

Adam NOWOSIELSKI, Łukasz CHODYŁA
 ZACHODNIOPOMORSKI UNIWERSYTET TECHNOLOGICZNY W SZCZECINIE,
 ul. Żołnierska 49, 71-210 Szczecin

Bezdotykowy interfejs klawiatury ekranowej dla systemów GNU/Linux

Dr inż. Adam NOWOSIELSKI

Ukończył studia na Wydziale Informatyki Politechniki Szczecińskiej w roku 2002 roku. Pracę doktorską obronił w 2006 r. Jego zainteresowania naukowe obejmują: biometrię, inteligentne systemy monitoringu, przetwarzanie i rozpoznawanie obrazów. Jest adiunktem w Katedrze Systemów Multimedialnych na Wydziale Informatyki ZUT oraz starszym wykładowcą w Instytucie Technicznym PWSZ w Gorzowie Wielkopolskim.



e-mail: anowosielski@wi.zut.edu.pl

Inż. Łukasz CHODYŁA

Ukończył studia w Instytucie Technicznym Państwowej Wyższej Szkoły Zawodowej w Gorzowie Wielkopolskim w 2013 roku. Jego zainteresowania naukowe obejmują: języki i technologie programowania obiektowego, nowoczesne systemy komputerowe, systemy komunikacji człowiek-komputer. Aktywnie uczestniczy w rozwoju projektów open-source.



e-mail: chodak166@op.pl

Streszczenie

Artykuł porusza problem bezdotykowego interfejsu klawiatury ekranowej przeznaczonego przede wszystkim dla osób niepełnosprawnych oraz kwestie realizacji projektu na odpowiedniej platformie sprzętowo-programowej. Proponowane rozwiązanie wpisuje się w obecną tendencję eliminowania fizycznych kontrolerów i zastępowania ich naturalnymi gestami użytkownika. W artykule omówiono podejścia stosowane w bezdotykowych interfejsach człowiek-komputer, przedstawiono ideę bezdotykowego interfejsu klawiatury ekranowej oraz kluczowe mechanizmy działania proponowanego rozwiązania.

Słowa kluczowe: bezdotykowe interfejsy, interfejsy dla niepełnosprawnych, komunikacja człowiek-komputer, klawiatura ekranowa, analiza obrazu.

A touchless onscreen keyboard interface for GNU/Linux systems

Abstract

In the paper the problem of a non-contact onscreen keyboard interface is addressed. There is a great number of people with disabilities who cannot have an effective control over electronic devices. They are unable to use standard input devices (mouse and keyboard) or take advantage from new sensors (e.g. Kinect). Communication with other people over the Internet and access to information are only possible with some control over computer equipment. There are four main approaches in touchless interfaces: eyetracking [1, 2], brain-computer interface [3, 4], speech recognition [5] and computer vision [6-9]. These techniques are addressed in Section 1 and the paper focuses on the computer vision approaches using them for a touchless onscreen keyboard – a visual input interface for disabled. The idea of the interface, named QVirtboard, is presented in Fig. 1 and described in Section 2. The gesture interpreter processes a video stream from a simple webcam. The user interaction is through movement (usually hand). User movements interpreted as gestures allow a key on the keyboard selection. Image analysis is based on colour image processing [12] and background modeling in video sequences [10, 11]. It is described in Section 3 and examples are shown in Fig. 2. To minimize the distance between each key, a special keyboard layout is proposed in Section 4 (Fig. 4). The layout takes into account the frequency of occurring the pairs of Polish language letters (Fig. 3 presents the appropriate diagram). The solution designed for GNU/Linux systems is open and extensible. The project is accessible through the sourceforge.net service <http://sourceforge.net/projects/qvirtboard/>.

Keywords: touchless interfaces, interface for disabled, human-computer interaction, onscreen keyboard, image analysis.

1. Bezdotykowe interfejsy

Nowej generacji interfejsy użytkownika umożliwiają sterowanie poprzez naturalne wskazywanie i gesty. Podstawę ich działania stanowią algorytmy przetwarzania i rozpoznawania obrazów pozyskiwanych z sensorów optycznych. Z drugiej strony w społeczeństwie funkcjonują osoby cierpiące na pewnego rodzaju dysfunkcje uniemożliwiające im jakiegokolwiek operowanie sprzętem

elektronicznym. Dedykowane dla takich osób rozwiązania często przekraczają ich zdolności finansowe. Otwartość oraz szerokie możliwości konfiguracji to cechy, które powinny wyróżniać tego typu podejścia.

Generalnie w bezdotykowych interfejsach sterowania sprzętem elektronicznym wyróżnić można cztery podstawowe techniki:

- śledzenie ruchów gałek ocznych (ang. *eyetracking*),
- interfejs mózg-komputer (ang. *brain-computer interface*),
- rozpoznawanie mowy (ang. *speech recognition*),
- podejścia widzenia maszynowego (ang. *computer vision approaches*).

Dzięki technikom *eyetrackingu* można określić miejsce, w które w danej chwili spogląda użytkownik. Nowoczesne systemy dokonują estymacji punktu spojrzenia opierając obliczenia na modelu oka człowieka, jego aparycji w świetle podczerwonym oraz punktach odniesienia [1]. Wyliczany wektor zmian pozwala oszacować punkt spojrzenia np. na ekranie komputera. *Eyetracking* stanowi szansę dla osób, które nie mają innej możliwości komunikacji z komputerem i światem zewnętrznym [2].

Interfejs mózg-maszyna pozwala na bezpośrednią komunikację pomiędzy mózgiem a odpowiednim urządzeniem zewnętrznym. Techniki BCI pomijają naturalny proces występujący przy sterowaniu lub komunikacji, gdzie intencja użytkownika aktywuje pewne obszary w mózgu, system nerwowy dokonuje transmisji sygnału do odpowiednich mięśni, a te z kolei realizują zdarzenie [3]. Mierzona aktywność mózgową użytkownika (obecność konkretnych wzorców) jest poddawana analizie i używana w roli sygnałów kontrolnych w interfejsie człowiek-maszyna [3]. Kluczową rolę spełniają tutaj metody przetwarzania sygnałów fal mózgowych [4].

Kolejne podejście do bezdotykowych interfejsów człowiek-maszyna to rozpoznawanie mowy. Technologia ta, rozwijana od lat pięćdziesiątych minionego wieku, uznawana jest obecnie za dojrzałą [5]. Charakteryzuje się ogromnym potencjałem w systemach dialogowych i systemach kontroli urządzeń (np. smartfonach).

Ostatnie z rozważanych podejść, wykorzystane także w niniejszym opracowaniu, to techniki widzenia maszynowego. W grupie tej występują rozwiązania operujące w paśmie światła widzialnego jak i podczerwieni. Kontrola komputera zapewniona jest poprzez śledzenie ruchów głowy lub specjalnego markera. Przykładem systemu zastępującego ruch myszy ruchami głowy jest Enable Viacam (eViacam) [6], operujący w świetle widzialnym bez markera. Przykładem systemu z markerem (umieszczanym na wybranej części ciała i śledzonym przez system) jest SmartNav [7] operujący w zakresie fal podczerwonych. Wiele z rozwiązań sterowania, wykorzystujących techniki widzenia maszynowego, wykorzystuje obserwację i analizę oczu użytkownika. Nie są to jednak systemy *eyetracking'owe*. Pozycja oczu w obrazie dwuwymiarowym jest wykrywana, a zmiany w położeniu przesyłane są do systemu kontroli [8]. Mruganie oczami przez użytkownika jest z kolei często wykorzystywane w roli potwierdzenia (np. w systemie b-Link [9]).

2. Idea bezdotykowego interfejsu klawiatury ekranowej

Klawiatura ekranowa (ang. *onscreen keyboard*, *virtual keyboard*) przenosi możliwości klawiatury sprzętowej na ekran monitora. W tradycyjnych rozwiązaniach obsługiwana jest z wykorzystaniem myszy komputerowej. Proponowane rozwiązanie QVirtboard rozszerza te funkcjonalności o możliwość kontroli bezdotykowej za pośrednictwem kamery cyfrowej.

Bezdotykowy interfejs klawiatury ekranowej w głównej mierze przeznaczony jest dla osób cierpiących na pewnego rodzaju (stałe lub czasowe) dysfunkcje uniemożliwiające jakiegokolwiek operowanie sprzętem elektronicznym. Funkcjonowanie takich osób w cyfrowym świecie uzależnione jest od dostępności specjalistycznych rozwiązań sprzętowo-programowych. Dedykowane rozwiązania często przekraczają zdolności finansowe osób potrzebujących. Otwartość oraz szerokie możliwości konfiguracji to cechy, które powinny charakteryzować tego typu podejścia.

Z wymienionych powyżej względów, jako docelowa platforma implementacji klawiatury ekranowej wybrany został system GNU/Linux. Jest on bezpłatny i oferuje szerokie możliwości konfiguracji i dostosowywania. Docelowo, utworzona została przenośna wersja systemu, wyposażona w proponowany interfejs QVirtboard, uruchamiana z nośnika USB lub CD. Przygotowane środowisko to zmodyfikowana dystrybucja Knoppix w wersji 7.0.5.

Idea funkcjonowania systemu opiera się na zastąpieniu tradycyjnej klawiatury za pomocą klawiatury ekranowej współpracującej z kamerą. Kamera rejestruje ruchy użytkownika, które są następnie interpretowane przez system i używane do sterowania i wyboru klawiszy. Proponowany bezdotykowy interfejs klawiatury ekranowej ma wizualną formę zaprezentowaną na rys. 1. Składa się z dwóch części: samej klawiatury (lewa część) oraz mapy akcji (prawa część).



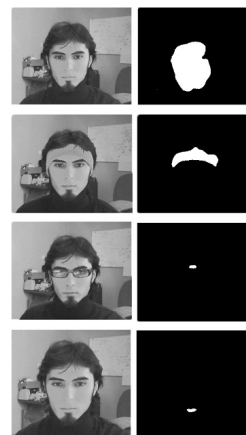
Rys. 1. Interfejs QVirtboard
Fig. 1. QVirtboard interface

Ruchy użytkownika rejestrowane są z użyciem mechanizmu markera (szczególnie w kolejnym punkcie). Projekcja ruchów markera odbywa się na mapie akcji. Mapa akcji podzielona jest na pięć obszarów. Wykryty w scenie marker otaczany jest neutralnym obszarem i w czasie inicjalizacji sytuowany jest w centralnym kwadracie. Wielkość tego obszaru neutralnego (nazywanego dalej „martwą strefą”) uzależniona jest od wielkości samego markera oraz od preferencji ruchowych użytkownika (jest to wartość w pełni konfigurowalna). Ruchy w granicach martwej strefy nie wyzwalają żadnej akcji w systemie. Wyjście poza strefę neutralną w jednym z czterech podstawowych kierunków (górną, lewo-prawo) i powrót interpretowane są jako akcja. Brak powrotu interpretowany jest jako zmiana pozycji użytkownika. Założono bowiem, że system nie może wymuszać na użytkowniku stałej i nieruchomej pozycji. Dlatego konieczne było wprowadzenie neutralnego obszaru w formie martwej strefy tak, aby dopuścić subtelne ruchy użytkownika.

Marker na ekranie pokazywany jest w formie animowanego kształtu w kształcie koła. Animacja pokazuje upływ czasu. Czas jest zerowany w momencie zmiany sektora przez marker lub kalibracji markera w czasie zmiany położenia użytkownika (brak powrotu do martwej strefy). Długość okresu jest konfigurowalna. Sama wizualizacja, poprzez zapewnienie sprzężenia zwrotnego, jest niezwykle pomocna użytkownikowi na etapie nauki obsługi systemu.

3. Wykrywanie markera

Podstawą skutecznego działania systemu stanowi założenie o posiadaniu przez użytkownika charakterystycznej barwnej cechy. Może to być naklejony punktowy marker na okularach lub części twarzy (np. czole). Może to być dodatkowy element ubioru, np. same okulary, opaska, czapka lub kolorowa końcówka mikrofonu (zakładanego na ucho ze słuchawkami). Wykorzystane mogą być także usta lub cała twarz. Taki wyróżniający się element może zostać skutecznie wykryty w obrazie z wykorzystaniem algorytmów modelowania tła [10, 11] i algorytmów segmentacji obrazów w przestrzeniach barwnych [12]. Wykorzystywana przestrzeń barwna musi separować sygnały luminancji i chrominancji. Taką przestrzenią barw nie jest np. RGB, ale jest HSV (wykorzystana w konstrukcji opisywanego systemu). Na rys. 2 przedstawione zostały przykłady przetwarzania obrazu wejściowego dla różnych kategorii elementów wybieranych przez użytkownika jako markery. Są to od góry: twarz, opaska, naklejka na okularach, usta. W lewej kolumnie przedstawione zostały obrazy z kamery, w prawej – rezultaty przetwarzania (wykrycia markera na etapie pracy systemu).



Rys. 2. Przykłady przetwarzania obrazów
Fig. 2. Examples of image processing

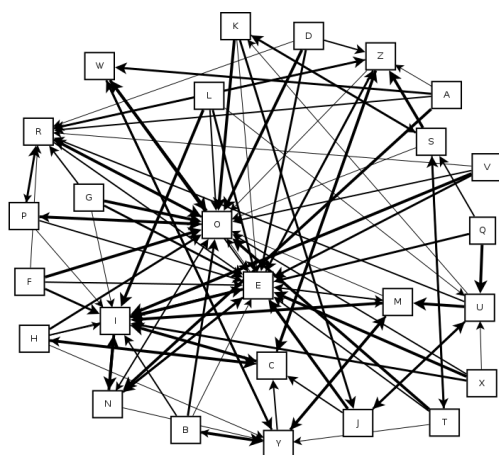
Marker wybierany jest na etapie kalibracji poprzez wskazanie przez użytkownika. Jest to automatyczny proces, w którym wybierane są piksele z sąsiedztwa. Są one dalej weryfikowane pod względem podobieństwa. Grupa podobnych pikseli jest używana do wygenerowania modelu kolorowego markera używanego przez użytkownika.

4. Dedykowany układ klawiszy QVirtboard

Kluczową cechą wszystkich klawiatur jest zastosowany układ klawiszy. Najpowszechniej stosowanym obecnie układem klawiatury jest układ QWERTY oraz jego warianty. Próbę utworzenia bardziej efektywnego układu, podjął w 1932 roku August Dvorak, aranżując klawisze w sposób uwzględniający najczęściej występujące pary liter w języku angielskim. Trzecim pod względem popularności na świecie, jest układ Colemak. Układ ten, podobnie jak Dvorak, ma zwiększać ergonomię pisania poprzez: umieszczenie najczęściej występujących znaków w środkowym rzędzie klawiatury, obniżając ilość przeskoków między rzędami oraz minimalizując ruchy małego palca.

Wymienione powyżej układy klawiatur uwzględniają w swej budowie głównie charakterystykę języka angielskiego. Brane pod uwagę są także cechy fizyczne, takie jak rozmieszczenie palców czy budowa maszyny do pisania. W przypadku klawiatury ekranowej, cechy te nie mają znaczenia. Wobec powyższych, podjęto próbę opracowania polskiego układu klawiatury ekranowej, dedykowanej bezdotykowemu interfejsowi. Taki sposób sterowania i wyboru klawiszy wymusza minimalizację długości przejść od jednego klawisza do kolejnego.

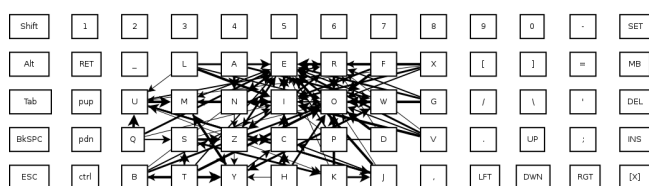
Przeprowadzono badania, których celem było wyznaczenie statystyk występowania par znaków z uwzględnieniem kolejności oraz bez jej uwzględnienia. Na podstawie sporządzonych statystyk posortowano znaki pod względem popularności oraz utworzono listę najczęściej występujących par. Popularność danego znaku mówi o tym, z jakimi innymi znakami i jak często on występuje. Za dane wejściowe posłużył zbiór leksemów języka polskiego zawartych w słowniku GNU Aspell (<http://aspell.net/>), udostępnianym na licencji LGPL. W celu trafniejszej interpretacji i możliwości porównania wyników, w zbiorze wejściowym oddzielono słownik z formami odmienionymi oraz podstawowymi. Do listy plików wejściowych dodano także losowo wybrane publikacje powyżej tysiąca stron, o treści zapisanej we współczesnym języku polskim. Rysunek 3 przedstawia diagram sporządzony na podstawie wyników przetwarzania słownika Aspell z odmianami. Każdy znak przedstawiony na diagramie, posiada cztery połączenia wskazujące na znak najczęściej po nim następujący. Wraz z malejącą grubością połączenia maleje częstość wystąpień sąsiada.



Rys. 3. Diagram sąsiedztwa znaków
Fig. 3. The character neighbourhood graph

Wyniki dla większości danych wejściowych wykazały wzajemną korelację. Wskazują, że najczęściej występującymi parami znaków w języku polskim są pary „N - I” oraz „I - E”. Wniosek ten pokrywa się z informacjami zawartymi we frekwencyjnym słowniku Korpus Języka Polskiego Wydawnictwa Naukowego PWN (<http://korpus.pwn.pl/sts/low.php>), gdzie w czołówce najczęściej występujących leksemów umieszczono „nie”. W badanym przypadku, ciąg ten brany był pod uwagę nie tylko jako przeczenie, ale również jako podciąg takich łańcuchów znaków jak „niebo”, „wzniesienie” itp. Wyniki pokazują także jednoznacznie, że najmniej popularnymi znakami, są znaki „Q”, „V”, oraz „X”.

W oparciu o rezultaty przeprowadzonych badań, sporządzono układ przedstawiony na rys. 4. Na etapie projektowania układu, uwzględniono selekcję kolumny/wiersza w celu wyboru znaku, oraz możliwość bezpośredniego przejścia między skrajnymi kolumnami/wierszami. Głównym kryterium wyznaczającym wzajemną pozycję wirtualnych klawiszy, była liczba koniecznych zmian kolumny lub wiersza.



Rys. 4. Układ klawiatury QVirtboard
Fig. 4. Layout of the QVirtboard keyboard

Ogromna popularność dominującego układu QWERTY sprawiła, że proponowany układ przewidziany jest jako element opcjonalny. Stanowi on alternatywę dla dostępnych w aplikacji innych układów. Dedykowany jest on dla osób, które zmuszone sytuacją lub ograniczeniami fizycznymi są w stanie przyswoić sobie nowy układ minimalizujący odległości przejść między klawiszami dla języka polskiego.

5. Wnioski

W artykule przedstawiono bezdotykowy interfejs klawiatury ekranowej. Interakcja z użytkownikiem odbywa się poprzez analizę jego ruchów, rejestrowanych przez kamerę i interpretowanych przez system. Proponowane rozwiązanie jest otwarte i ma szerokie możliwości konfiguracji i dostrajania. Pomimo zapewnienia mechanizmów automatyzujących proces początkowej kalibracji (np. automatyczne generowanie modelu markera po wskazaniu go myszą) konieczna jest pomoc osoby sprawnej na etapie ustawień systemu. Projekt dostępny jest przez serwis sourceforge.net (<http://sourceforge.net/projects/qvirtboard/>). Przykłady działania systemu dostępne są na stronie domowej projektu: <http://qvirtboard.sf.net>.

6. Literatura

- [1] Mantiuk R., Kowalik M., Nowosielski A., Bazyluk B.: Do-It-Yourself Eye Tracker: Low-Cost Pupil-Based Eye Tracker for Computer Graphics Applications. Lecture Notes in Computer Science 7131, 115–125, 2012.
- [2] Jacob R. J. K., Karn K. S.: Eye tracking in human-computer interaction and usability research: Ready to deliver the promises. The Minds Eye: Cognitive and Applied Aspects of Eye Movement Research, Elsevier Science, Oxford, UK, 2003.
- [3] Graimann B., Allison B. Z., Pfurtscheller G.: BrainComputer Interfaces: a gentle introduction. In: Graimann B, Allison B Z, Pfurtscheller G (red.) Brain Computer Interfaces Revolutionizing Human Computer Interaction. New York: Springer Publishing, 2010.
- [4] Bashashati A., Fatourehchi M., Ward R. K., Birch G. E.: A survey of signal processing algorithms in brain-computer interfaces based on electrical brain signals. Journal of Neural engineering 4(2), R35–57, 2007.
- [5] Rebman C. M., Aiken M. W., Cegielski C. G.: Speech recognition in the human-computer interface. Information & Management 40(6), 509–519, 2003.
- [6] Loba C. M.: Enable Viacam (strona projektu). <http://eviacam.sourceforge.net/index.php> (dostęp: 2013-01-19).
- [7] NaturalPoint: SmartNav (strona produktu). <http://www.naturalpoint.com/smartnav/> (dostęp: 2013-01-16).
- [8] Santis A., Iacoviello D.: Robust real time eye tracking for computer interface for disabled people. Computer Methods and Programs in Biomedicine 96(1), 1–11, 2009.
- [9] Orange Labs Poland, Politechnika Łódzka: b-Link (strona projektu). <http://b-link.sourceforge.net/> (dostęp: 2013-01-19).
- [10] Piccardi M.: Background Subtraction Techniques: A Review. IEEE International Conference on Systems, Man and Cybernetics 4, 3099–3104, 2005
- [11] Bouwmans T.: Recent Advanced Statistical Background Modeling for Foreground Detection: A Systematic Survey. Recent Patents on Computer Science 4(3), 147–176, 2011.
- [12] Bhattacharyya S.: A Brief Survey of Color Image Preprocessing and Segmentation Techniques. Journal of Pattern Recognition Research 1 (2011), 120–129, 2011.

otrzymano / received: 09.05.2013

przyjęto do druku / accepted: 03.07.2013

artykuł recenzowany / revised paper