

Stanowisko laboratoryjne do badań algorytmów sterowania robotami mobilnymi z wizyjnym sprzężeniem zwrotnym

Daniel Figurowski, Michał Brasel, Michał Kubicki

Zachodniopomorski Uniwersytet Technologiczny w Szczecinie, Wydział Elektryczny, ul. Sikorskiego 37, 70-313 Szczecin

Streszczenie: W artykule poruszono tematykę związaną z opracowaniem platformy badawczo-dydaktycznej przeznaczonej dla robotów mobilnych. Omawiane stanowisko zostało zbudowane na Wydziale Elektrycznym Zachodniopomorskiego Uniwersytetu Technologicznego w Szczecinie jako platforma przeznaczona do projektowania i testowania algorytmów sterowania z wizyjnym sprzężeniem zwrotnym. W artykule poruszono przyczyny budowy takiej platformy wraz z odniesieniem się do istniejących rozwiązań, które powstały w innych ośrodkach badawczych. W dalszej części opisano wymagania postawione opracowanej platformie, a także omówiono konstrukcję prototypu platformy, zastosowane roboty mobilne oraz komponenty sprzętowe systemu. Następnie poruszono zagadnienie wizji maszynowej oraz przedstawiono opracowany algorytm analizy obrazu 2D. Artykuł kończy się podsumowaniem, w którym zamieszczono perspektywę rozwoju stanowiska oraz zarysowano plan wykorzystania opracowanej platformy do badań naukowych, jak i celów dydaktycznych.

Słowa kluczowe: roboty mobilne, pojazdy kołowe, system wizyjny, analiza obrazów 2D, stanowisko laboratoryjne

1. Wprowadzenie

W ostatnich latach obszar wiedzy związany z robotami mobilnymi jest jednym z najprężniej rozwijających się w dziedzinie robotyki. Różnorodność zastosowań robotów mobilnych zarówno w przemyśle (m.in. magazynowanie, transport, diagnostyka), jak i w życiu codziennym (sprzątanie, rozrywka) powoduje wzrost znaczenia badań nad metodami sterowania tymi obiektami oraz potrzebę poszukiwania nowych rozwiązań. To rozległe spektrum aplikacji robotów mobilnych przekłada się również na coraz większe ich rozpowszechnienie, a tym samym na zainteresowanie tą tematyką ze strony społeczeństwa. Producenci robotów konkurując ze sobą, nie tylko starają się poszerzyć zakres zadań realizowanych przez ich produkty, ale również dążą do maksymalnego uproszczenia ich obsługi od strony użytkownika. Ten ostatni aspekt powoduje, że roboty takie sprzedawane są często jako systemy zamknięte (z punktu widzenia modyfikacji algorytmów sterowania) z dostępnym i łatwym w obsłudze interfejsem użytkownika (HMI). To utrudnia wprowadzanie własnych modyfikacji sprzętowo-programowych, które są niezbędne w prowadzeniu

badań naukowych. Projektowanie i testowanie własnych algorytmów sterowania automatycznego w odniesieniu do robotów mobilnych wymaga zatem bogatego zasobu wiedzy inżynierskiej i odpowiedniego stanowiska badawczego. Stanowisko, które spełniałoby odpowiednie wymagania elastyczności sprzętowo-programowej najczęściej trzeba zbudować od podstaw. Działania takie są podejmowane na całym świecie przez różne ośrodki, a stworzone platformy służą zarówno jako stanowiska badawcze, jak i narzędzie dydaktyczne [8, 9, 13, 14].

W artykule przedstawiono opis stanowiska laboratoryjnego przeznaczonego do badań algorytmów sterowania robotami mobilnymi z wizyjnym sprzężeniem zwrotnym, które powstało w ramach działalności koła naukowego SARIS (ang. *Smart Autonomous Robots and Intelligent Systems*) działającego przy Katedrze Automatyki Przemysłowej i Robotyki na Wydziale Elektrycznym Zachodniopomorskiego Uniwersytetu Technologicznego w Szczecinie. Główną motywacją dla opracowanej platformy była chęć stworzenia rozwiązania, które byłoby modułarne, łatwo rozszerzalne i umożliwiałoby realizację badań w zakresie robotyki mobilnej i dziedzin z nią powiązanych. Idea takiej otwartej platformy jest pokazana w realizacjach [13, 14]. Istnieją jednak pewne istotne różnice w porównaniu z opracowanym systemem. W publikacji [13] implementacja oparta jest na czujnikach optycznych oraz IR, a nacisk położony jest na autonomię robotów, które mogą komunikować się ze sobą. Informacja o otaczającym świecie w publikacji [14] uzyskiwana jest z wykorzystaniem odometrii, a warstwa obliczeniowa jest mocno związana z pojedynczym robotem i jego konstrukcją. W przypadku platformy prezentowanej w niniejszym artykule, dominującym problemem jest wykorzystanie wizyjnego sprzężenia zwrotnego jako jedynego źródła informacji o świecie i wykorzystaniu go do sterowania robotami w tym świecie funkcjonującymi. Stąd też

Autor korespondujący:

Michał Brasel, michal.brasel@zut.edu.pl

Artykuł recenzowany

nadesłany 1.09.2016 r., przyjęty do druku 29.09.2016 r.



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



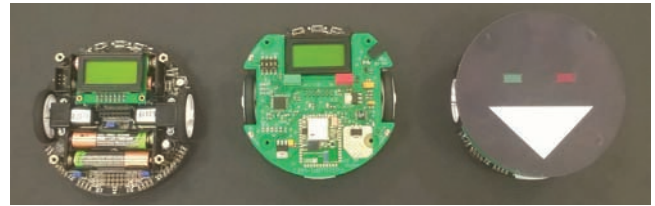
Rys. 1. Stół z umieszczonymi na nim robotami mobilnymi
Fig. 1. Test bench with mobile robots placed on it

bliżej jej do rozwiązań prezentowanych w [8] i [9], w stosunku do nich postawiono jednak na większą otwartość architektury opracowanego systemu. Całość sterowania realizowana jest po stronie klienta, w środowisku, które użytkownik uznaje za najbardziej dla siebie dogodną. Klienci mogą zarządzać pojedynczym robotem lub grupą robotów, przy czym z samej platformy może korzystać wielu klientów jednocześnie. Rola programu działającego cyklicznie na robocie mobilnym została ograniczona do niezbędnego minimum – realizuje zadania komunikacyjne oraz przesyła wypracowane sygnały na napędy kół. Nie występuje tu konieczność wgrywania programu użytkownika bezpośrednio na roboty. W porównaniu do [8] w opracowanym systemie nie zastosowano bezpośredniej komunikacji pomiędzy robotami, gdyż za ich koordynację odpowiada klient. W celu zminimalizowania opóźnień transportowych nacisk położono też na funkcjonowanie komponentów platformy w sieci lokalnej. Również pozostawiono użytkownikowi możliwość tworzenia własnych algorytmów przetwarzania obrazu 2D i wykorzystania ich jako źródła danych dla stworzonego systemu sterowania. W artykule szczególną uwagę zwrócono na opracowany algorytm analizy obrazu 2D, który podzielony został na dwie części: lokalizację oraz identyfikację robotów mobilnych. Stanowi on jeden z najważniejszych aspektów funkcjonowania platformy. Opisano też podejście zastosowane w budowie warstwy programowej oraz wykorzystane biblioteki do zrealizowanych celów. W podsumowaniu artykułu zarysowany został plan wykorzystania platformy do badań naukowych, jak i celów dydaktycznych. Przedstawiono również koncepcję rozwoju platformy oraz potencjalnych usprawnień w warstwie programowej oraz sprzętowej.

2. System wizyjny

Zadaniem systemu wizyjnego (zwanego często także wizją maszynową), jest udostępnienie maszynom możliwości postrzegania i analizowania otoczenia, podobnego do jednego z ludzkich zmysłów – wzroku. Należy jednak zauważyć, iż w pewnych aspektach systemy wizyjne są od niego doskonalsze – możliwość pracy w szerszym spektrum barw: noktowizja, termowizja, ultrafiolet. Systemy takie składają się z kamery pozyskującej informacje o aktualnym stanie obiektów na płaszczyźnie roboczej, oświetlacz oraz komputera analizującego zebrane dane przy użyciu odpowiednich metod analizy obrazów i systemów decyzyjnych.

Wizja maszynowa znalazła szerokie zastosowanie w przemyśle i nie tylko, gdzie realizuje m.in. takie zadania jak: liczenie, pomiar, identyfikacja, sortowanie, ocena jakości. Stosowana jest też do szerokiego wachlarza materiałów, produktów końcowych, kompleksowych systemów, a nawet może wspomagać zarządzanie zakładem przemysłowym [12]. Dynamiczny rozwój wizji maszynowej trwa począwszy od lat osiemdziesiątych XX wieku, na co składa się wiele czynników – spadek kosztów produkcji systemu, rozwój technologii LED, rozwój algorytmów sztucznej inteligencji, wzrost mocy obliczeniowej czy doskonalszych matryc. Metody



Rys. 2. Trzy roboty mobilne: z i bez WLAN oraz z trójkątem do identyfikacji

Fig. 2. Three mobile robots: with and without WLAN and one with identification triangle

przetwarzania obrazu również nie pozostały w tyle, znajdując implementację w wielu powszechnie dostępnych, dobrze udokumentowanych bibliotekach programistycznych. Nie oznacza to jednak, że rozwój tej dziedziny uległ spowolnieniu, wręcz przeciwnie – kamery o coraz większej rozdzielczości i ilości rejestrowanych klatek na sekundę wymagają nowych pokładów mocy obliczeniowej. Również sposób postrzegania otoczenia się zmienia – z próby zidentyfikowania w przestrzeni zdefiniowanych obiektów, na warstwę bardziej abstrakcyjną, a to z kolei wymaga zmiany w sposobie myślenia i stosowanych algorytmach.

3. Platforma naukowo-dydaktyczna

3.1. Założenia

Przed przystąpieniem do projektowania stanowiska laboratoryjnego zdefiniowano cele i założenia jakie musi spełniać projektowany system. Wyboru odpowiednich założeń dokonuje się najczęściej na drodze kompromisu między różnymi ograniczeniami: konstrukcyjnymi, wydajnościowymi i ekonomicznymi. Wybór ten w dużej mierze jest kwestią indywidualną. W przypadku omawianego stanowiska, przeznaczonego do badań algorytmów sterowania robotami mobilnymi, głównymi wymaganiami przyjętymi podczas jego projektowania były: możliwość detekcji pojazdów w ruchu oraz dostateczna odporność algorytmów detekcji i komunikacji. W toku realizacji projektu przyjęto również dodatkowe kryteria – dokładność pozycjonowania na poziomie ± 5 mm i położenia kąтового na poziomie $\pm 5^\circ$, obsługa do 10 robotów oraz niskie koszty konstrukcyjne. Komunikacja z robotami powinna odbywać się bezprzewodowo, z wykorzystaniem popularnego protokołu komunikacyjnego, tak aby możliwe było tworzenie nowego oprogramowania w innych środowiskach.

3.2. Stanowisko laboratoryjne

Głównym elementem konstrukcyjnym stanowiska jest stół roboczy o wymiarach $2\text{ m} \times 2\text{ m}$ pokryty antypoślizgową matą PCV o czarnej matowej powierzchni. Mata ta została przyklejona do podkładu z płyt MDF pełniącego funkcję blatu opartego na konstrukcji wsporczej. Szkielet podtrzymujący blat został wykonany z profili firmy Bosch Rexroth, przy czym cała strefa robocza została otoczona barierą zapobiegającą wypadnięciu robota poza krawędź stołu. Konstrukcję zbudowaną w ten sposób postawiono na regulowanych stopkach. Całość przedstawiono na rys. 1.

3.3. Warstwa sprzętowa

Objektami sterowania użytymi w początkowych badaniach testowych są dwukołowe roboty mobilne „3pi” firmy Pololu, zbudowane jako jednostka eksperymentalna z jedną centralną osią napędową ze zdolnością do wykonywania ruchu skrętnego w miejscu. Na spodniej krawędzi robota zostały umieszczone czujniki zbliżeniowe. Pojazd ten jest zasilany czterema akumulatorami typu AAA i może osiągnąć maksymalną prędkość 1 m/s. Szczegó-

lowe parametry zastosowanych robotów można znaleźć w dokumentacji producenta [1]. Oryginalna konstrukcja robota „3pi” nie ma jednak modułu komunikacji Wi-Fi, dlatego została rozbudowana o ten moduł według projektu zrealizowanego w pracy dyplomowej [2]. Moduł ten ma dodatkowo diody wykorzystywane w procesie wykrywania przez opracowany system wizyjny znajdujących się na stole pojazdów. Na rys. 2 przedstawiono trzy roboty mobilne, kolejno: bez modułu komunikacji, z modułem WLAN oraz ze znacznikiem w postaci białego trójkąta.

Główną rolę w zaprojektowanym systemie wizyjnym odgrywa kamera firmy Imaging Sources model DFK23GP031 z interfejsem komunikacyjnym Gigabit Ethernet, którym przesyłany jest obraz oraz ustawienia parametrów akwizycji danych. Przechwytuje ona obraz kolorowy w maksymalnej rozdzielczości wynoszącej $2592 \text{ px} \times 1944 \text{ px}$ przy 15 klatkach na sekundę, przy czym istnieje możliwość ustawienia mniejszej rozdzielczości, dzięki czemu można skorzystać z szybszej akwizycji obrazów. Wielkość piksela wynosi $2,2 \text{ mm} \times 2,2 \text{ }\mu\text{m}$, natomiast czas migawki ustawiany jest w zakresie od 50 μs do 30 s. Ponadto producent udostępnia pakiet SDK, który oferuje programiście szerokie możliwości w zakresie konfiguracji połączenia z kamerą oraz parametrów otrzymywanego obrazu. Więcej szczegółów można znaleźć w dokumentacji producenta [3, 4]. Na rys. 3 przedstawiono kamerę zamontowaną na regulowanym uchwycie przymocowanym do sufitu nad stołem. Uchwyt ten wykonany z profili firmy Bosch Rexroth zapewnia manualną regulację w sześciu stopniach swobody ruchu z ograniczeniem liniowym w zakresie 150 mm oraz kątowym 90° w każdym ramieniu.

Komputer klasy PC (rys. 4) pełni rolę serwera obliczeniowo-komunikacyjnego. Jego zadaniem jest zarówno analiza obrazu z kamery, jak i komunikacja z robotami. Składa się on z procesora Intel Core 2 Duo E8400 o częstotliwości pracy 3 GHz i 4 GB pamięci RAM. Kolejnym istotnym elementem jest zastosowanie odpowiedniej karty graficznej wyposażonej w technologię CUDA. Optymalną cenę pod względem oferowanych możliwości zapewniał akcelerator graficzny Nvidia GeForce GTX 750 Ti. Techno-



Rys. 3. Kamera DFK23GP031 wykorzystana do akwizycji obrazu
Fig. 3. DFK23GP031 camera used for image acquisition



Rys. 4. Zastosowany komputer i router WiFi
Fig. 4. Applied computer and WiFi router

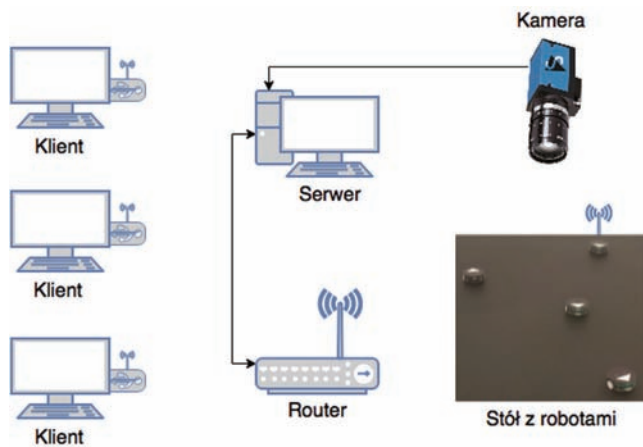
logia CUDA to równoległa architektura obliczeniowa, w której setki prostych procesorów realizuje obliczenia o charakterze równoległym [5]. Procesor ten ma 640 rdzeni CUDA zapewniających 1,4 TFLOPS. Dodatkowo do komunikacji z kamerą została zamontowana karta rozszerzeń z portem Gigabit Ethernet.

Łączność bezprzewodową w systemie zapewnia router LINKSYS WRT54GL (rys. 4). Jest to model zapewniający łączność w standardzie IEEE 802.11g, co przekłada się na teoretyczną prędkość transmisji danych na poziomie 54 Mb/s, przy czym moc nadawcza wynosi maksymalnie 18 dBm. Ponadto zapewnia on szereg możliwości w zakresie zabezpieczenia sieci oraz konfiguracji jej parametrów.

3.4. Zastosowany system wizyjny

W przypadku opracowanej platformy określenie położenia robotów (ich współrzędnych) jest wynikiem działania zaawansowanych algorytmów analizy obrazu. Dzięki zebranych w ten sposób i odpowiednio przetworzonym informacjom możliwe jest wykorzystanie systemów wizyjnych w roli sprzężenia zwrotnego w układach sterowania (tzw. *visual servoing*). Informacje o położeniu pojazdów przekazywane są przez podsystem komunikacyjny do użytkownika, który wykorzystuje je w algorytmie sterowania do wypracowania odpowiednich sygnałów sterujących (w tym przypadku są to sygnały prędkości poszczególnych kół pojazdów znajdujących się na stole). Wypracowane wartości sygnałów sterujących przesyłane są następnie do serwera, który po ich odebraniu, przetwarza je i wysyła z wykorzystaniem komunikacji bezprzewodowej bezpośrednio do robotów. Schemat całej platformy przedstawia rys. 5. Możliwe inne role i konfiguracje kamery w sterowaniu można znaleźć w [10], natomiast przegląd algorytmów w [12].

Dla prawidłowej detekcji pojazdów w ruchu, niezbędne było osiągnięcie dużego współczynnika liczby klatek na sekundę dla kamery przechwytyjącej obraz. Cel ten spełniono dzięki obniżeniu rozdzielczości sensora optycznego o połowę, co przełożyło się na zdolność do akwizycji 25 klatek na sekundę. Problem „rybiego oka”, który mógłby mieć niekorzystny wpływ na poprawne nanieśienie współrzędnych, niwelowany jest przez telecentryczny obiektyw kamery. Dodatkowo kamera umieszczona jest centralnie nad stołem roboczym i prostopadle do niego, a obraz jest przycinany i obracany. Pozwala to na uniknięcie konieczności kalibracji kamery, problemu przedstawionego w [11]. W efekcie rozdzielczość robocza wynosi $964 \text{ px} \times 964 \text{ px}$. Należy jednak pamiętać, że zmniejszenie rozdzielczości sensora wiąże się z obniżeniem dokładności pozycjonowania. W tym przypadku uzyskane war-



Rys. 5. Schemat opracowanej platformy
Fig. 5. Devised platform schema

tości dokładności pozycjonowania nadal mieszczą się przyjętym w punkcie 2.1 zakresie. Dodatkowo, dzięki takiemu ustawieniu uzyskano synchronizację migawki z częstotliwością sieci zasilającą lampy fluorescencyjne, co pozwoliło zniwelować efekt migotania.

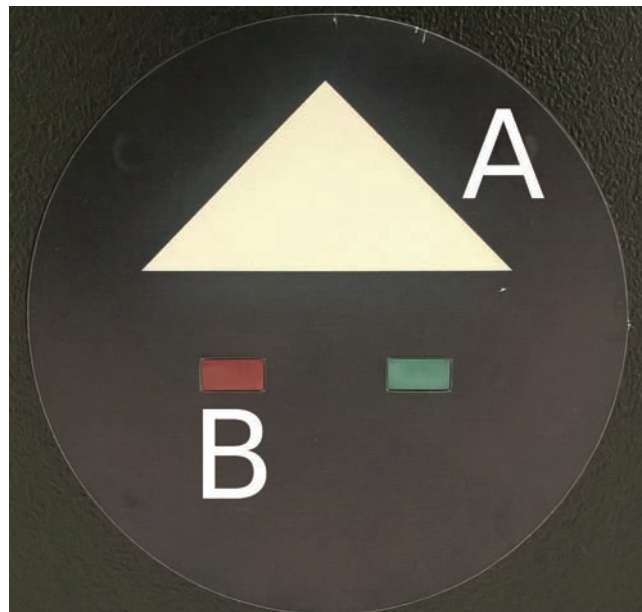
3.5. Algorytm analizy obrazu 2D

W opracowanej platformie system wizyjny jest wykorzystywany do dwóch zadań: określenia współrzędnych robota oraz do jego identyfikacji. Zadania te w zakresie przetwarzania obrazu są niezależne od siebie, ale samo nadanie identyfikatora robotowi może nastąpić dopiero po określeniu jego współrzędnych.

Określenie współrzędnych oraz orientacji robota bazuje na wykryciu kształtu białego trójkąta (rys. 6A) umieszczonego na górnej powierzchni robota. Serwer na bieżąco analizuje obraz i stara się wykryć wszystkie obiekty przypominające ten kształt, po czym oblicza współrzędne robotów bazując też na ich właściwościach fizycznych. W celu znalezienia środka wykorzystuje się stałą odległość trójkąta od centrum robota, a położenie katowe determinowane jest na podstawie położenia trójkąta względem układu współrzędnych związanych ze stołem. Kolejne kroki są zdefiniowane następująco:

- Konwersja obrazu na skalę szarości.
- Prognozowanie obrazu – w efekcie otrzymywana jest macierz binarna.
- Znalezienie wszystkich konturów.
- Sprawdzenie czy obszar znalezionej konturu mieści się w zdefiniowanych ograniczeniach (więcej niż 100 px, ale mniej niż 500 px).
- Wyznaczenie najmniejszego możliwego trójkąta zawierającego dany kontur.
- Znalezienie najdłuższego boku trójkąta (podstawy).
- Określenie rotacji trójkąta względem układu współrzędnych.

Identyfikacja robota polega na powiązaniu adresu sieciowego robota z fizycznym obiektem znajdującym się na stole. Serwer w sposób cykliczny stara się nawiązać komunikację z pewną pulą adresów sieciowych, które jeszcze nie zostały zidentyfikowane. Po udanym nawiązaniu połączenia taki niezidentyfikowany robot zapala swoją czerwoną diodę LED (rys. 6B). Zadaniem systemu wizyjnego jest wykrycie tego sygnału i powiązanie go z koordynatami robota (wcześniejsze zadanie) bezpośrednio w jego otoczeniu. Jednocześnie tak oznaczonemu obiektowi nadawany jest unikalny identyfikator, który ułatwia dalszą interakcję z nim za pośrednictwem serwera. Sam algorytm działa w sposób następujący:



Rys. 6. Obszary charakterystyczne robota wykorzystywane do:
A) znajdowania położenia, B) identyfikacji robota

Fig. 6. Characteristic zones of mobile robot used in: A) position finding, B) identification

- Konwersja obrazu z palety RGB na HSV i rozłożenie go na kanały reprezentujące poszczególne składowe Kanały te są reprezentowane przez macierze, których komórki odpowiadają za poszczególne piksele.
- Zastosowanie zestawu filtrów na poszczególnych kanałach pozwalających znaleźć obiekty reprezentujące świecąca diodę LED. Dzięki temu otrzymujemy trzy zbinaryzowane tablice.
- Wykonanie serii operacji logicznych na uzyskanych trzech macierzach binarnych, w efekcie czego otrzymujemy jedną macierz z informacją binarną – czy dana komórka ma charakterystykę odpowiednią dla świecącej się diody.
- Z macierzy uzyskanej w poprzednim punkcie dokonujemy wyszukania konturów. W celu wyeliminowania zakłóceń w postaci pojedynczych pikseli przyjęto, że obszary identyfikowane jako dioda LED powinny składać się co najmniej z pięciu pikseli.
- Wyznaczenie momentu centralnego znalezionej obszaru, który uznawany jest za środek obszaru diody LED.

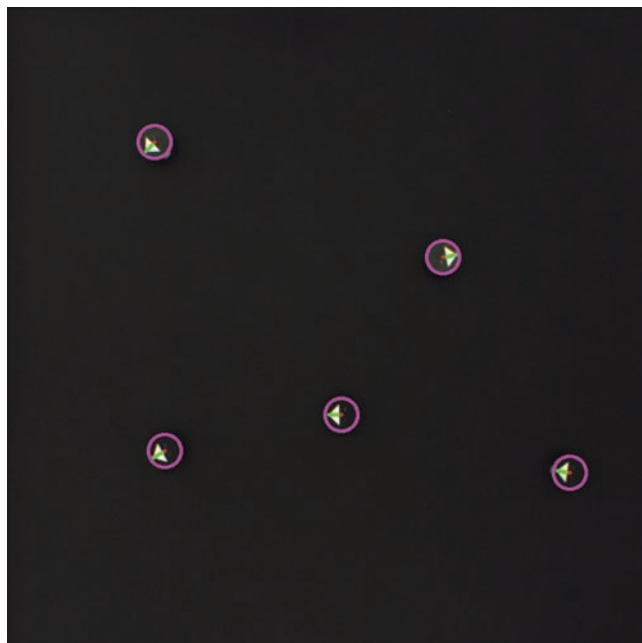
Tak jak wspomniano wcześniej, po zidentyfikowaniu robota należy go powiązać z koordynatami, w których się znajduje. Dokonywane jest to przez znalezienie najmniejszego dystansu między jego środkiem, a środkiem obszaru diody LED. Na rys. 7 można zobaczyć przetworzony obraz z naniesionymi markerami graficznymi dla rozpoznanych robotów.

3.6. Warstwa programowa

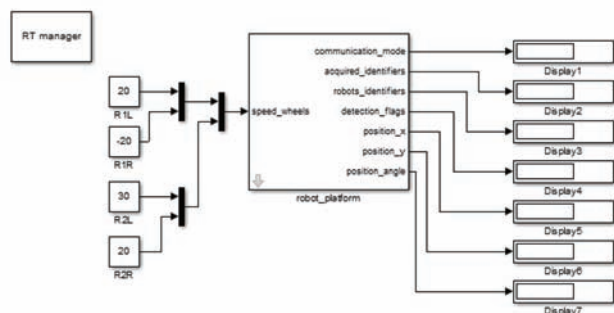
Na część programową systemu składają się trzy aplikacje. Każda z nich pełni inną rolę w środowisku opracowanej platformy:

- Algorytm analizy obrazów (napisany został w języku C++, korzysta z biblioteki OpenCV). Jego zadaniem jest identyfikacja i lokalizacja robotów.
- Komunikacja z robotami mobilnymi (zrealizowana w języku Python).
- Zarządzanie klientami i przydzielanie robotów poszczególnym klientom w C#.

Jak widać, każda z tych aplikacji została napisana w innym języku programowania. Spowodowane to było m.in. tym, iż nad kodem źródłowym pracowało kilku autorów, z których każdy miał własny bagaż doświadczeń związanych z określonym języ-



Rys. 7. Przenalizowany obraz 2D z rozpoznanymi robotami
Fig. 7. Analyzed 2D image with recognized robots



Rys. 8. Schemat w środowisku obliczeniowym MATLAB/Simulink
Fig. 8. MATLAB/Simulink model

kiem, dostępnością bibliotek czy kwestiami wydajnościowymi. Do interakcji między stworzonymi aplikacjami wykorzystywana jest biblioteka ZMQ, natomiast komunikacja z robotami oraz ze środowiskiem MATLAB/Simulink odbywa się po protokole TCP/IP.

Szczególnie dużą rolę odgrywa biblioteka OpenCV. Zawiera szeroki wachlarz funkcji przydatnych w realizowaniu zadań związanych z analizą obrazu, które zostały szeroko zastosowane w opisywanym wcześniej algorytmie. Jej popularność oraz dobra dokumentacja przekładają się na zestaw przetestowanych przez innych programistów funkcji, które są łatwe do implementacji.

Do realizacji celów badawczych oraz dydaktycznych opracowana została biblioteka w środowisku MATLAB/Simulink (rys. 8), która pozwala korzystać z opracowanego stanowiska. Za jej pośrednictwem można sterować pojedynczym robotem, bądź grupą robotów realizując zaawansowane algorytmy sterowania. Wykorzystany jest blok Matlab System, który pozwala na korzystanie z funkcji Matlab w środowisku MATLAB/Simulink. Dodatkowo zastosowany został model oparty na programowaniu zorientowanym obiektowo.

Cały system zbudowany jest z komponentów realizujących pojedyncze zadania, a informacje z nich pochodzące dostępne są dla większej liczby klientów. Rozwój warstwy programowej może zatem następować dwójako: zarówno przez doskonalenie istniejących już usług, jak i przez tworzenie nowych narzędzi, korzystających z udostępnionych już informacji.

5. Podsumowanie

Sterowanie robotami mobilnymi w układzie ze sprzężeniem wizyjnym jest problemem interdyscyplinarnym wymagającym biegłości w zagadnieniach związanych z analizą obrazu 2D, komunikacją oraz syntezą odpowiednich algorytmów sterowania przewidzianych dla różnych zadań. Warunkiem koniecznym prowadzenia badań dotyczących sterowania robotami mobilnymi jest dostęp do odpowiedniego stanowiska badawczego. Opisana platforma badawczo-dydaktyczna dedykowana pojazdom kołowym umożliwia realizację różnych zadań sterowania, m.in.: dynamiczne pozycjonowanie robotów, sterowanie robotami po zadanej trajektorii ruchu, a także sterowanie formacją robotów. W kręgu zainteresowań badawczych autorów artykułu znajdują się zagadnienia związane m.in. ze sterowaniem odpornym, identyfikacją modeli obiektów sterowania, algorytmami antykolizyjnymi, dynamiczną optymalizacją trajektorii ruchu obiektów, a także optymalizacji zużycia energii przez autonomiczne pojazdy. Na opracowanej platformie przeprowadzane będą również zajęcia dydaktyczne ze studentami pozwalające im zapoznać się z aspektami sterowania robotami mobilnymi. Realizowane zajęcia poruszają problematykę z zakresu kinematyki i dynamiki robotów mobilnych oraz generacji trajektorii.

Planowany jest dalszy rozwój opracowanego systemu. Jedną z proponowanych zmian jest wykorzystanie kamery o większej rozdzielczości i z szybszą akwizycją obrazu. Przy dostatecznej mocy obliczeniowej, zwiększona częstotliwość próbkowania (prędkość akwizycji) wraz ze zwiększeniem dokładności mierzonej pozycji (rozdzielczość) powinna pozwolić na sterowanie, które będzie bardziej precyzyjne. System budowany jest modułowo, wobec czego wymiana pojedynczego komponentu (np. robotów) jest prosta. W tym zakresie planuje się zarówno modyfikację istniejących robotów przez wprowadzenie silnika BLDC z enkoderami, jak i wykorzystanie nowych rodzajów jednostek o innej geometrii. Na stole planuje się umieścić stację dokującą służącą do ładowania akumulatorów w sposób stykowy i bezprzewodowy. Kolejnym udoskonaleniem systemu analizy obrazu będzie sposób detekcji przeszkód, zarówno przez fizyczne naklejenie markerów na stole, jak i przez programowe symulowanie labiryntu. Usprawnienia są również planowane w warstwie programowej – większa integracja z platformą .NET oraz unifikacja rozwiązań w postaci wspólnych bibliotek, mają na celu dalsze ułatwienie tworzenia nowego oprogramowania dla opracowanej platformy. Dla zwiększenia wydajności obliczeniowej wykorzystana zostanie technologia CUDA, która pozwoli na zminimalizowanie czasu wykonania algorytmu.

Bibliografia

1. Pololu Corporation, *Pololu 3pi Robot User's Guide*, 2013.
2. Spychała M., *Integracja modułu WLAN ze sterownikiem robota mobilnego*, Praca dyplomowa, Zachodniopomorski Uniwersytet Technologiczny w Szczecinie, 2015.
3. The Imaging Sources, *DFK23GP031 Color Camera*, 2013.
4. The Imaging Sources, *Camera SDK manual*, http://www.theimagingsource.com/en_US/support/documentation/icimagingcontrol-dotnet/, dostęp w 2016.
5. Nvidia, GeForce GTX 750 Ti, <http://www.geforce.com/hardware/desktop-gpus/geforce-gtx-750-ti>, dostęp w 2016.
6. Nvidia, CUDA Zone, <https://developer.nvidia.com/cuda-zone>, dostęp w 2016.
7. Figurowski D., *System wizyjny dla dynamicznego pozycjonowania robotów mobilnych*, Praca dyplomowa, Zachodniopomorski Uniwersytet Technologiczny w Szczecinie, 2015.
8. Fabregas E., Farias G., Dormido-Canto S., Guinaldo M., Sánchez J., Dormido Bencomo S., *Platform for Teaching Mobiles Robotics*. "Journal of Intelligent & Robotic Systems", Vol. 81, Issue 1, 2016, 131–143.

9. Witkowski U., Emad Monier, Rückert U., *An Automated Platform for Minirobots Experiments*, 10th Intl. Conf. on Control, Automation, Robotics and Vision, 2008.
10. Tepe Tufan, *Mobile Robot Navigation Using Visual Servoing*, MSc Internship TU/e, www.mate.tue.nl/mate/pdfs/11494.pdf, dostęp w 2016.
11. Gregor Klančar, Marko Lepetič, Matej Kristan, Rihard Karba, *Vision system design for mobile robot tracking*, Mobile robots: new research. New York: Nova Science, 2005.
12. Batchelor G. B. (editor), *Machine Vision Handbook*, Rozdziały: I, XIV–XVII, Springer, 2012.
13. Rostkowska M., Topolski M., Skrzypczyński P., *A modular mobile robot for multi-robot applications*, „Pomiary Automatyka Robotyka”, Vol. 17, No. 2/2013, 288–293.
14. Piątek P., Zieliński M., Skrzypczyński P., *Modułowy robot mobilny do celów dydaktycznych i badawczych*, „Pomiary Automatyka Robotyka”, R. 15, Nr 2/2011, 411–419.

Laboratory Station for Studying Visual Feedback Control Systems for Mobile Robots

Abstract: The article presents the development of an academic platform dedicated to mobile robots. This stand was built in West Pomeranian University of Technology in Szczecin as a platform on which control systems with visual feedback could be devised and tested. In the article a motivation for such station was presented with acknowledging already existing solutions developed by other research groups. Further, platform requirements were presented as well as its construction, mobile robots and hardware components. A general outline of the machine vision problem is given, as well as a detailed description of a 2D image analysis algorithm devised for the designed platform. The article ends with a summary in which a concept of using the platform for research and didactics is given as well as its further development.

Keywords: mobile robots, wheeled vehicles, machine vision, 2D image analysis, laboratory station

mgr inż. Daniel Figurowski

daniel-figurowski@zut.edu.pl

Doktorant w Katedrze Automatyki Przemysłowej i Robotyki na Wydziale Elektrycznym Zachodniopomorskiego Uniwersytetu Technologicznego w Szczecinie. Zainteresowania naukowe obejmują robotykę mobilną, analizę obrazów oraz projektowanie układów elektronicznych.



mgr inż. Michał Brasel

michal.brasel@zut.edu.pl

Asystent w Katedrze Automatyki Przemysłowej i Robotyki na Wydziale Elektrycznym Zachodniopomorskiego Uniwersytetu Technologicznego w Szczecinie. Główne zainteresowania naukowe to: analiza i synteza wielowymiarowych układów sterowania, synteza sterowania nieliniowego i optymalnego, sterowanie ruchem obiektów pływających.



mgr inż. Michał Kubicki

michal.kubicki@zut.edu.pl

Asystent w Katedrze Automatyki Przemysłowej i Robotyki na Wydziale Elektrycznym Zachodniopomorskiego Uniwersytetu Technologicznego w Szczecinie. Zainteresowania naukowe obejmują metody sztucznej inteligencji, optymalizację, modelowanie oraz teorię gier.

