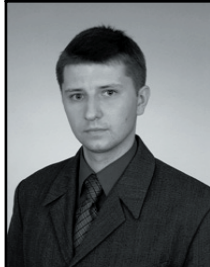


Andrzej SKALSKI¹, Mirosław SOCHA¹, Mariusz DUPLAGA²¹AKADEMIA GÓRNICZO-HUTNICZA W KRAKOWIE, KATEDRA METROLOGII²UNIwersytet Jagielloński w Krakowie, COLLEGIUM MEDICUM**Algorytm wyznaczania ścieżki nawigacyjnej do automatycznego ruchu wirtualnej kamery w strukturach anatomicznych****Mgr inż. Andrzej SKALSKI**

Absolwent wydziału Elektrotechniki, Automatyki, Informatyki i Elektroniki Akademii Górniczo-Hutniczej w Krakowie. Obecnie doktorant w Katedrze Metrologii wyżej wymienionego wydziału. Zainteresowania naukowe koncentrują się wokół zagadnień związanych z analizą i przetwarzaniem obrazów medycznych oraz z modelowaniem systemów pomiarowych.



e-mail: skalski@agh.edu.pl

Mgr inż. Mirosław SOCHA

Absolwent wydziału Elektrotechniki, Automatyki, Informatyki i Elektroniki Akademii Górniczo-Hutniczej na specjalności Automatyka i Metrologia kierunku Elektrotechnika. Obecnie pracuje na stanowisku asystenta w Katedrze Metrologii AGH i zajmuje się dydaktyką w zakresie elektronicznych układów pomiarowych oraz pomiarów wielkości nieelektrycznych. W pracy naukowej zajmuje się głównie problemami wizualizacji oraz przetwarzania danych medycznych.



e-mail: socha@agh.edu.pl

Dr med. Mariusz DUPLAGA

Specjalizacja z chorób wewnętrznych. Obecnie na stanowisku adiunkta w Zakładzie Informacji Naukowej Instytutu Zdrowia Publicznego CMUJ oraz w Klinice Pulmonologii CMUJ. Uczestnik lub lokalny kierownik kilku międzynarodowych projektach badawczo-rozwojowych Unii Europejskiej (obecnie udział w projektach HEALTHWARE (6FP - SPACE) and e-Health ERA 6FP). Główne obszary zainteresowań: systemy komputerowe wspomagające zabiegi endoskopowe, e-zdrowie, telemedycyna, telemonitorowanie.

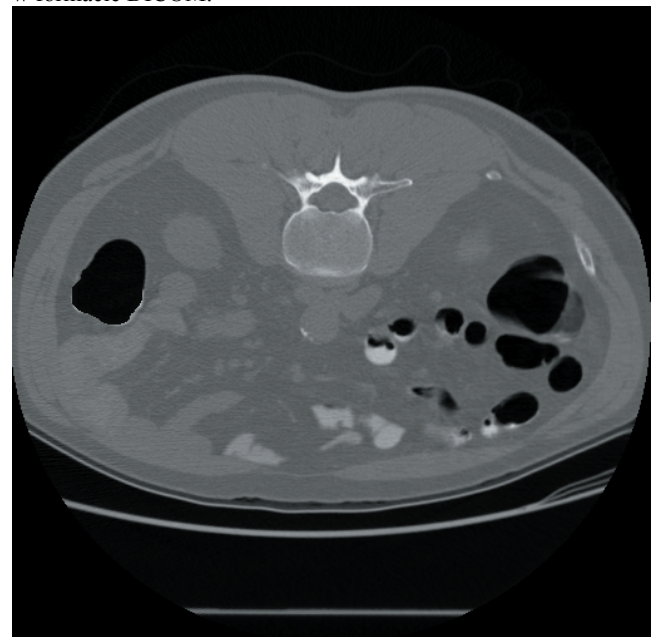


e-mail: mmduplag@cyf-kr.edu.pl

1. Wstęp

Obecnie, ze względu na nieinwazyjność oraz komfort pacjenta, coraz częściej wykonuje się „zabiegi wirtualne”. Idea wirtualnych zabiegów sprowadza się do wizualizacji struktur anatomicznych oraz symulacji ruchu kamery endoskopowej wewnątrz tej struktury. Symulacja ruchu wymaga zdefiniowania toru, po którym porusza się kamera. Podejście takie nie wymaga specjalnego przygotowania pacjenta do zabiegu, redukuje ryzyko perforacji oraz nie wpływa ujemnie na komfort pacjenta jak to ma miejsce w przypadku klasycznej kolonoskopii. Przekładem takiego badania jest wirtualna kolonoskopia (*virtual colonoscopy*). Z wyżej wymienionych powodów pacjenci bardzo często nie podają się zabiegowi klasycznej kolonoskopii, co powoduje obniżenie wykrywalności raka jelita grubego we wczesnym stadium. Warto zaznaczyć, iż wczesne wykrycie raka jelita grubego prowadzi do zwiększenia poziomu średniego 5-letniego przeżycia do 90% [1].

Dane otrzymane po przeprowadzeniu badania radiologicznego z wykorzystaniem tomografu komputerowego zapisywane są w formacie DICOM.



Rys.1. Przykładowy przekrój danych pochodzący z CT
Fig.1. Example of CT slice

Streszczenie

W pracy przedstawiono nowy algorytm wyznaczania centralnej ścieżki nawigacyjnej w trójwymiarowych danych pochodzących z tomografii komputerowej. Algorytm wykorzystuje transformacje odległości, iteracyjny schemat przetwarzania charakterystycznych punktów ścieżki oraz interpolację metodą *cubic spline* do wyznaczeni gładkiej przestrzennej krzywej nawigacyjnej. Wygenerowanie ścieżki pozwala lekarzowi na łatwe przeglądanie przestrzennego komputerowego modelu struktur anatomicznych, takich jak drzewo oskrzelowe (wirtualna bronchoskopia) czy jelito grube (wirtualna kolonoskopia), a dzięki temu poszerza możliwości diagnostyczne lekarza. W artykule szczegółowo zaprezentowano algorytm przetwarzania danych prowadzący do wyznaczenia ścieżki oraz wyniki działania prezentowanych algorytmów w zastosowaniu do wirtualnej kolonoskopii.

Słowa kluczowe: ścieżka centralna, wirtualna kolonoskopia, CT,

Algorithm of centre path tracing for virtual camera automatic movement in anatomical structure**Abstract**

The paper presents a new algorithm of centre path computing of CT data. The algorithm uses distance transformation, characteristic points interaction processing and cubic spline interpolation. The centre path defining allows for easily determination of three-dimensional model of anatomical structures that might be viewed by physician. The described method is a novel approach at enhancing diagnostic capabilities of - for example - bronchial tree (v.b. - virtual bronchoscopy) or colon (v.c. - virtual colonoscopy). The authors present detailed process of practical application of given algorithm, focusing upon v.c.

Keywords: centre path, virtual colonoscopy, CT

W przypadku wirtualnej kolonoskopii wykonuje się badanie obejmujące jamę brzuszną pacjenta. W efekcie otrzymujemy około 500 przekrojów (*slices*) pacjenta o typowym rozmiarze 512 na 512 pikseli. Każdy piksel ma przypisaną wartość osłabienia promieniowania RTG w skali Hounsfielda (HU).

W celu redukcji zajmowanej pamięci, w formacie DICOM dodaje się stałą wartość (zwykle 1024) do zapisywanych danych, aby otrzymać 16-bitowy całkowitoliczbowy format bez znaku. Na ich podstawie możliwa jest komputerowa rekonstrukcja wewnętrznych struktur anatomicznych oraz opracowanie komputerowych metod diagnostycznych (np. algorytmy wykrywania polipów) [2-6].

Jednym z zastosowań danych pochodzących z CT jest wirtualna, automatyczna nawigacja w zobrazowanej strukturze anatomicznej, na przykład w drzewie oskrzelowym [7-9] lub jelicie grubym [10-12]. Pożądane jest, aby wygenerowana ścieżka przebiegała w środku wizualizowanej struktury. Metody generacji ścieżki nawigacyjnej można podzielić na 3 zasadnicze grupy. Pierwsza metoda polega na ręcznym wskazaniu punktu należącego do ścieżki w każdym przekroju struktury (*manual extraction*). Metoda jest bardzo czasochłonna oraz nie gwarantuje centralnego położenia punktu w każdym z przekrojów. W przypadku wirtualnej kolonoskopii zadanie jest dodatkowo utrudnione ze względu na skomplikowany kształt oraz różną orientację fragmentów jelita w przestrzeni. Kolejna grupa algorytmów (*topological thinning*) [10,11] bazuje na wyznaczeniu morfologicznego szkieletu obiektu, a następnie poszukiwaniu ścieżki w ramach wyznaczonego szkieletu. Ostatnia grupa metod, do której zalicza się prezentowany algorytm, wykorzystuje transformację odległości (*distance transformation*) [12].

2. Wyznaczanie komputerowego modelu jelita grubego

Wyznaczenie komputerowego modelu jelita grubego w procesie trójwymiarowej segmentacji pozwala, z jednej strony przedstawić trójwymiarowy model, a z drugiej zastosować automatyczną metodę generacji ścieżki nawigacyjnej.

W artykule zaprezentowano wyniki działania algorytmu na wstępnie zdecydowanych danych.

2.1. Segmentacja

Celem procesu segmentacji jest podział voxelów (*volumetric pixel*) należących do danych CT na grupy reprezentujące poszczególne struktury anatomiczne. Piksele w przekrojach danych CT przyjmują wartości z przedziału -1000 do 1000 reprezentujące znormalizowane wartości osłabienia promieniowania RTG zwane skalą Hounsfielda. Powietrze reprezentowane jest przez voxele o wartościach mniejszych niż -700 HU (HU – *Hounsfield Unit*), woda 0 ± 5 HU, kości powyżej 130 HU. Właściwość ta pozwala wyodrębnić struktury anatomiczne o określonej gęstości.

Najprostszym algorytmem segmentacji jest progowanie (*thresholding*), które polega na rozdzieleniu danych na podstawie wartości progu, voxele o wartości mniejszej od progu są klasyfikowane jako tło, natomiast większej bądź równej jako obiekt. Poza podstawową zaletą, jaką jest prostota implementacji, taki sposób segmentacji ma istotną wadę: powoduje zaklasyfikowanie voxelów o tych samych wartościach jednak należących do innych struktur jako ten sam obiekt. Wynikiem progowania przeprowadzonego na danych z tomografii komputerowej jest zazwyczaj zbiór wielu rozdzielnych przestrzennie grup voxelów reprezentujących różne organy. Dodatkowo, zmiany wartości voxelów przy granicach obiektów są znaczne, w związku z tym progowanie prowadzi do błędnej klasyfikacji voxelów leżących blisko takich granic. Istnieje modyfikacja wyżej wymienionego algorytmu wyznaczająca

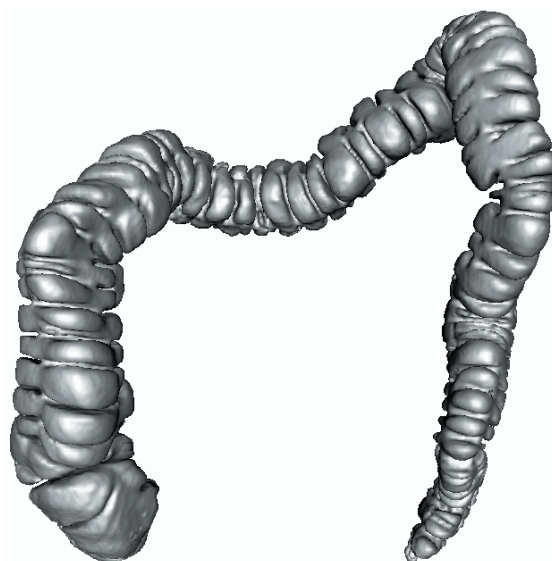
wartość progu automatycznie na podstawie histogramu obrazów, jednakże nie eliminuje ona podstawowych wad tej metody.

Typowym algorytmem stosowanym w wirtualnej kolonoskopii jest algorytm *region growing* [13-15]. Metoda ta, poza wartościami poszczególnych voxelów, uwzględnia również ich wzajemne położenie w przestrzeni. Zazwyczaj analizowane jest najbliższe sąsiedztwo voxelów – jeżeli sąsiednie voxele spełniają przyjęte kryterium przynależności, wówczas są dołączane do segmentu. W metodzie tej ograniczono problem niespójnej przestrzennie klasyfikacji voxelów o tych samych wartościach, jednakże pojawiają się trudności z doбором odpowiedniego kryterium przynależności poszczególnych voxelów do konkretnego segmentu. Algorytmy tego rodzaju wymagają zazwyczaj pewnych warunków początkowych (np. punktu startowego należącego do konkretnej struktury anatomicznej), co skutecznie utrudnia automatyzowanie procesu segmentacji.

Autorzy proponują zastosowanie metody trójwymiarowej segmentacji wododziałowej (*3D watershed*) [16], zastosowanej do trójwymiarowej mapy modułów gradientów wyznaczonych przy pomocy masek Sobel'a. Typowy problem nadsegmentacji, występujący przy wykorzystaniu tej metody, wyeliminowano używając mapy gradientów połączonych z zalewaniem obrazu z wcześniej wyznaczonych markerów. Do głównych zalet tej metody można zaliczyć to, iż nie wymaga ona nadzoru użytkownika. Wyniki są powtarzalne dla różnych zestawów danych oraz jest relatywnie szybka.

Ideę algorytmu można zobrazować na przykładzie zalewania dwuwymiarowej powierzchni topograficznej. Wyobraźmy sobie obraz jako pewną powierzchnię topograficzną odniesioną np. do poziomu morza, dodatkowo minima obszarowe będziemy interpretować jako źródła wody. Woda wypływająca ze źródła zaczyna zalewać baseny poczynając od poziomu najniższego, wypełnia je aż w pewnych momentach może nastąpić zlewanie się wody pochodzącej z dwóch różnych minimów. Aby temu zapobiec musimy budować tamę w odpowiednim miejscu. Oczywiście należy podwyższać taką tamę wraz ze wzrostem poziomu wody. System tam tworzy w obrazie linie wododziału, które oddzielają wysegmentowane obiekty.

W prezentowanym algorytmie proponujemy wprowadzenie sztucznych minimów (markerów), od których będzie inicjalizowany proces zalewania. Jako markery przyjmujemy voxele posiadające wartość odpowiadającą tkankom miękkim oraz powietrzu. Ponieważ „zalewanie” odbywa się na gradientach danych, krawędzie obiektów są poprawnie segmentowane, co jest dużą zaletą algorytmu. Wynik działania algorytmu został przedstawiony na rysunku 2.



Rys.2. Wysegmentowane jelito grube
Fig.2. Colon after segmentaion process

2.2. Generacja ścieżki nawigacyjnej

Generacja ścieżki nawigacyjnej jest operacją niezmiernie ważną. Pozwala wyznaczyć tor poruszania się wirtualnej kamery w wyznaczonym przestrzennym modelu struktury anatomicznej. Ścieżka powinna leżeć wewnątrz struktury w taki sposób, aby zapewnić szerokie pole widzenia kamery. Warunek ten można zapewnić generując ścieżkę przechodzącą przez centralne punkty przekroju osiowego struktury. Dodatkowo, algorytm generacji powinien wymagać minimalnej interakcji ze strony użytkownika. Symulacja ruchu kamery pozwala zredukować czas, jaki lekarz poświęca na analizę każdego z przekrojów otrzymanych z tomografu komputerowego (CT). Segmentacja w połączeniu z generacją toru ruchu kamery umożliwia zobrazowanie wnętrza segmentowanych struktur anatomicznych. Trójwymiarowa wizualizacja ułatwia interpretację i poszukiwanie zmian chorobowych w strukturach.

W artykule przedstawiono algorytm oparty o transformację odległości, co umożliwia wyznaczenie położenia centralnego piksela w przekroju. Transformacja odległości wyznacza odległość od najbliższego voxela o wartości 1. W artykule zaprezentowano ścieżki wyznaczone dla 3 metryk.

Danymi wejściowymi w algorytmie jest binarna macierz L reprezentująca obiekt (1 – obiekt, 0 – tło). Pierwszym etapem jest wyznaczenie dopełnienia IL macierzy L , co pozwoli wyznaczyć odległość voxelu wewnątrz obiektu do jego ścianek.

Następnie przyjmowana jest liczba punktów ścieżki N , które mają zostać wygenerowane przez iteracyjny algorytm. Zamiast liczby punktów można również przyjąć wartość reprezentującą minimalną odległość punktu ścieżki od ścianki obiektu, przy której obliczenia są przerywane.

W każdej iteracji wyznaczany jest jeden punkt ścieżki według następującej procedury. Dla macierzy IL wyznaczana jest transformata odległości, następnie wyszukiwana jest wartość maksymalna transformaty $maxD$, której współrzędne (x,y,z) są zapamiętywane w macierzy punktów ścieżki $path_{iter}=(x,y,z)$.

W transformacji odległości zbadano trzy różne metryki, zdefiniowane następująco:

- Odległość Czebyszewa (*chessboard*)

$$d_{ch} = \max(|x_i - x_j|, |y_i - y_j|, |z_i - z_j|) \quad (1)$$

- Odległość Euklidesowa (*Euclidean*)

$$d_{eu} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2} \quad (2)$$

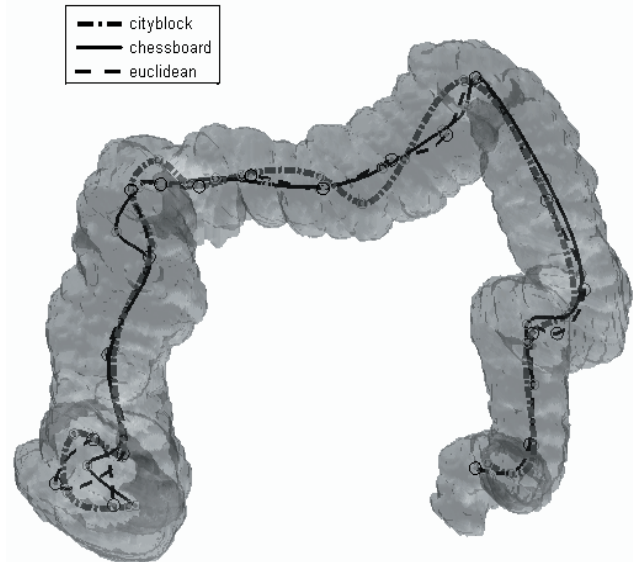
- Odległość miejska (*city block*)

$$d_c = |x_i - x_j| + |y_i - y_j| + |z_i - z_j| \quad (3)$$

Kolejnym krokiem algorytmu jest usunięcie najbliższego otoczenia wyznaczonego punktu z macierzy IL . Dla metryki Euklidesowej (2) przez wpisanie kuli o promieniu $maxD$, środka (x,y,z) i wartościach równych 1. Dla pozostałych metryk (1),(3) zamiast kuli stosuje się sześcian. Operacja ta sprowadza się do wykasowania części obiektu. Procedura jest powtarzana aż do spełnienia warunku końca, np. gdy liczba iteracji osiągnie zadaną wartość N .

W wyniku otrzymujemy macierz $path$ zawierającą współrzędne określające położenie N punktów ścieżki centralnej. Punkty te muszą zostać posortowane by mogły posłużyć do wyznaczenia trajektorii kamery. Pierwszym krokiem sortowania jest wyznaczenie początku (lub końca) ścieżki. Zazwyczaj jest nim punkt o minimalnej współrzędnej z . Następnie wyliczana jest odległość od punktu początkowego do pozostałych punktów.

Punkt najbliższy położony klasyfikowany jest jako kolejny punkt ścieżki. Powtarzając obliczenia wyznaczana jest prawidłowa kolejność punktów w przestrzeni, umożliwiającą poprowadzenie przez nie krzywej. Ostatnim krokiem jest wyznaczenie przez interpolację *cubic spline* gładkiej krzywej przechodzącej przez wyliczone punkty. Na rysunku 3 przedstawiono przykład wyznaczonej ścieżki nawigacyjnej dla trzech różnych metryk stosowanych do obliczenia transformacji odległości.



Rys.3. Przykład wyznaczonej ścieżki nawigacji
Fig.3. Example of computed navigation path

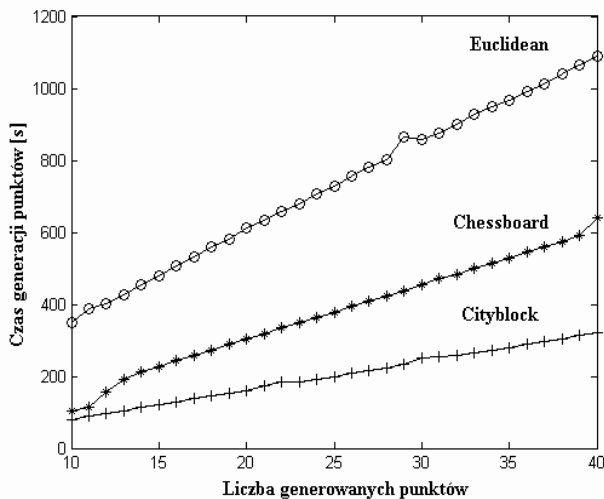
W tabeli 1 przedstawiono czasy wyliczenia poszczególnych etapów algorytmu dla danych 255x255x236 voxelu, odległości *city block*, 25 punktów ścieżki, zaimplementowanych w środowisku Matlab [17] na komputerze PC Intel Celeron 2 GHz, 512 MB RAM. Najbardziej czasochłonny jest proces generacji punktów ścieżki. Pozostałe operacje trwają mniej niż 6% czasu działania całego algorytmu.

Tab. 1. Czas wyznaczenia poszczególnych etapów algorytmu
Tab. 1. Algorithm phase computing time

	Czas operacji [s]
Dopełnienie obrazu	0.47
Generacja punktów	203.68
Sortowanie punktów	0.07
Interpolacja	10.79

Wybór normy zastosowanej do wyliczenia transformacji odległości ma wpływ nie tylko na przebieg ścieżki (rys. 3.), ale także na czas wyznaczenia ścieżki, co można zaobserwować na rysunku 4. Najszybciej generowane są punkty przy normie *city block*. Wynika to z najprostszej formuły na odległość. Ceną zaletą tego rozwiązania jest to, iż obliczenia są przeprowadzane na liczbach całkowitych, co zmniejsza ilość zajmowanej pamięci i dodatkowo przyspiesza algorytm. Najpoważniejszą wadą norm *city blok* i Czebyszewa jest znaczna anizotropowość w obliczanej transformacji odległości, co może prowadzić do błędnie wyznaczanych punktów ścieżki. Zastosowanie izotropowej normy Euklidesowej powoduje ponad trzykrotne zwiększenie czasu obliczeń.

Liczba generowanych punktów ma wpływ na przebieg ścieżki. Za małą ich liczbą spowoduje, iż interpolowana krzywa może nie zawierać się całkowicie wewnątrz segmentowanej struktury. Możliwe jest wprowadzenie ograniczeń przy interpolacji, jednakże znacznie zwiększy to czas wykonywania obliczeń algorytmu. Z drugiej strony, zadana zbyt duża liczba punktów



Rys. 4. Czas wyznaczenia voxelu przy różnych metrykach oraz zmiennej liczbie generowanych punktów ścieżki

Fig. 4. Voxel generation time for different number of path points

prowadzi do powstania błędnych punktów nie wyznaczających linii centralnej.

Możliwy jest wówczas przypadek, w którym punkty szukane są w „resztkach” danych – w każdej iteracji następuje częściowe wykasowanie obiektu, co zazwyczaj prowadzi do powstania mniejszych jego fragmentów. Kasowanie fragmentów obiektu, w kolejnych iteracjach, zapobiega zakwalifikowaniu jako punkt ścieżki voxelu leżącego blisko poprzednio wyznaczonych punktów. W przypadku jelita grubego i decymowanych danych o wymiarach 255x255x236 voxelu zalecana liczba punktów mieści się w przedziale od 22 do 26 punktów.

W zależności od segmentowanej struktury optymalna liczba punktów będzie inna. W zastosowaniu do innych struktur zaleca się przeprowadzanie wstępnych badań, które umożliwią wyznaczenie optymalnej liczby punktów ścieżki, co zapewni poprawność wyznaczonej krzywej oraz krótki czas obliczeń.

Odmiernym podejściem jest przyjęcie jako warunek przerwania działania algorytmu minimalnej odległości, w jakiej powinien znajdować się każdy punkt ścieżki od ścianki wysegmentowanej struktury. Nie ma konieczności wyboru z góry ustalonej liczby punktów, jednakże konieczny jest wybór takiej wartości odległości, która będzie uwzględniała normę oraz przybliżony rozmiar segmentowanego obiektu.

3. Podsumowanie

W artykule zaprezentowano nowy algorytm wyznaczania ścieżki nawigacyjnej na przykładzie modelu jelita grubego wysegmentowanego z danych CT. Algorytm wykorzystuje transformację odległości oraz interpolację *cubic spline* w celu wyznaczenia centralnej ścieżki nawigacyjnej. Zaprezentowano przykładowe ścieżki dla 3 metryk w jelicie grubym (rys. 3). Zaproponowano algorytm segmentacji wykorzystujący metodę wododziałową wykorzystującą zalewanie ze sztucznie wygenerowanych minimów umieszczonych w trójwymiarowej mapie modułów gradientów, zapewniający poprawną segmentację ścian struktur anatomicznych z danych CT. Przedstawione wyniki potwierdzają małą czasochłonność proponowanej metody w porównaniu z innymi rozwiązaniami. W przypadku zastosowania algorytmu generacji ścieżki do struktur anatomicznych posiadających rozgałęzienia konieczne jest wprowadzenie modyfikacji wybierającej punkty zżądanego odgałęzienia. W przypadku drzewa oskrzelowego wystarczy wskazać punkt początkowy i końcowy oraz zmodyfikować algorytm tak, aby

odległości były wyznaczane od jednego i drugiego punktu równocześnie.

Przedstawiona metoda generowania ścieżki centralnej, ze względu na przewidywany relatywnie krótki czas obliczeń po przeniesieniu do języka C, może zostać zastosowana w oprogramowaniu wspomagającym bezinwazyjne badania diagnostyczne, np. w wirtualnej kolonoskopii pacjentów, u których lekarz podejrzewa zmiany chorobowe w jelicie grubym.

Zaprezentowany algorytm można rozbudować o dedykowaną metodę realizującą automatyczną detekcję zmian chorobowych w badanych strukturach.

4. Literatura

- [1] S. M. Frentz, R. M. Summers, "Current Status of CT Colonography", *Academic Radiology*, vol. 13, 2006, pp. 1517-1531.
- [2] R. M. Summers, M. Franaszek, M. Miller, P. J. Pickhardt, J. Choi, W. R. Schindler, "Computer-aided detection of polyps on oral contrast-enhanced CT colonography", *American Journal of Roentgenology*, vol. 184(1), Jan 2005, pp. 105-108.
- [3] N. Sezille, R. J. T. Sadleir, P. F. Whelan, "Automated synthesis, insertion and detection of polyps for CT colonography", *Opto-Ireland 2002, Proceedings of the SPIE*, vol. 4877, 2003, pp. 183-191.
- [4] H. Yoshida, J. Näppi, P. MacEaney, D. T. Rubin, A. H. Dachman, "Computer-aided Diagnosis Scheme for Detection of Polyps at CT Colonography", *Radiographics*, vol. 22, 2002, pp. 963-979.
- [5] P. J. Pickhardt, A. J. Taylor, D. V. Gopal, "Surface Visualization at 3D Endoluminal CT Colonography: Degree of Coverage and Implications for Polyp Detection", *Gastroenterology*, vol. 130, 2006, pp. 1582-1587.
- [6] Zhuowen Tu Zhou, X.S. Bogoni, L. Barbu, A. Comaniciu, D., "Probabilistic 3D Polyp Detection in CT Images: The Role of Sample Alignment", *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference*, vol. 2, 2006, pp. 1544-1551.
- [7] K. Duda, M. Duplaga, "Automatyczna generacja ścieżki nawigacyjnej w drzewie oskrzelowym", *PAK 5bis 2006, Sympozjum Modelowanie i Pomiar w Medycynie*, Krynica, 14-18 V 2006 str. 115-118.
- [8] A. Shahrokni, H. Soltanian-Zadeh, R. A. Zoroofi, "Fast skeletonization algorithm for 3D elongated objects", *Proceedings of the SPIE, Medical Imaging*, vol. 4322, 2001, pp. 323-330.
- [9] J. P. Helferty, A. J. Sherbondy, A. P. Kiraly, W. E. Higgins, "System for Live Virtual-Endoscopic Guidance of Bronchoscopy", *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Workshops*, vol. 3, pp. 68-76.
- [10] W. Xie, R. P. Thompson, R. Perucchio, "A topology-preserving parallel 3D thinning algorithm for extracting the curve skeleton", *Pattern Recognition*, vol. 36, 2003, pp. 1529-1544.
- [11] R. Sadleir, P. Whelan, "Fast colon centreline calculation using optimised 3D topological thinning", *Computerized Medical Imaging and Graphics*, vol. 29(4), 2005, pp. 251-258.
- [12] M. Sabry Hassouna, A.A. Farag, "3D Path Planning For Virtual Endoscopy," *Proceedings of Computer Assisted Radiology and Surgery (CARS'05)*, Berlin, Germany, 2005, pp.115-120
- [13] W. Xie, R. P. Thompson, R. Perucchio, "A topology-preserving parallel 3D thinning algorithm for extracting the curve skeleton", *Pattern Recognition*, vol. 36, 2003, pp. 1529-1544.
- [14] R. Sadleir, P. Whelan, "Fast colon centreline calculation using optimised 3D topological thinning", *Computerized Medical Imaging and Graphics*, vol. 29(4), 2005, pp. 251-258.
- [15] A. Vilanova, A. König, E. Gröller, "VirEn: A Virtual Endoscopy System", *Machine Graphics & Vision*, Vol. 8(3), 1999, pp. 469-487.
- [16] L. Vincent, P. Soille, "Watersheds in digital spaces: An efficient algorithm based on immersion simulations", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13(6), 1991, pp. 583-598.
- [17] *Image Processing Toolbox For Use with MATLAB – User's Guide Version 5*, www.mathworks.com