

Wykorzystanie systemu wizyjnego w sterowaniu robotem przemysłowym za pomocą gestów

Use of the vision system to control industrial robot with gestures

WOJCIECH KACZMAREK
MICHAŁ TOMASZUK *

Materiały z XX SKWPIE, Jurata 2016 r.
DOI: 10.17814/mechanik.2016.7.139

Artykuł przedstawia możliwość wykorzystania systemu wizyjnego w sterowaniu robotem przemysłowym za pomocą gestów. W artykule przedstawiono elementy tworzenia oprogramowania do sterowania robotem w trybie offline i online. Do sterowania robotem przemysłowym wykorzystano urządzenie Kinect.

SŁOWA KLUCZOWE: robot przemysłowy, sterowanie, system wizyjny

The article presents the possibility of using a vision system to control industrial robot with gestures. The article presents the components of software development for robot control in the offline and online mode. To control the industrial robot Kinect device was used.

KEYWORDS: industrial robot, control, vision system

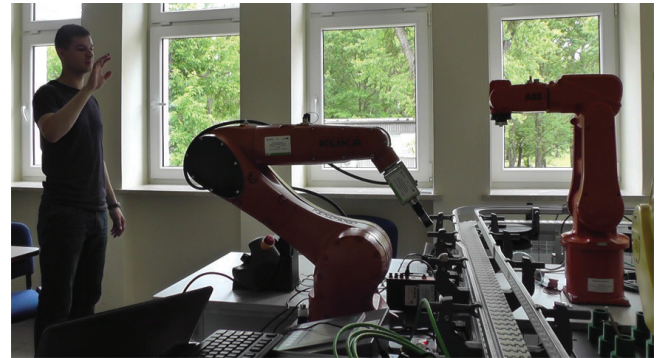
W ostatnim czasie sterowanie urządzeniami za pomocą gestów znalazło szerokie zastosowanie w różnych obszarach nauki [1]. Śledząc zmiany w sposobie interakcji człowieka z urządzeniami elektronicznymi, można zauważyć dynamiczny rozwój w tej dziedzinie. Przykładem zastosowania obsługi urządzeń za pomocą gestów mogą być nowoczesne telewizory, gry komputerowe, komputery, systemy automatyki w budynkach, roboty mobilne itp. Rozwój elektroniki oraz systemów wizyjnych przyczynił się do powstania nowej drogi interakcji pomiędzy człowiekiem a urządzeniem, co stwarza zupełnie nowe możliwości w zakresie projektowania oraz nowych zastosowań aplikacji komputerowych. Integracja systemów wizyjnych z urządzeniami elektronicznymi zmienia dotychczasowe podejście do obsługi urządzeń elektronicznych.

Koncepcja stanowiska

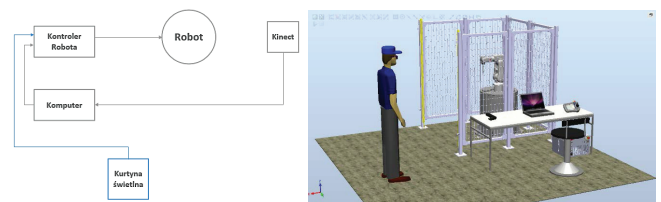
Zrobotyzowane stanowisko do sterowania za pomocą gestów składa się z robota przemysłowego, systemu wizyjnego, kontrolera robota, komputera oraz – dla bezpieczeństwa stanowiska – kurtyny świetlnej (rys. 1, 2).

W omawianym przypadku jako system wizyjny wykorzystano urządzenie Microsoft Kinect. Kontroler ruchu Kinect zawdzięcza swoje właściwości integracji odpowiednich komponentów (rys. 3). Urządzenie zostało wyposażone w kamerę RGB, emiter światła podczerwonego, kamerę głębi oraz zestaw mikrofonów, umożliwiający interakcję za pomocą mowy. Zastosowanie emitera podczerwieni oraz kamery głębokości wyposażonej w filtr światła podczerwonego pozwala na otrzymanie reprezentacji 3D przetwarzanej sceny. Emiter w sposób ciągły wysyła pseudolosowe wzory światła podczerwonego, które, odbijając się od powierzchni, zniekształca się, a następnie trafia do kamery głębi i tworzy obraz 3D. Zastosowanie światła podczerwonego ogranicza wpływ zmieniającej się iluminacji sceny.

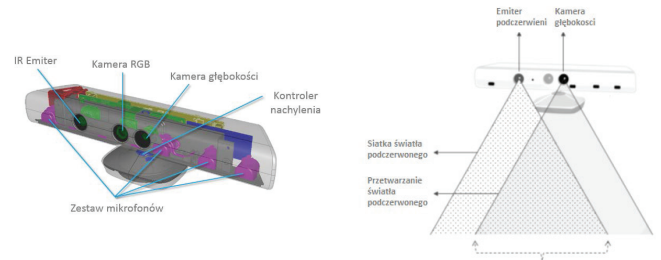
Kluczowym rozwiązaniem dla tego projektu jest detekcja punktów ludzkiego ciała za pomocą urządzenia Microsoft Kinect. Wykorzystując odpowiedni język programowania, kontroler ruchu udostępnia programiście odczytanie współrzęd-



Rys. 1. Sterowanie robotem przemysłowym w trybie śledzenia dłoni



Rys. 2. Schemat stanowiska oraz wizualizacja stanowiska wirtualnego



Rys. 3. Budowa urządzenia Kinect oraz schemat przetwarzania sceny 3D

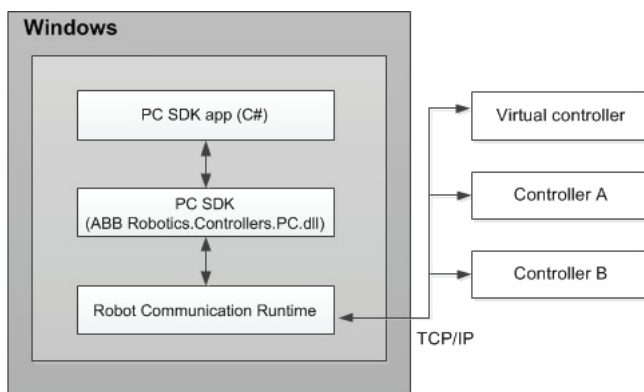
nych wybranych punktów ludzkiego ciała w przestrzeni 3D uchwyconej sceny. Idąc w tym kierunku, została opracowana pełna logika przetwarzania obrazu w celu uzyskania sterowania robotem przemysłowym za pomocą gestów.

Projekt aplikacji

Zintegrowanie takich urządzeń jak robot przemysłowy oraz urządzenie Microsoft Kinect wymaga napisania odpowiedniej aplikacji umożliwiającej komunikację pomiędzy urządzeniami wchodzącymi w skład stanowiska.

Powyższy schemat przedstawia uproszczoną strukturę aplikacji oraz przepływy danych pomiędzy poszczególnymi modułami. Opracowano aplikację sterującą w języku C#, która jest odpowiedzialna za przetwarzanie obrazu, komunikację pomiędzy kontrolerem Kinect, komputerem a kontrolerem robota [2÷5]. Aplikacja została wyposażona również w interfejs graficzny użytkownika, umożliwiający swobodną obsługę bez konieczności ingerencji w kod programu. Z drugiej strony zostały napisane programy sterujące dla robota przemysłowego IRB 120 firmy ABB w języku RAPID [8÷10].

* Ppłk dr inż. Wojciech Kaczmarek (wojciech.kaczmarek@wat.edu.pl) – Wojskowa Akademia Techniczna; mgr inż. Michał Tomaszuk (m.tomaszusk@hotmail.com) – Develley Polska Sp. z o.o.



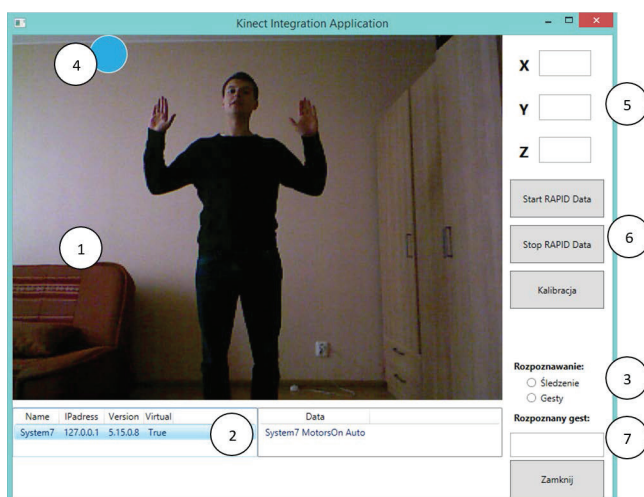
Rys. 5. Ogólny schemat komunikacji aplikacji z kontrolerem robota

Opracowana aplikacja komunikuje się z robotem poprzez sieć Ethernet, korzystając z protokołu transmisji danych TCP/IP (rys. 5). Dołączenie bibliotek ABB w aplikacji C# umożliwia swobodną komunikację z kontrolerem robota. Zgodnie z powyższym schematem aplikacja, wykorzystując biblioteki ABB, przesyła dane do modułu Robot Communication Run Time, który jest odpowiedzialny za synchronizację kodu oraz przesyłanie danych z języka programowania C# do języka Rapid w obu kierunkach [11, 12].

Wykonanie aplikacji

Oprócz pełnej funkcjonalności aplikacji umożliwiającej sterowanie robotem przemysłowym za pomocą gestów opracowano intuicyjny interfejs graficzny, który pozwala na łatwe uruchomienie stanowiska i sterowanie nim.

W celu napisania aplikacji wykorzystano środowisko Microsoft Visual Studio 2012 oraz język programowania C#. Aplikacja umożliwia użytkownikowi sterowanie robotem przemysłowym poprzez ruch dłoni w przestrzeni 3D i nadawanie trajektorii robota oraz za pomocą gestów (rys. 6). Zdefiniowano gesty umożliwiające uruchomienie oraz zatrzymanie stanowiska, a także gesty nadające kierunek ruchu robota w prawo, lewo, górę i dół.



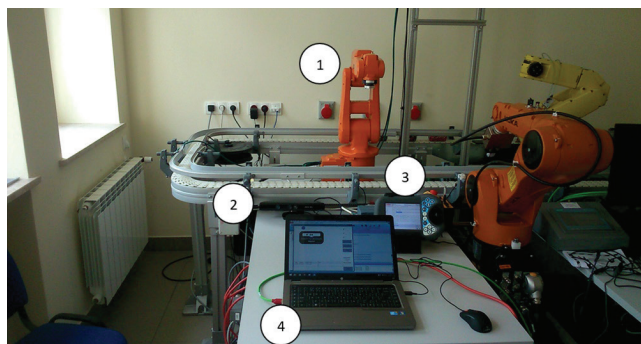
Rys. 6. Interfejs graficzny opracowanej aplikacji. Interfejs składa się z następujących komponentów: 1 – transmisja obrazu kamery RGB urządzenia Kinect, 2 – okno wyświetlające podstawowe informacje o podłączonych kontrolerach, natomiast okno obok wyświetla informację o stanie pracy kontrolera oraz o załączeniu silników robota, 3 – sekcja przeznaczona do wyboru trybu sterowania robotem przemysłowym, 4 – grafika, która podąża za ruchem prawej dłoni w przypadku wyboru trybu śledzenia, 5 – pola tekstowe wyświetlające współrzędne śledzonej dłoni, 6 – przyciski do komunikacji z kontrolerem robota, 7 – pole tekstowe wyświetlające rozpoznany gest

Testy aplikacji

Testy uruchomieniowe aplikacji zostały przeprowadzone na wirtualnym kontrolerze w środowisku RobotStudio oraz na fizycznym robocie przemysłowym ABB IRB 120 w Laboratorium Robotyki w Wojskowej Akademii Technicznej (rys. 7).

Testy wykazały, że sterowanie posiada niewielkie opóźnienie w stosunku do wykonanego gestu lub ruchu dłonią.

Testy fizyczne stanowiska dały rezultat zgodny z testami na wirtualnym kontrolerze robota. Stanowisko może być sterowane za pomocą śledzenia dłoni lub gestów bez większych problemów. Testy fizyczne potwierdziły, że opracowana aplikacja działa poprawnie, nie zgłaszając przy tym żadnych błędów przetwarzania kodu aplikacji.



Rys. 7. Stanowisko do sterowania robotem przemysłowym za pomocą gestów, 1 – robot przemysłowy ABB IRB 120 [13], 2 – kontroler ruchu Kinect, 3 – panel operatorski FlexPendant, 4 – komputer

Podsumowanie

Interakcja człowieka z robotem przemysłowym przy wykorzystaniu systemu wizyjnego wydaje się być ciekawym oraz obiecującym rozwiązaniem w kierunku rozwoju robotyki. Nauczanie trajektorii ruchu robotów przemysłowych za pomocą gestów może okazać się w przyszłości jedną z metod programowania robotów. Ponadto ogromną zaletą okazało symulowanie ruchu robota w środowisku offline. Pokazuje to efektywność wykorzystywania tego typu narzędzi oraz przede wszystkim oszczędność czasu, a także kosztów, co jest znaczące we współczesnym przemyśle.

LITERATURA

1. Giorio C., Fascinari M. "Kinect in Motion - Audio and Visual Tracking by Example", Packt Publishing 2013.
2. Greene J., Stellman A. „C#. Rusz głową!”. Wydanie drugie, Wydawnictwo Helion 2011.
3. Hejlsberg A., Torgersen M., Wiltamuth S., Golde P. „Język C# Programowanie”. Wydanie III, Wydawnictwo Helion 2010.
4. Jana A. "Kinect for Windows SDK Programming Guide". Packt Publishing 2012.
5. Kruk S. „Sterowanie Windows 8 gestami jak w raporcie mniejszości”. [online] [dostęp 25.06.2015] <http://www.chip.pl/news/oprogramowanie/systemy-operacyjne/2012/11/sterowanie-windows-8-gestami-jak-w-raporcie-mniejszosci>.
6. Premaratne P. "Human Computer Interaction Using Hand Gestures". Singapore 2014, Wydawnictwo Springer.
7. Zimmerman T., Lanier J., Blanchard Ch., Bryson S., Harvill Y. "A Hand gesture interface Device". Redwood City 1987.
8. Application manual PC SDK – Operating Manual Rev. A, Sweden 2012, ABB.
9. Introduction to RAPID – Operating Manual, Rev. A, Sweden 2012, ABB.
10. Kinect [online] [dostęp 25.06.2015] <https://en.wikipedia.org/wiki/Kinect>.
11. Pternesas V. "Implementing Kinect Gestures". [online] [dostęp 25.06.2015] <http://www.codeproject.com/Articles/716741/Implementing-Kinect-gestures>.
12. SDK Documentation (Kinect for Windows) [online] [dostęp 25.06.2015] <https://msdn.microsoft.com/en-us/library/hh855347.aspx>.
13. Technical data for the IRB 120 industrial robot [online] [dostęp 25.06.2015] <http://new.abb.com/products/robotics/industrial-robots/irb-120/irb-120-data>.