejestruje obraz przedstawiający przestrzeń przed mikrosamolotem. Następnie obraz jest poddawany analizie, która polega na detekcji krawędzi i wyznaczeniu najdłuższych linii prostych, które reprezentują krawędzie jezdni, chodnika dla pieszych lub budynków oraz wykonywany jest pomiar położenia zidentyfikowanych linii prostych względem punktu kontrolnego „A”. Wyznaczone wartości położenia linii względem punktu kontrolnego na obrazie są przeliczane na dystanse rzeczywistych obiektów od mikrosamolotu, wykorzystując w tym celu dane o położeniu i orientacji mikrosamolotu uzyskane z układu

autopilota. Innymi słowy rzeczywista lokalizacja obiektów ograniczających bezpieczny obszar lotu jest estymowana przez algorytm na podstawie obrazu z kamery. Mikrosamolot powinien być sterowany tak, aby zawsze zachowywał on bezpieczną odległość od tych obiektów. Sygnały odległości powinny być filtrowane by wyeliminować losowe zakłócenia, które mogą powstać w wyniku drgań obrazu, ograniczonej prędkości rejestrowania sekwencji obrazów czy też z ograniczonej rozdzielczości obrazu i wynikającej z tego dokładności lokalizacji krawędzi i linii prostych.

Etapy analizy obrazu, dzięki którym możliwa jest detekcja linii prostych i określenie ich pozycji względem punktu kontrolnego „A” są przedstawione w postaci schematu na Rys. 9.



Rys. 9. Etapy analizy obrazu opracowanej koncepcji autonomicznego sterowania lotem MAV w kanionach ulic

2.1. Analiza obrazu

Pierwszym etapem analizy obrazu jest detekcja krawędzi połączona z jednoczesnym progowaniem, które pozwala na eliminację zakłóceń w postaci punktów reprezentujących drobne krawędzie. Do detekcji krawędzi na obrazie wykorzystuje się operatory Sobela (Rys. 10).

1	2	1	-1	0	1
0	0	0	-2	0	2
-1	-2	-1	-1	0	1
0°			90°		

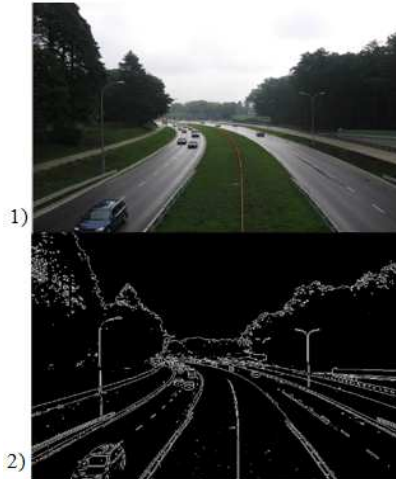
Rys. 10. Operatory Sobela

Wynik zastosowania maski operatora Sobela wykorzystujący przykładowy obraz ulicy jest ukazany na Rys. 11. Ponieważ operator Sobela wykrywa wszystkie możliwe krawędzie pomiędzy obszarami o różnych wartościach intensywności, należy zastosować równoczesne progowanie, które wyeliminuje drobne krawędzie mogące mieć wpływ na pozostałe etapy analizy obrazu. W ten sposób może być tracona również informacja o istotnych, ale mało wyraźnych krawędziach, które mogą reprezentować mniejsze przeszkody zlokalizowane wzdłuż drogi. Z drugiej jednak strony wzrasta pewność prawidłowej detekcji krawędzi jezdni, chodników, ścieżek rowerowych lub krawędzi budynków, które mogą być wykorzystane w autonomicznym sterowaniu lotem MAV.

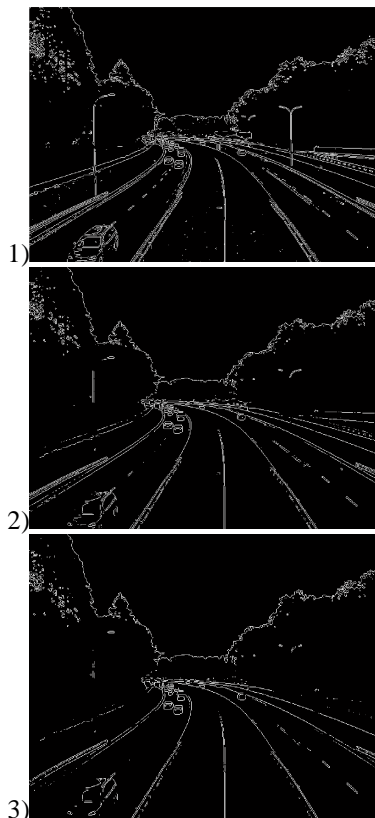
Na Rys. 12 przedstawiono efekty zastosowania różnych wartości progów. Przy większych wartościach progów widać mniejszą ilość drobnych krawędzi a nawet brak krawędzi

reprezentujących pewne elementy obrazu np. latarni ulicznych.

Kolejnym etapem analizy obrazu jest identyfikacja linii prostych na obrazie przedstawiającym krawędzie wyznaczone z użyciem maski Sobela. Narzędziem, które to umożliwia jest transformata Hougha.



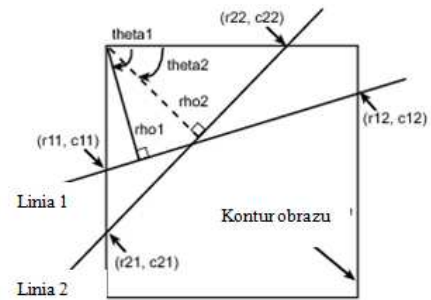
Rys. 11. Detekcja krawędzi - zastosowanie maski operatora Sobela, program MATLAB - SIMUINK



Rys. 12. Efekty zastosowania różnych wartości progów podczas detekcji krawędzi maską Sobela

Transformata Hougha wyznacza parę parametrów opisujących jednoznacznie linię prostą, tj. kąt pomiędzy osią OX a normalną do zidentyfikowanej linii prostej oraz jej najmniejszą odległość do początku układu współrzędnych

OXY (Rys. 13). Parametry te tworzą tzw. Przestrzeń Hougha.



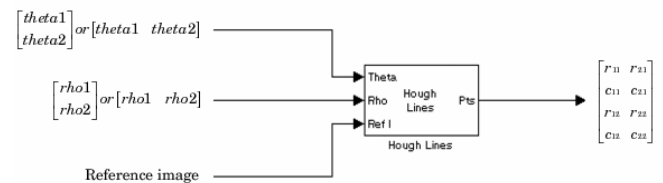
Rys. 13. Graficzna reprezentacja przestrzeni Hougha – parametrów (θ , ρ) opisujących linie proste wyznaczone przez transformatę Hougha (punkty (r_{ij}, c_{ij}) są punktami przecięcia się linii prostej z konturem obrazu)

Równanie opisujące linię prostą w przestrzeni Hougha jest następujące:

$$\rho = x \cdot \cos(\theta) + y \cdot \sin(\theta), \quad (1)$$

gdzie: x, y – punkt obrazu należący do linii prostej, ρ – odległość linii prostej od początku układu współrzędnych, θ – kąt pomiędzy normalną do prostej a osią OX.

W programie MATLAB-SIMULINK istnieje Toolbox *Video & Image Processing* umożliwiający wykorzystanie transformaty Hougha w modelu SIMULINKa. W tym celu zaimplementowano dwa bloki. Pierwszy *Hough Transform* wyznacza parametry przestrzeni Hougha, zaś drugi *Hough Lines* wyznacza współrzędne punktów przecięcia się zidentyfikowanych linii prostych z konturem obrazu (Rys. 13). Dla każdej zidentyfikowanej linii prostej wyznaczone są dwa punkty przecięcia z konturem obrazu, stąd znajomość współrzędnych tych punktów umożliwia wyznaczenie równania tej prostej (2).



Rys. 14. Blok *Hough Lines* z pakietu narzędziowego *Video & Image Processing Toolbox* programu MATLAB – SIMULINK

$$y - r_{i1} = \frac{r_{i2} - r_{i1}}{c_{i2} - c_{i1}} \cdot (x - c_{i1}), \quad (2)$$

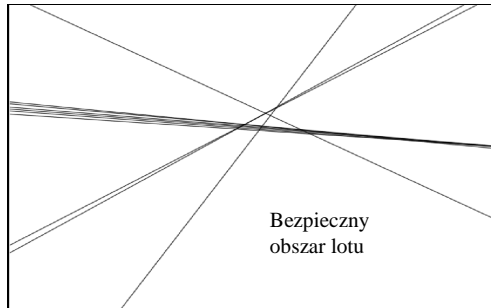
gdzie: r_{i1}, c_{i1} – współrzędne 1-ego punktu przecięcia się linii prostej z konturem obrazu, r_{i2}, c_{i2} – współrzędne 2-ego punktu przecięcia się linii prostej z konturem obrazu.

Rys. 15 przedstawia przykładowe zdjęcia ulicy z wyznaczonymi najdłuższymi liniami prostymi i z wykorzystaniem transformaty Hougha.

Natomiast zlokalizowane linie proste bez obrazu wejściowego jako tła są przedstawione na Rys. 16. Na rysunku 16 jest również zaznaczony bezpieczny obszar lotu. Punkt kontrolny „A” powinien się znajdować wewnątrz tego obszaru.



Rys. 15. Najdłuższe linie proste zlokalizowane na obrazach z wykorzystaniem transformaty Hougha



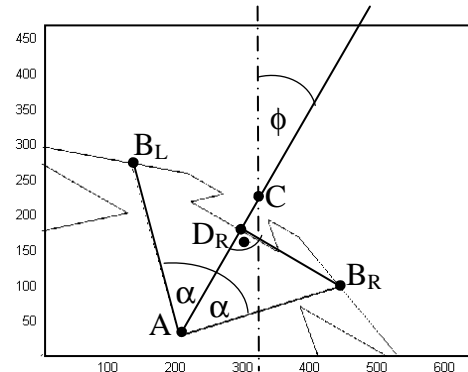
Rys. 16. Linie proste wyznaczone na podstawie obrazu drogi. (bezpieczny obszar lotu jest ograniczony dolnym konturem obrazu i trzema liniami prostymi. Wewnątrz tego obszaru powinien znajdować się punkt kontrolny „A”)

Kolejnym etapem analizy jest wyznaczanie odległości punktu kontrolnego „A” od prostych ograniczających bezpieczny obszar lotu. Położenie punktu kontrolnego „A” na obrazie jest określane w następujący sposób. Odległość punktu kontrolnego „A” od środka obrazu „C” jest stała i równa połowie wysokości obrazu w pikselach P_y (Rys. 17). Ponadto odcinek AC tworzy z osią symetrii obrazu kąt równy aktualnemu kątowi przechylenia ϕ . W ten sposób położenie punktu kontrolnego „A” względem linii prostych będzie niezależne od przechylenia mikrosamolotu. Odległość punktu kontrolnego od linii prostych ograniczających bezpieczny obszar lotu jest wyznaczana w kierunkach oznaczonych prostymi tworzącymi z odcinkiem AC kąt α . Równania tych prostych są następujące:

$$\begin{aligned} 1: y - A_y &= \tan(\pi - \phi - \alpha) \cdot (x - A_x) \\ 2: y - A_y &= \tan(\pi - \phi + \alpha) \cdot (x - A_x) \\ A_x &= \frac{P_x}{2} - \cos(\phi) \cdot \frac{P_y}{2} \\ A_y &= \frac{P_y}{2} - \sin(\phi) \cdot \frac{P_y}{2} \end{aligned} \quad (3)$$

gdzie: A_x, A_y – współrzędne punktu kontrolnego „A”, P_x, P_y – rozmiary obrazu w pikselach, ϕ – kąt przechylenia mikrosamolotu.

Kąt ten determinuje on minimalną szerokość przeszkody od przeszkody, od której powinien rozpocząć się manewr omijania zapewniający bezkolizyjny lot. Innymi słowy wartość kąta α ma wpływ na czas wyprzedzenia reakcji układu sterowania na zbliżającą się przeszkodę ograniczającą bezpieczny obszar lotu.



Rys. 17. Wyznaczanie położenia linii prostych ograniczających bezpieczny obszar lotu względem punktu „A”. „C” – środek obrazu, $|AB_R|, |AB_L|$ – odległość prostych od punktu kontrolnego „A”, $|D_R B_R|$ – odległość prostej ograniczającej bezpieczny obszar lotu z prawej strony od linii wyznaczającej bieżący kurs lotu. Obraz z symulacji SIMULINKA

W celu określenia położenia linii prostych względem punktu kontrolnego „A” wyznacza się punkty B_R, B_L, D_R i D_L oraz następujące odległości zgodnie z równaniami (4). Długości odcinków z indeksem p są wyrażone w pikselach, natomiast bez indeksu p w metrach.

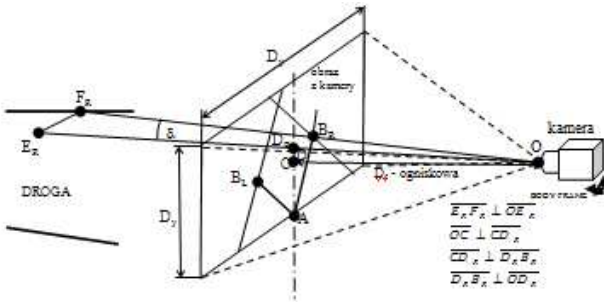
$$\begin{aligned} |\overline{CD}_i| &= \left(|\overline{AC}|_p - |\overline{AD}_i|_p \right) \cdot k, \\ |\overline{AC}|_p &= \frac{1}{2} \cdot P_y, \\ |\overline{AD}_i|_p &= |\overline{AB}_i|_p \cdot \cos(\alpha), \\ |\overline{D_i B}_i|_p &= |\overline{AB}_i|_p \cdot \sin(\alpha), \\ k &= \frac{D_x}{P_x} = \frac{D_y}{P_y} \\ |\overline{OC}| &= D_f, \end{aligned} \quad (4)$$

gdzie: P_x, P_y – wymiary obrazu w pikselach, D_x, D_y – wymiary obszaru reprezentującego pełny obraz, prostopadłego do osi optycznej kamery i oddalonego od kamery o odległość D_f , D_f – odległość obszaru reprezentującego pełny obraz kamery, prostopadłego do jej osi optycznej, $|AB_i|_p$ – odległość punktu „A” od linii prostej ograniczającej bezpieczny obszar lotu z prawej strony, $|AD_i|_p$ – odległość punktu „A” od linii prostej w kierunku osi OY, $|D_i B_i|_p$ – odległość punktu „A” od linii prostej w kierunku osi OX, k – współczynnik określający długość odcinka należącego do obszaru reprezentowanego przez obraz,

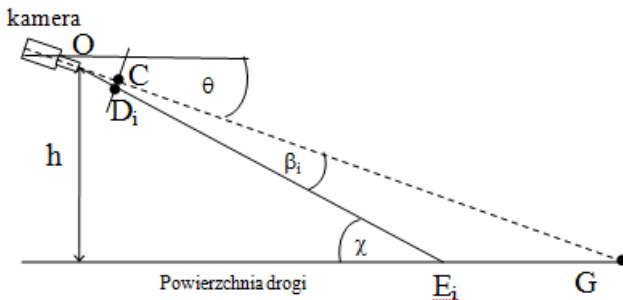
prostopadłego do osi optycznej kamery i oddalonego od niej o odległość D_f , wyrażoną w metrach odpowiadającą jednemu pikselowi na obrazie, $i - L$ – lewa strona lub R – prawa strona.

Zależności (4) wylicza się dla obu punktów B_R i B_L . Punkty B_R i B_L są punktami przecięcia się prostych opisanych równaniami (2) i (3). Punkty D_R i D_L są odpowiednio rzutami punktów B_R i B_L na prostą wyznaczoną odcinkiem AC.

Znając długości odcinków $|D_i B_i|$, $|CD_i|$, $|OC|$ ($i \in \{L, R\}$) i posiadając informacje o kącie pochylenia mikrosamolotu oraz o jego wysokości lotu nad terenem można oszacować położenie obiektów reprezentowanych przez linie proste na obrazie. Sposób oszacowania jest przedstawiony na Rys. 18 i 19 oraz opisany zależnościami (5).



Rys. 18. Geometryczne zależności wykorzystane w oszacowaniu położenia obiektów reprezentowanych przez linie proste na obrazie



Rys. 19. Wykorzystanie informacji o kącie pochylenia mikrosamolotu oraz jego wysokości na terenie do oszacowania położenia obiektów reprezentowanych przez linie proste na obrazie

$$d_i = |E_i F_i| = |OE_i| \cdot \tan(\delta_i) = |OE_i| \cdot \frac{|D_i B_i|_p \cdot k}{|OD_i|},$$

$$|OD_i| = \sqrt{(|OC|)^2 + (|CD|)^2},$$

$$|OE_i| = \frac{h}{\sin(\chi_i)},$$

$$\chi_i = \beta_i + \theta$$

$$\beta = \arctan\left(\frac{|CD_i|}{|OC|}\right),$$

$$E_i = D_{safe} - |E_i F_i|$$

gdzie: θ – kąt pochylenia mikrosamolotu, h – wysokość lotu nad terenem, χ_i – kąt pomiędzy powierzchnią terenu a

prostą wyznaczoną przez punkty O i E_i , β_i – kąt pomiędzy odcinkami OD_i i OC , δ_i – kąt pomiędzy odcinkami OB_i i OD_i , k – współczynnik określający długość odcinka należącego do obszaru reprezentowanego przez obraz, prostopadłego do osi optycznej kamery i oddalonego od niej o odległość D_f , wyrażoną w metrach odpowiadającą jednemu pikselowi na obrazie, $i - L$ – lewa strona lub R – prawa strona, $|E_i F_i|$ – odległość d_i obiektu reprezentowanego przez linie proste na obrazie od mikrosamolotu mierzona w kierunku prostopadłym do aktualnego kursu. Jest to również wartość wykorzystywana w wyznaczaniu uchybów E_L i E_R (Rys. 8).

Wartość odległości d_R i d_L są końcowym wynikiem etapu analizy obrazu i określają odległość granic bezpiecznego obszaru lotu od bieżącego toru lotu wyznaczonego aktualnym kursem. Ponieważ algorytm autonomicznego sterowania lotem powinien ingerować w sterowanie lotem mikrosamolotu jedynie w momencie, gdy wyznaczone odległości będą mniejsze niż bezpieczny dystans D_{safe} , na wyznaczone wartości powinny być nałożone ograniczenia zgodnie z następującymi zależnościami (Rys. 8):

$$\begin{cases} E_i = d_i^* - D_{safe} & d_i^* - D_{safe} < 0 \\ E_i = 0 & d_i - D_{safe} \geq 0 \end{cases} \quad (6)$$

gdzie: d_i – dystans do obiektu ograniczającego bezpieczny obszar lotu oszacowany na etapie analizy obrazu, E_i – uchyb, różnica pomiędzy dystansem d_i , a wartością bezpieczną D_{safe} , $i - L$ – lewa strona lub R – prawa strona. Wartości uchybów E_R i E_L są sygnałami wejściowymi do regulatorów PID, zaś cały proces sterowania lotem polega na ich minimalizacji (Rys. 6).

Niezbędnym elementem struktury systemu sterowania jest moduł logiczny. Jego zadaniem jest decydowanie o aktualnym priorytecie sterowania tzn. czy mikrosamolotem steruje układ autonomicznego lotu w kanionie ulic zapewniając bezkolizyjny lot, czy układ autopilota, który kieruje lotem według zaplanowanej trasy przelotowej. Stąd też kontroler logiczny powinien dysponować zarówno sygnałami $\phi_{PIDleft}$ i $\phi_{PIDright}$ z regulatorów PID, wartościami dystansów d_L , d_R jak i sygnałem kąt przechylenia $\phi_{autopilot}$ generowanym przez układ autopilota. Układ autonomicznego sterowania lotem w kanionach ulic kontroluje kierunek lotu jedynie w sytuacji, gdy wartość oszacowanych dystansów będzie mniejszą od wartości bezpiecznej D_{safe} . W przeciwnym przypadku lotem steruje autopilot kierując mikrosamolot w stronę kolejnego punktu przelotowego. Zależności opisujące działanie modułu logicznego są następujące (Kownacki, 2009):

$$\phi^C = \begin{cases} \phi_{autopilot} & : \phi_{PIDright} = 0 \cap \phi_{PIDleft} = 0 \\ \phi_{PIDright} + \phi_{PIDleft} & : \phi_{PIDright} \neq 0 \cap \phi_{PIDleft} \neq 0 \cap |\phi_{PIDright} - \phi_{PIDleft}| > W \\ \phi_{PIDright} & : (\phi_{PIDright} \neq 0 \cap \phi_{PIDleft} = 0) \cup \\ & \left(\phi_{PIDright} \neq 0 \cap \phi_{PIDleft} \neq 0 \cap |\phi_{PIDright} - \phi_{PIDleft}| \leq W \cap \right. \\ & \left. d_L > d_R \right) \\ \phi_{PIDleft} & : (\phi_{PIDright} = 0 \cap \phi_{PIDleft} \neq 0) \cup \\ & \left(\phi_{PIDright} \neq 0 \cap \phi_{PIDleft} \neq 0 \cap |\phi_{PIDright} - \phi_{PIDleft}| \leq W \cap \right. \\ & \left. d_L \leq d_R \right) \end{cases} \quad (7)$$

$$W = 0.1 \cdot \max(\phi_{PIDright}, \phi_{PIDleft})$$

gdzie: ϕ^C – zadany kąt przechylenia, $\phi_{\text{autopilot}}$ – zadany kąt przechylenia wyznaczony przez autopilota, ϕ_{PIDleft} – zadany kąt przechylenia wyznaczony przez regulator PID_L z uchybem E_L , ϕ_{PIDright} – zadany kąt przechylenia wyznaczony przez regulator PID_R z uchybem E_R , d_R , d_L – dystanse obiektów ograniczających bezpieczny obszar lotu od aktualnego toru lotu, W – współczynnik określający wartość graniczną różnicy $\phi_{\text{PIDright}} - \phi_{\text{PIDleft}}$, powodującą zmianę sposobu sterowania, tzn. jeśli wartość różnicy jest mała wówczas wartości d_R i d_L decydują o tym który regulator steruje kątem przechylenia. W przeciwnym przypadku kąt przechylenia jest wypadkową sygnałów obu regulatorów.

Jeśli wyjścia obu regulatorów PID tzn. lewego i prawego są równe zero wówczas autopilot steruje kierunkiem lotu mikrosamolotu. W przypadku, gdy te sygnały są różne od zera oraz posiadają różne wartości a różnica jest większa od W – sygnał sterujący kątem przechylenia jest sumą sygnałów wyjściowych obu regulatorów PID. W tym momencie należy wspomnieć, że oba regulatory mają identyczne nastawy, zaś sygnały wyjściowe obu regulatorów PID różnią się znakiem, gdyż każdy regulator kontroluje skręt mikrosamolotu w przeciwnych kierunkach – dodatnie lub ujemne wartości kąta przechylenia. Kolejnym przypadkiem jest sytuacja, gdy sygnał wyjściowy lewego regulatora PID_L jest równy zero lub gdy sygnały obu regulatorów mają wartości różne od zera a ich różnica jest mniejsza od W oraz jednocześnie dystans $d_L > d_R$. Wówczas kątem przechylenia będzie sterował prawy regulator PID_R . W odwrotnej sytuacji, gdy sygnał wyjściowy z prawego regulatora PID_R jest równy zero lub sygnały z obu regulatorów mają wartości różne od zera a ich różnica jest mniejsza od W oraz jednocześnie dystans $d_R > d_L$, kąt przechylenia jest kontrolowany przez regulator PID_L .

3. SYMULACJA ALGORYTMU AUTONOMICZNEGO STEROWANIA LOTEM MIKROSAMOLOTU

Badania symulacyjne opracowanego algorytmu autonomicznego sterowania z wykorzystaniem kamery przeprowadzono w środowisku MATLAB – SIMULINK. Jednakże z braku możliwości wykorzystania rzeczywistego obrazu lotu mikrosamolotu w kanionie ulicy zbudowano model wyznaczający binarny obraz widoku perspektywicznego na mapę otoczenia zapisaną w pliku BMP. Na uzyskanym obrazie wyszukiwano linie krawędzi obszaru ulicy i następnie dalszą analizę przeprowadzano zgodnie z opracowanym algorytmem autonomicznego sterowania. Stąd w zbudowanym modelu nie uwzględniono etapu wykorzystującego maskę Sobela i transformatę Hougha. Model wykorzystuje binarny obraz krawędzi i wyznacza położenie tych krawędzi względem punktu kontrolnego „A” zgodnie z przedstawionymi zależnościami (4) i (5).

W badaniach symulacyjnych wykorzystano model mikrosamolotu w układzie delta. Wprowadzono jednak równanie opisujące ruch samolotu w kierunku z tzn. wprowadzono dodatkowy parametr w postaci wysokości lotu. Parametr ten jest niezbędny podczas wyznaczania obrazu perspektywicznego mapy otoczenia. Równania opisujące

zastosowany model mikrosamolotu są następujące (Kownacki, 2009):

$$\begin{aligned}\dot{x} &= V \cdot \cos \psi \cdot \cos \theta, \\ \dot{y} &= V \cdot \sin \psi \cdot \cos \theta, \\ \dot{z} &= V \cdot \sin \theta,\end{aligned}\tag{8}$$

$$\psi = \frac{g}{V} \cdot \tan \phi,$$

$$\dot{V} = \alpha_v (V^C - V),$$

$$\dot{\phi} = \alpha_\phi (\phi^C - \phi),$$

$$\dot{\theta} = \alpha_\theta (\theta^C - \theta),$$

gdzie: x , y , z – aktualne współrzędne mikrosamolotu, V – aktualna prędkość mikrosamolotu, V^C – zadana prędkość mikrosamolotu, ψ – aktualny kurs mikrosamolotu, ϕ – aktualny kąt przechylenia, ϕ^C – zadany kąt przechylenia, α_ϕ , α_v , α_θ – stałe czasowe dynamiki mikrosamolotu, g – stała grawitacyjna, θ – aktualny kąt pochylenia, θ^C – zadany kąt pochylenia,

Model zakłada, że lot mikrosamolotu jest kontrolowany przez następujące parametry: prędkość lotu V^C , kąt przechylenia ϕ^C i kąt pochylenia θ^C . Kąt przechylenia jest odpowiedzialny za zmianę kierunku lotu, a pochylenia za zmianę wysokości lotu. W przeprowadzonych symulacjach założono, że prędkość lotu oraz wysokość lotu jest stała, zaś algorytm kontroluje jedynie kierunek lotu.

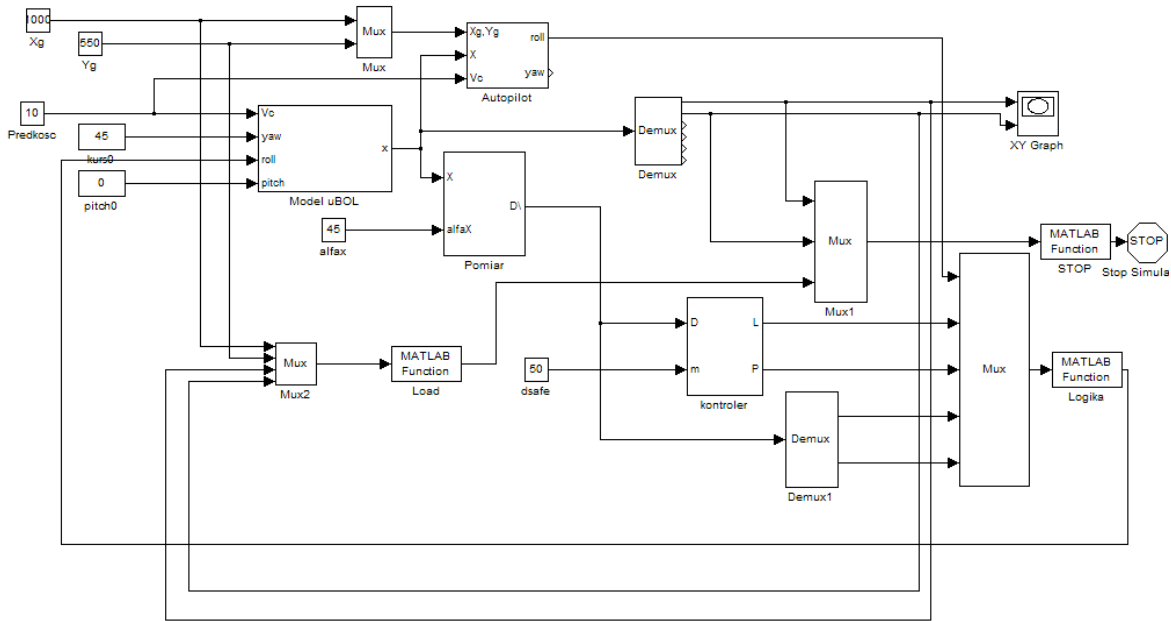
Do przeprowadzania symulacji niezbędny był również model autopilota. Przyjęty model autopilota opisany jest zależnością 9. Kontroluje on kierunek lotu przez minimalizację uchybu pomiędzy aktualnym zadanym kursem na kolejny punkt przelotowy, a aktualnym kursem mikrosamolotu. Model autopilota opisują równania 9 (Kownacki, 2009):

$$\begin{aligned}\psi_{\text{err}} &= \psi_{\text{WP}} - \psi, \\ \psi_{\text{WP}} &= \arctan\left(\frac{y_{\text{WP}} - y}{x_{\text{WP}} - x}\right) \\ \phi_{\text{autopilot}} &= \text{PID} \cdot \psi_{\text{err}}\end{aligned}\tag{9}$$

gdzie: ψ_{err} – uchyb kursu, ψ – aktualny kurs mikrosamolotu, ψ_{WP} – aktualny kurs na kolejny punkt przelotowy, x , y – aktualne współrzędne mikrosamolotu, x_{WP} , y_{WP} – współrzędne kolejnego punktu przelotowego, PID – transmitancja regulatora PID, $\phi_{\text{autopilot}}$ – zadany kąt przechylenia wyznaczony przez autopilota.

Model opracowanego algorytmu autonomicznego sterowania lotem mikrosamolotu w z wykorzystaniem kamery wykonany w SIMUKINKU jest przedstawiony na Rys. 20.

W badaniach symulacyjnych wykorzystano syntetyczne mapy otoczenia miejskiego. Zostały one przygotowane w postaci plików graficznych w formacie bitmapy wczytywanych do przestrzeni roboczej programu MATLAB. Następnie obliczany był obraz rzutu perspektywicznego mapy w taki sposób, że dwuwymiarowa mapa widziana była jako powierzchnia terenu z zaznaczoną granicą jezdni – otoczenie (Rys. 7). Wyznaczony obraz perspektywiczny jest funkcją położenia i orientacji samolotu. W ten sposób uzyskano symulację rzeczywistego obrazu krawędzi / linii prostych, które są lokalizowane przez opracowany algorytm na etapie analizy obrazu.

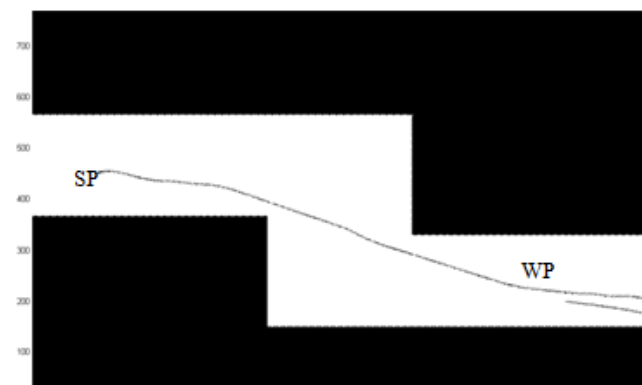


Rys. 20. Model opracowanego algorytmu autonomicznego sterowania lotem mikrosamolotu z wykorzystaniem kamery wykonany w SIMULINKU

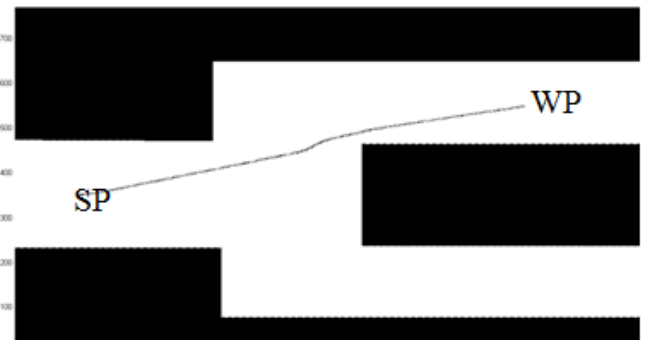
4. WYNIKI BADAŃ SYMULACYJNYCH

W celu weryfikacji działania opracowanej koncepcji autonomicznego sterowania lotem w kanionach ulic przeprowadzono badania symulacyjne. Dwie mapy przedstawiają kaniony ulic, natomiast dwie dalsze prezentują przeszkodę, jaka może się pojawić bezpośrednio na trasie lotu mikrosamolotu. Wyniki symulacji przedstawione na Rys. 21-24 są wizualizacją trajektorii lotu mikrosamolotu. Kolor biały oznacza bezpieczny obszar lotu, zaś czarny jest obszarem tworzącym kanion ulicy lub przeszkodę, która powinna być ominięta. Granice mapy są również traktowane jako obszar przeszkody. W każdym z przypadków symulowany lot mikrosamolotu rozpoczyna się w punkcie startowym SP z ustalonym początkowym kursem, a kończy się w kolejnym punkcie przelotowym WP.

zaś dystans bezpieczny $D_{safe}=50m$. Zmiennymi parametrami symulacji były współrzędne punktu początkowego, początkowy kurs oraz współrzędne punktu końcowego.



Rys. 21. Trajektoria lotu mikrosamolotu w pierwszej symulacji lotu w kanionie ulicy. Początkowy kurs 300, punkt startu SP – $x=100, y=450$, kolejny punkt przelotowy WP – $x=800, y=200$

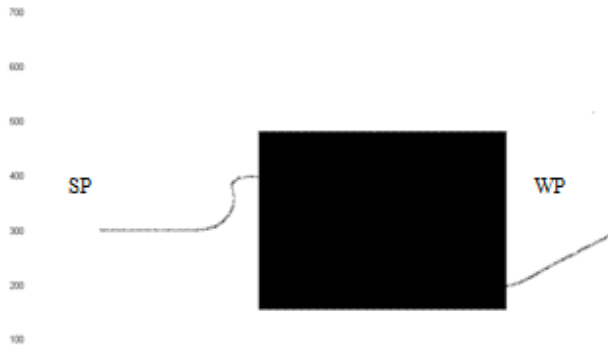


Rys. 22. Trajektoria lotu mikrosamolotu w drugiej symulacji lotu w kanionie ulicy. Początkowy kurs 00, punkt startu SP – $x=100, y=350$, kolejny punkt przelotowy WP – $x=800, y=550$



Rys. 23. Trajektoria lotu mikrosamolotu w pierwszej symulacji zadania omijania przeszkody. Początkowy kurs – 250, punkt startu SP – $x=100, y=700$, kolejny punkt przelotowy WP – $x=900, y=200$

W symulacji przyjęto stałą prędkość lotu $V^C=12m/s$, stałą wysokość lotu $z=50m$, wartość kąta α wynosi 60°



Rys. 24. Trajektoria lotu mikrosamolotu w drugiej symulacji zadania omijania przeszkody. Początkowy kurs 00, punkt startu SP – $x=100$, $y=300$, kolejny punkt przelotowy WP – $x=800$, $y=300$

Z zaprezentowanych wyników można wywnioskować, że opracowana koncepcja autonomicznego sterowania może być wykorzystana w autonomicznym locie w kanionach ulic, co potwierdzają Rys. 22-22. Natomiast w sytuacji manewru omijania przeszkody i konieczności wykonywania zwrotów o 90^0 (Rys. 23, 24) algorytm nie był w stanie sterować lotem w sposób bezkolizyjny. Przyczyną takich rezultatów jest ograniczone pole widzenia krawędzi bezpiecznego obszaru lotu. W sytuacji, gdy krawędź drogi, jezdnii itp. nie jest widoczna na obrazie kamery układ sterowania przełącza się na sterowanie kierunkiem lotu poprzez autopilota, mimo, że nadal nie ma wolnej przestrzeni lotu. Następuje wówczas „skoszenie zakrętu”. Pole widzenia kamery zależy od dwóch parametrów lotu:

- wysokości lotu;
- od ogniskowej obiektywu, tj. kąta widzenia kamery.

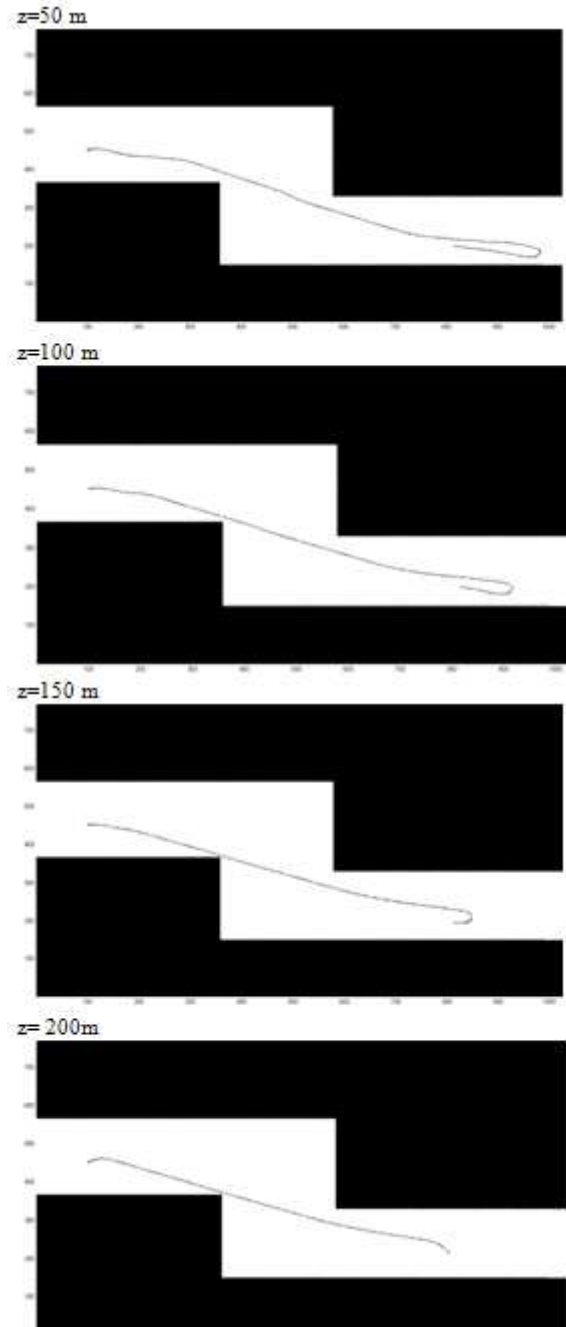
Dodatkowym parametrem mającym wpływ na manewr omijania jest kąt α , gdyż reguluje on wyprzedzenie, z którym mikrosamolot reaguje na zmiany otoczenia. Mniejsza wartość kąta α powoduje, że mikrosamolot reaguje na dalsze otoczenie kanionu ulicy. Natomiast jeśli wartość kąta jest większa to o kierunku lotu mikrosamolotu decyduje jego bliższe otoczenie. Również bezpieczny dystans D_{safe} ma wpływ na właściwą reakcję układu autonomicznego sterowania. Większa wartość tego parametru powoduje lot w większej odległości od przeszkody.

Należy jednak zwrócić uwagę, że symulowana perspektywa mapy otoczenia użyta w badaniach jest pewnym uproszczeniem, gdyż nie uwzględnia ona informacji o pionowych krawędziach budynków, które również ograniczają bezpieczny obszar lotu i mogą być zlokalizowane przez algorytm. Dlatego też sytuacje przedstawione na rysunkach 23 i 24 mogą nie mieć miejsca w rzeczywistym locie mikrosamolotu. Aby zwizualizować wpływ wysokości lotu oraz wartości kąta α przeprowadzono kolejne badania symulacyjne, których wyniki są przedstawione na kolejnych rysunkach 25 i 26. Trajektorie lotu przedstawione na rysunkach 25 i 26 potwierdzają wpływ wysokości lotu oraz kąta α na stopień bezpieczeństwa lotu. Otóż większa wysokość lotu ($z=200m$) lub mniejsza wartość kąta α ($\alpha=20^0$) powoduje, że trajektoria lotu niebezpiecznie zbliża się do granicy bezpiecznego obszaru lotu. Niższa wysokość lotu lub większa wartość kąta α powoduje natomiast wzrost odległości

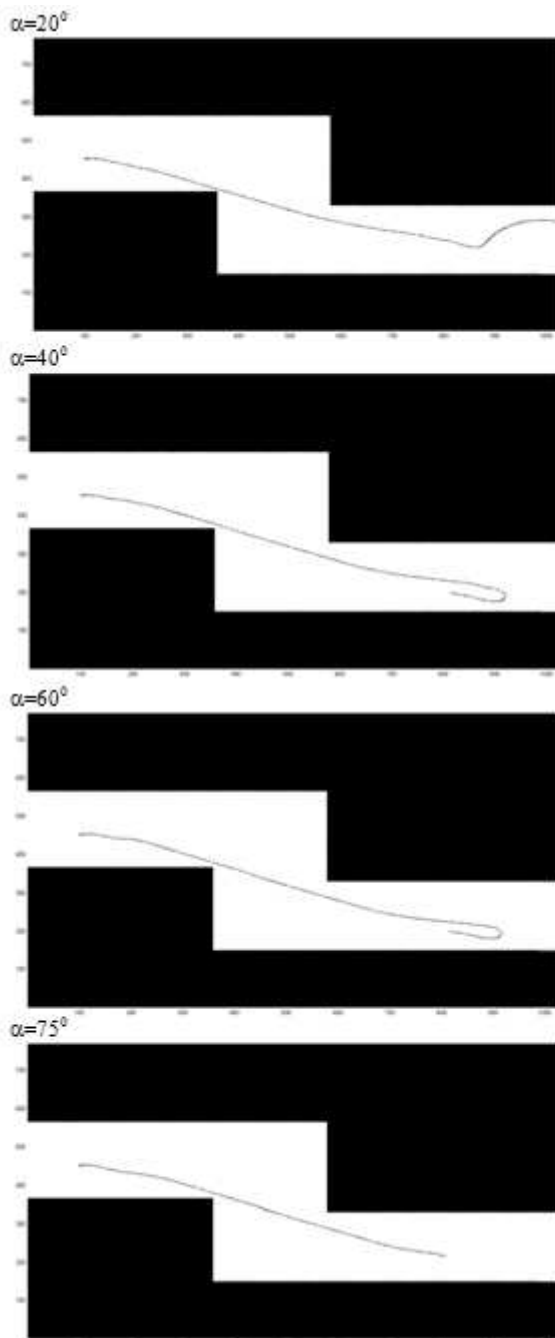
między trajektorią lotu a granicą bezpiecznego obszaru lotu. Aby trajektoria lotu była niezależna od wysokości lotu można wprowadzić relację pomiędzy kątem α a wysokością lotu. Z przedstawionych rysunków wynika, że ta relacja jest odwrotnie proporcjonalna (Rathinam i inni, 2006):

$$\alpha = f\left(\frac{1}{z}\right) \quad (10)$$

gdzie: α – kąt z zależności (4), z – wysokość lotu mikrosamolotu.

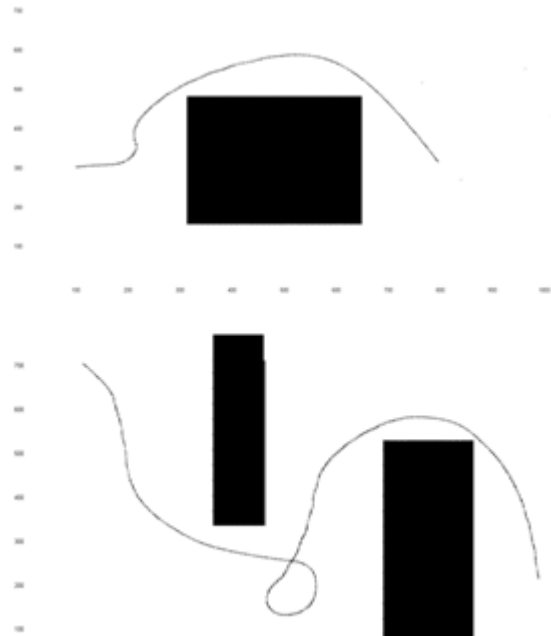


Rys. 25. Trajektorie lotu mikrosamolotu w kanionie ulicy na różnych wysokościach nad powierzchnią terenu. Początkowy kurs 300, punkt startu SP – $x=100$, $y=450$, kolejny punkt przelotowy WP – $x=800$, $y=200$



Rys. 26. Trajektorie lotu mikrosamolotu w kanionie ulicy z różnymi wartościami kąta α . Początkowy kurs 300, punkt startu SP – $x=100, y=450$, kolejny punkt przelotowy WP – $x=800, y=200$

Przeprowadzono również symulację lotu w zadaniu omijania przeszkody z większą wartością bezpiecznego dystansu $D_{safe}=100m$ oraz większą wartością kąta $\alpha=65^{\circ}$. Wyniki przedstawiają już znacznie lepsze sterowanie lotem, gdyż przy identycznych pozostałych parametrach jak na rysunkach 23 i 24 udało się uzyskać bezkolizyjną trajektorię lotu. Niestety jedna z trajektorii jest wciąż położona zbyt blisko przeszkody, by mogła być uznana za trajektorię bezpieczną. Wniosek jest następujący, że skuteczność działania algorytmu zależy od przyjętych wartości α i D_{safe} , które należy wyznaczyć dla konkretnego modelu dynamiki mikrosamolotu.



Rys. 27. Trajektorie lotu mikrosamolotu w zadaniu omijania przeszkód z $\alpha=65$ i $D_{safe}=100m$

5. PODSUMOWANIE

Metody autonomicznego sterowania lotem mikrosamolotów z wykorzystaniem systemów wizyjnych należą do rozwiązań, które będą w przyszłości szeroko rozwijane i stosowane. Obraz otoczenia mikrosamolotu posiada dużą ilość informacji, która może być wykorzystana w pełni autonomicznym systemie kontroli lotu. Zidentyfikowanie, odczytanie i przetworzenie tych informacji wymaga jednak opracowania skomplikowanych algorytmów, których wymagania sprzętowe znacznie przekraczają możliwości mobilnych urządzeń, jakie mogłyby być zainstalowane na pokładzie mikrosamolotu. Stąd też opracowanie algorytmu sterującego autonomicznym lotem w kanionach ulic powinno uwzględniać możliwości obliczeniowe mikroprocesorów, które znajdują się na pokładzie mikrosamolotu.

Opracowana koncepcja wykorzystująca informację o położeniu linii prostych na rejestrowanym obrazie sprawdziła się w badaniach symulacyjnych. Uzyskane trajektorie symulujące lot w kanionach ulic mieszczą się w granicach bezpiecznego obszaru lotu. Również po odpowiednim doborze wartości parametrów α i D_{safe} algorytm był w stanie bezkolizyjnie pokierować lotem w manewrze ominięcia przeszkody leżącej na trasie lotu.

Jednakże wyniki uzyskane podczas badań symulacyjnych potwierdzają zależność trajektorii lotu od szeregu parametrów takich jak wysokość lotu, wartość bezpiecznego dystansu D_{safe} oraz kąta α . Badania nad doбором wartości wymienionych parametrów powinny być przeprowadzone jeszcze przed implementacją algorytmu na podstawie znajomości dynamiki rzeczywistego obiektu. Osobnym zagadnieniem jest również dobór nastaw regulatorów PID, który również należy wykonać przy dobrej znajomości dynamiki sterowanego obiektu.

Podsumowując należy stwierdzić, że opracowany algorytm autonomicznego sterowania lotem w kanionach ulic

z wykorzystaniem systemu wizyjnego ma szansę być zaimplementowany w rzeczywistym układzie sterowania mikrosamolotem. Detekcja krawędzi, transformata Hougha oraz pozostałe niezbędne obliczenia nie są zbyt skomplikowane i mogą być zaprogramowane w energooszczędnych mikroprocesorach. Natomiast wszystkie pozostałe procedury sterowania, kontroli parametrów lotu oraz nawigacji będą wykorzystywały pokładowy autopilot, który jest urządzeniem zaprojektowanym do montażu w platformach latających klasy mikro.

LITERATURA

1. **Anderson E.** (2002), *Extremal control and unmanned air vehicle trajectory generation*, Master's thesis, Brigham Young University.
2. **Dong T., Liao X. H., Zhang R., Sun Z., Song Y. D.** (2005), Path Tracking and Obstacle Avoidance of UAVs - Fuzzy Logic Approach, *FUZZ '05, The 14th IEEE International Conference on Fuzzy Systems*, Reno, 43-48.
3. **Frew E., Langelan J.** (2005), Receding Time Horizon Control for Passive, Non-cooperative UAV See-and-Avoid, *IEEE International Conference on Robotics and Automation*, Barcelona, Spain.
4. **Frew E., Sengupta R.** (2004), Obstacle Avoidance with Sensor Uncertainty for Small Unmanned Aircraft, *43rd IEEE Conference on Decision and Control*, Paradise Island, Bahamas.
5. **Frew E., Spry S., Howell A., Hedrick J. K., Sengupta R.** (2004), Flight Demonstrations of Self-Directed Collaborative Navigation of Small Unmanned Aircraft, *AIAA 3rd Unmanned Unlimited Technical Conference, Workshop, & Exhibit*, Chicago, IL.
6. **Frew E.** (2004a), Vision-Based Road Following Using a Small Autonomous Aircraft, *IEEE Aerospace Conference*, Big Sky, MT.
7. **Frew W.** (2004b), Stereo-Vision-Based Control of a Small Autonomous Aircraft Following a Road, *Second Annual Swarming Conference*, Crystal City, MD.
8. **He Z.** (2006), Venkataraman Iyer R., Chandler P.R.: Vision-based UAV flight control and obstacle avoidance, *IEEE Automatic Control Conference*.
9. **Kownacki C.** (2009), Guidance and obstacle avoidance of MAV in uncertain urban environment, *European Micro Aerial Vehicle Conference and Flight Competition 2009: EMAV'2009*, Delft Holandia.
10. **Rathinam S., Kim Z., Sengupta R.** (2006), *Vision-Based Following of Structures Using an Unmanned Aerial Vehicle (UAV)*, RESEARCH REPORT UCB-ITS-RR-2006-1, Institute of Transportation Studies University of California at Berkeley.
11. **Saunders J. B., Call B., Curtis A., Beard R. W., McLain T. W.** (2005), Static and Dynamic Obstacle Avoidance in Miniature Air Vehicles, *Infotech@Aerospace, AIAA*, Arlington, Virginia.
12. **Zufferey J.-C., Beyeler A., Floreano D.** (2009), *Optic Flow to Steer and Avoid Collisions in 3D in Flying Insects and Robots*, Berlin, Springer.
13. **Zufferey J.-C., Klaptocz A., Beyeler A., Nicoud J.-D., Floreano, D.** (2007), A 10-gram Vision-based Flying Robot, *Advanced Robotics*, 21(14), 1671-1684.

CONTROL ALGORITHM OF MICRO AERIAL VEHICLE FLIGHT IN STREETS' CANYONS BASED ON VISION SYSTEM

Abstract: The paper presents a control algorithm of autonomous flight of micro aerial vehicle in streets' canyon using camera as a source of information about surrounding environment. The algorithm structure and the video processing routines were designed in the way which enables possibility of the algorithm realization using available devices such as micro cameras, advanced autopilots and DSP processors. Basing on the designed algorithm model a simulation experiment was conducted. The results confirm the effectiveness of the proposed control algorithm of micro aerial vehicle autonomous flight in streets' canyons. The simulation didn't use the real video signal, but only simplified model of perspective view on street map was employed. That is why the research should be continued including all steps of video processing routine.

Pracę wykonano w ramach realizacji projektu rozwojowego nr 0059/R/T00/2008/06 finansowanego ze środków Ministerstwa Nauki i Szkolnictwa Wyższego.