

MODELOWANIE REALISTYCZNEJ PRZESTRZENI OPERACYJNEJ NA PODSTAWIE OBRAZÓW DICOM

MODELLING OF REALISTIC SURGICAL SPACE BASED ON DICOM IMAGES

Ryszard Leniowski*, Jakub Meissner

Politechnika Rzeszowska, Wydział Elektrotechniki i Informatyki,
Katedra Informatyki i Automatyki, 35-959 Rzeszów, al. Powstańców Warszawy 12

*e-mail: lery@prz-rzeszow.pl

STRESZCZENIE

Praca przedstawia metody modelowania realistycznej przestrzeni medycznej na podstawie danych diagnostyki obrazowej zgodnych ze standardem DICOM. Zawiera krótki przegląd współczesnych zastosowań trójwymiarowych wizualizacji w medycynie oraz opisuje metodę pozwalającą na odczytywanie zdjęć diagnostyki obrazowej zapisanych w plikach DICOM. Dla ciągu dwuwymiarowych obrazów następuje wyznaczenie konturów ludzkich narządów. Kontury łączy się względem osi Z i w efekcie otrzymuje się trójwymiarową siatkę obiektów w przestrzeni ograniczonej powierzchnią skóry. Jednym z tych obiektów jest również przestrzeń operacyjna. Implementację komputerową metody wykonano w języku C++ przy wykorzystaniu bibliotek SFML, SFGUI, Boost, DCMTK oraz OpenGL. Aplikacja odczytuje ciągi zdjęć procedury diagnostycznej, a następnie określa granice interesującego obszaru, wylicza wierzchołki węzłowe siatki modelu trójwymiarowego i renderuje scenę.

Słowa kluczowe: przetwarzanie obrazów medycznych, DICOM, modelowanie ciała człowieka, oprogramowanie medyczne

ABSTRACT

The aim of this work was to develop methods that are able to model realistic surgical site, using DICOM-compliant images. The theoretical issues include significance of 3D visualizations in modern medicine for diagnostic and surgical use. The practical part of this work has been focused on creating a C++ application, using SFML, SFGUI, DCMTK, boost, and OpenGL libraries for reading sequences of images stored in DICOM standard files, and then using them for tracing contours of human organs. Contours were then used for 3D visualization of surgical site. patient.

Keywords: medical image processing, DICOM, human body modelling, medical software

1. Wstęp

Modelowanie ludzkich narządów, staje się powoli podstawą skomplikowanych operacji, wymagających milimetrowej precyzji. Pozwala na nieinwazyjne rozpoznanie przestrzeni, którą chirurg wykonujący ten sam zabieg 30 lat wcześniej ujrzałby dopiero po przecięciu skóry np. brzucha. Wizualizacje są wykorzystywane w procedurach wykonywanych przy rdzeniu kręgowym, na sercu i innych czułych narządach. Realistyczne i dokładne wizualizacje wymagają jednak olbrzymiej ilości danych diagnostycznych. Po ich pozyskaniu są one zapisywane w plikach i bazach danych działających na podstawach

standardu DICOM (ang. *Digital Imaging and Communications in Medicine*), który można tłumaczyć jako Cyfrowe Obrazowanie i Komunikacja w Medycynie. Jest on absolutną podstawą medycznego obrazowania [1]. Pozwala on na pobieranie, przechowywanie i przetwarzanie informacji medycznych na tysiącach urządzeń, działających w szpitalach i klinikach całego świata.

Za początek historii wizualizacji 3D możemy uznać daty pojawienia się pierwszych urządzeń, będących w stanie zbierać cyfrowe dane diagnostyki obrazowej. Pierwsze komercyjne tomografy, wyprodukowane przez firmę EMI i zaprojektowane przez Godfreya Hounsfielda pojawiły się w 1972 roku. Dla magnetycznego rezonansu MRI, łatwiej wskazać datę i jest to rok 1980 kiedy firma FONAR zademonstrowała pierwszy komercyjny aparat wykorzystujący tę technologię.

Współczesne aparaty są w stanie wykonywać zdjęcia 320 przekrojów, pięć razy w ciągu sekundy. Przy takiej ilości danych i prędkości współczesnych komputerów, możliwe jest tworzenie filmów wizualizujących bijące serce w czasie rzeczywistym. Dzięki postępom modelowania ciała człowieka, możliwa jest diagnostyka osób, których postawa ciała nie pozwala na przeprowadzenie standardowych zabiegów. Dla wielu kobiet trudno uzyskać jasne wyniki przy tradycyjnej mammografii, ze względu na osteoporozę, zapalenie stawów lub inne schorzenia. Tomosynteza 3D umożliwia obrazowanie tkanki piersi pod różnymi kątami i uzyskując wyraźne obrazy dla różnych głębokości narządu. Kolejnym problemem diagnostycznym, który łatwiej obejść dzięki wizualizacjom 3D jest mierzenie przepływu krwi w żyłach i tętnicach, w szczególności u pacjentów po atakach serca lub wylewach. Połączenie modelowania wraz z tomografią rezonansu magnetycznego, umożliwia badanie przepustowości nawet pojedynczych naczyń krwionośnych. Przy tak szczegółowych informacjach problemu nie stanowi diagnozowanie ogólnego stanu układu krwionośnego, w tym serca.

2. Wybrany przegląd narzędzi i oprogramowania

Jednym z narzędzi w pełni korzystającym z postępów w wizualizacji ciała człowieka jest konfigurowalny aparat CT Toshiba *AquilionOne*. Pozwala on, w zależności od wersji, na wykonywanie 320 lub 640 zdjęć przekrojów przestrzeni zajmującej 8 lub 16 centymetrów, 3–4 razy na sekundę. Taka ilość informacji umożliwia dokładną wizualizację ciała człowieka z szybkim odświeżaniem zachodzących wewnątrz zmian. Dostarczane wraz z aparatem oprogramowanie udostępnia funkcje wizualizacji różnych grup tkanek ciała. Dla zmiennych potrzeb badania, możliwym jest generowanie modeli kości i naczyń krwionośnych, generowanie dodatkowych widoków na narządy zawierające w danej chwili kontrasty i odpowiednie ich oznaczanie.

Technologią łączącą informacje z tomografu komputerowego oraz USG w jednoczesnej wizualizacji 3D jest *eSieFusion*, wdrażana w ultrasonografach Siemens ACUSON S3000. Wczytują one model przestrzeni stworzony we wcześniejszym badaniu CT i wykorzystują go, jako swoistą mapę, oznaczając punkty kluczowe planowanego zabiegu. Ultrasonograf porównuje zaimportowaną wizualizację ze swoimi zbieranymi danymi, nakładając je na siebie. Dzięki takiemu rozwiązaniu (technologia hybrydowa), lekarz może w czasie rzeczywistym śledzić swoje manewry, dokonując procedury medycznej w mało inwazyjny sposób. Technologię tę można także wykorzystać podczas zabiegów ablacji.

Jednym z podstawowych narzędzi diagnostyki obrazowej są przeglądarki plików DICOM. Najczęściej używaną na świecie przeglądarką jest *OsiriX* i wspiera pełną integrację z procesami diagnostycznymi. Jest ona certyfikowana do przeprowadzania realnych badań medycznych i diagnostycznych. Aplikacja implementuje funkcje przetwarzania obrazów, zarówno w dwóch jak i trzech wymiarach, oraz narzędzia komunikacji pozwalające na łączenie do, lub tworzenie, baz danych standardu DICOM. Możliwym jest tworzenie autorskich „pluginów”, wspierających i rozbudowujących funkcje w zależności od potrzeb. Program jest dostępny w wersji darmowej, nie certyfikowanej do użytku medycznego. *OsiriX* oferuje wszelkie współczesne metody wizualizacji 3D badań, takie jak wielopłaszczyznowa rekonstrukcja (MPR), trójwymiarowe renderowanie zarówno przez wolumetryczne tekstuowanie jak i projekcję maksymalnej intensywności. Sceny utworzone za pomocą programu można eksportować, jako filmy i obrazy.

3. Generowanie rastrowych konturów warstw przestrzeni

Jednym z kluczowych etapów budowy trójwymiarowych obiektów geometrycznych na podstawie uporządkowanego ciągu płaskich obrazów jest generowanie konturów interesujących obiektów. Standardowe postępowanie polega na przetworzeniu wejściowego obrazu za pomocą operacji splotu, która w praktyce jest realizowana poprzez „przemiatanie” maską (macierz współczynników) wszystkich wierszy i kolumn i wyznaczeniu nowej wartości (np. koloru obrazu) na podstawie pikseli sąsiadujących [2]. Dla maski o rozmiarze 3×3 o współczynnikach:

$$\begin{bmatrix} p_1 & p_2 & p_3 \\ p_4 & p_5 & p_6 \\ p_7 & p_8 & p_9 \end{bmatrix} \quad (1)$$

gdzie elementem centralnym jest p_5 , wyjściowy obraz jest wyznaczany z równania:

$$\begin{aligned} I_{out}(m, n) = & I_{in}(m-1, n-1)p_1 + I_{in}(m-1, n)p_2 + I_{in}(m-1, n+1)p_3 + \\ & I_{in}(m, n-1)p_4 + I_{in}(m, n)p_5 + I_{in}(m, n+1)p_6 + \\ & I_{in}(m+1, n-1)p_7 + I_{in}(m+1, n)p_8 + I_{in}(m+1, n+1)p_9 \end{aligned} \quad (2)$$

W przypadku detekcji krawędzi wykorzystuje się filtrację kierunkową (horyzontalną i azymutalną) z dwoma maskami dla obu kierunków. W przypadku filtru Sobela maski (3×3) mają postać:

$$P_h = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}, \quad P_v = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \quad (3)$$

Zaś obraz wynikowy jest kwadratową średnią z filtrowanych obrazów I_{inh} , I_{inv} na obu kierunkach, czyli:

$$I_{out}(m, n) = \sqrt{I_{inh}^2(m, n) + I_{inv}^2(m, n)} \quad (4)$$

Zbliżone własności ma filtr Prewita o maskach:

$$P_h = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}, \quad P_v = \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix} \quad (5)$$

W celu uzyskania ostrych krawędzi można zastosować jedną z symetrycznych masek Laplace’a, np.:

$$P_1 = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}, \quad P_2 = \begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix}, \quad P_3 = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix} \quad (6)$$

Podane filtry pozwalają określić grubość ścianek narządów. W przypadkach przestrzeni operacyjnej wystarczy wyznaczyć jedynie jej krawędź, stąd w obliczeniach można zastosować filtr progowy, który wytwarza binarny obraz przestrzeni.

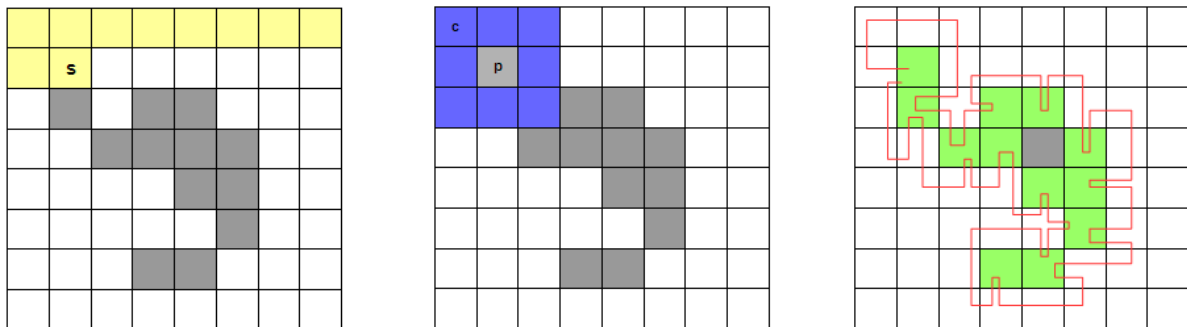
Określenie brzegu figury na binarnym obrazie dokonano za pomocą algorytmu Moore’a z modyfikacją Jacoba dla ośmiokierunkowego modelu sąsiedztwa [3].

Danymi wejściowymi algorytmu jest kwadratowa tessellacja \mathbf{T} zawierająca ciągły obiekt \mathbf{P} pikseli określonego koloru \mathbf{K} . Na wyjściu zwracany jest zbiór \mathbf{G} pikseli tworzących kontur. Zbiorem $\mathbf{M}(\mathbf{a})$

będziemy oznaczać sąsiedztwo Moore'a piksela **a**. Przez symbol **p** będziemy rozumieć rozważany element konturu, natomiast **c** będzie oznaczało obecnie rozważany element sąsiedztwa Moore'a tego punktu, tj. $M(p)$. Kolejnymi krokami algorytmu są:

- 1) Ustaw zbiór wyjściowy **G** na pusty.
- 2) Przesuwaj się w prawo a następnie w dół kolejnych elementów tessellacji **T** aż napotkasz piksel **s** o kolorze **K**. Dodaj go jako pierwszy element zbioru **G**.
- 3) Ustaw obecny punkt konturu **p** tak aby wskazywał na **s**. Wróć do piksela, z którego przesunąłeś się do piksela **s**.
- 4) Poruszając się zgodnie z ruchem wskazówek zegara po sąsiedztwie Moore'a $M(p)$ wybierz piksel **c**.
- 5) Tak długo jak **c** nie jest równe **s** i elementem wejścia do **s** nie jest ten sam co w punkcie 4:
 - Jeśli **c** ma kolor **K** dodaj je do zbioru **G**, ustaw **p** tak aby wskazywał na **c**, wróć do piksela, z którego przesunąłeś się do **p**.
 - w innym wypadku ustaw **c** na kolejny, rozważany w ruchu zgodnym z wskazówkami zegara, piksel w sąsiedztwie $M(p)$.

Na rysunku 1 pokazano kolejne etapy działania algorytmu. Żółtym kolorem zaznaczono szukanie pierwszej komórki **s** koloru **K**, kolor niebieski to sąsiedztwo Moore'a elementu **p**, wraz z pierwszym badanym pikselem **c**. Czerwona łamana reprezentuje ścieżkę badanych elementów, a zielone pola – wyznaczoną granicę.

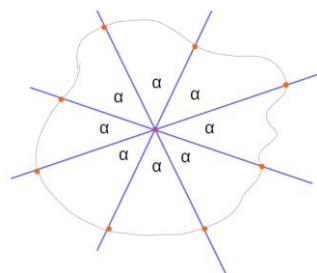


Rys. 1. Kolejne etapy działania algorytmu Moore'a

Algorytm można rozszerzyć na struktury zagnieżdżone. Wymagałoby to wprowadzenia do algorytmu mechanizmu usuwania przestrzeni już znalezionej. W konsekwencji należałoby dodać funkcję budowania dość skomplikowanej struktury drzewiastej, która zawierałaby informację o zagnieżdżeniach kolejnych figur. Do wyznaczenia przestrzeni operacyjnej wystarczającym okazało się zastosowanie mniej zaawansowanego wariantu algorytmu.

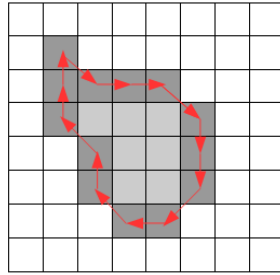
4. Transformacja rastrowych konturów do przestrzeni wektorowej

Transformacja rastrowych konturów do przestrzeni wektorowej opiera się na metodzie polegającej na wyznaczeniu geometrycznego środka ciężkości zamkniętego konturu i poprowadzeniu z niego pęku półprostych, co określony kąt, pokazany na rysunku 2.



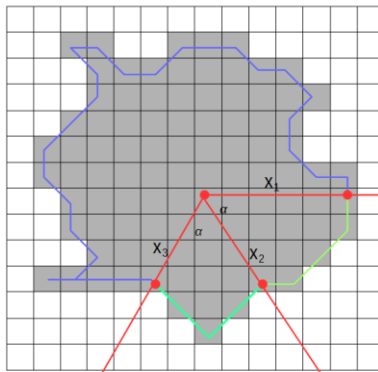
Rys. 2. Koncepcja wyznaczania punktów węzłowych

Z wyznaczonych punktów przecięć utworzono ciąg wektorów prowadzonych z każdego piksela granicy do jej następnego elementu, co pokazuje rysunek 3.



Rys. 3. Translacja granicy w postaci bitmapowej do postaci wektorowej

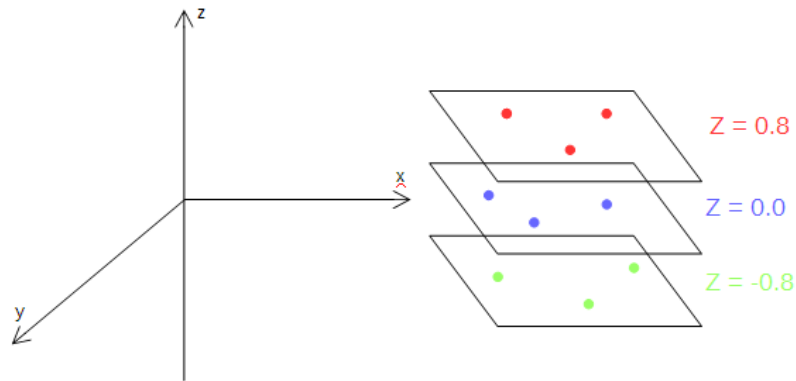
Ponieważ algorytm śledzenia sąsiedztwa Moore'a zwraca wierzchołki posortowane w kolejności ich wystąpienia, nie były wymagane dodatkowe operacje porządkujące. Aby nie sprawdzać istnienia wspólnych wierzchołków dla wszystkich wektorów i półprostych, wykorzystano własność algorytmu szukania granicy, dzięki której jej elementy były posortowane w kolejności zgodnej z ruchem wskazówek zegara. Zaczynając od półprostej prowadzonej pod kątem 0° , szukano punktu przecięcia dla kolejnych obiektów, zwiększano kąt o wartość wybraną przez użytkownika i iteracja po wektorach konturów przebiegała bez powrotu do wartości startowej. Proces ten pokazano na rysunku 4.



Rys. 4. Sprawdzanie punktów przecięcia zgodnie z ruchem wskazówek zegara dla kolejnych półprostych

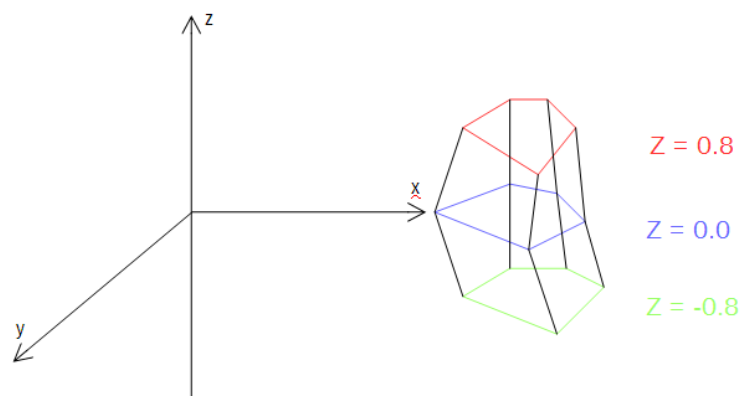
5. Łączenie płaskich konturów w obiekt wolumetryczny

Po wyznaczeniu wszystkich punktów węzłowych dla każdego zdjęcia, dokonano skalowania danych rastrowych do wymiarów rzeczywistych, wykorzystując dane zawarte w pliku DICOM. Współczynniki skali dla osi Z są zawarte w metce o pozycji (0x0020, 0x1041), zaś współczynniki skali dla osi X oraz w metce (0x0028, 0x0030). Otrzymane dane są wyrażane w milimetrach. Po tym etapie utworzono stos warstw z punktami węzłowymi, co pokazano na rysunku 5.



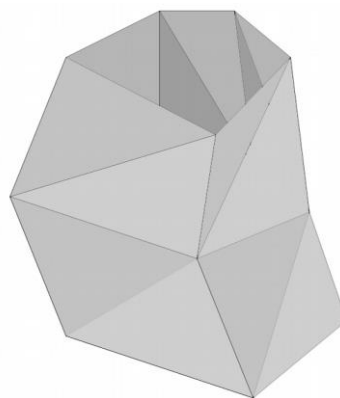
Rys. 5. Tworzenie stosu warstw z punktami węzłowymi

Mając położenie punktów węzlowych w trzech wymiarach, połączono je krawędziami uzyskując model szkieletowy, pokazany na rysunku 6.



Rys. 6. Tworzenie siatki dla kolejnych warstw wierzchołków

Ostatnim etapem było podzielenie czworokątnych elementów szkieletu na dwa trójkąty, tak, aby zawsze można było wyznaczyć trójkątną ścianę. Efekt działania pokazano na rysunku 7.



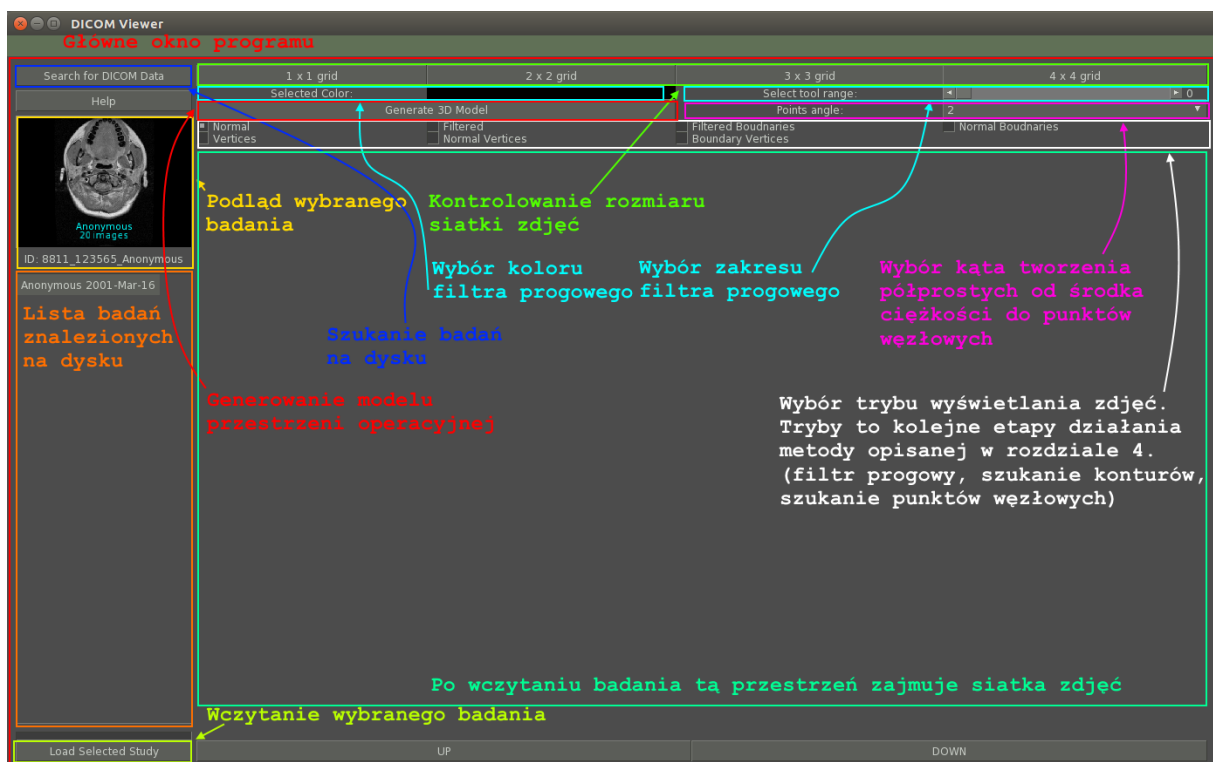
Rys. 7. Wygenerowana powierzchnia o trójkątnej siatce

Wygenerowany model oparty na trójkątnej siatce uzupełniono wektorami normalnymi w celu symulacji odbicia światła, a tym samym zwiększenia realizmu modelu przestrzeni operacyjnej.

6. Autorskie narzędzia programowe

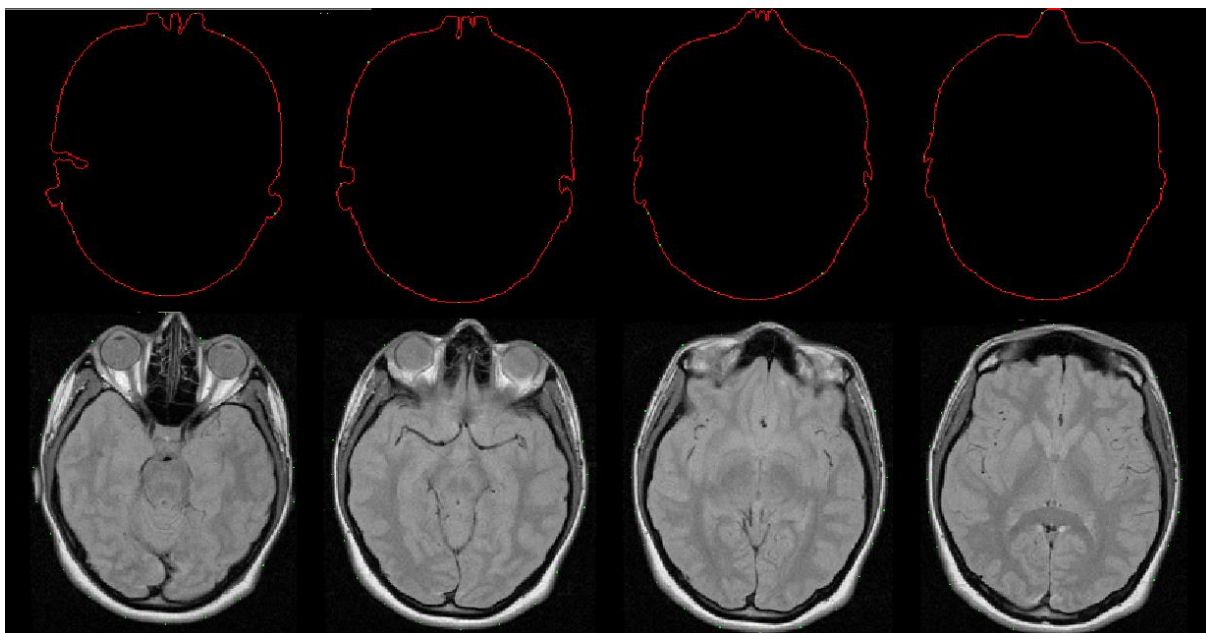
Opracowane oprogramowanie wykorzystuje kilka bibliotek wspieranych uznanymi standardami i jest kolejnym etapem prac autorów [Error! Bookmark not defined.]. Ze względu na wymogi szybkości działania i niewielkiego zapotrzebowania na skomplikowane elementy interfejsu zdecydowano się na użycie bibliotek SFML oraz SFGUI, które wspierają budowanie interfejsu użytkownika oraz operacje graficzne. Pierwsza działa bezpośrednio na API bibliotek OpenGL i wykorzystuje w sposób bezpośredni możliwości obliczeniowe kart graficznych. Druga działa bezpośrednio na funkcjach poprzedniej i zawiera wszelkie elementy potrzebne do stworzenia interfejsu aplikacji. Obsługę standardu DICOM zrealizowano przy wykorzystaniu biblioteki DCMTK, umożliwiającej dalszy rozwój aplikacji także jako narzędzia typu „closed source”. W aplikacji wykorzystano także funkcje z kilku modułów biblioteki Boost [4].

Główne okno programu udostępnia zestaw elementów ekranu, których zadaniem jest prezentacja danych DICOM, oraz wszelkie narzędzia niezbędne do realizacji modelowania realistycznej przestrzeni medycznej. Widok interfejsu przedstawia rysunek 8.



Rys. 8. Główne okno programu, z opisanymi elementami interfejsu

Kolejny rysunek pokazuje efekty działania programu. Na rysunku 9 pokazano wyznaczone punkty węzłowe na kolejnych przekrojach w widoku "na zdjęciu" i "na konturze".



Rys. 10. Wyznaczone punkty węzłowe (kolor zielony) na kolejnych przekrojach w widoku "na zdjęciu" i "na konturze"

7. Podsumowanie

Powyższa praca przedstawia zagadnienia teoretyczne i praktyczne związane z wizualizacją ciała człowieka i jest efektem kilkumiesięcznej pracy autorów. Efektem końcowym jest opracowane oprogramowanie, które posiada wszystkie założone funkcjonalności i pozwala wyznaczyć automatycznie trójwymiarowy model przestrzeni operacyjnej. Osiągnięcie wspomnianych rezultatów było możliwe dzięki zrozumieniu struktur danych standardu DICOM i połączeniu tej wiedzy z doświadczeniem w zakresie grafiki komputerowej. Autorzy zdają sobie również sprawę z pewnych braków zastosowanej metody, w odniesieniu do analizy obiektów zagnieżdżonych i wklęsłych o skomplikowanym kształcie. Jak wspomniano w jednym z punktów, do aplikacji można byłoby włączyć modelowanie grubości tkanek.

LITERATURA

- [1] O.S. Pinykh: *Digital Imaging and Communications In Medicine (DICOM). A Practical Introduction and Survival Guide*, Springer, 2007.
- [2] R. M. Haralick, L.G. Shapiro: *Computer and Robot Vision*, Addison-Wesley, 1991.
- [3] R. Leniowski, J. Meissner: *Zastosowanie standardu DICOM do wyznaczania przestrzeni operacyjnej*, Wydawnictwo Uniwersytet Rzeszowski, Rzeszów 2013.
- [4] Discussion Forum for OFFIS DICOM Tools: <http://forum.dcmk.org>

otrzymano / submitted: 20.11.2016
zaakceptowano / accepted: 20.12.2016