

# ZESZYTY NAUKOWE

**Wydawnictwo Wyższej Szkoły Agrobiznesu  
w Łomży**



Wydawnictwo Wyższej Szkoły Agrobiznesu w Łomży

Seria:

**Zeszyty Naukowe**

**Nr 45**

Łomża 2010

**WYŻSZA SZKOŁA AGROBIZNESU W ŁOMŻY  
ACADEMY OF AGROBUSINESS IN ŁOMŻA**

**REDAKTOR TECHNICZNY**

mgr inż. Marek Pawłowski

**RECENZENCI**

prof. dr hab. Leonid Kompanets  
dr hab. Edward Oczeretko prof. WSA  
dr inż. Janusz Bobulski  
dr inż. Tomasz Huścio

Skład wykonano z gotowych materiałów dostarczonych przez Autorów.  
Wydawca nie ponosi odpowiedzialności za dostarczony materiał graficzny.

**ISBN 978-83-87492-83-0**

**Copyright © by Wyższa Szkoła Agrobiznesu w Łomży  
Łomża 2010**

Wszelkie prawa zastrzeżone. Publikowanie lub kopiowanie w części lub w całości  
wyłącznie za zgodą Wydawcy.

Wydawnictwo Wyższej Szkoły Agrobiznesu w Łomży  
18-402 Łomża, ul. Studencka 19  
Tel. +48 (86) 216 94 97, fax +48 (86) 215 11 89  
E-mail: rektorat@wsa.edu.pl

## Spis treści

1. <b>Janusz Bobulski</b>	
Zastosowanie transformaty falkowej w ekstrakcji cech twarzy .....	5
2. <b>Tomasz Hućcio, Adam Radziejewski</b>	
Multimedialny podręcznik programowania w Visual Basic for Applications .....	15
3. <b>Tomasz Hućcio, Janusz Pawczyński</b>	
Wirtualna pracownia programowalnych systemów sterowania .....	23
4. <b>Tomasz Hućcio, Łukasz Kobyliński</b>	
Figury geometryczne - oprogramowanie do nauki geometrii .....	34
5. <b>Edward Oczeretko</b>	
Przekształcenie Hilberta w przetwarzaniu wybranych sygnałów biomedycznych .....	40
6. <b>Wiesław Póljanowicz</b>	
Tworzenie multimedialnych materiałów dydaktycznych	
- przegląd istniejących rozwiązań .....	54



# Zastosowanie transformaty falkowej w ekstrakcji cech twarzy

*Janusz Bobulski*

## Streszczenie

W artykule zaprezentowano jedno z zagadnień dotyczących dziedziny identyfikacji twarzy. Skuteczna ekstrakcja cech twarzy stanowi podstawę do prawidłowego jej zakwalifikowania w procesie identyfikacji. Proponowane podejście bazuje na transformacie falkowej (TF), wykorzystywanej dotychczas w rozpoznawaniu mowy. Zastosowanie tej metody ekstrakcji cech, w połączeniu z odpowiednią metodą klasyfikacji danych pozwala na zbudowanie skutecznego systemu identyfikacji osób. W porównaniu do innych metod ekstrakcji cech, np. transformata Fouriera, transformata falkowa wykazuje lepsze właściwości oraz uniezależnia system od wpływu czynników zewnętrznych, typu zmiana natężenia oświetlenia oraz pozwala na lokalną analizę sygnału i wydobywanie z niego szczegółów, które są tracone w analizie globalnej oferowanej przez FFT.

**Słowa kluczowe:** biometria, ekstrakcja cech, identyfikacja twarzy, transformata falkowa

## Wprowadzenie

Rozpoznawanie twarzy jest jedną z dziedzin wyodrębnioną z rozległej nauki – biometrii, która obecnie jest czołowym zagadnieniem wielu ośrodków badawczo-rozwojowych. Zainteresowanie tą dziedziną wynika z potencjalnych możliwości praktycznego zastosowania nowych rozwiązań w identyfikacji osób w systemach wymagających autoryzacji dostępu osób uprawnionych do korzystania z potencjalnych zasobów, a także w celu wykorzystania ich w prewencji, ściganiu przestępców, a także autoryzacji dostępu [1].

Systemy identyfikacji twarzy składają się z trzech zasadniczych modułów: obróbki wstępnej, ekstrakcji cech, oraz reguły decyzyjnej. W fazie obróbki wstępnej wczytywany jest obraz zawierający twarz, a moduł odpowiada za wydzielenie i znormalizowanie twarzy. W drugim etapie następuje ekstrakcja cech twarzy, podobnie

jak w rozpoznawaniu dowolnych wzorców należy wybrać charakterystyczne właściwości badanego obiektu oraz uniezależnić je od czynników zakłócających i zewnętrznych. Etap ten jest najtrudniejszy, gdyż warunkuje on skuteczność systemu. W przypadku wyboru niewłaściwych cech lub tych, które nie różnią się znacząco u poszczególnych osób nie będzie możliwa poprawna weryfikacja obrazu twarzy. W ostatniej fazie procesu rozpoznawania twarzy następuje podjęcie decyzji o podobieństwie obiektów. Do tego celu można zastosować sieci neuronowe, ukryte modele Markowa i wiele innych. Jednak żadne z tych narzędzi nie da nam zadowalających wyników, jeśli w drugim etapie nie będzie zastosowana odpowiednia procedura.

### Transformata falkowa

Matematycznie, proces analizy Fouriera jest reprezentowany przez transformatę Fouriera (1),

$$F(\omega) = \int_{-\infty}^{\infty} f(t)e^{-j\omega t} dt \quad (1)$$

w której sumą są wszystkie czasy sygnału  $f(t)$  pomnożonego przez zespolony wykładniczy. Skutkami przekształceń Fouriera są *współczynniki*, które kiedy są pomnożony przez sinusoidę o odpowiedniej częstotliwości, wydają składowe sinusoidalne części oryginalnego sygnału. Graficznie cały proces został przedstawiony na Rys.1:

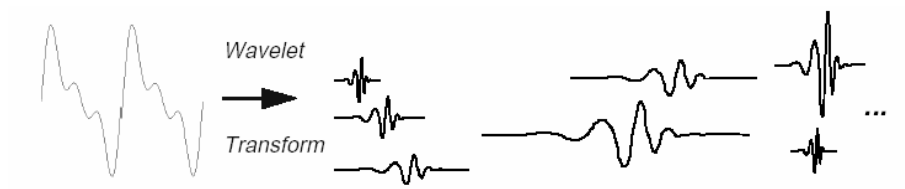


Rys.1. Proces otrzymywania składowych sinusoidalnych różnych częstotliwości [3].

Podobnie, ciągła transformata falkowa (CWT) jest zdefiniowana jako suma wszystkich czasów sygnału pomnożonego przez skalę (2).

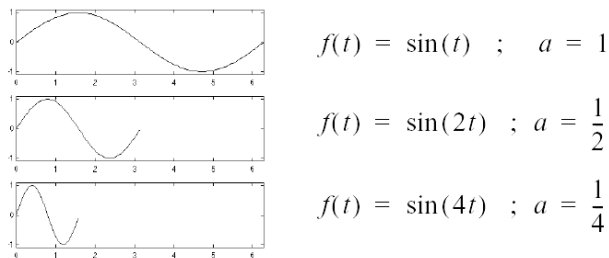
$$C(\text{skala}, \text{pozycja}) = \int_{-\infty}^{\infty} f(t)\psi(\text{skala}, \text{pozycja})dt \quad (2)$$

Wynikiem CWT jest wiele *współczynników falki C*, które są funkcjami skali i położenia. Pomnażając każdy współczynnik przez właściwie skalowaną i przesuniętą falkę zyskujemy składowe falki oryginalnego sygnału. Graficznie cały proces został przedstawiony na Rys.2:



Rys.2. Proces otrzymywania falki oryginalnego sygnału [3].

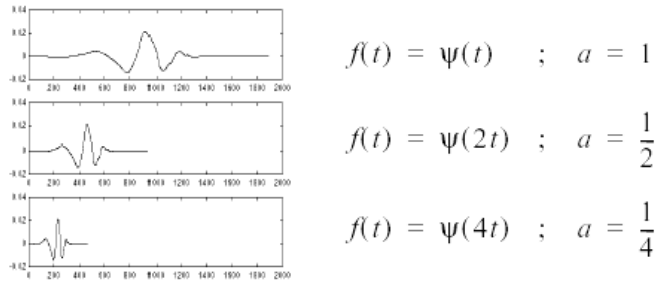
Skalowanie falki oznacza potocznie jej rozciąganie (albo kompresowanie). Aby wyjść poza potoczne określenie takie jak "rozciągnąć", wprowadzamy współczynnik skalowania, często oznaczony przez literę  $a$ . W przypadku sinusoid, skutek współczynnika skalowania jest bardzo jasny (Rys.3).



Rys.3. Skalowanie sinusoid [3].

Współczynnik skalowania pracuje dokładnie tak samo z falkami, współczynnik mniejszej skali, bardziej „kompresuje” falkę.





Rys. 4. Skalowanie falki.

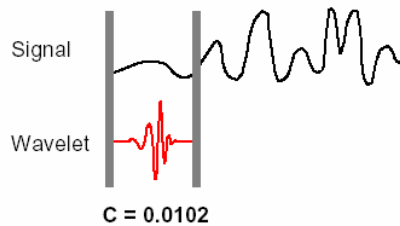
Przesunięcie falki po prostu znaczy opóźnianie (albo przyspieszenie) jej początku. Matematycznie, opóźnienie funkcji  $f(t)$  przez  $k$  jest przedstawione przez  $f(t-k)$ . Proces ten przedstawiony został na Rys.5



Rys. 5. Przesunięcie falki [3].

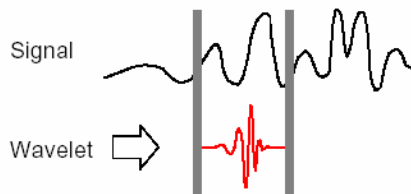
Ciągła transformata falkowa jest sumą całego czasu sygnału pomnożonego przez skalowanie, przesuniętą wersję falki. Ten proces daje współczynniki falki, które są funkcją skali i położenia. Proces tworzenia ciągłej Transformaty falkowej obejmuje następujące etapy [3]:

1. Porównujemy falkę do odcinka na początku oryginalnego sygnału.
2. Obliczamy numer  $C$ , który przedstawia jak dokładnie zestawiamy falkę z tym odcinkiem sygnału. Im Wyższe jest  $C$ , tym większe podobieństwo. Skutki zależą od kształtu falki, którą wybieramy (Rys.6).



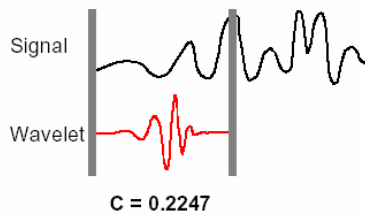
Rys.6. Zestawienie falki z początkowym odcinkiem sygnału [3].

3. Przesuwamy falkę w prawo i powtarzamy kroki 1 i 2 dopóki nie przykryją całego sygnału (Rys.7).



Rys.7. Przesunięcie falki [3].

4. Skalujemy (rozciągamy) falkę i powtarzamy kroki 1 do 3 (Rys.8).



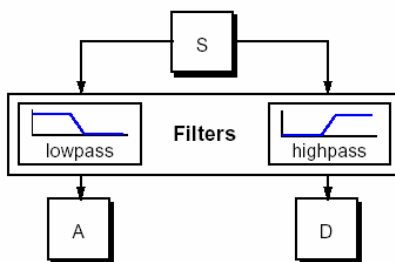
Rys.8. Skalowanie falki [3].

5. Powtarzamy kroki 1 do 4 dla wszystkich skalowań.

Po wykonaniu tych pięciu kroków, współczynniki są przedstawione w różnych skalach przez różne odcinki sygnału. Współczynniki tworzą skutki nawrotu oryginalnego sygnału wykonanego na falkach.

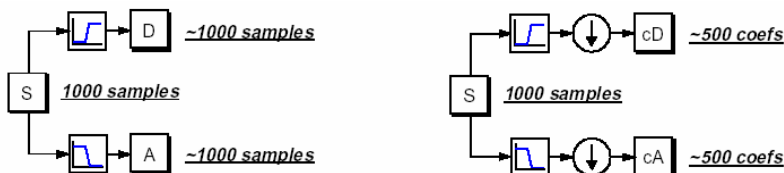
Każdy przetwarzany sygnał wykonywany na komputerze używającym prawdziwych danych musi być wykonany na sygnale dyskretnym - to jest, na sygnale, który został zmierzony w dyskretnych przedziałach czasu. Tak samo jest w przypadku ciągłej transformaty falkowej, która też działa w dyskretnym czasie.

Dla wielu sygnałów, istota niskiej częstotliwości jest najważniejszą częścią. To jest to, co daje sygnałowi jego tożsamość. Na przykład ludzki głos. Jeśli usuniemy składnik wysokiej częstotliwości, głos brzmi odmiennie, ale możemy nadal odróżnić, co jest mówione. Jednak, jeśli usuniemy składnik niskiej częstotliwości, słyszymy bełkot. W analizie falek, często rozmawiamy o *przybliżeniach* i *szczegółach*. Przybliżenia są wysokiej skali, niskiej częstotliwości składowymi sygnału. Szczegóły są niskiej skali, wysokiej częstotliwości składowymi sygnału. Proces filtrowania, w jego najbardziej zasadniczym poziomie przedstawiono na Rys.9.



Rys.9. Schemat procesu filtrowania [3].

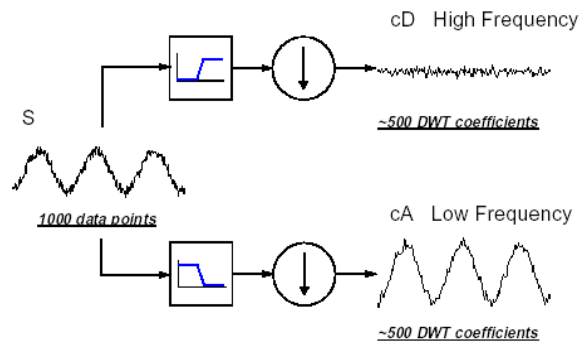
Oryginalny sygnał, S, przebiega przez dwa uzupełniające się filtry i wyłania się jako dwa sygnały. Niestety, jeśli ta operacja wykonywana jest na rzeczywistym cyfrowym sygnale, to jest używanych dwa razy więcej danych. Aby poprawić ten problem, wprowadzono pojęcie *downsampling* (Rys.10), które oznacza wyrzucanie, co drugiego punktu danych. Robiąc to wprowadzamy *schodkowanie* w elementach sygnału.



Rys.10. Sygnał przed i po downsamplingu [3].

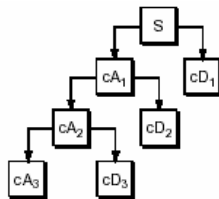
Proces po prawej stronie, który zawiera downsampling, produkuje współczynniki DWT.

Lepsze zrozumienie procesu umożliwi wykonanie jednoetapowej dyskretnej transformaty falkowej sygnału. Sygnał będzie zwykłą sinusoidą z wysokoczęstotliwościowym hałasem dodanym do niego. Rys.11 przedstawia schemat procesu z rzeczywistymi sygnałami.



Rys.11 Schemat jednoetapowej dyskretnej transformaty falkowej sygnału [3].

Proces rozkładu może zostać powtórzony, wraz z kolejnym przybliżeniem będąc rozłożonym w układzie, tak żeby jeden sygnał został rozbity na dużo niższe składniki rozwiązania. Taki proces nazywany jest *drzewem rozkładu falki* (Rys1.12).

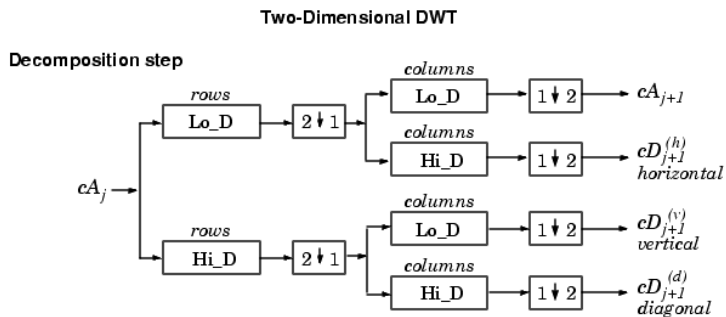


Rys.12. Drzewo rozkładu sygnału falki [3].

### Dwuwymiarowa transformata falkowa

W przypadku sygnałów dwuwymiarowych do przeprowadzenia transformaty stosuje się wydzielony bank filtrów. Zazwyczaj stosuje się parę filtrów – górno i dolno przepustowy. Splot z filtrem dolnoprzepustowym daje w efekcie tak zwany obraz

przybliżony, natomiast spłot z kombinacją filtrów pasmowych zwraca obrazy z wydzielonymi szczegółami w określonych kierunkach: poziomym, pionowym i diagonalnym. Schemat przetwarzania obrazu przedstawiono na Rys. 13.



Rys. 13. Schemat dwuwymiarowej transformaty falkowej [3].

Efekt zastosowania transformaty falkowej na rzeczywistym obrazie twarzy przedstawiono na Rys. 14. Rozmiar obrazu wynikowego jest równy rozmiarowi obrazu wejściowego, poprzez zastosowanie *downsamplingu*, czyli odrzuceniu co drugiej kolumny i co drugiego wiersza. Operacja ta nie zwiększa ilości danych, jednocześnie nie powodując istotnej utraty informacji.



Rys.14. Obraz oryginalny i jego transformata falkowa.

Tak jak w przypadku transformaty jednowymiarowej tak i przy przetwarzaniu obrazów, proces może być powtórzony przez zastosowanie rekurencji [2]. W rezultacie można uzyskać bardziej szczegółowe dane dotyczące przetwarzanej informacji. Sposób tworzenia n-poziomowej transformaty można opisać następująco:

$$A_n = [H_x * [H_y * A_{n-1}] \downarrow_{2,1}] \downarrow_{1,2}$$

$$D_{n1} = [H_x * [G_y * A_{n-1}] \downarrow_{2,1}] \downarrow_{1,2}$$

$$D_{n2} = [G_x * [H_y * A_{n-1}] \downarrow_{2,1}] \downarrow_{1,2}$$

$$D_{n3} = [G_x * [G_y * A_{n-1}] \downarrow_{2,1}] \downarrow_{1,2}$$

Gdzie: \* - oznacza splot funkcji,  $\downarrow_{2,1}$  ( $\downarrow_{1,2}$ ) oznacza próbkowanie co drugi rząd (kolumnę),  $A_0$  - oryginalny obraz.

### Podsumowanie

Największą zaletą transformaty falkowej jest jej precyzja, którą można skalować zależnie od potrzeb. Dotychczas transformata falkowa stosowana była głównie w przetwarzaniu sygnałów jednowymiarowych np. przy analizie dźwięków w procesie rozpoznawania mowy. Można jednak z powodzeniem stosować ją w przetwarzaniu i analizie sygnałów dwuwymiarowych, np. obrazów twarzy, a w połączeniu z sieciami neuronowymi, modelami Markowa lub innymi regułami decyzyjnymi, pozwala na zbudowanie skutecznych systemów rozpoznawania twarzy.

### Literatura

1. Bobulski J., 2(2)2002, Metoda rozpoznawania twarzy z wykorzystaniem pseudo-entropijnego kryterium, Informatyka teoretyczna i stosowana, Zeszyty naukowe Instytutu Matematyki i Informatyki na Politechnice Częstochowskiej, s. 69-74
2. Gacia C., Zikos G., Tziritas G., 2000, Wavelet packet analysis for face recognition, Image and Vision Computing 18, pp 289-297
3. Misiti M., Oppenheim G., Poggi J.M., Misiti Y., 1996, [Wavelet Toolbox User's Guide](#), MatLab 5.0, The MathWorks

**Abstract**

There were presented problems concerning to the face identification in this paper. The base of correctly classification of face is an effective extraction of the face features. The proposed approach bases on Wavelet Transform (WT) which is used in speech recognition. The application of this method together with proper data classification method enables building an effective person identification system. Wavelet transform reveals better properties than other features extraction methods e.g. Fourier Transform. WT makes the system independent on the impact of external conditions like changing of light intensity. WT allows to local signal analysis and details extracting which are lost in global analysis with FFT.

dr inż. Janusz Bobulski

Wyższa Szkoła Agrobiznesu w Łomży

e-mail: januszb@icis.pcz.pl

# **Multimedialny podręcznik programowania w Visual Basic for Applications**

*Tomasz Hućcio, Adam Radziejewski*

## **Streszczenie**

W artykule przedstawiono program WirtualVBA. WirtualVBA jest multimedialnym podręcznikiem z zakresu programowania w Visual Basic. Program oprócz podstaw teoretycznych z dziedziny programowania w VBA zawiera zestaw ćwiczeń, zadania oraz ich rozwiązania. Wykorzystanie zasady interaktywności umożliwia szybsze i skuteczniejsze przyswajanie wiedzy z zakresu programowania w Visual Basic.

**Słowa kluczowe:** multimedialny podręcznik, interaktywność,

## **Wprowadzenie**

WirtualVBA jest interaktywnym programem multimedialnym, którego zasadniczym przeznaczeniem jest wspomaganie procesu nauczania w zakresie programowania w Visual Basic.

Program oprócz podstaw teoretycznych z dziedziny programowania w VBA (Visual Basic for Applications – system programowania aplikacji) zawiera zestaw ćwiczeń do każdej lekcji oraz rozwiązania do tych ćwiczeń. Z założenia powinien on być elementem dopełniającym (wspomagającym) pracę studenta (ucznia) na zajęciach.

Program WirtualVBA został opracowany w programie Flash MX firmy Macromedia (aktualnie Adobe). Oprogramowanie Flash MX z wewnętrznym językiem programowania ActionScript tworzy środowisko programistyczne do tworzenia interaktywnych animacji, które reagują na działania użytkownika. Dzięki wykorzystaniu oprogramowania Flash MX multimedialny podręcznik z zakresu programowania w Visual Basic zajmuje tylko 1,89 MB w pamięci na dysku twardym komputera.



## Prezentacja programu WirtualVBA

Multimedialny podręcznik z zakresu programowania w Visual Basic spełnia następujące założenia:

- łatwa nawigacja po poszczególnych stronach podręcznika,
- atrakcyjna, przyciągająca swym wyglądem grafika,
- intuicyjny interfejs,
- treść podręcznika obejmuje swym zasięgiem podstawy teoretyczne oraz ćwiczenia z zakresu programowania w Visual Basic,
- łatwa instalacja i obsługa programu.

Poszczególne strony multimedialnego podręcznika zawierają:

- elementy informacyjne (teksty, tabele, fotografie, itd.),
- elementy sterujące (przyciski nawigacyjne).

Na rysunku 2.1 przedstawiono główną stronę multimedialnego podręcznika WirtualVBA. Multimedialny podręcznik składa się z następujących elementów:

- *Kurs* (rysunek 2.2) zawierający:
  - *Wprowadzenie* – podstawy teoretyczne (rysunek 2.3),
  - *Visual Basic* – opis środowiska projektowego (okno programu VBA)(rysunek 2.4),
  - instrukcje do ośmiu zajęć (rysunek 2.5).
- *Zadania* – rozwiązania zadań z zakresu programowania w VBA, których treści podano w kolejnych instrukcjach z działu *Kurs* (rysunek 2.6).
- *Procedury* – alfabetycznie zestawiony spis procedur Visual Basic (rysunek 2.7).



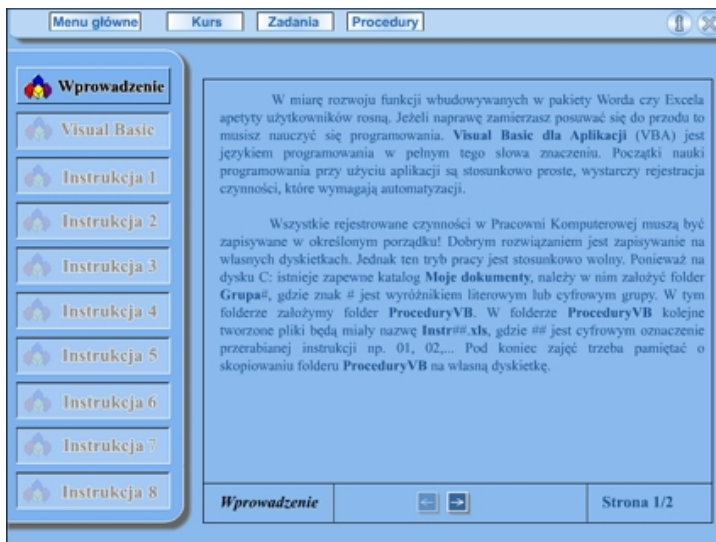
Rys. 2.1. Główna strona multimedialnego podręcznika WirtualVBA.



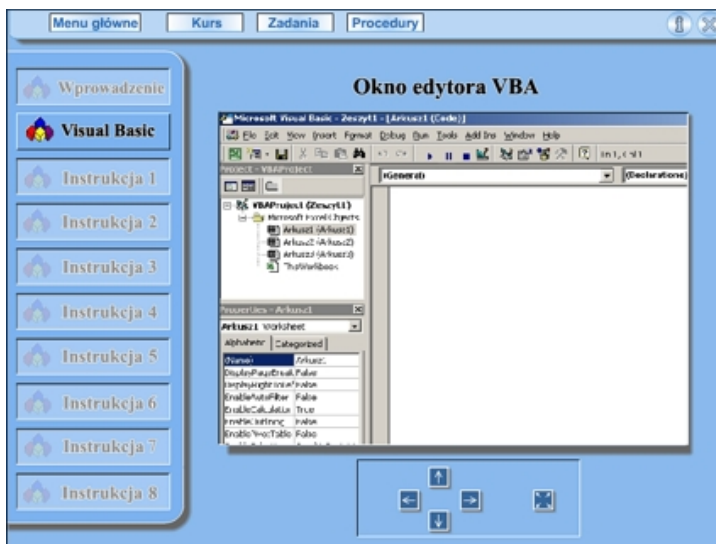
Rys. 2.2. *Kurs* - strona multimedialnego podręcznika WirtualVBA.

Interfejs strony *Kurs* zawiera menu górne z przyciskami do pozostałych części podręcznika (*Zadania*, *Procedury*), przycisk *Menu główne* do przejścia na stronę główną podręcznika, przyciski do poszczególnych stron kursu (*Wprowadzenie*, *Visual Basic*,

instrukcje do zajęć), przycisk informacji o programie oraz przycisk do zamykania programu.



Rys. 2.3. *Wprowadzenie* - strona multimedialnego podręcznika WirtualVBA.



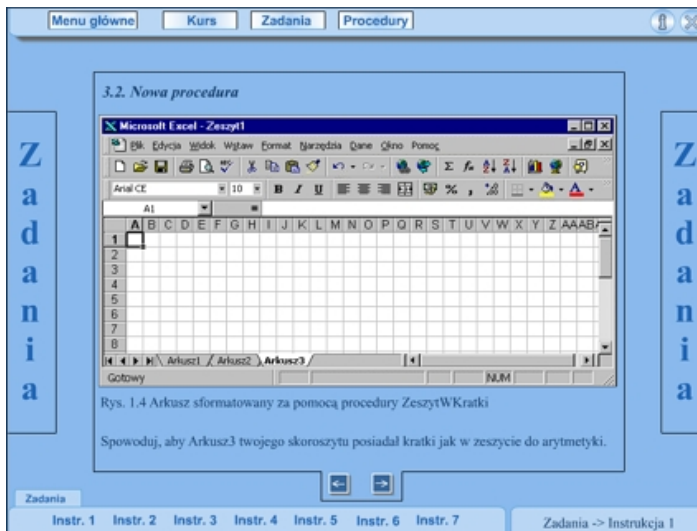
Rys. 2.4. *Visual Basic* - strona multimedialnego podręcznika WirtualVBA.



Rys. 2.5. *Instrukcja 1* - strona multimedialnego podręcznika WirtualVBA.

Pod obszarem z treścią poszczególnych instrukcji wstawiono stopkę składającą się z trzech pól:

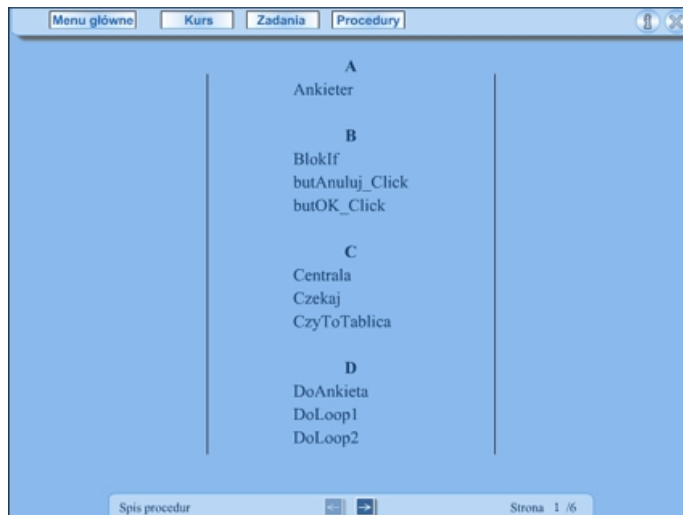
- pole z nazwą wybranego przycisku,
- pole z przyciskami do przewijania stron podręcznika (instrukcji) w przód i w tył,
- pole z numeracją stron danej instrukcji.



Rys. 2.6. *Zadania* - strona multimedialnego podręcznika WirtualVBA

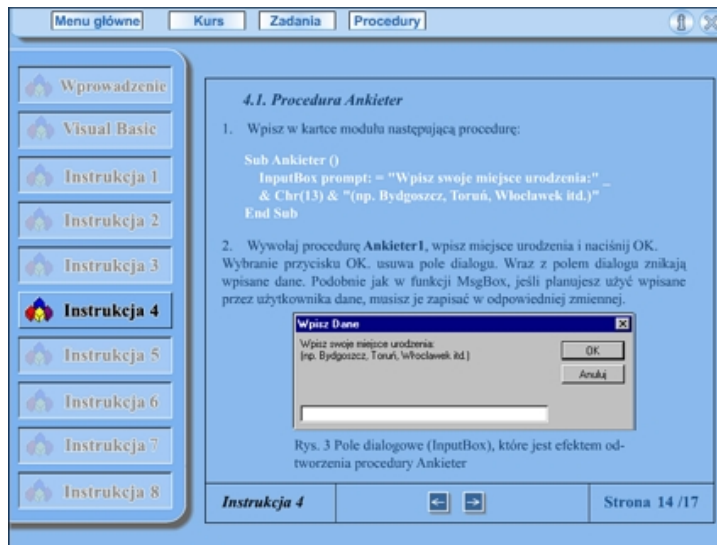
Pod obszarem z treścią poszczególnych zadań znajdują się:

- przyciski do przewijania stron podręcznika w przód i w tył,
- menu z przyciskami umożliwiającymi przejście do pozostałych zadań,
- pole informujące, z której instrukcji pochodzi przedstawione zadanie.



Rys. 2.7. *Procedury* - strona multimedialnego podręcznika WirtualVBA.

Po kliknięciu na wybraną procedurę otwiera się strona z jej opisem (rysunek 2.8). Pod obszarem z treścią wstawiono stopkę, która zawiera pola analogiczne jak w instrukcjach podręcznika.



Rys. 2.8. *Procedura Ankieter* - strona multimedialnego podręcznika WirtualVBA.

## Podsumowanie

Program WirtualVBA przeznaczony jest dla uczniów, studentów oraz osób, które chcą poznać zasady programowania w Visual Basic. Ze względu na dobrą przejrzystość oraz intuicyjne menu z łatwością mogą z niego korzystać osoby młodsze a także osoby niemające dużego doświadczenia w obsłudze komputera.

Program oprócz podstaw teoretycznych z dziedziny programowania w VBA zawiera zestaw ćwiczeń do każdej lekcji, rozwiązania do tych ćwiczeń oraz zestaw procedur stosowanych przy programowaniu w Visual Basic. W przyszłości można modyfikować jego strukturę poprzez dodanie nowych stron (np. z dodatkowymi ćwiczeniami), zadań oraz procedur.

### **Literatura**

1. Radziejewski A., 2005, Multimedialny podręcznik programowania w Visual Basic for Applications, praca inżynierska WSA, Łomża.
2. Keating J., Fig Leaf Software, 2002, Flash MX Vademecum profesjonalisty, Wydawnictwo Helion, Gliwice.
3. Sanders B., 2001, Flash ActionScript f/x, Wydawnictwo Helion, Gliwice.
4. Bargieł D., 2002, Flash MX Ćwiczenia praktyczne, Wydawnictwo Helion, Gliwice.
5. Czasopismo design Extra! Flash nr 04/2003.

### **Abstract**

The paper describes the computer program WirtualVBA. A WirtualVBA is a multimedia textbook concerning programming in Visual Basic. Apart from theoretical bases of programming in Visual Basic, computer program WirtualVBA comprises set of exercises with answers to each lesson. Usage of the principle of the interaction enables assimilating knowledge of programming in Visual Basic faster and more effective.

dr inż. Tomasz Huścio

Wyższa Szkoła Agrobiznesu w Łomży

e-mail: tomek@pb.edu.pl

# **Wirtualna pracownia programowalnych systemów sterowania**

*Tomasz Hućcio, Janusz Pawczyński*

## **Streszczenie**

W artykule przedstawiono istotę pracy w wirtualnej pracowni programowalnych systemów sterowania. Proces tworzenia oraz analiza działania programowalnych systemów sterowania z wykorzystaniem wirtualnej pracowni staje się bardziej obrazowy i interesujący. Wykorzystanie zasady interaktywności umożliwia projektantowi układów ze sterownikami programowalnymi (PLC) szybsze i skuteczniejsze (baz konieczności posiadania elementów) sprawdzenie poprawności ich działania.

**Słowa kluczowe:** wirtualna pracownia, interaktywność,

## **Wprowadzenie**

Wirtualna pracownia programowalnych systemów sterowania jest interaktywnym programem multimedialnym, którego zasadniczym przeznaczeniem jest wspomaganie procesu nauczania w zakresie programowalnych sterowników PLC (Programmable Logic Controllers). Wirtualna pracownia może być przeznaczony dla uczniów i studentów szkół technicznych w zakresie przedmiotu programowalne systemy sterowania. Program z powodzeniem można wykorzystać w realizacji ćwiczeń na zajęciach laboratoryjnych oraz do samodzielnego przygotowania studenta do ćwiczeń laboratoryjnych. Pomimo, iż program ten może istnieć samodzielnie, z założenia jest elementem dopełniającym pracę studenta w pracowni rzeczycwistej.

Treść programu jest zgodna z tematyką przedmiotu programowalnych systemów sterowania realizowanych w Wyższej Szkole Agrobiznesu w Łomży. Oprócz podstawowych informacji teoretycznych, dotyczących budowy i działania sterowników, dostarcza studentowi także wiedzy na temat języków programowania sterowników oraz programu VersaPro niezbędnego do programowania sterowników. Najważniejszą część programu stanowi zestaw ćwiczeń laboratoryjnych oraz przykładowe zadania symulujące



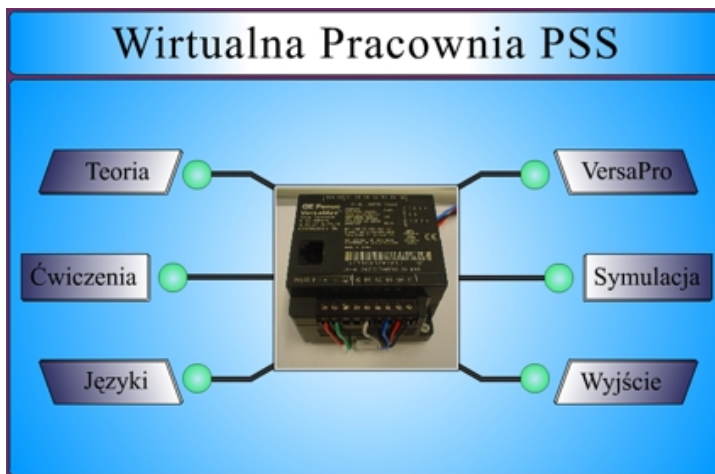
pracę sterownika w wybranych układach. Materiał przedstawiony w programie oparty jest na programowalnym sterowniku VersaMax Nano firmy GeFanuc.

Wirtualna pracownia programowalnych systemów sterowania została opracowana w programie Flash MX firmy Macromedia (aktualnie Adobe). Oprogramowanie Flash MX z wewnętrznym językiem programowania ActionScript tworzy środowisko programistyczne do tworzenia interaktywnych animacji, które reagują na działania użytkownika. Dzięki wykorzystaniu oprogramowania Flash MX wirtualna pracownia programowalnych systemów sterowania zajmuje tylko 3,83 MB w pamięci na dysku twardego komputera.

### Prezentacja wirtualnej pracowni programowalnych systemów sterowania

Na rysunku 2.1 przedstawiono interfejs wirtualnej pracowni. Wirtualna pracownia programowalnych systemów sterowania podzielona jest na moduły, które zawierają:

- podstawowe informacje wprowadzające studenta w tematykę danego zagadnienia,
- ćwiczenia wstępne, przygotowujące ogólnie do pracy w laboratorium,
- ćwiczenia będące symulacją zadań wykonywanych w realnie istniejącym laboratorium.

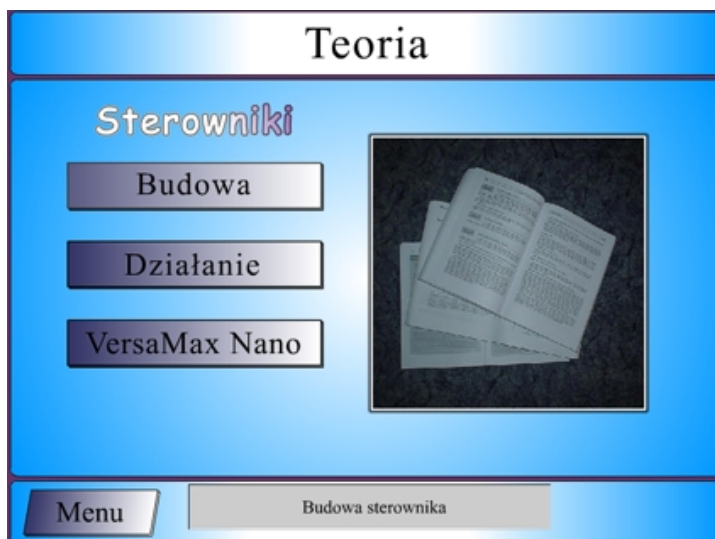


Rys. 2.1. Interfejs wirtualnej pracowni programowalnych systemów sterowania.

Wirtualna pracownia składa się z następujących modułów:

1. *Teoria* - moduł zawiera podstawy teoretyczne z zakresu programowalnych systemów sterowania. Informacje podzielone są na trzy logiczne części, dostępne z poziomu menu podrzędnego (rysunek 2.2):

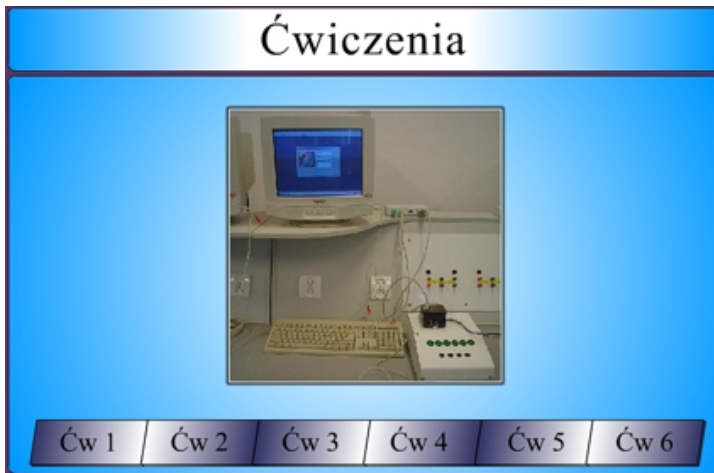
- *Budowa* - w której przedstawiono budowę sterownika,
- *Działanie* - w której ukazano sposób działania sterownika,
- *VersaMax Nano* - w której przedstawiono sterownik VersaMax z grupy Nano.



Rys. 2.2. Interfejs wirtualnej pracowni – moduł *Teoria*.

2. *Ćwiczenia* - moduł zawiera zestaw ćwiczeń laboratoryjnych, pogrupowanych tematycznie (rysunek 2.3):

- Ćwiczenie nr 1 - *Funkcje logiczne* (rysunek 2.4),
- Ćwiczenie nr 2 - *Układy pamięciowe*,
- Ćwiczenie nr 3 - *Układy czasowe*,
- Ćwiczenie nr 4 - *Liczniki*,
- Ćwiczenie nr 5 - *Funkcje matematyczne*,
- Ćwiczenie nr 6 - *Operacje na bitach*.

Rys. 2.3. Interfejs wirtualnej pracowni – moduł *Ćwiczenia*.

Rys. 2.4. Wirtualna pracownia – Ćwiczenie nr 1: Funkcje logiczne.

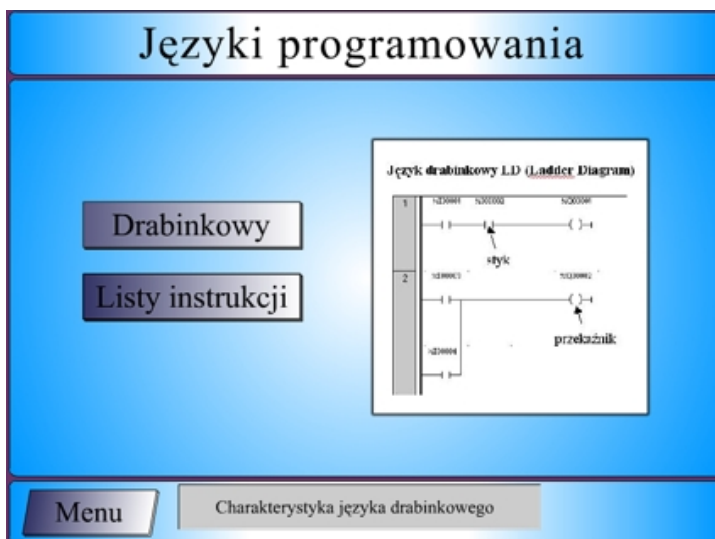
Wszystkie ćwiczenia laboratoryjne zawierają informacje podstawowe, wprowadzające studenta w tematykę danego ćwiczenia, przykłady oraz zadania do samodzielnego opracowania.

Przejsięcie do dalszej lub powrót do wcześniejszej części lekcji możliwy jest po wciśnięciu przycisków nawigacyjnych. Powrót do głównego menu wirtualnej pracowni następuje po wciśnięciu przycisku *Menu*.

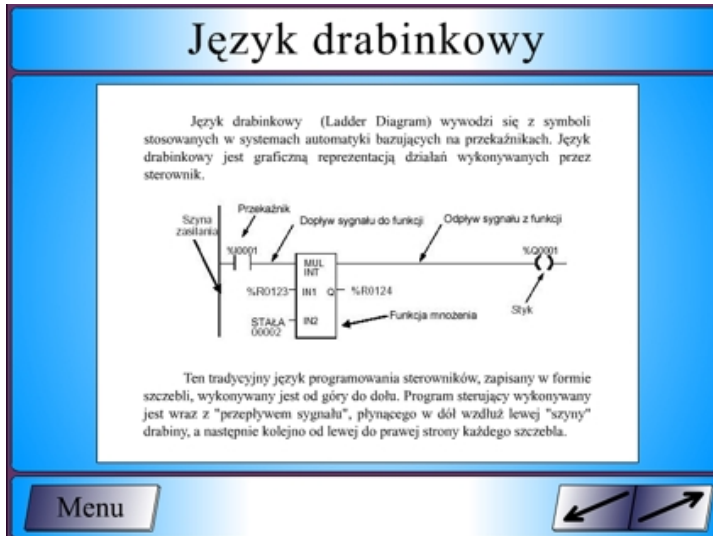
3. *Języki* - moduł dotyczy języków programowania sterowników (rysunek 2.5).

W tej części programu przedstawione zostały następujące języki:

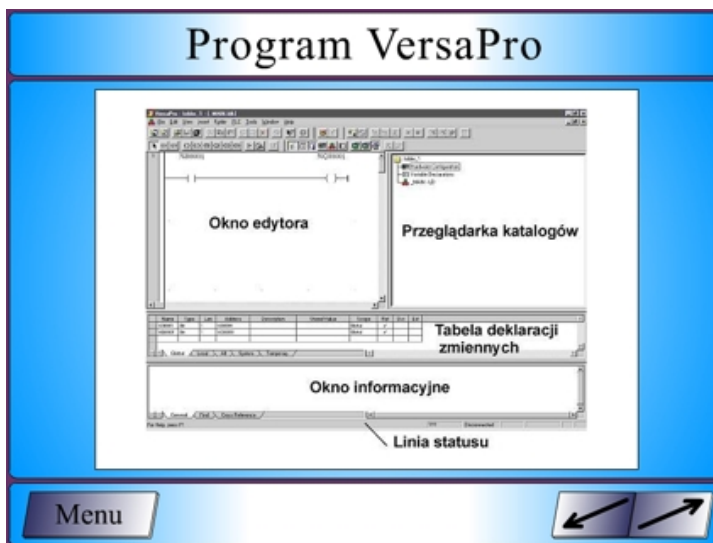
- drabinkowy (rysunek 2.6),
- listy instrukcji.



Rys. 2.5. Interfejs wirtualnej pracowni – moduł *Języki*.

Rys. 2.6. Wirtualna pracownia – lekcja *Język drabinkowy*.

4. *VersaPro* - moduł zawiera podstawowe informacje na temat programu *VersaPro* służącego do programowania sterowników (rysunek 2.7).

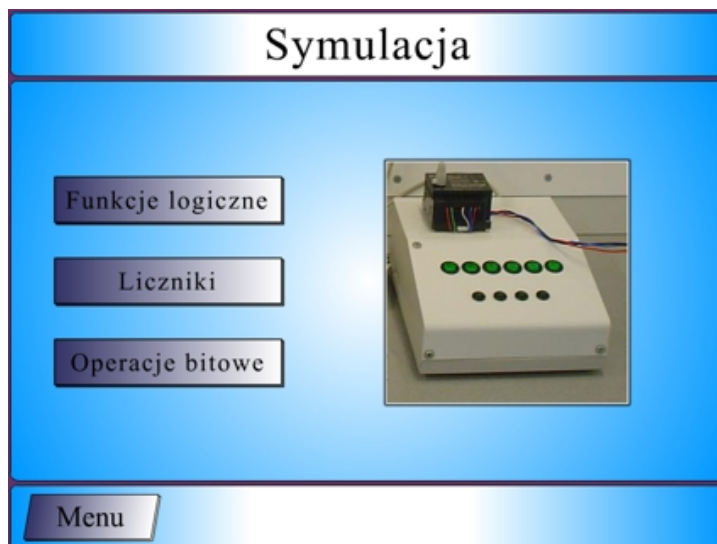
Rys. 2.7. Interfejs wirtualnej pracowni – moduł *VersaPro*.

5. *Symulacja* - moduł zawiera aplikację *MiniSterCadJP*, która umożliwia symulację działania sterownika. Moduł zawiera trzy części (rysunek 2.8):

- *Funkcje logiczne*,
- *Liczniki*,
- *Operacje bitowe*.

*MiniSterCad* jest częścią programu, w której użytkownik może sam wprowadzać dane niezbędne do zaprogramowania sterownika w odpowiednim układzie i poprzez symulację wyciągnąć odpowiednie wnioski.

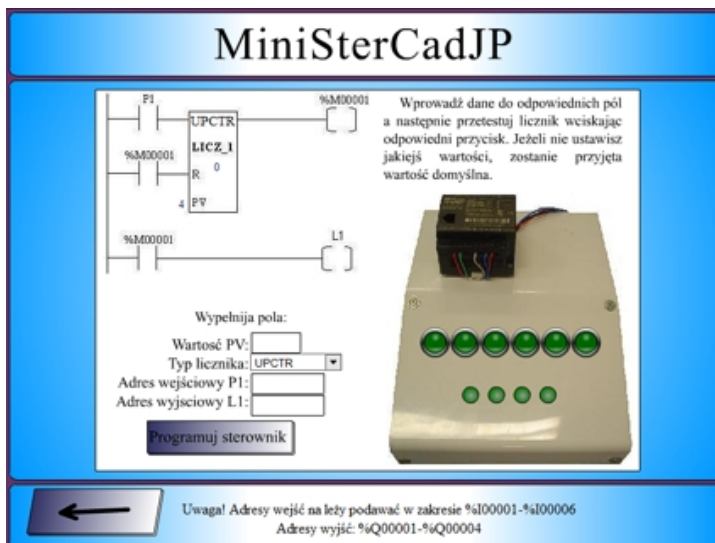
Użytkownik może wprowadzać lub modyfikować dane dotyczące: adresów wejść i wyjść sterownika, wartości zadanej licznika (PV), rodzaju licznika, rodzaju funkcji logicznych oraz rodzaju operacji bitowych.



Rys. 2.8. Interfejs wirtualnej pracowni – moduł *Symulacja*.

### **Przykładowa symulacja działania układu ze sterownikiem programowalnym przeprowadzona w wirtualnej pracowni**

Na rysunku 3.1 przedstawiono interfejs użytkownika aplikacji *MiniSterCadJP* z lekcją do symulacji układu z wykorzystaniem instrukcji licznika.



Rys. 3.1. Interfejs *MiniSterCadJP* – lekcja *Liczniki*.

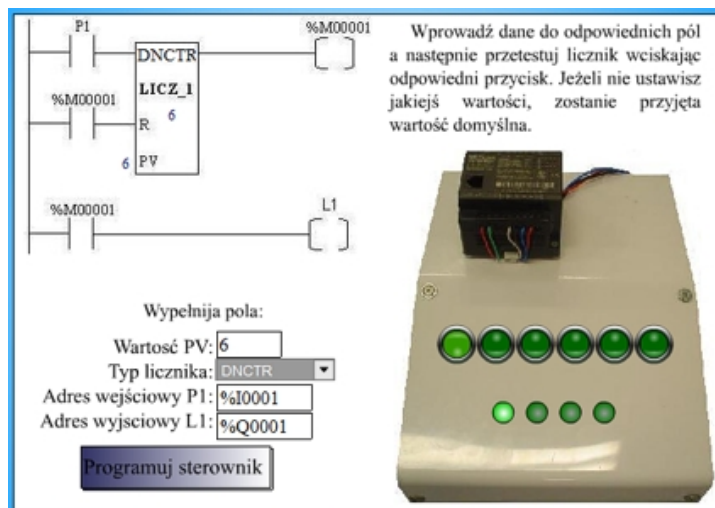
Symulacja zostanie przeprowadzona zgodnie z podanym programem sterującym, zapisanym w języku drabinkowym.

Symulację działania układu należy poprzedzić wypełnieniem określonych pól:

- *Wartość PV* – w celu przypisania do sterownika wartości zadanej licznika (liczba całkowita),
- *Typ licznika* – do wyboru licznik zliczający w dół - DNCTR i licznik zliczający w górę -UPCTR,
- *Adres wejściowy P1* – w celu określenia, który z przycisków sterownika ma być aktywny (adres %I0001 odpowiada przyciskowi pierwszemu, adres %I0002 odpowiada przyciskowi drugiemu, itd. do %I0006),
- *Adres wyjściowy L1* – w celu określenia, która lampka na panelu kontrolnym ma się zapalić gdy klikniemy na określony przycisk tyle razy ile wynosi wartość zadana PV (adres %Q0001 odpowiada pierwszej lampce, adres %Q0002 odpowiada drugiej lampce, itd. do %Q0004),

W celu przeprowadzenia symulacji działania programowalnego systemu sterowania należy wcisnąć przycisk *Programuj sterownik*.

Po tej operacji wciskamy określony przycisk na panelu kontrolnym i sprawdzamy czy lampka zapali się zgodnie z podanym programem sterującym. Wizualizacja zliczania ile razy wcisnięto przycisk jest możliwa dzięki zmianie wartości bieżącej licznika widocznej w programie sterującym (liczba całkowita pod nazwą licznika LICZ\_1). Natomiast wcisnięcie przycisku oraz zapalenie lampki sygnalizowane jest sygnałem świetlnym poprzez zmianę koloru przycisku lub lampki (rysunek 3.2).



Rys. 3.2. Symulacja działania układu w programie *MiniSterCadJP* – lekcja *Liczniki*.

### Podsumowanie

Wirtualne laboratorium jest elementem coraz częściej wykorzystywanym w edukacji. Wirtualne laboratorium pozwala na samodzielne istnienie eksperymentalnych stanowisk dydaktycznych związanych z praktycznym wykonywaniem dowolnych procesów technicznych (fizycznych), które przebiegają w naturalnych warunkach. Takie laboratorium powinno wzorować się na dwóch typach programów.

1. Programy przygotowujące do pracy w laboratorium (zawierające część teoretyczną - „podręczniki wirtualne”, ćwiczenia wstępne, testy sprawdzające wiedzę).
2. Programy symulujące ćwiczenia laboratoryjne wykonywane w rzeczywistym laboratorium na rzeczywistych elementach oraz w konkretnych ustalonych



warunkach pracy przez osoby wykonujące dane doświadczenie. [Półjanowicz, Citko 2005]

Wirtualną pracownię programowalnych systemów sterowania z powodzeniem można zaliczyć do nowej generacji multimedialnych interaktywnych metod nauczania. Dzięki możliwościom programu Flash MX i wbudowanemu językowi skryptowemu ActionScript wirtualna pracownia programowalnych systemów sterowania jest w pełni interaktywnym programem multimedialnym, wyróżniającym się atrakcyjną grafiką i animacją. Wirtualne laboratorium dostarcza studentom podstawowej wiedzy z zakresu programowania sterowników oraz przygotowuje do pracy w rzeczywistej pracowni z fizycznym sprzętem. Wykorzystanie zasady interaktywności umożliwi szybsze i skuteczniejsze przyswajanie wiedzy.

Wirtualna pracownia programowalnych systemów sterowania może być w przyszłości rozbudowana o dodatkowe elementy zgodnie z wymogami przyszłych użytkowników.

### **Literatura**

1. Pawczyński J., 2005, Wirtualna Pracownia Programowalnych Systemów Sterowania, praca inżynierska WSA, Łomża
2. Półjanowicz W., Citko U., 2005, Wykorzystanie „wirtualnych laboratoriów” w edukacji studentów, VII Ogólnopolskie Forum SNTI Gdynia, 03-05.06.2005
3. Keating J., Fig Leaf Software, 2002, Flash MX Vademecum profesjonalisty, Wydawnictwo Helion, Gliwice
4. Sanders B., 2001, Flash ActionScript f/x, Wydawnictwo Helion, Gliwice
5. Bargieł D., 2002, Flash MX Ćwiczenia praktyczne, Wydawnictwo Helion, Gliwice

### **Abstract**

The paper describes the Virtual laboratory of the programmable control systems. A creation process and analysis of the working of the programmable control systems is more vivid and interesting. Usage of the principle of the interaction enables a designer of

the systems with programmable logic controllers, faster and more effective checking of the working correctness of the systems, and eliminates necessity of possession of the elements.

dr inż. Tomasz Huścio

Wyższa Szkoła Agrobiznesu w Łomży

e-mail: [tomek@pb.edu.pl](mailto:tomek@pb.edu.pl)

# **Figury geometryczne - oprogramowanie do nauki geometrii**

*Tomasz Hućcio, Łukasz Kobyliński*

## **Streszczenie**

W artykule przedstawiono program komputerowy Figury geometryczne, który może być multimedialną pomocą do nauki geometrii w szkołach podstawowych i gimnazjalnych. Program oprócz podstaw teoretycznych dotyczących podstawowych płaskich i przestrzennych figur geometrycznych zawiera moduł do obliczeń ich pól, obwodów i objętości. Wykorzystanie zasady interaktywności umożliwia szybszą i skuteczniejszą naukę geometrii.

**Słowa kluczowe:** multimedialny podręcznik, interaktywność,

## **Wprowadzenie**

Figury geometryczne jest interaktywnym programem multimedialnym, którego zasadniczym przeznaczeniem jest wspomaganie procesu nauczania w zakresie nauki geometrii w szkołach podstawowych i gimnazjalnych. Program z powodzeniem można wykorzystać w realizacji ćwiczeń na lekcjach matematyki oraz do samodzielnego przygotowania ucznia do lekcji. Z założenia program Figury geometryczne jest elementem dopełniającym pracę ucznia na lekcjach matematyki.

Program, oprócz podstawowych informacji teoretycznych o figurach płaskich i przestrzennych, umożliwia uczniom także przeprowadzanie obliczeń obwodu, pola powierzchni i objętości wybranych figur geometrycznych.

Program Figury geometryczne został opracowany w programie Flash MX firmy Macromedia (aktualnie Adobe). Oprogramowanie Flash MX z wewnętrznym językiem programowania ActionScript tworzy środowisko programistyczne do tworzenia interaktywnych animacji, które reagują na działania użytkownika. Dzięki wykorzystaniu oprogramowania Flash MX program zajmuje tylko 1,24 MB w pamięci na dysku twardym komputera.

## Prezentacja programu Figury geometryczne

Na rysunku 2.1 przedstawiono interfejs programu Figury geometryczne.



Rys. 2.1. Interfejs użytkownika programu Figury geometryczne.

Program Figury geometryczne spełnia następujące założenia:

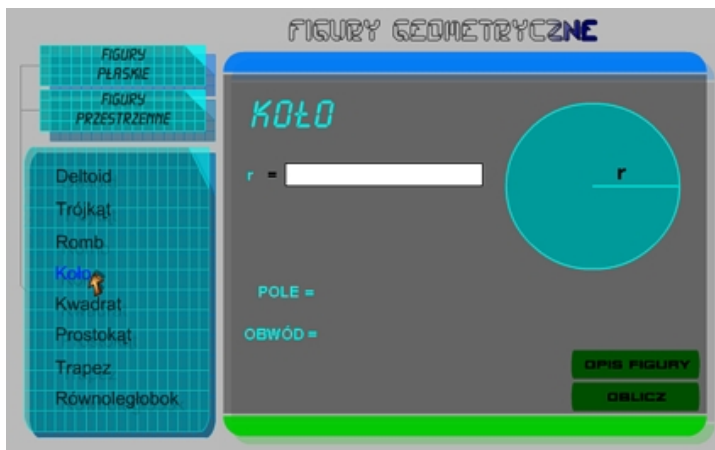
- prezentacja wybranych figur płaskich i przestrzennych,
- możliwość obliczania obwodu, pola powierzchni i objętości figur,
- opis poszczególnych figur,
- udźwiękowanie – lektor czyta nazwy wybranych figur.

W programie można wyróżnić następujące obszary:

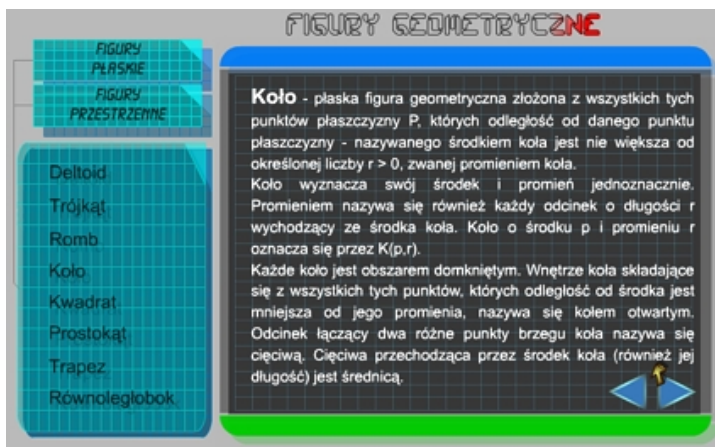
- główne menu programu zawierające dwa przyciski *Figury płaskie* i *Figury przestrzenne*,
- podmenu z przyciskami umożliwiającymi otwarcie obszaru z treścią: pole do obliczeń i opis danej figury geometrycznej,
- obszar z treścią z przyciskami *Opis Figury* i *Oblicz*.

Na rysunku 2.2 przedstawiono wygląd programu po kliknięciu z podmenu figur płaskich przycisku *Koło*. Po kliknięciu przycisku *Opis figury* otworzy się okno z opisem

wybranej figury (rysunek 2.3) z dwoma przyciskami służącymi do nawigacji pomiędzy stronami opisu.



Rys. 2.2. Interfejs użytkownika programu Figury geometryczne – okno figury płaskie: Koło.

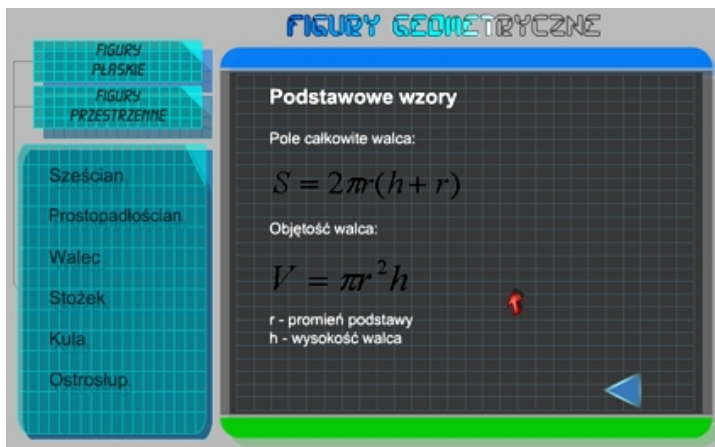


Rys. 2.3. Interfejs użytkownika programu Figury geometryczne – okno figury płaskie – opis figury koło.

Na rysunku 2.4 przedstawiono wygląd programu po kliknięciu z podmenu figur przestrzennych przycisku *Ostrosłup*. Po kliknięciu przycisku *Opis figury* otworzy się okno z opisem wybranej figury. Na rysunku 2.5 przedstawiono opis walca.



Rys. 2.4. Interfejs użytkownika programu Figury geometryczne  
– okno Figury przestrzenne: Ostrosłup.



Rys. 2.5. Interfejs użytkownika programu Figury geometryczne  
– okno Figury przestrzenne – opis figury walec

## Obliczenia przeprowadzone w programie Figury geometryczne

W celu przeprowadzenia obliczeń na figurach geometrycznych należy uzupełnić odpowiednie pola celem nadania wartości zadeklarowanych zmiennych (długość boku, promienia lub wysokości). Wciśnięcie przycisku *Oblicz* spowoduje, iż program wykona odpowiednie działanie i zwróci w wyniku obliczoną wartość: pola powierzchni, obwodu, objętości.

Na rysunku 3.1 przedstawiono okno kuli z przeprowadzonymi obliczeniami.



Rys. 3.1. Obliczenia pola całkowitego i objętości kuli.

## Podsumowanie

Program Figury geometryczne z powodzeniem można zaliczyć do nowej generacji multimedialnych interaktywnych metod nauczania. Dzięki możliwościom programu Flash MX i wbudowanemu językowi skryptowemu ActionScript Figury geometryczne są w pełni interaktywnym programem multimedialnym, wyróżniającym się intuicyjnym interfejsem, atrakcyjną grafiką i animacją. Multimedialność programu wpływa na jakość przekazu informacji, natomiast interaktywność zdecydowanie zwiększa aktywność ucznia w procesie nauczania. Proces nauczania z wykorzystaniem multimedialnego oprogramowania staje się bardziej obrazowy i interesujący, ponieważ

różni się od tradycyjnych systemów nauczania. Wykorzystanie zasady interaktywności umożliwia szybsze i skuteczniejsze przyswajanie wiedzy.

Program Figury geometryczne może być w przyszłości rozbudowany o dodatkowe figury geometryczne zgodnie z wymogami przyszłych użytkowników.

### Literatura

1. Kobyliński Ł., 2007, Oprogramowanie do nauki geometrii, praca inżynierska WSA, Łomża
2. Reinhardt R., Snow D., 2004, Macromedia Flash MX 2004 Biblia, Wydawnictwo Helion, Gliwice
3. Derek F, Jobe M., 2004, Flash MX 2004 actionscript – Oficjalny podręcznik, Wydawnictwo Helion, Gliwice
4. Bargiel D., Flash MX 2004, 2004, Ćwiczenia praktyczne, Wydawnictwo Helion, Gliwice
5. Ulrich K., 2004, Po prostu Flash MX 2004, Helion, Gliwice
6. Shane E., 2004, *Flash MX 2004 Skuteczne rozwiązania*. Helion. Gliwice

### Abstract

The paper describes the computer program Geometric Figures, which is a multimedia aid for learning geometry in primary school and secondary school. Apart from the theoretical bases of the basic plane figures and 3D figures, computer program Geometric Figures comprises also computer module for calculating areas, perimeters and volumes of them. Usage of the principle of the interaction enables learning geometry faster and more effective.

dr inż. Tomasz Huścio

Wyższa Szkoła Agrobiznesu w Łomży

e-mail: tomek@pb.edu.pl



# **Przekształcenie Hilberta w przetwarzaniu wybranych sygnałów biomedycznych**

*Edward Oczeretko*

## **Streszczenie**

Bardzo ważnym zagadnieniem jest badanie synchronizacji dwóch lub więcej jednocześnie rejestrowanych sygnałów biomedycznych. Przy pomocy przekształcenia Hilberta można szacować wskaźnik synchronizacji fazowej  $\gamma_H$  obliczany bezpośrednio z różnicy faz oraz inny wskaźnik  $\gamma_{H-Sh}$  szacowany z entropii Shannona rozkładu różnicy faz. Badano sygnały reprezentujące aktywność skurczową macicy. Zmiany tej aktywności często towarzyszą różnym patologiom położniczym i schorzeniom ginekologicznym. Analizowano także sygnały reprezentujące trzy wybrane parametry u pacjenta cierpiącego na bezdech (częstotliwość akcji serca w uderzeniach na minutę, częstotliwość oddechową oraz stężenie tlenu we krwi). Potwierdzono użyteczność przekształcenia Hilberta do badań synchronizacji.

## **Wprowadzenie**

Sygnał biomedyczny otrzymywany przez przyrządy pomiarowe jest odbiciem, bądź jak niektórzy twierdzą „cieniem” procesu, który go wywołuje. Badając sygnał otrzymujemy informacje o procesie. Przykładowymi sygnałami tego typu są sygnały EEG, EKG, czy EMG. Sygnały biomedyczne charakteryzują się zazwyczaj dużym stopniem złożoności, są one dalekie od stacjonarności. Badanie synchronizacji wielu jednocześnie rejestrowanych sygnałów biomedycznych jest ważne zarówno z teoretycznego jak i praktycznego punktu widzenia. Zmiany wartości parametrów synchronizacji mogą być bardzo użytecznymi parametrami diagnostycznymi. Ogólnie metody badania synchronizacji dzielimy na liniowe i nieliniowe. Do metod liniowych należą przykładowo: korelacja wzajemna, koherencja [Kitlas 2009]. Wśród metod nieliniowych znajdujemy entropię przybliżoną wzajemną, informację wzajemną, wymiar korelacyjny wzajemny i nieliniowe współzależności [Borowska 2010, Oczeretko 2010]. Ważną rolę w badaniu zjawisk synchronizacji odgrywa liniowe przekształcenie Hilberta.

Proste i odwrotne przekształcenia Hilberta są określone wzorami:

$$\hat{x}(t) = H[x(t)] = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{x(\tau)}{t-\tau} d\tau, \quad 1$$

$$x(t) = H^{-1}[\hat{x}(t)] = -\frac{1}{\pi} \int_{-\infty}^{\infty} \frac{\hat{x}(\tau)}{t-\tau} d\tau. \quad 2$$

Powyższe całki są rozumiane w sensie wartości głównej Cauchy'ego, ponieważ, dla  $t = \tau$  funkcje podcałkowe w tych wzorach są nieograniczone.

Wzory powyższe możemy zapisać za pomocą splotów:

$$\hat{x}(t) = \frac{1}{\pi t} * x(t),$$

$$x(t) = -\frac{1}{\pi t} * \hat{x}(t).$$

Korzystając z definicji splotu możemy przecież pierwsze z dwóch powyższych równań zapisać w postaci:

$$\hat{x}(t) = \frac{1}{\pi t} * x(t) = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{x(\tau)}{t-\tau} d\tau = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{x(t-\tau)}{\pi\tau} d\tau.$$

Weźmy pod uwagę proste przekształcenie Hilberta. Wykonajmy transformatę Fouriera obu jego stron, wykorzystując przy tym twierdzenie o splotcie (transformata Fouriera splotu dwóch funkcji jest iloczynem ich transformat). W rezultacie:

$$F[\hat{x}(t)] = F\left[\frac{1}{\pi x}\right] F[x(t)].$$

Ponieważ [Toland, 1997]:

$$F\left[\frac{1}{\pi x}\right] = -i \operatorname{sgn}(\omega),$$

gdzie  $\operatorname{sgn}(f)$  jest tzw. funkcją znaku:

$$\operatorname{sgn}(\omega) = \begin{cases} -1 & \text{dla } \omega < 0 \\ 1 & \text{dla } \omega \geq 0 \end{cases},$$

więc:  $F[\hat{x}(t)] = -i \operatorname{sign}(\omega) \cdot F(\omega)$ .

Przy pomocy przekształcenia Fouriera możemy obliczać transformatę Hilberta. Po pomnożeniu transformaty Fouriera sygnału przez  $-i \operatorname{sign}(\omega)$  i wykonaniu

odwrotnego przekształcenia Fouriera otrzymanego wyrażenia otrzymujemy przekształcenie Hilberta:

$$\hat{x}(t) = F^{-1}[-i \operatorname{sign}(\omega) \cdot F(\omega)].$$

### Własności przekształcenia Hilberta

Niech  $\hat{x}(t)$  i  $\hat{y}(t)$  będą przekształceniami Hilberta sygnałów  $x(t)$  i  $y(t)$

$$\hat{x}(t) \leftrightarrow x(t)$$

$$\hat{y}(t) \leftrightarrow y(t)$$

Do podstawowych właściwości przekształcenia Hilberta należą:

1. Własność podobieństwa  $x(at) \leftrightarrow \hat{x}(at)$ ;
2. Własność przesunięcia  $x(t-t_0) \leftrightarrow \hat{x}(t-t_0)$ ;
3. Liniowość  $H[ax(t)+by(t)] \leftrightarrow a\hat{x}(t)+b\hat{y}(t)$ ;
4. Przekształcenie Hilberta sygnału parzystego jest sygnałem nieparzystym i odwrotnie;
5.  $\hat{\hat{x}} = -x$ ;
6. Splot  $x(t)*y(t) \leftrightarrow \hat{x}(t)*y(t) = x(t)*\hat{y}(t)$ ;  $x(t)*y(t) = -\hat{x}(t)*\hat{y}(t)$ ;
7. Równość iloczynów skalarnych  $(x, y) = (\hat{x}, \hat{y})$ , czyli  $\int_{-\infty}^{\infty} x(t)y^*(t)dt = \int_{-\infty}^{\infty} \hat{x}(t)\hat{y}^*(t)dt$ ;
8. Równość energii (mocy)  $(x, x) = (\hat{x}, \hat{x})$ ;
9. Dla sygnałów rzeczywistych ortogonalność sygnału i transformaty  $(\hat{x}, \hat{y}) = 0$ ;
10. Równość funkcji autokorelacji  $\int_{-\infty}^{\infty} x(t)x^*(t-\tau)dt = \int_{-\infty}^{\infty} \hat{x}(t)\hat{x}^*(t-\tau)dt$ .

Przekształcenie Hilberta jest związane z przesunięciem fazy analizowanych sygnałów o  $90^\circ$ . Stąd też sygnał sinusoidalny staje się cosinusoidalnym i na odwrót. Poniżej mamy kilka przykładowych par transformat Hilberta (par, ponieważ po lewej stronie mamy sygnał, po prawej jego przekształcenie):

$$\sin \omega_0 t \leftrightarrow -\cos \omega_0 t$$

$$\cos \omega_0 t \leftrightarrow \sin \omega_0 t$$

$$\delta(t) \leftrightarrow \frac{1}{\pi t}$$

$$\text{rect}(t) \leftrightarrow \frac{1}{\pi} \ln \left| \frac{t + \frac{1}{2}}{t - \frac{1}{2}} \right|.$$

Przekształcenie Hilberta znalazło bardzo szerokie zastosowanie w analizie sygnałów biomedycznych: EEG [Etevenon 1999], EKG [Ivanov 1996, Singh 2010], EGG – elektrogastrogramów [Liang 2000], ECoG – elektrokortykogramów [Kozma 2008]. Zastosowania wiążą się z pojęciami sygnału analitycznego (analytic signal), fazy chwilowej i częstotliwości chwilowej IF (instantaneous frequency). Elementarnym wstępem do tych zagadnień są prace Boashasha [Boashash 1992a, 1992b]. W pracach tych znajdziemy rys historyczny rozwoju pojęcia częstotliwości chwilowej jak również algorytmy służące do jej oszacowania.

Na początek wprowadzimy pojęcie sygnału analitycznego:

$$z(t) = x(t) + iH[x(t)] = x(t) + i\hat{x}(t),$$

gdzie  $H[x(t)]$  jest operatorem transformaty Hilberta.

Sygnał analityczny  $z(t)$  jest to sygnał zespolony. Sygnał pierwotny  $x(t)$  stanowi jego część rzeczywistą. Transformata Hilberta sygnału pierwotnego tworzy część urojoną. Amplitudą chwilową (obwiednią) sygnału  $x(t)$  nazywamy moduł jego sygnału analitycznego:

$$A(t) = \sqrt{x(t) \cdot x(t) + \hat{x}(t) \cdot \hat{x}(t)}.$$

Faza chwilowa  $\varphi(t)$  to:

$$\varphi(t) = \arctan \left\{ \frac{\hat{x}(t)}{x(t)} \right\}.$$

Definicja częstotliwości chwilowej  $f(t)$  jest związana z szybkością zmian fazy chwilowej:

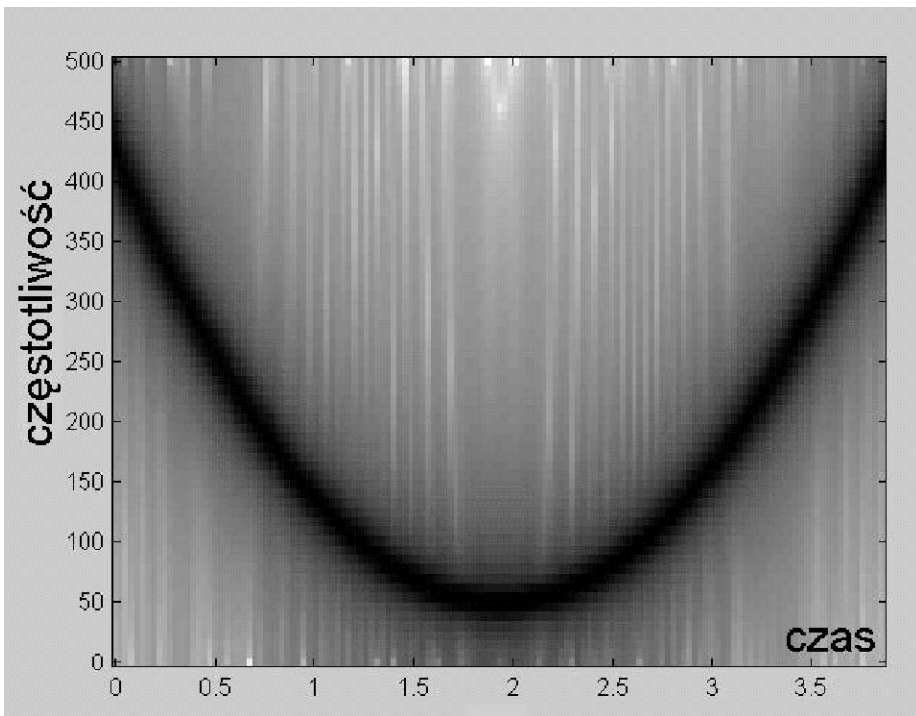
$$f(t) = \frac{1}{2\pi} \frac{d\varphi(t)}{dt}.$$

Szacowanie częstotliwości chwilowej dla danego sygnału wydaje się na pierwszy rzut oka bardzo proste. W MATLAB-ie mamy funkcję `hilbert`, która zastosowana do sygnału  $x$ , daje nam sygnał analityczny  $y = x + i\hat{x}$ . Jednakże zastosowanie instrukcji:

```
f = diff(angle(hilbert(x))),
```

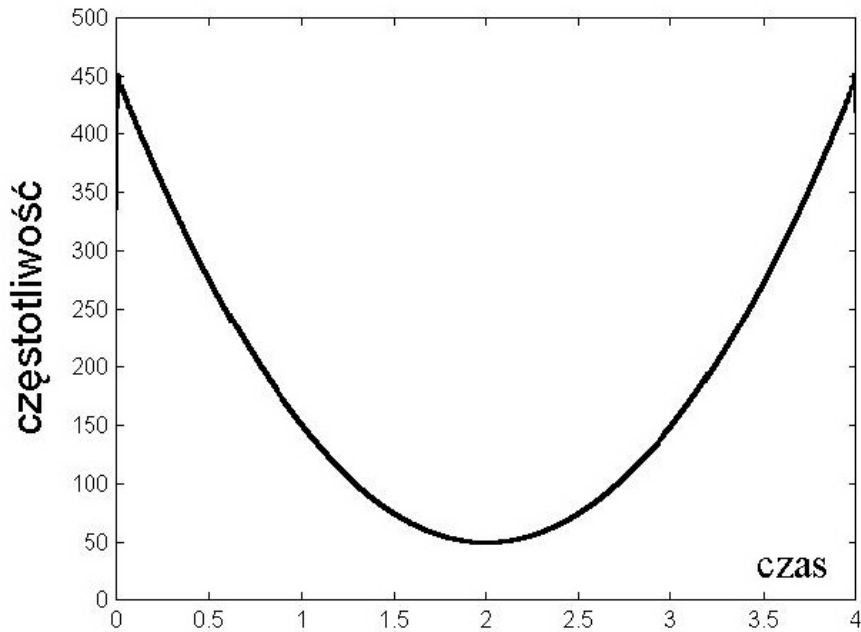
(gdzie funkcja `angle` zwraca kąt fazowy w radianach zawarty w przedziale od  $-\pi$  do  $\pi$ , natomiast `diff` zwraca wektor różnic sąsiednich elementów wektora będącego argumentem jej wywołania prowadzi) do nieoczekiwanych rezultatów. Nawet dla sygnałów sinusoidalnych otrzymujemy w tym przypadku bardzo nierównomierny przebieg częstotliwości chwilowej z pikami sięgającymi wartości ujemnych. Przyczyną tego jest szum, który wzmacniany jest przez operację `diff`. Dlatego też, jak sugeruje Kootsookos na stronie internetowej Feldmana poświęconej transformacji Hilberta (<http://hitech.technion.ac.il/~feldman/hilbert2.html>), należy tę operację przeprowadzić wewnątrz funkcji `angle`, tak jak to mamy w poniższej instrukcji:

```
f = angle(conv(ones(1,5),y(2:length(y)).*conj(y(1:length(y)-1))))).
```



Rysunek 1. Spektrogram sygnału o zmieniającej się kwadratowo częstotliwości.

Na rysunku 1 mamy spektrogram sygnału o zmieniającej się kwadratowo częstotliwości. Rysunek 2 ilustruje z kolei przebieg zmian częstotliwości chwilowej tego sygnału oszacowany przy zastosowaniu koncepcji sygnału analitycznego. Mimo, że częstotliwość chwilowa dostarcza nam informacji na temat wewnętrznej struktury sygnału, bardzo ważna rzecz jak lokalizacja energii w przestrzeni czas-częstotliwość jest tu poza naszym zasięgiem. Dla sygnału złożonego z kilku częstotliwości w wyniku zastosowania tej metody otrzymamy jedynie częstotliwość maksymalną, którą przyjmujemy za wartość częstotliwości chwilowej. W sygnale może przecież występować wiele różnych częstotliwości zmieniających się w czasie jego trwania. Należy, więc sygnał badany dekomponować na pewne składowe. W każdej z takich składowych występują jedynie częstotliwości z pewnego wąskiego przedziału częstotliwości sygnału. Jednym ze sposobów dekompozycji jest metoda falkowa, która nie wymaga stacjonarności i zachowuje fourierowską informację o fazie [Ivanow 1996]. Dla każdego sygnału składowego jest obliczana częstotliwość chwilowa przy wykorzystaniu idei sygnału analitycznego. Huang i wsp. [Huang 1998] wprowadzili bardzo interesujący algorytm tzw. empirycznego rozkładu sygnału na mody (EMD – empirical mode decomposition). Lai [Lai 1998] badał przejście do chaosu deterministycznego w układach dynamicznych w kategoriach sygnału analitycznego. Badany sygnał dekomponował przy pomocy metody EMD na skończoną liczbę składowych. Stwierdził, że częstości chwilowe w obszarze chaotycznym są zawarte w pewnym stosunkowo wąskim paśmie, w przeciwieństwie do szerokiego pasma spektrum Fouriera.



Rysunek 2. Częstotliwość chwilowa otrzymana po zróżniczkowaniu fazy chwilowej sygnału analitycznego.

Przekształcenie Hilberta można łatwo uogólnić na przypadek dwuwymiarowy. Proste i odwrotne przekształcenia wyrażają się wtedy wzorami:

$$H[f(x, y)] = \hat{f}(x, y) = \int_{-\infty}^{\infty} dy' \int_{-\infty}^{\infty} \frac{f(x', y')}{(x - x')(y - y')} dx'$$

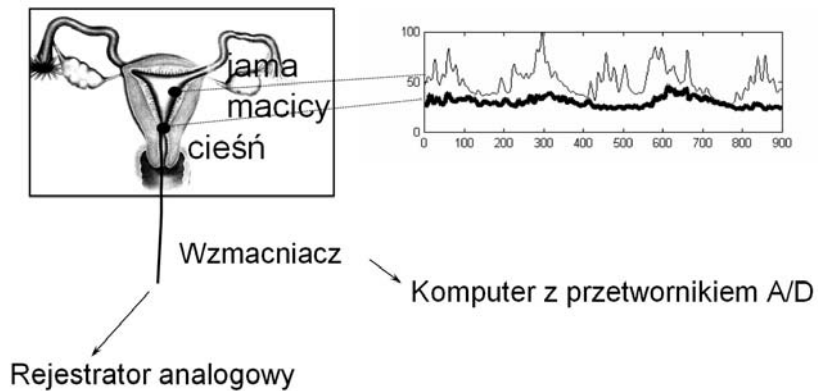
$$f(x, y) = H^{-1}[\hat{f}(x, y)] = \int_{-\infty}^{\infty} dy' \int_{-\infty}^{\infty} \frac{\hat{f}(x', y')}{(x - x')(y - y')} dx'.$$

2 - D transformata Hilberta znalazła zastosowanie w medycynie w analizie mikroskopowych obrazów preparatów histologicznych. Wizualnie prawie nierozróżnialne, lub często wręcz niewidoczne przezroczyste obiekty znajdujące się w takim preparacie wpływają na fazę (modulacja fazy) przechodzącej przez nie fali świetlnej. Z tego powodu noszą one nazwę obiektów fazowych. Przedmiot wprawdzie jest przezroczysty, ale lokalne zmiany jego grubości czy współczynnika załamania muszą mieć wpływ i mają na fazę. Zastosowanie przekształcenia Hilberta pozwala na wizualizację takich obiektów. Potwierdzili to Arnison i wsp. [Arnison 2000] stosując,

dwuwymiarowe przekształcenie Hilberta dla mikroskopowych obrazów przedstawiających chromosomy. W pracy tej możemy znaleźć także dokładny opis algorytmu realizującego to podejście.

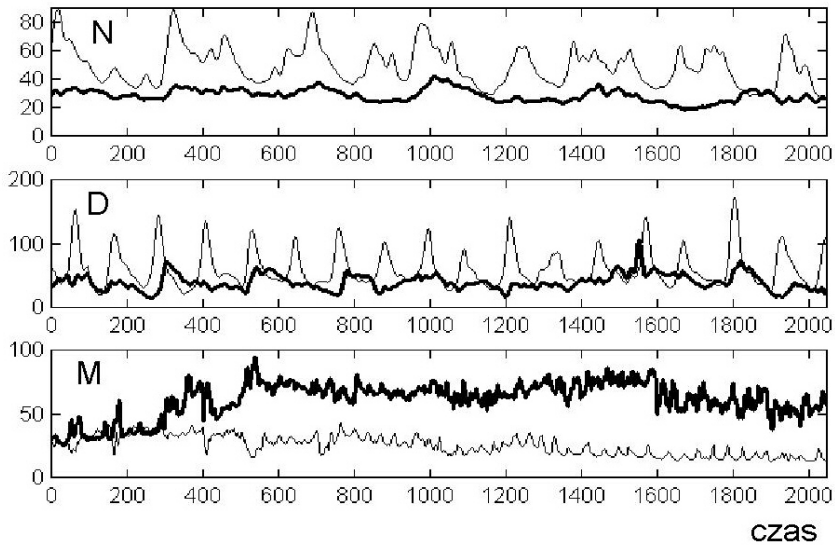
### Material i metody

Analizowane sygnały reprezentujące czynność skurczową macicy zostały otrzymane metodą pomiaru ciśnienia wewnątrzmacicznego. Badania zostały zaakceptowane przez Komisję etyczną Uniwersytetu Medycznego w Białymstoku i zostały przeprowadzone w Klinice Perinatologii UMB w Białymstoku. Układ pomiarowy pokazany na rysunku 3 składał się z cewnika z czujnikami, wzmacniacza i komputera. Mamy tu równoczesny zapis z cieśni i jamy macicy.



Rysunek 3. Zapis ciśnienia wewnątrzmacicznego cewnikiem z mikroprzetwornikami ciśnienia (macica nieciążerna). Równoczesny zapis sygnałów z cieśni i jamy macicy.



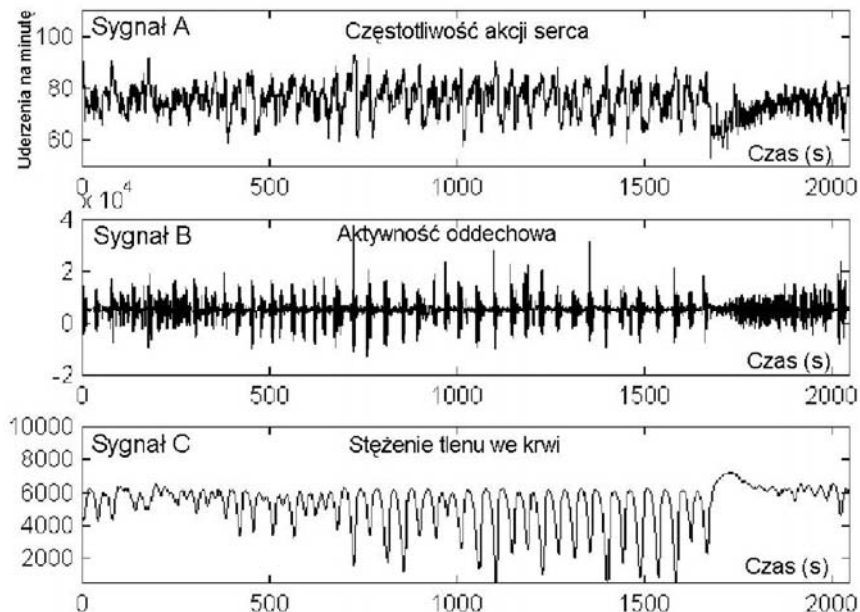


Rysunek 4. Sygnały reprezentujące czynność skurczową macicy uzyskane od:  
 N) kobiety zdrowej, D) kobiety cierpiącej na bolesne miesiączkowanie,  
 M) kobiety cierpiącej na mięśniaki. Linią pogrubioną oznaczono sygnał z cieśni macicy.

Na cewniku znajdowały się w odległości 30 mm dwa czujniki tensometryczne – ciśnieniowe. Czujniki produkowały sygnały elektryczne o zakresie napięciowym  $-5V$ ,  $+5V$ , o napięciu proporcjonalnym do rejestrowanego ciśnienia. Sygnał kierowano do komputera IBM PC z 12 bitowym przetwornikiem analogowo-cyfrowym – karta Advantech PCL – 711S posiadająca 8 kanałów pomiarowych. Do akwizycji służył program GENIE firmy Advantech. Częstość próbkowania wynosiła 2 Hz (można ją oczywiście zmieniać stosownie do potrzeb badania). Jeden czujnik mierzył ciśnienie na dnie macicy, drugi czujnik był umieszczony w pobliżu szyjki macicy. Uzyskano trzy zapisy (rysunek 4): zapis N pochodzący od kobiety zdrowej, zapis D – od pacjentki z bolesnym miesiączkowaniem i zapis M od pacjentki cierpiącej na mięśniaki macicy.

Przy pomocy Physionetu (<http://www.physionet.org>) uzyskano (rysunek 5): sygnał A reprezentujący częstotliwość akcji serca w uderzeniach na minutę, sygnał B reprezentujący częstotliwość oddechową (można ją mierzyć zmianami objętości klatki piersiowej) i sygnał C przedstawiający stężenie tlenu we krwi zmierzone przy pomocy usznej oksymetrii. Wartości sygnałów B i C podane są w jednostkach

niewykalibrowanych. Wszystkie trzy szeregi czasowe były jednocześnie rejestrowane z częstotliwością próbkowania 2 Hz u pacjenta cierpiącego na bezdech. Bezdech polega na przerwach w oddychaniu w czasie snu. W ciągu nocy mogą być setki przerw, których długość może wynosić nawet kilkadziesiąt sekund. Długości sygnałów wynoszą 17000 próbek. Do naszej analizy wybraliśmy trzy szeregi 4096 elementowe.



Rysunek 5. Badane sygnały pochodzące od pacjenta cierpiącego na bezdech.

Sygnał A – częstotliwość akcji serca, sygnał B aktywność oddechowa,  
sygnał C – stężenie tlenu we krwi.

Przy pomocy przekształcenia Hilberta szacowano wskaźniki synchronizacji fazowej  $\gamma_H$  obliczany bezpośrednio z różnicy faz, oraz wskaźnik synchronizacji fazowej  $\gamma_{H-Sh}$  szacowany z entropii Shannona rozkładu różnicy faz. Różnicę faz obliczano przy pomocy wzoru:

$$\varphi_{xy}(t) = n\varphi_x(t) - m\varphi_y(t), \text{ dla } n = 1 \text{ i } m = ,$$

gdzie  $\varphi_x(t)$  i  $\varphi_y(t)$  - to fazy chwilowe dla sygnałów  $x$  i  $y$ .

Wskaźnik  $\gamma_H$  szacowano przy pomocy wzoru;

$$\gamma_H = \left\langle \left| e^{ip_{xy}^H(t)} \right| \right\rangle_t$$

gdzie symbol  $\langle \rangle$  oznacza średnią.

Wskaźnik  $\gamma_{H-sh}$  obliczano przy pomocy równania:

$$\gamma_{H-sh} = \frac{S_{max} - S}{S_{max}}$$

$$\text{gdzie: } S = -\sum_{k=1}^N p_k \cdot \ln p_k \quad i \quad S_{max} = \ln N$$

Obliczenia przeprowadzono w środowisku MATLAB, korzystając z pakietu do przetwarzania sygnałów (Signal Processing Toolbox).

## Rezultaty

Wyniki obliczeń przedstawiono w Tabelach 1 i 2.

Tabela 1.

Wartości wskaźników synchronizacji par sygnałów pochodzących od:  
 N) kobiety zdrowej, D) kobiety cierpiącej na bolesne miesiączkowanie,  
 M) kobiety cierpiącej na mięśniaki.

1.	Sygnaly	2.	$\gamma_H$	3.	$\gamma_{H-sh}$
4.	N	5.	0,5551	6.	0,5169
7.	D	8.	0,8198	9.	0,7632
10.	M	11.	0,3685	12.	0,3762

Tabela 2.

Wartości wskaźników synchronizacji par sygnałów pochodzących od pacjenta cierpiącego na bezdech: (Sygnał A – częstotliwość akcji serca, sygnał B aktywność oddechowa, sygnał C – stężenie tlenu we krwi).

13.	Pary sygnałów	14.	$V_H$	15.	$V_{H-Sh}$
16.	A-B	17.	0,8271	18.	0,6315
19.	A-C	20.	0,9444	21.	0,6889
22.	B-C	23.	0,7920	24.	0,6220

### Dyskusja i wnioski

Badanie synchronizacji wielu jednocześnie rejestrowanych sygnałów jest ważne zarówno z teoretycznego jak i praktycznego punktu widzenia. Dotyczy to wielu typów sygnałów EEG, aktywności skurczowej macicy, sygnałów elektrokortykograficznych, sygnałów kardiokortykograficznych, elektromiograficznych i elektrohisteryograficznych. Nieprawidłowości w synchronizacji w przypadku skurczów macicy towarzyszą różnym patologiom położniczym i schorzeniom ginekologicznym jak: poród przedwczesny, bolesne miesiączkowanie, bezpłodność, endometrioza, mięśniaki, nowotwory. Dla sygnałów EEG praktycznie każde zaburzenie (padaczka, omdlenia, choroba Alzheimera) charakteryzuje się zmianami w synchronizacji [Borowska, 2010]. W naszej pracy największe wartości wskaźników synchronizacji rejestrowano u pacjentki cierpiącej na bolesne miesiączkowanie ( $\langle V_H \rangle = 0.8198$ ,  $\langle V_{H-Sh} \rangle = 0.7632$ ), zaś najniższe u pacjentki z mięśniakami ( $\langle V_H \rangle = 0.3685$ ,  $\langle V_{H-Sh} \rangle = 0.3762$ ). Różnica jest bardzo wyraźna. W przypadku pacjenta cierpiącego na bezdech najwyższe wartości współczynników były dla pary sygnałów A i C i dość wysokie dla wszystkich par sygnałów. Oznacza to wysoki stopień zależności między tymi sygnałami.

Powyższe przykłady potwierdzają użyteczność przekształcenia Hilberta do badań synchronizacji sygnałów medycznych.

### Literatura

1. Arnison M.R., Cogswell C.J., Smith N.I., Fekete P.W., Larkin K.G. Using the Hilbert transform for 3D visualization of differential interference contrast microscope images. *J. Microsc.* 2000. 199. 79-84.
2. Boashash B (a). Estimating and Interpreting the instantaneous frequency of a signal-Part 1: Fundamentals. *Proceedings of the IEEE.* 1992. 80(4). 520-538.
3. Boashash B (b). Estimating and Interpreting the instantaneous frequency of a signal-Part 2: Algorithms and applications. *Proceedings of the IEEE.* 1992. 80(4). 539-568.
4. Borowska M, Oczeretko E., Sobaniec P., Sobaniec W. Nonlinear synchronization analysis of the EEG signals. W: C.T. Lim I J.C.H. Goh (ed.) *WCB 2010, IFMBE Proceedings*, 2010, 31, 1362-1365.
5. Etevenon P., Lebrun N., Clochon P., Perchey G., Eustache F., Baron J-C. High temporal resolution dynamic mapping of instantaneous EEG amplitude modulation after tone-burst auditory stimulation. *High temporal resolution dynamic mapping of instantaneous EEG amplitude modulation after tone-burst auditory stimulation.* 1999. 12(2), 129-137.
6. Huang N.E., Shen Z., Long S.R., Wu M.L. C., Shih H.H., Zheng Q.N., Yen N.C., Tung. The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis. *Proc. Roy. Soc. London, Ser A.* 1998. 454. 903-995.
7. Ivanov P.Ch., Rosenblum M.G., Peng C.-K., Mietus J., Havlin S., Stanley H.E., Goldberger A.R. Scaling behaviour of heartbeat intervals obtained by wavelet-based time-series analysis. *Nature.* 1996. 383. 323-327.
8. Kitlas A., Oczeretko E., Świątecka J. Borowska M., Ludański T.: Uterine contraction signals – application of the linear synchronization measures, *Eur. J. Obstet. & Gynecol. Reprod. Biol.* 2009, 144S, S61-S64.
9. Kozma R., Freeman W.J., Intermittent spatio-temporal desynchronization and sequenced synchrony in ECoG signals, *Chaos*, 2008, 18(3), 037131.
10. Lai Y-C. Analytic signals and the transition to chaos in deterministic flows. *Phys. Rev. E.* 1998. 58. 6911-6914.

11. Liang H., Lin Z., McCallum R.W. Artifact reduction in electrogastrogram based on empirical mode decomposition method. *Med. Biol. Eng. Comput.* 2000. 38. 35-41.
12. Oczeretko E., Borowska M., Kitlas A., Laudański P., Laudański T. Synchronization in the bivariate intrauterine pressure signals – nonlinear dynamics methods, *Reproductive Sciences*, 2010, 17(7), 667-672
13. Singh H.R., Sharma R., Sahgal N., Sethi P., Kushwah R., Kachhawa P. An improved method of measurement of ECG parameters for online medical diagnosis. *Stud Health Technol Inform.* 2010, 156, 40-46.
14. Toland J.F. A few remarks about the Hilbert transform. *J. Funct. Anal.* 1997. 145. 151-174.

### **Abstract**

In physiological research, we often study multivariate data sets, containing two or more simultaneously recorded time series. It is important to examine synchronization in these kinds of signals. By means of Hilbert transform we can assess phase synchronization index  $\gamma_H$  and other synchronization measure  $\gamma_{H-Sh}$  directly calculated from Shannon entropy of the phase difference distribution. Spontaneous uterine contraction activity was recorded. Analysis of synchronization of such time series may have potential research importance in the biology and pathophysiology of uterine contractility. The signals recorded from the patient in the sleep apnea also were studied (heart rate variability, respiration rate and blood oxygen concentration). The usefulness of the Hilbert transform in the examination of the synchronization of biomedical signals was confirmed.

prof. nadzw. dr hab. Edward Oczeretko  
Wyższa Szkoła Agrobiznesu w Łomży  
e-mail: eddoczer@ii.uwb.edu.pl

# **Tworzenie multimedialnych materiałów dydaktycznych - przegląd istniejących rozwiązań**

*Wiesław Póljanowicz*

## **Streszczenie**

W publikacji przedstawiono przykłady tworzenia multimedialnych materiałów dydaktycznych z wykorzystaniem darmowych, dostępnych aplikacji istniejących na rynku programów do zastosowań multimedialnych. Przeanalizowano zasadę działania wybranych aplikacji do tworzenia obiektów multimedialnych (filmy, dźwięki) oraz interaktywnych (animacje, podręczniki). Zaprezentowano programy do nagrywania sekwencji wideo z pulpitu komputera, obróbki audio i wideo, tworzenia interaktywnych krzyżówek i podręczników multimedialnych.

**Słowa kluczowe:** materiał multimedialny, obróbka audio/wideo, interaktywność, nagrywanie sekwencji wideo,

## **Wprowadzenie**

W aspekcie kształcenia studentów coraz częściej sięgamy po nowoczesne metody i środki (np. komputer, przekaz multimedialny), aby ułatwić sobie zadania związane z nowoczesną edukacją. Elektroniczne formy przekazu treści multimedialnych są istotnym elementem uzupełniającym wiedzę praktyczną studentów. Zainteresowanie takim modelem kształcenia jest, między innymi spowodowane coraz szerszym dostępem do aplikacji, za pomocą których możemy w dość prosty sposób tworzyć multimedialne materiały do dydaktyki i nie tylko, wykorzystywane zarówno przy prowadzeniu zajęć jak i przy poszerzeniu zakresu wiedzy ogólnej. Motorem rozwoju tej formy przekazu jest też Internet jako źródło zasobów multimedialnych wykorzystywanych w szeroko pojętej edukacji.

Przygotowane multimedialne materiały dydaktyczne potrafią przedstawić np. szereg złożonych procesów dynamicznych z dużą dokładnością i z bardzo dużą wiernością w stosunku do ich rzeczywistego przebiegu. Dotyczy to praktycznie każdej

dziedziny wiedzy. Przygotowany multimedialny materiał dydaktyczny posiada wiele zalet między innymi: selektywny wybór odpowiednich partii materiału, wiedza jest przedstawiona na przykładach w postaci ilustracji, animacji, wykresów, dodatkowych opisów i wyjaśnień [Półjanowicz 2005] co skutkuje szybszym przyswojeniem jej przez studenta. Istotnym elementem jest możliwość kilkakrotnego powtórzenia wybranego fragmentu materiału co pozwala osobie uczącej się lepiej ugruntować wiedzę z danego zakresu tematycznego. Kolejną zaletą nauczania wspomaganego komputerem jest to, iż osoba ucząca się przyswaja sobie zadany materiał w kolejności i tempie określonym przez nią samą. Prowadzi to do nauki samodzielnej i wysoce zindywidualizowanej [Półjanowicz 2005].

Wszystkie te zmiany powodują, że nauczanie z wykorzystaniem technik multimedialnych zyskuje coraz większą popularność. Obecnie na rynku można spotkać bardzo dużo narzędzi do tworzenia multimedialnych materiałów, które możemy wykorzystać w edukacji. Są to aplikacje do nagrywania sekwencji wideo z pulpitu komputera wykorzystywane w celu tworzenia demonstracji i pokazów, programy do: obróbki audio i wideo, tworzenia animacji Flash, programy umożliwiające przygotowanie interaktywnego podręcznika, gier i krzyżówek.

### **Przegląd aplikacji do tworzenia materiałów multimedialnych**

#### ***CamStudio***


Jest to darmowy program służący do nagrywania sekwencji wideo z pulpitu komputera głównie w celu tworzenia demonstracji. Wraz z sekwencjami wideo możemy także nagrać dźwięk z mikrofonu. Program rejestruje wszystkie operacje wykonywane przez użytkownika - ruchy kursorem myszy, uruchamianie aplikacji, wpisywanie tekstu, również wyłącznie na wydzielonej części pulpitu. Wersja 2.0 dodatkowo umożliwia konwersję generowanych standardowo plików AVI do animacji w formacie Flash (SWF) (rys. 1).





Rys 1. Okno główne programu CamStudio.

*Źródło: Opracowanie własne.*

Nagrywanie ekranu realizuje się poprzez wybranie opcji Record lub przycisku nagrywania  po wcześniejszym wyborze obszaru nagrywania. Aby ustawić obszar nagrywania należy z menu programu wejść w zakładkę Region (rys. 2).



Rys 2. Wybór obszaru nagrywania programie CamStudio.

*Źródło: Opracowanie własne.*

Program umożliwia wybór trybu nagrywania: nagrywanie z podkładem dźwiękowym lub bez. Aby ustalić ustawienia audio należy z menu programu wybrać zakładkę Options. Pojawia się wówczas okno opcji programu. Po wybraniu „Do not record audio” dźwięk nie jest rejestrowany.

Przy wyborze opcji „Record audio from microphone” nagrywany jest dźwięk z mikrofonu, a przy wyborze „Record audio from speakers” z głośników komputera (rys. 3).



Rys 3. Wybór trybu nagrywania programie CamStudio.


*Źródło: Opracowanie własne.*

Program ma możliwość dodawania adnotacji do nagrywanego materiału. Aby wstawić adnotacje, czyli tzw. „chmurki” na pulpit należy z menu programu wybrać zakładkę Tools, a następnie opcje Screen Annotations (rys. 4).



Rys 4. Okno wyboru adnotacji wraz z dostępnymi adnotacjami programie CamStudio.

*Źródło: Opracowanie własne.*

Program oferuje nam nagrywanie filmów w dwóch formatach: AVI oraz SWF. W standardowo ustawione jest nagrywanie w formacie AVI. Jeśli chcemy zmienić format nagrywania na Flash (SWF) należy kliknąć na ikonkę  znajdującą się w głównym oknie programu.

Wówczas w głównym oknie powinien pojawić się napis „Record to SWF” (rys. 5).

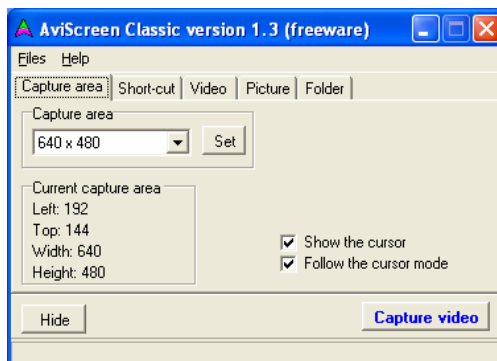


Rys 5. Ustawienie nagrywania plików w formacie Flash (swf) w programie CamStudio.

*Źródło: Opracowanie własne.*

### ***AviScreen Classic***

Jest to bezpłatna aplikacja, która umożliwi nagrywanie tego co dzieje się na pulpicie lub wybranej części ekranu do plików wideo w formacie AVI. W przypadku nagrywania wyłącznie określonego obszaru ekranu program oferuje unikalną i bardzo przydatną funkcję "follow the cursor" (podążaj za kursorem), dzięki czemu przesuwając kursor muszy automatycznie przesuwany jest także obszar nagrywania. Niestety program nie oferuje nagrywania dźwięku, posiada jednak opcję wykonywania zrzutów ekranowych (rys. 6).

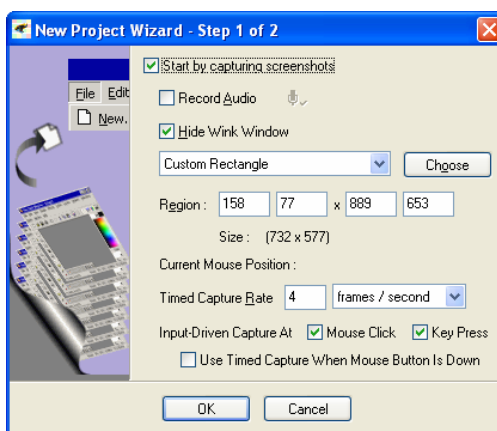


Rys 6. Okno główne programu AviScreen Classic.

*Źródło: Opracowanie własne.*

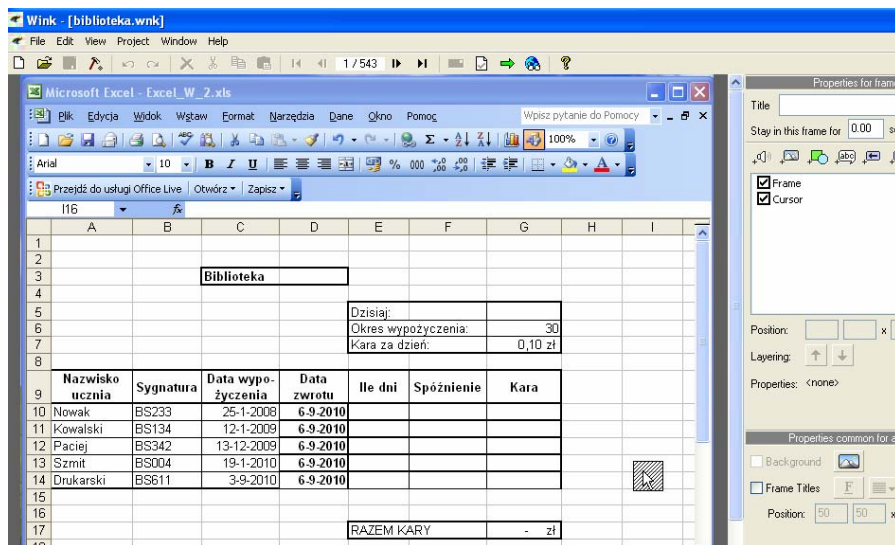
### **Wink**

Jest to bezpłatny program służący do tworzenia demonstracji szkoleniowych, prezentacji (rys. 7). Przy pomocy Winka można stworzyć sekwencję pojedynczych zrzutów ekranowych z wybranego obszaru Pulpit (lub z całego ekranu), dodawać tzw. dymki z komentarzami, przyciski, nagrywać głos lektora. Program pozwala na utworzenie animacji w formacie Flash (SWF) jak i w postaci wykonywalnego pliku EXE, który może być dystrybuowany w celu pobierania na dysk twardy komputera (rys. 8). Wink umożliwia także na tworzenie plików PDF lub HTML.



Rys 7. Okno nowego projektu w programie Wink 2.0.

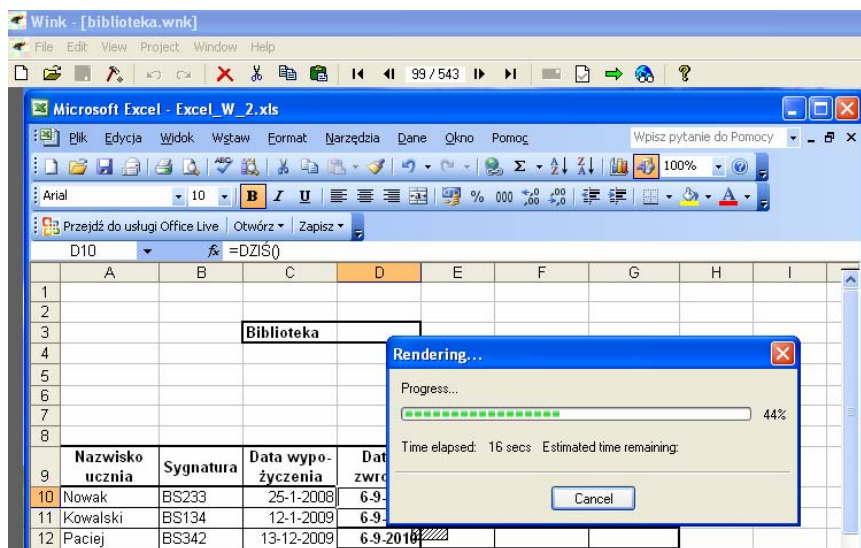
*Źródło: Opracowanie własne.*



Rys 8. Okno główne programu Wink 2.0.

*Źródło: Opracowanie własne.*

Po nagraniu materiału mamy możliwość edycji tego materiału (kopiowanie, usuwanie, duplikowanie pojedynczych sekwencji (klatek). Ważnym elementem programu jest dodawanie objaśnień w postaci opisu tekstowego, ustawianie interaktywnych przycisków sterujących pokazem, wstawianie grafiki. Po edycji materiału wideo przechodzimy do opcji renderowania materiału → (rys.9).



Rys 9. Okno renderowania (przetwarzania) informacji w programie Wink 2.0.

*Źródło: Opracowanie własne.*

Następnym krokiem jest podgląd obiektu po renderowaniu oraz zapis do wcześniej wybranego formatu wyjściowego np. exe czy swf.

### **VirtualDub**

Rozbudowany program do liniowej obróbki i kompresji filmów w formatach avi, mpeg czy choćby XviD (można sobie wybrać własny kodek). Podstawowe możliwości programu to zrzut dowolnych klatek do pliku, zrzut ścieżki dźwiękowej do \*.wav, wycinanie i wstawianie dowolnych scen z filmów, łączenie plików filmowych, konwersja filmów ze starych kodeków na nowe, zmiana rozdzielczości na wyższą, zmiana formatu dźwięku, konwersje plików wideo, możliwość wstawiania efektów do filmu (rys. 10).



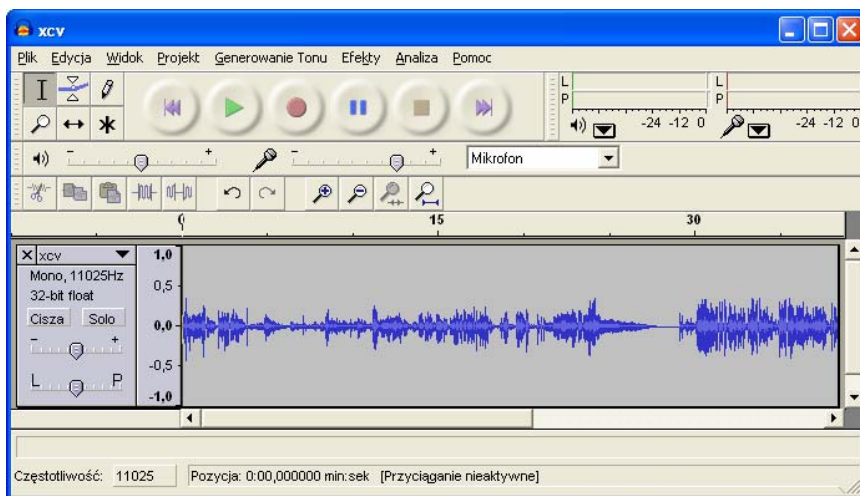
Rys 10. Okno główne programu VirtualDub 1.9.7.

*Źródło: Opracowanie własne.*

VirtualDub jest programem wykorzystywanym również do przechwytywania obrazu wideo jako narzędzie wykorzystywane do 32-bitowych i 64-bitowych platform Windows (98/ME/NT4/2000/XP/Vista/7), na podstawie licencji GNU General Public License (GPL).

### **Audacity**

Audacity jest darmowym i łatwym w użyciu programem do obróbki dźwięku (rys. 11). Jego główne zadania to nagrywanie i odtwarzanie dźwięków, import i eksport plików w formatach MP3, WAV, AIFF, Ogg Vorbis i innych. Umożliwia on także edycję ścieżek dźwiękowych z wykorzystaniem wycinania, kopiowania i wklejania, miksowania ścieżek oraz dodawania efektów (echo, usuwanie szumu, normalizacja itp.).



Rys 11. Okno główne programu Audacity.

*Źródło: Opracowanie własne.*

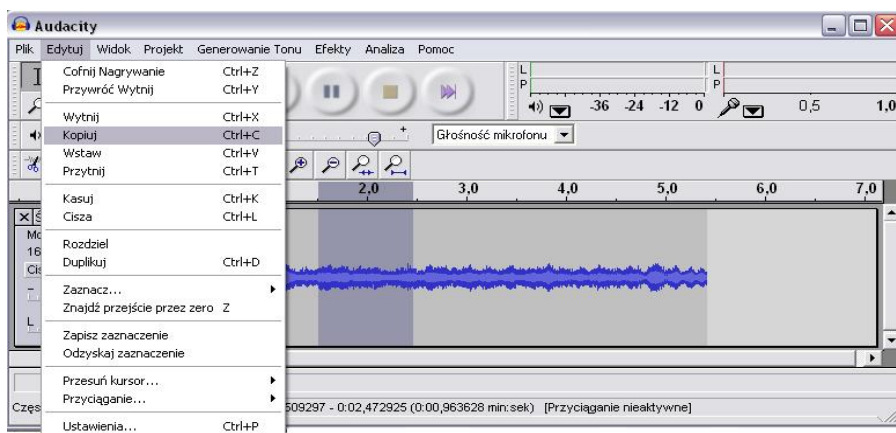
Menu programu składa się z 8 zakładek:

- Plik - W tej zakładce dostępne są podstawowe operacje takie jak zapis, odczyt, wydruk, utworzenie nowego projektu oraz eksport plików w formatach MP3, WAV, AIFF, Ogg Vorbis
- Edytuj – Zakładka zawiera opcje edycji projektu: wycinanie, kopiowanie, wklejanie, usuwanie, zaznaczanie, cofanie i przywracanie operacji, a także rozdzielanie i duplikowanie ścieżki dźwiękowej oraz ustawienia programu
- Widok – Dzięki opcjom oddalanie, przybliżania, dopasowania do okna, dopasowania w pionie użytkownik może wybrać najbardziej odpowiadający mu widok
- Projekt – W tej zakładce zawarte są opcje związane z tworzeniem projektu takie jak import pliku audio, dodawanie, usuwanie oraz wyrównywanie ścieżek dźwiękowych
- Generowanie Tonu – Dostępne są opcje Biały szum, Silence (cisza) oraz Ton, która generuje ton o podanej przez użytkownika długości, częstotliwości i amplitudzie



- Efekty – Zakładka umożliwia dodawanie efektów do ścieżki dźwiękowej np. usuwanie szumu, echo, normalizację, wyciszanie, wzmacnianie, zmianę tempa, prędkości i wysokości
- Analiza – Umożliwia szczegółową analizę częstotliwości ścieżki dźwiękowej np. poprzez narysowanie widma
- Pomoc – W tej zakładce znajdziemy funkcje pomocy przydatne użytkownikowi

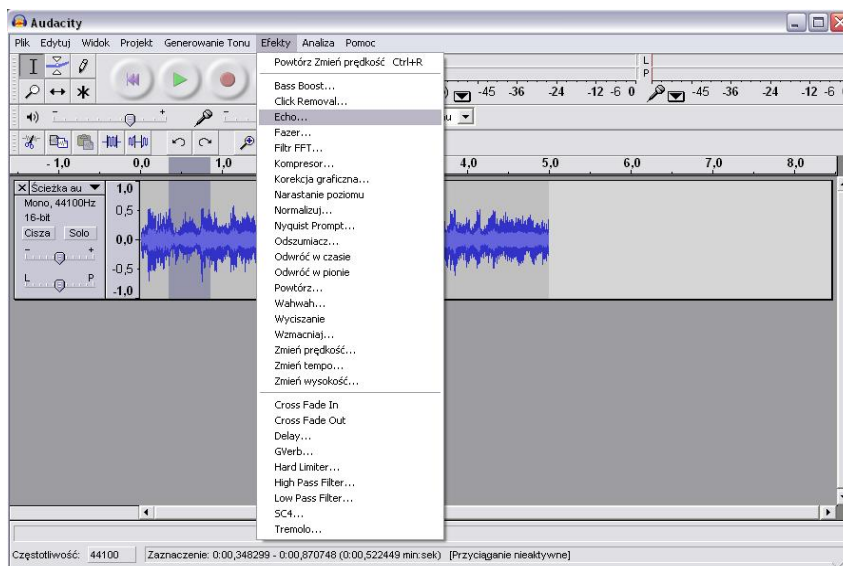
Za pomocą programu Audacity można edytować ścieżki dźwiękowe za pomocą takich opcji jak kopiowanie, wycinanie i wklejanie oraz usuwanie (rys. 12).



Rys 12. Kopiowanie fragmentu ścieżki w programie Audacity.

*Źródło: Opracowanie własne.*

W programie można dodawać różne efekty. Aby dodać do ścieżki dźwiękowej np. efekt echa należy zaznaczyć fragment ścieżki a następnie wybrać z menu Efekty opcję Echo (rys. 13).



Rys 13. Dodawanie efektu echo do fragmentu ścieżki w programie Audacity.

*Źródło: Opracowanie własne.*

Można dodawać wiele innych efektów widocznych na (rys. 13) np. zmiana prędkości. Pojawia się wówczas okno zmiany prędkości. Znajduję się w nim suwak, którym możemy regulować procentową zmianę prędkości (rys. 14). Opcja Podgląd umożliwia wysłuchanie tego jakby brzmiał dźwięku po dokonaniu zmiany. Aby zaakceptować zmianę prędkości wciskamy przycisk OK..

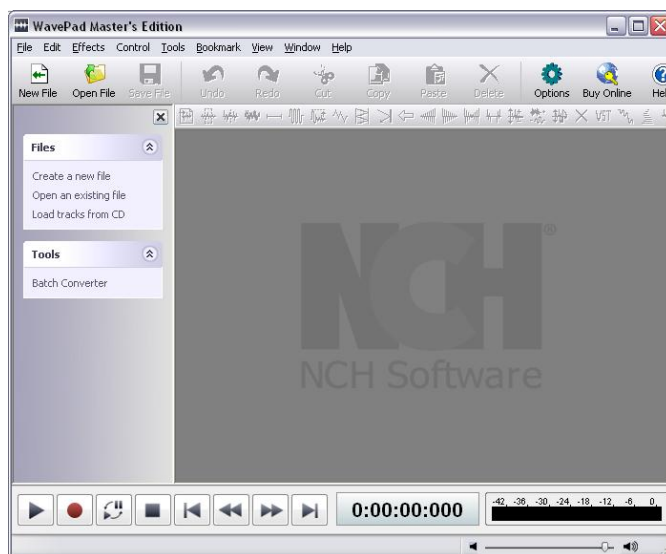


Rys 14. Zmiana prędkości fragmentu ścieżki w programie Audacity.

*Źródło: Opracowanie własne.*

### ***WavePad Sound Editor***

Jest bezpłatnym prostym programem umożliwiającym tworzenie i edycję plików dźwiękowych w formatach wav, mp3, vox, gsm, real audio, au, aif, flac, ogg i wielu innych. Umożliwia m.in. wycinanie fragmentów i łączenie nagrań, miksowanie kilku utworów, dodanie do utworu echa, redukcję szumów, wzmacnianie, wyciszanie itp. Posiada ponadto możliwość zgrywania utworów z płyt CD-Audio oraz nagrywania stworzonych utworów na płytę CD. Program może służyć także jako narzędzie do konwersji plików pomiędzy różnymi formatami (rys. 15).

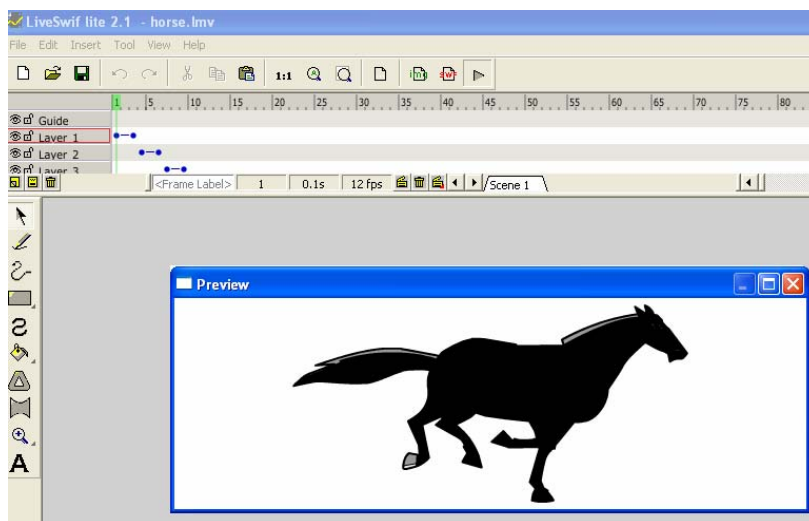


Rys 15. Okno główne programu WavePad.

*Źródło: Opracowanie własne.*

### ***Liveswif Lite***

Program Liveswif Lite to bezpłatne oprogramowanie do tworzenia animacji Flash, za pomocą którego można generować animowane przyciski czy ruchome obrazki. Do programu dołączona została baza gotowych grafik. Możemy także wczytywać własne pliki graficzne oraz dołączać pliki dźwiękowe (rys.16).

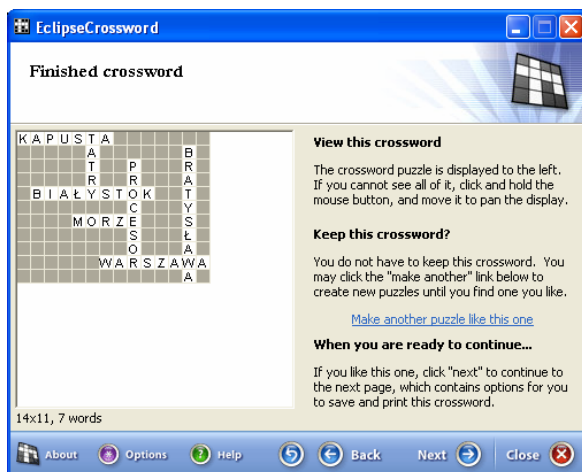


Rys 16. Okno główne programu Liveswift Lite.

*Źródło: Opracowanie własne.*

### ***EclipseCrossword***

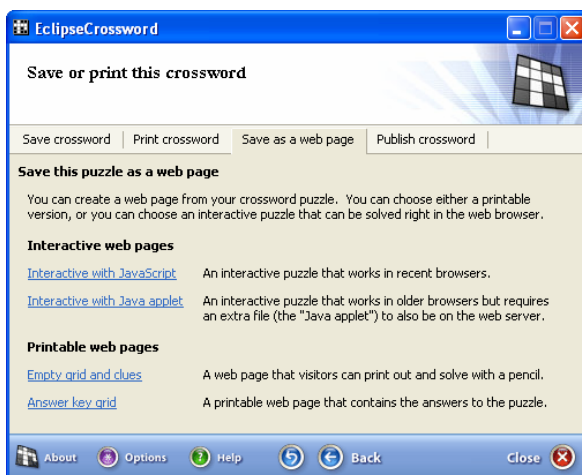
W programie tym można przygotowywać i tworzyć krzyżówki, które możemy umieszczać na stronach WWW jak również w prezentacjach multimedialnych. Budując krzyżówkę tworzymy listę nowych słów (opcja zaznaczona domyślnie). Zaznaczając drugą opcję wykorzystamy wcześniej przygotowaną krzyżówkę. Następnie wpisujemy listę słów wraz z ich opisem oraz tytuł krzyżówki ewentualnie jej twórcę. W kolejnym kroku powinniśmy określić wymiary krzyżówki w pionie i poziomie. Program planuje sam rozmieszczenie układu słów w krzyżówce, które możemy zmienić automatycznie (rys. 17).



Rys 17. Rozmieszczenie słów w krzyżówce Eclipse Crossword.

*Źródło: Opracowanie własne.*

Gotową krzyżówkę zapisujemy jako stronę internetową wybierając zakładkę *Save as web page*, a następnie *Interactive with JavaScript* (rys.18).



Rys 18. Zapisanie krzyżówki jako strony internetowej .

*Źródło: Opracowanie własne.*

### Quandary

Program Quandary firmy *Half Baked* umożliwia przygotowanie interaktywnego podręcznika, coś w rodzaju kursu na zadany temat, mający formę pytań i odpowiedzi. Student przechodzi przez nie jak przez labirynt, dotąd aż udzieli wszystkich poprawnych odpowiedzi. Można w ten sposób na bieżąco sprawdzać zrozumienie kolejnych partii materiału (rys.19).



Rys 19. Przykład wykorzystania programu Quandary .

Źródło: <http://www.hlw-dl.at/content/maltaexperience/>.

### Materialy i metody

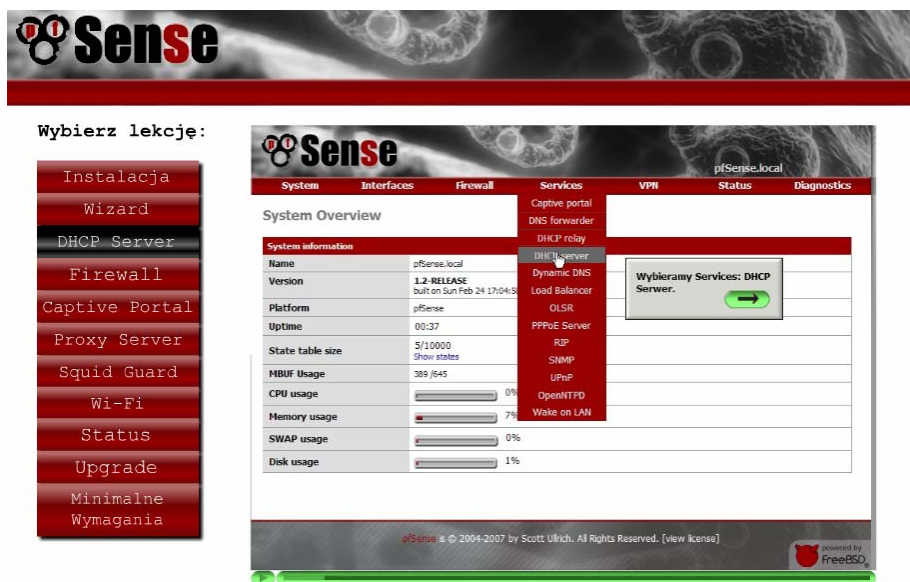
Badanie programów do zastosowań multimedialnych przeprowadzono pod względem ich użyteczności i funkcjonalności w pozyskiwaniu i obróbce elementów multimedialnych. Opisano zasadę działania programów oraz podstawowe funkcje użyteczne przy tworzeniu obiektów multimedialnych. Do analizy wykorzystano głównie darmowe programy służące do nagrywania sekwencji wideo z pulpitu komputera takie jak: CamStudio 2.0, AviScreen Clasic 1.3, Wink 2.0.

W zakresie liniowej obróbki wideo i kompresji filmów badania przeprowadzono na przykładzie programu Virtual Dub. Natomiast programy do nagrywania i edytowania

dźwięku, które zostały wykorzystane do opisu to: Audacity oraz Wave Pad Sound Editor (oba freeware). Przedstawiono również przykłady zastosowań innych programów tworzących krzyżówki i interaktywne podręczniki.

## Rezultaty

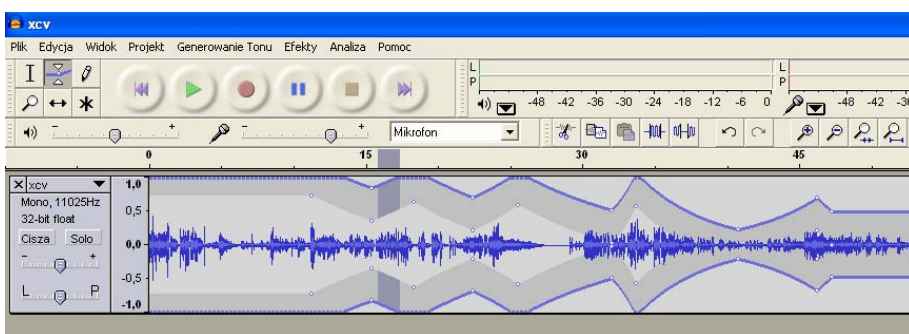
W grupie programów bezpłatnych do nagrywania sekwencji wideo z pulpitu komputera najlepiej wypadł program Wink 2.0. Posiada on dość rozbudowany lecz przyjazny interfejs, ma spore możliwości techniczne oraz takie funkcjonalności jak: tworzenie sekwencji pojedynczych zrzutów ekranowych z wybranego obszaru pulpitu (lub całego ekranu), dodawanie „chmurek” z komentarzami, przyciski akcji, tytuły, nagrywanie głosowe lektora. Przygotowaną demonstrację można zapisać w jednym z kilku formatów. Program pozwala na utworzenie animacji w formacie Flash (swf), którą można umieścić na stronie WWW oraz w postaci wykonywalnego pliku EXE. Wink umożliwia także tworzenie plików PDF lub HTML, co pozwala na tworzenie prezentacji oraz dokumentacji technicznej np. instrukcji obsługi danego oprogramowania (rys. 20).



Rys 20. Okno utworzonej dokumentacji technicznej za pomocą programu Wink 2.0.

Źródło: Opracowanie własne.

W analizie funkcjonalnej programów służących do nagrywania i zaawansowanej obróbki dźwięków największymi możliwościami charakteryzuje się darmowy program Audacity. Posiada on następujące funkcje: nagrywanie dźwięku z zewnętrznych źródeł między innymi z mikrofonu, wejścia linowego, odczyt i zapis plików w wielu formatach audio np. WAV, AIF, AU, Ogg Vorbis. Eksport do plików MP3. Edycja plików audio z wykorzystaniem wycinania, kopiowania i wklejania, miksowania ścieżek oraz dodawania różnych efektów np. echo, usuwanie szumu, normalizacja, wyciszenie i wzmacnianie, zmiana prędkości itp. (rys.21).

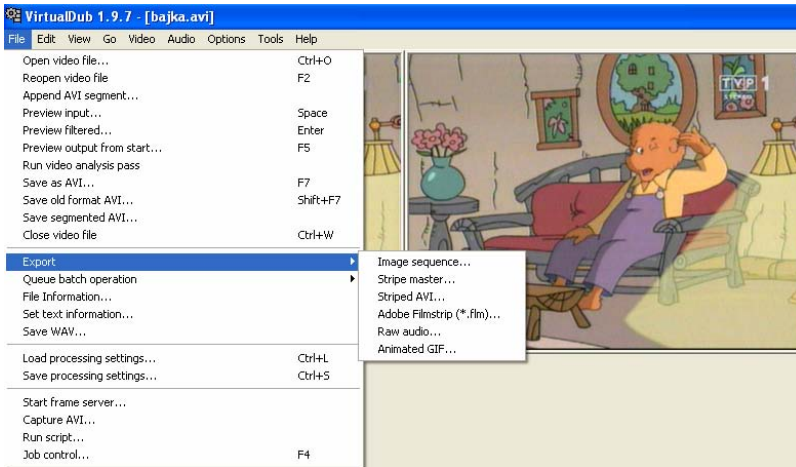


Rys 21. Przykład wykorzystania narzędzia obwiedni w programie Audacity.

*Źródło: Opracowanie własne.*

Niezastąpiony w obróbce liniowej i kompresji filmów jest bezpłatny program Virtual Dub lub jego polska wersja Virtual Dub Mod. Możliwości jakie prezentuje program to zrzut dowolnych klatek do pliku \*.bmp, \*.tga, \*.png, zrzut ścieżki dźwiękowej do pliku \*.wav, wycinanie, kopiowanie i wstawianie dowolnych scen z filmów, łączenie plików filmowych, konwersja filmów ze starych kodeków na nowe, zmiana rozdzielczości, konwersje plików wideo, możliwość wstawiania efektów (filtrów) do filmu (np. blur more, emboss, grayscale, resize, rotate, sharpen, smoother TV, itd.), zmiana kontroli szybkości klatek wideo, przechwytywanie obrazu (capture AVI), eksport materiału do różnych formatów zapisu pliku (rys. 22).

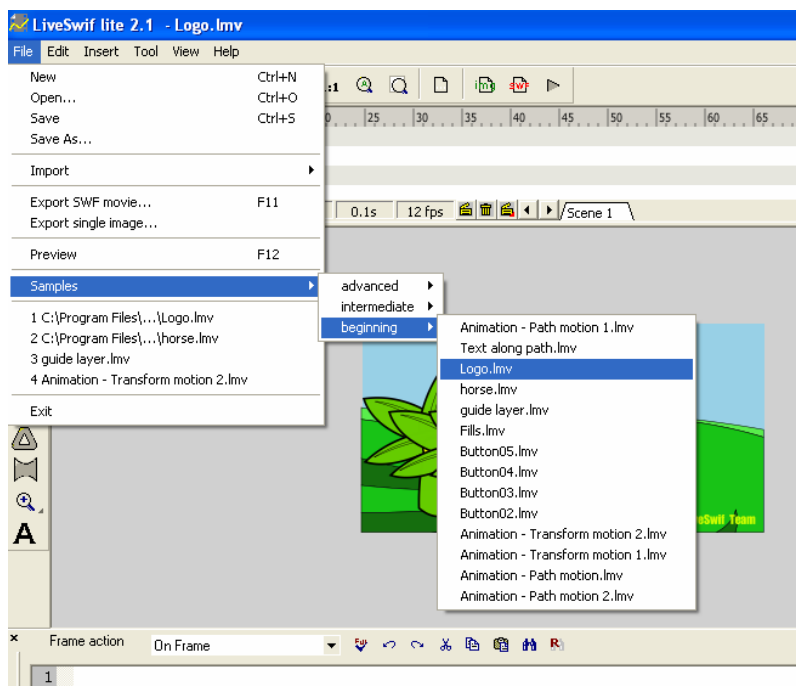




Rys. 22. Eksport materiału filmowego do różnych formatów zapisu pliku.

*Źródło: Opracowanie własne.*

Programem bezpłatnym do tworzenia animacji Flash został wybrany Liveswift Lite. Jest to aplikacja, za pomocą której można generować animowane przyciski czy ruchome obrazki. Można dokonywać składu obiektów multimedialnych wczytując własne pliki graficzne, dołączając pliki dźwiękowe oraz korzystać z gotowej bazy grafik, którą posiada też ta aplikacja (rys. 23).



Rys. 23. Wczytanie przykładowej grafiki z gotowej bazy grafik w programie Liveswift Lite.

Źródło: Opracowanie własne.

## Dyskusja i wnioski

Tradycyjne metody nauczania oparte o wzorzec „nauczyciel-uczeń”, przeprowadzone w klasycznych warunkach sali wykładowej, mają swoją ugruntowaną od wieków pozycję. Rozwój mediów elektronicznych oraz nowoczesnych środków przekazu informacji umożliwia wprowadzenie do zajęć multimedialnych treści i form nauczania, w których wykorzystujemy animację danego zjawiska, film pokazowy, demonstrację działania zjawiska, dźwięki (muzyka, nagrania lektorskie), tekst [Atras2005]. Przegląd zaprezentowanego oprogramowania oraz analiza jego funkcjonalności pozwoli na lepszy dobór poszczególnych aplikacji do tworzenia multimedialnych materiałów dydaktycznych, które mogą być wykorzystywane przez osoby prowadzące zajęcia w praktycznie dowolnej dziedzinie. Przygotowując multimedialne materiały dydaktyczne zwiększamy atrakcyjność i poziom merytoryczny

zajęć. Popularność form multimedialnych może wynikać z interaktywnego dostępu do wiedzy [Bremer2002]. Studenci mają możliwość wielokrotnie zapoznać się z wcześniej analizowanym zagadnieniem, prześledzić go dokładnie, lepiej go przyswoić. Istotą publikacji jest wskazanie dróg realizacji multimedialnych form przekazu informacji i środków jakimi można to uczynić. Może to stanowić ważny element w nauczaniu przedmiotu Techniki multimedialne na studiach zawodowych z informatyki. Studenci tworząc aplikacje multimedialne mogą wykorzystać wyżej omawiane programy w celu pozyskiwania obiektów multimedialnych, które stanowią będą content projektowanej aplikacji.

Aplikacje multimedialne zawierające animacje komputerowe, sekwencje wideo, dźwięki, przedstawiające elementy przesuwające się po ekranie, zmieniające swą postać, kształt i kolor wraz z odpowiednim stanem zachodzącego zjawiska (procesu) z odpowiednimi komentarzami to bogactwo multimediiów.

Najważniejszym elementem jest interaktywność użytkownika na przebieg pokazu, prezentowanego zjawiska [Huścio2004]. To stanowi o sile takich aplikacji multimedialnych. Oprócz tego ważnym elementem jest tu atrakcyjna forma przekazywanych treści, co pozwala uczącym się osiągnięcie pozytywnej motywacji do nauki, a twórcom materiałów multimedialnych większą pomysłowość w realizacji oraz skuteczność nauczania niż przy zastosowaniu tradycyjnych metod.

### **Literatura**

1. Półjanowicz W., Citko U., 2005, Wykorzystanie „wirtualnych laboratoriów” w edukacji studentów, Gdynia, VII Ogólnopolskie Forum SNTI, 26-31.
2. Atras T., 2005, Multimedialne wspomaganie nauczania. Szkolna biblioteka multimedialna, Gdynia, VII Ogólnopolskie Forum SNTI, 66- 69.
3. Huścio T. , Kuźmierowski T., 2004, Wirtualne laboratorium z napędów i sterowania pneumatycznego, Białystok, Heden Media, 2-15.
4. Bremer A., Sławik M., 2002, Technologia informacyjna z informatyką, Katowice, Videograf, 63-108.

### **Abstract**

The paper presents examples of teaching materials of multimedia using free, available applications existing on the market for multimedia uses. There was analyzed the principle of operation of selected applications used in creating multimedia objects (videos, sounds) and interactive (animation, manuals). Programs for video recording of computer desktop, working with audio and video, creating interactive crosswords and multimedia textbooks were presented.

mgr inż. Wiesław Póljanowicz  
Wyższa Szkoła Agrobiznesu w Łomży  
e-mail: wiepol@wp.pl

