

Marcin Janaszewski^{*,**}, Michał Postolski^{*,**,***},
Laurent Babout^{*}, Łukasz Jopek^{*,**}

Zamykanie topologicznych i wolumetrycznych otworów w obiektach 3D

1. Wprowadzenie

Obraz binarny 3D to funkcja, która przypisuje każdemu punktowi z przestrzeni dyskretnej 3D wartość 0 lub 1. Punkty mające wartość 1 zwane są czarnymi punktami, zaś punkty o wartości 0 – białymi punktami. Zbiór wszystkich czarnych punktów nazywamy wolumetrycznym obiektem 3D lub krócej obiektem 3D. Binarne obrazy 3D są często wynikiem segmentacji obrazów tomograficznych, z których wyodrębniane są woksele interesującego obiektu, natomiast pozostałe zaliczane są do tła.

W artykule rozważamy dwa typy otworów w obiektach 3D, które nie są proste do zdefiniowania i wizualizacji. Kong w [13] zaproponował koncepcję detekcji otworu topologicznego, zwanego inaczej tunelem, zgodnie z którą w obiekcie X występuje otwór (otwory) topologiczny wtedy, gdy istnieje pętla wokseli należących do X , którą nie można iteracyjnie transformować wewnątrz X , z wykorzystaniem elementarnych lokalnych transformacji, do jednego punktu należącego do X . Elementarne, lokalne transformacje zdefiniowano precyzyjnie w [13]. Nieformalnie opisując, można powiedzieć, że elementarna transformacja to deformacja pętli wokseli w ramach objętości o boku 3 wokseli, która nie powoduje rozłączenia pętli. Zgodnie z opisywaną koncepcją kula nie posiada tunelu, torus posiada jeden tunel, pusty torus ma dwa tunele.

Drugi typ otworu – otwór geometryczny jest szeroko rozważany w literaturze, np. [11, 14] opisującej przetwarzanie obiektów reprezentowanych za pomocą siatki trójkątów – zbioru wierzchołków i zbioru zorientowanych trójkątów, które łączą te wierzchołki. Zakładając, że prawidłowa siatka powinna być zamkniętą rozmaitością, otwory geometryczne mogą być łatwo wykrywane przez wyszukiwanie nieciągłości tzn. krawędzi, które należą do tylko jednego

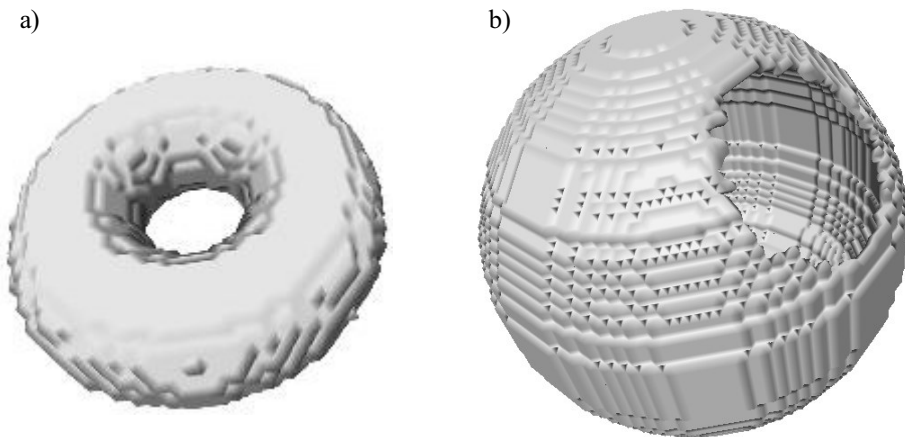
* Katedra Informatyki Stosowanej, Politechnika Łódzka, Polska

** Katedra Systemów Ekspertowych i Sztucznej Inteligencji, Wyższa Szkoła Informatyki w Łodzi, Polska

*** Universite Paris-Est, LIGM, A3SI, ESIEE, Noisy le Grand, Francja

trójkąta, zwanych krawędziami granicznymi. Tego typu reprezentacje generowane są przez laserowe skanowanie rzeczywistych obiektów. Otwory powstają w wyniku przysłaniania jednych fragmentów obiektu innymi.

W przypadku obiektów 3D (w przestrzeni wokseli) detekcja otworów geometrycznych nie jest już taka oczywista. Intuicyjnie otwór geometryczny w przestrzeni wkselowej może być rozumiany podobnie jak w przypadku obiektów powierzchniowych jako nieciągłość w powierzchni zewnętrznej obiektu zbudowanej w tym przypadku z wokseli. Różnicę między otworem topologicznym a geometrycznym wygodnie jest zaprezentować na podstawie przykładu. Sfera S z wyciętym dyskiem posiada otwór geometryczny, gdyż występuje nieciągłość jej powierzchni zewnętrznej (patrz rys. 1b). Nie jest to tunel, gdyż każda pętla zbudowana z wokseli tej sfery może być transformowana wewnątrz S za pomocą przekształceń elementarnych do jednego punktu zawartego w S . Z drugiej strony wypełniony torus T posiada otwór topologiczny, gdyż nie posiada nieciągłości wokseli budujących jego powierzchnię zewnętrzną, ale potrafimy wskazać przynajmniej jedną pętlę, którą nie można transformować wewnątrz T do jednego punktu zawartego w T (patrz rys. 1a).



Rys. 1. Izopowierzchnia torusa (a), izopowierzchnia sfery o ścianie grubości jednego woksela z wyciętym otworem geometrycznym (b)

W kolejnym rozdziale zaprezentowano przegląd literatury z dziedziny zamykania otworów geometrycznych i topologicznych. W rozdziale 3 przytoczono pojęcia topologii i geometrii dyskretnej wykorzystane w dalszej części artykułu. Rozdział 4 prezentuje algorytm zamykania tuneli, który stanowi podstawę nowego algorytmu zaprezentowanego w niniejszym artykule. W rozdziale 5 opisano zasadę działania nowego algorytmu, jego właściwości i wyniki testowania na dwóch obrazach. Ostatni rozdział stanowi podsumowanie niniejszej pracy. W dalszej części artykułu otwór geometryczny będzie często, w skrócie, nazywany otworem zaś otwór topologiczny tunelem.

2. Podobne prace

Istnieje duża liczba artykułów prezentujących różne podejścia do zamykania otworów w obiektach reprezentowanych za pomocą powierzchni zewnętrznych złożonych z wielokątów.

Algorytmy zamykania otworów dzielą się na dwie grupy. Pierwsza grupa to algorytmy dokonujące uzupełnienia braków w siatce wielokątów reprezentujących powierzchnię zewnętrzną. Druga grupa algorytmów transformuje zadanie do przestrzeni wokselowej, gdzie dokonuje się identyfikacji i zamykania otworów w powierzchni zewnętrznej obiektu.

Typowe metody grupy pierwszej iteracyjnie wypełniają brakujące fragmenty powierzchni zewnętrznej, stosując kryterium minimalizujące obszar zajmowany przez łąkę zamykającą otwór np. [14]. Bardziej zaawansowane metody grupy pierwszej dokonują zamknięcia przez odnalezienie w obiekcie fragmentu, który przypomina obszar w pobliżu otworu. Po odnalezieniu takiego fragmentu następuje jego kopiowanie w miejsce otworu, np. [8, 18]. Inne podejścia poszukują łąk najbardziej pasujących do zamykania otworów przez poszukiwanie obiektów podobnych w bazie danych np. [2, 6].

W drugiej grupie metod znajdują się podejścia, które w przestrzeni wokselowej klasyfikują przynależność każdego woksela do wnętrza i powierzchni zewnętrznej obiektu lub do tła, np. [10, 16]. W wyniku tej klasyfikacji otrzymujemy wypełniony obiekt bez otworów, na bazie którego możemy łatwo wyznaczyć powierzchnię zewnętrzną w postaci siatki wielokątów za pomocą algorytmu *marching cubes* [15]. Inne podejścia w tej grupie dokonują symulacji dyfuzji ciepła, uzyskując naturalną, gładką kontynuację powierzchni z otoczenia otworu, np. [11, 17].

W odróżnieniu od tematu zamykania otworów, literatura na temat zamykania tuneli jest bardzo skąpa. Zgodnie z wiedzą autorów istnieje tylko kilka publikacji na ten temat: [1, 19]. W artykułach tych zamykanie tuneli realizowane jest przez iteracyjne kasowanie wokseli wypełnionego prostopadłościanu Y zawierającego obiekt wejściowy X . Kasowaniu ulegają te woksle z Y/X , których usunięcie nie powoduje stworzenia tunelu w $Y \setminus X$. Niestety otwory nie są jednocześnie zamykane.

Podsumowując analizę literatury, a w szczególności artykuły przeglądowe [7, 12], należy stwierdzić, że autorzy nie znaleźli żadnej publikacji prezentującej algorytm zamykający zarówno otwory, jak i tunele. Stąd głównym celem tego artykułu jest prezentacja zasady działania i właściwości nowego podejścia do zamykania zarówno tuneli, jak i otworów.

3. Pojęcia topologii i geometrii dyskretnej

Podstawowe pojęcia topologii i geometrii dyskretnej dotyczące obrazów 3D reprezentujących obiekty posiadające objętość są zdefiniowane w innych publikacjach, np. [3, 13]. W tym rozdziale zostaną zaprezentowane jedynie wybrane zagadnienia niezbędne do zrozumienia dalszej części artykułu.

Oznaczmy przez Z zbiór liczb całkowitych, przez N zbiór dodatnich liczb całkowitych. Niech ponadto $E = Z^3$.

Pomijając pewne szczegóły techniczne, załóżmy, że $A(x, X)$ będzie zbiorem punktów z $X \setminus \{x\}$ należących do sąsiedztwa punktu x . Niech $Ab(x, X)$ będzie zbiorem punktów dopełnienia X (tło) należących do sąsiedztwa x . Oznaczmy odpowiednio przez $T(x, X)$ (odpowiednio $Tb(x, X)$) liczbę połączonych komponentów $A(x, X)$ (odpowiednio $Ab(x, X)$). Nieformalnie, *punktem prostym* p dyskretnego obiektu $X \subset E$ nazywamy punkt, który „nie ma wpływu” na topologię X tzn. można usunąć taki punkt i topologia obiektu X się nie zmieni. Bertrand w [3] udowodnił, że punkt jest prosty dla X wtedy i tylko wtedy gdy: $T(x, X) = Tb(x, X) = 1$. Ponadto, jeżeli $Tb(x, X) = 1$, to usunięcie punktu z X nie tworzy nowego tunelu w X . Pojęcie punktów prostych jest kluczowe dla budowy transformacji zachowujących topologię w przestrzeniach dyskretnych [3, 13]. Na bazie liczb topologicznych można rozpoznawać inne typy punktów. Punkt $x \in X$ taki, że $T(x, X) = 0$ jest punktem izolowanym. Jeżeli $Tb(x, X) \neq 0$, to x należy do brzegu X . Jeżeli $T(x, X) \geq 2$, to usunięcie x z X powoduje lokalne rozłączenie X . Taki punkt nazywamy przesmykiem 1D (*1D isthmus*). Podobnie punkt, dla którego $Tb(x, X) \geq 2$ jest nazywany przesmykiem 2D (*2D isthmuses*) gdyż jego skasowanie lokalnie łączy komponenty tła obrazu. Punkt taki, że $T(x, X) = 2$ i $Tb(x, X) = 1$, nazywamy prostym przesmykiem 1D (*simple 1D isthmus*). Podobnie mamy punkt prosty 2D jeżeli $Tb(x, X) = 2$ i $T(x, X) = 1$.

Niech X będzie skończonym podzbiorem E . Zbiór $Y \subseteq X$ jest **homotopologicznym pocienianiem** (*homotopic thinning*) zbioru X , jeżeli Y można uzyskać poprzez iteracyjne usuwanie punktów prostych. Ponadto, Y jest **ostatecznym homotopologicznym szkieletem** (*ultimate homotopic skeleton*) X , jeżeli Y jest homotopologicznym pocienianiem X i Y nie zawiera punktów prostych. Mówimy, że Y jest **ostatecznym homotopologicznym szkieletem X ograniczonym przez zbiór C** , jeżeli $C \subseteq Y$, Y jest homotopologicznym pocienianiem X i zbiór $Y \setminus C$ nie zawiera punktów prostych. Zbiór C zwany jest **zbiorem ograniczającym** tak wygenerowanego szkieletu.

4. Algorytm zamykania tuneli (AZT)

W niniejszym rozdziale zaprezentujemy zasadę działania oraz właściwości oryginalnego algorytmu zamykania tuneli AZT dla obiektów wolumetrycznych zaprezentowanego w [1].

Pierwszym krokiem algorytmu jest wyznaczenie wypełnionego prostopadłościanu Y , który jest minimalnym prostopadłościanem zawierającym obiekt wejściowy X . Następnie algorytm iteracyjnie kasuje punkty ze zbioru $Y \setminus X$, które są punktami brzegowymi i jednocześnie nie są przesmykami 2D (patrz rozdział 3). Jeżeli punkt jest przesmykiem 2D, to może być usunięty tylko wtedy, gdy jego odległość od zbioru X jest większa niż *a priori* zdefiniowana wartość r reprezentująca „promień”(rozmiar) zamykanego tunelu. Ostatni warunek prowadzi do tego, że

tylko tunele o „rozmiarze” mniejszym lub równym r zostaną zamknięte. Selekcja punktów do kasowania jest sterowana przez odległość punktu $x \in Y \setminus X$ od zbioru X oznaczaną $d(x, X)$. Stąd najpierw są kasowane punkty najdalej oddalone od X . Dzięki temu łąty zamykające tunele dla prostych geometrycznie obiektów znajdują się w ich centrum.

Pseudokod procedury zamykania tuneli może być przedstawiony następująco:

AZT (Wejście X, r , Wyjście Y)

Generacja prostopadłoscianu Y zawierającego X

Powtarzaj do momentu, gdy brak punktu do skasowania:

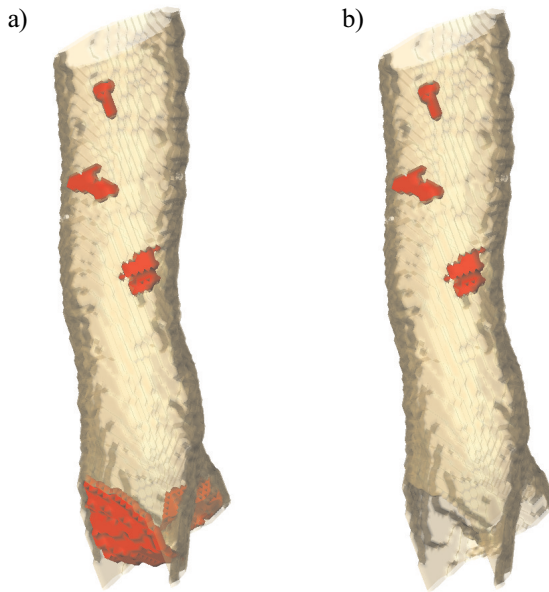
Wybierz punkt $p \in Y \setminus X$ taki, że: $Tb(p, X) = 1$

lub taki, że: $Tb(p, X) = 2$ i $d(p, X) > r$

który jest w największej odległości od X

$Y := Y \setminus p$

Rezultat: Y



Rys. 2. Wizualizacja wyniku zastosowania AZT dla tchawicy: a) Wynik działania AZT zamykającego wszystkie tunele. Łąty zamykające tunele odrysowano ciemnym odcieniem szarości na tle półprzezroczystej jasnej tchawicy. Widać, że AZT zamknął nie tylko niepożądane tunele w ścianie tchawicy, ale również prześwit tchawicy, co jest działaniem niepożądanym. b) Wynik działania AZT przy ustalonym parametrze $r = 5$. Niepożądane tunele zostały zamknięte. Prześwit tchawicy pozostał otwarty

Parametr r zastosowany w algorytmie ma istotne znaczenie praktyczne, gdyż umożliwia zamykanie tylko tych tuneli, dla których maksymalna odległość pikseli łąty zamykającej od

pikseli obiektu wejściowego jest mniejsza lub równa r . Zwykle niepożądane tunele w obiektach wolumetrycznych powstałe w wyniku szumów są niewielkich rozmiarów. Oprócz takich tuneli obiekt może jeszcze posiadać tunele, które są jego naturalną cechą i nie powinny być zamykane. Takie tunele zwykle są większe od tych powstałych w wyniku zaszumienia. Odpowiednie ustawienie parametru r w AZT, umożliwia zamknięcie tylko niepożądanych tuneli, przy pozostawieniu tuneli większych będących istotnymi cechami badanego obiektu. Sytuacja taka została zaprezentowana na rysunku 2. Należy jednak podkreślić, że AZT nie zamyka otworów.

5. Algorytm zamykania otworów i tuneli (AZOT)

W niniejszym rozdziale zaprezentujemy zasadę działania algorytmu służącego do zamykania zarówno otworów, jak i tuneli, w skrócie: AZOT.

Algorytm na wejście przyjmuje binarny obraz 3D O , natomiast zwraca obiekt L będący sumą łąt zamykających otwory i tunele (patrz dalej zamieszczony pseudokod AZOT). Na pierwszym etapie AZOT realizuje szkieletyzację powierzchniową z wykorzystaniem algorytmu opisanego w [5]. W ostatnich dziesięcioleciach zaproponowano wiele algorytmów szkieletyzacji powierzchniowej. Jednak niewiele z nich jest odpowiednich do naszego zastosowania, w którym wymagane jest, aby każdy woksel szkieletu należał do jednego z dwóch zbiorów: woksele krawędziowe ($T(x, X) = 1$) i przesmyki 2D ($Ab(x, X) > 1$ patrz rozdział 3). Przykładowy szkielet spełniający opisany warunek zaprezentowano na rysunku 3. Autorzy zastosowali solidny algorytm z [5], oparty na dobrze zdefiniowanych pojęciach topologii i geometrii dyskretnej. Przykładowy wynik pierwszego etapu działania algorytmu zaprezentowano na rysunku 4b.

Drugi etap AZOT polega na wyodrębnieniu wokseli krawędziowych ze szkieletu powierzchniowego. Ten etap można zrealizować łatwo poprzez sprawdzenie wartości liczb topologicznych dla każdego wokselu szkieletu. Przykładowy wynik działania etapu wyodrębniania wokseli krawędziowych zaprezentowano na rysunku 4c. W pseudokodzie etap ten realizowany jest w linii 02. Kolejny etap polega na wyodrębnieniu połączonych komponentów wokseli krawędziowych (patrz linia 03 pseudokodu). W analizowanym przykładzie mamy dwa komponenty. Czwarty etap dotyczy rozpoznawania, które z komponentów są pętlami, a które łukami. Ponownie zastosowano analizę liczb topologicznych dla każdego wokselu komponentu. Jeżeli i -ty komponent ma przynajmniej dwa tzw. punkty końcowe ($T(x, K_i) = 1$), to oznacza, że jest łukiem (linie 05, 06 pseudokodu). W przypadku łuków trzeba najpierw dokonać ich uzupełnienia do pętli, potem będzie można zamknąć uzupełnione łuki, wykorzystując AZT. W tym celu generujemy ostateczny homotopologiczny szkielet S_i obiektu S ograniczony przez komponent K_i (patrz rys. 4d i algorytm OHS w linii 08 pseudokodu). Szkieletyzacja realizowana jest przez algorytm pocieniania sterowany przez funkcję odległości D_i wokseli S od komponentu K_i (patrz linie 07 i 08 pseudokodu). Dzięki funkcji odległości sterującej procesem szkieletyzacji

uzyskujemy uzupełnienie komponentu K_i do pętli poprzez możliwie najkrótszą ścieżkę, zawartą w S , łączącą punkty końcowe tego komponentu (rys. 4d). W przypadku gdy komponent jest pętlą, nie trzeba stosować uzupełniania, stąd pomijane są linie 07–09 pseudokodu. Uzupełnione łuki oraz pętle są następnie zamykane z wykorzystaniem AZT (patrz linia 10 pseudokodu oraz rys. 4e). Ostatecznie AZOT zwraca obiekt L reprezentujący sumę łąt zamykających kontury (linia 11 pseudokodu).

Pseudokod algorytmu zamykającego tunele i otwory zaprezentowano poniżej:

AZOT(Wejście O , Wyjście L)

01 $S = \text{AlgSzkieletyzacjiPowierzchniowej}(O)$

02 $K = \{x \in S : T(x, S) = 1 \text{ i } T_b(x, S) = 1\}$

03 $\{K_i\} = \text{EtykietowaniePołączonychKomponentów}(K)$

04 Dla każdego i -tego komponentu wyznacz:

05 $PK_i = \{x \in K_i : T(x, K_i) = 1\}$

06 Jeżeli $PK_i \geq 2$ to:

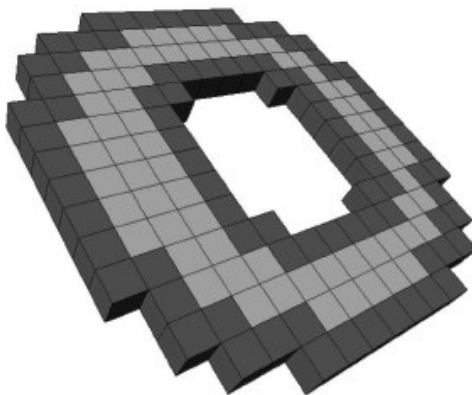
07 $D_i = \text{GeodezyjnaOdległość}(S, K_i)$

08 $S_i = \text{OHS}(S, K_i, D_i)$

09 $K_i = S_i$

10 $\text{AZT}(K_i, 1000, L_i)$

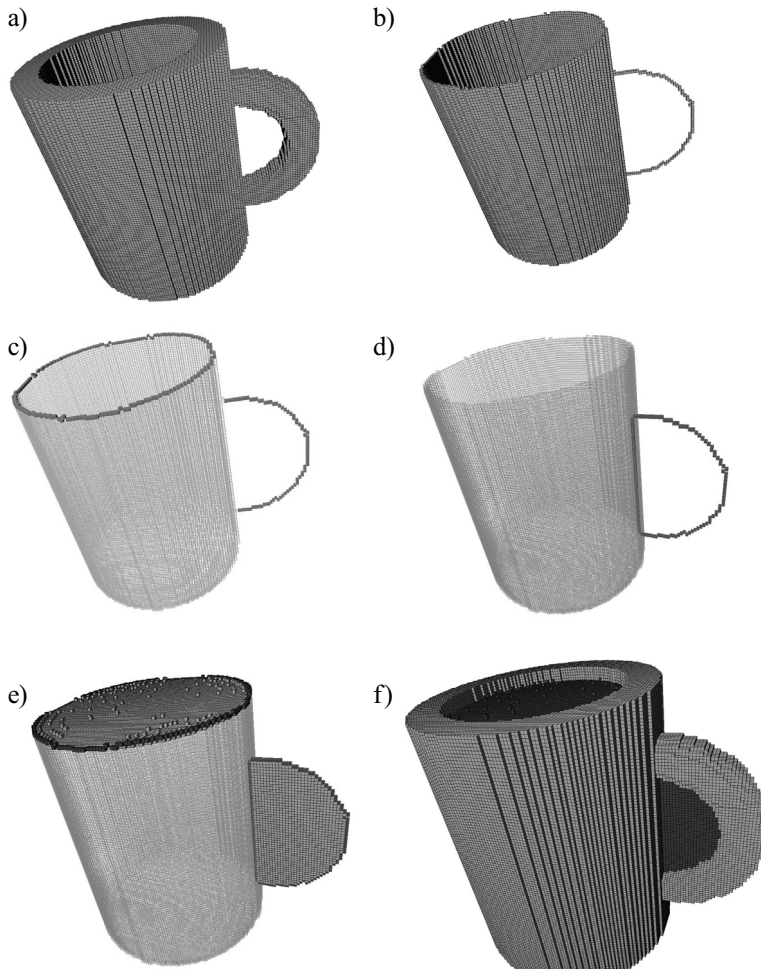
11 $L = \bigcup_i L_i$



Rys. 3. Wizualizacja przykładowego szkieletu powierzchniowego wygenerowanego algorytmem [5].

Woksele będące przesmykami 2D zaznaczono jasnym odcieniem szarości.

Woksele krawędziowe zaznaczono ciemnym odcieniem szarości



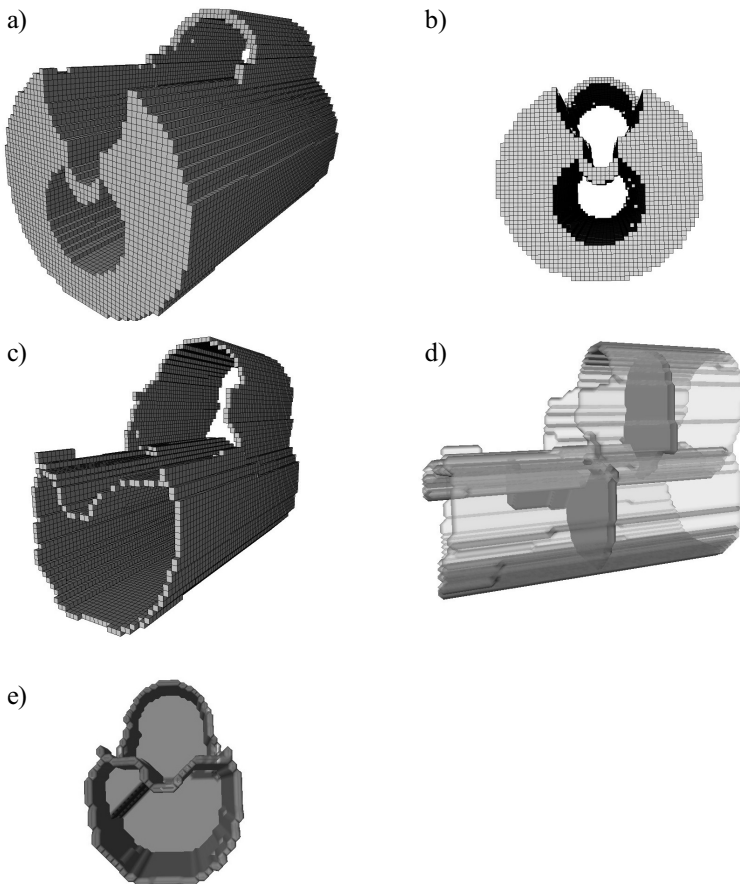
Rys. 4. Prezentacja zasady działania algorytmu zamykającego tunele i otwory:

a) obiekt wejściowy – kubek, którego ucho tworzy tunel, a niezamknięta powierzchnia kubka tworzy otwór; b) szkielet powierzchniowy kubka; c) kontury tunelu kubka zaznaczono ciemnym odcieniem szarości na tle jasnego szkieletu kubka; d) uzupełniony do pętli kontur szkieletu ucha dzięki zastosowaniu algorytmu ostatecznej szkieletyzacji szkieletu kubka; zastosowany algorytm jest sterowany funkcją odległości od konturu ucha, dzięki czemu fragment uzupełniający to najkrótsza ścieżka łącząca punkty końcowe konturu ucha; e) wynik zamykania konturów kubka (ciemny odcień szarości) na tle kubka (jasny odcień szarości); f) wynik zamykania otworu i tunelu na tle kubka

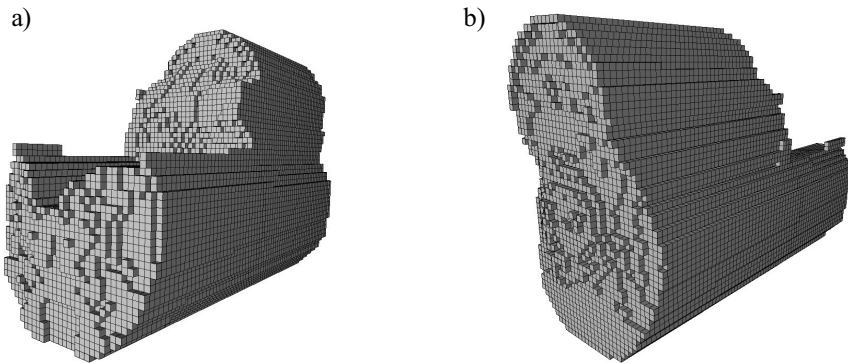
Pobieżna analiza zasady działania AZOT na podstawie przykładu z rysunku 4 może doprowadzić do wątpliwości dotyczących słuszności uzupełniania konturu ucha do pętli. Po zastosowaniu AZT do szkieletu kubka (rys. 4b) tunel utworzony przez ucho zostanie zamknięty. Rzeczywiście w przypadku zaprezentowanym na rysunku 4 można by pominąć operację

uzupełnienia konturu i zastosować AZT do szkieletu kubka, a później po wyodrębnieniu konturu otworu ponownie zastosować AZT w celu jego zamknięcia. Można jednak zaprezentować przykłady obiektów, dla których wyodrębnianie konturów i ich uzupełnianie ma istotne zastosowanie.

Rozważmy obiekt zaprezentowany na rysunku 5a, przypominający dwie stykające się ściankami rurki. Występują tutaj dwa tunele. Po zastosowaniu AZT otrzymujemy wynik zaprezentowany na rysunku 5d. Formalnie otwory zostały zamknięte, ale łaty zamykające otwory znajdują się w środku rurek. Nie zawsze takie rozwiązanie jest zadowalające z praktycznego punktu widzenia. Często interesuje nas zamknięcie tuneli na końcach. Wynik zamykania tuneli z wykorzystaniem AZOT pokazano na rysunku 6. To dzięki wyodrębnianiu konturów i ich późniejszemu zamykaniu uzyskano łaty usytuowane na końcach rurek.



Rys. 5. Przykład zamykania obiektu z dwoma tunelami: a) wizualizacja obiektu wejściowego; b) widok obiektu wejściowego z przodu; c) szkielet obiektu wejściowego; d) wynik zamykania szkieletu obiektu wejściowego, z wykorzystaniem AZT na tle półprzezroczystego szkieletu; e) widok z przodu pokazujący łaty zamykające otwory na tle szkieletu obiektu wejściowego



Rys. 6. Wynik zamykania obiektu z rysunku 5a. b) Widok z tyłu

Krok 01 AZOT można zaimplementować w postaci algorytmu o liniowej złożoności obliczeniowej. Krok 02 również ma liniową złożoność, ponieważ ustalenie liczby topologicznej woksela wymaga badania najbliższego sąsiedztwa tego piksela. Rozmiar tego sąsiedztwa jest stały, liczba wykonywanych operacji też jest stała i nie zależy od liczby wokseli obrazu. Podobnie etykietowanie połączonych komponentów można zrealizować za pomocą algorytmu liniowego w czasie. Algorytmy: GeodezyjnaOdległość, OHS i AZT są procedurami o liniowej złożoności obliczeniowej [1, 4]. Podsumowując, można stwierdzić, że wszystkie składowe AZOT mają liniową złożoność obliczeniową, stąd AZOT jest również liniowy w czasie.

Zaprezentowane algorytmy zaimplementowano w języku C, korzystając z biblioteki PINK (Pink Is Not Knor) [9], o otwartych źródłach, która zawiera szereg algorytmów analizy i przetwarzania obrazów 3D na bazie topologii dyskretnej i geometrii dyskretnej. Do testów zastosowano komputer klasy PC z procesorem czterordzeniowym. Dla testowych obrazów o rozmiarze $200 \times 200 \times 200$ pikseli czas działania AZOT nie przekraczał jednej minuty.

6. Wnioski

W niniejszym artykule autorzy zaprezentowali wydajny algorytm zamykania otworów i tuneli w obiektach wolumetrycznych, oparty na dobrze zdefiniowanych pojęciach matematycznych. Wyniki testowania algorytmu dla dwóch obrazów wolumetrycznych zaprezentowano w artykule. Istotną zaletą algorytmu jest liniowa złożoność obliczeniowa. Zgodnie z wiedzą autorów jest to pierwszy algorytm zamykający tunele i otwory w binarnych obrazach 3D.

Literatura

- [1] Aktouf Z., Bertrand G., Perroton L., *A three-dimensional holes closing algorithm*. Pattern Recognition Letters, Elsevier, nr 23, t. 5, 2002, 523–31.
- [2] Allen B., Curless B., Popovic Z., *The space of human body shapes: reconstruction and parameterization from range scans*. ACM Transactions on Graphics, t. 3, nr 22, 2003.

- [3] Bertrand G., *Simple points, topological numbers and geodesic neighborhoods in cubic grids*. Pattern Recognition Letters, t. 10, nr 15, 1994, 1003–1011.
- [4] Bertrand G., Couprie M., *Transformations topologiques discretées*. Géométrie discrete et images numériques. D. Coeurjolly, A. Montanvert and J.-M. Chassery, Hermes, 2007, 187–209.
- [5] Bertrand G., Couprie M., *On parallel thinning algorithms: minimal non-simple sets, P-simple points and critical kernels*. Journal of Mathematical Imaging and Vision, t. 1, nr 35, 2009, 23–35.
- [6] Blanz V., Mehl A., Vetter T., *et al.*, *A statistical method for robust 3d surface reconstruction from sparse data*. Proc. of the 3D Data Processing, Visualization, and Transmission, 2nd International Symposium, 2004.
- [7] Breckon T.P., Fisher R.B., *Amodal volume completion: 3d visual completion*. Computer Vision and Image Understanding, t. 3, nr 99, 2005, 499–526.
- [8] Breckon T.P., Fisher R.B., *Three-Dimensional Surface Relief Completion Via Nonparametric Techniques*. IEEE Transactions on Pattern Analysis and Machine Intelligence, t. 12, nr 30, 2008, 2249–2255.
- [9] Couprie M., Najman L., Daragon X., *et al.*, PINK library. Computer Program 2011s, <http://www.esiee.fr/~coupriem/pink.tgz>.
- [10] Curless B., Levoy M., *A volumetric method for building complex models from range images*. SigGraph-96.
- [11] Davis J., Marschner S.R., Garr M., *et al.*, *Filling Holes in Complex Surfaces Using Volumetric Diffusion*. First International Symposium on 3D Data Processing, Visualization, and Transmission, 2001.
- [12] Ju T., *Fixing geometric errors on polygonal models: a survey*. Journal of Computer Science and Technology, t. 1, nr 24, 2009, 19–29.
- [13] Kong T.Y., Rosenfeld A., *Digital topology: Introduction and survey*. Computer Vision, Graphics, and Image Processing, t. 3, nr 48, 1989, 357–393.
- [14] Liepa P., *Filling holes in meshes*. SGP '03: Proceedings of the 2003 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing, Aire-la-Ville, Switzerland, Switzerland, 2003.
- [15] Lorensen W.E., Jolesz F.A., Kikinis R., *The Exploration of Cross-Sectional Data with a Virtual Endoscope*, 1995.
- [16] Sagawa R., Ikeuchi K., *Hole Filling of a 3D Model by Flipping Signs of a Signed Distance Field in Adaptive Resolution*. IEEE Transactions on Pattern Analysis and Machine Intelligence, nr 30, t. Journal Article, 2008, 686–699.
- [17] Verdera J., Caselles V., Bertalmio M. *et al.*, *Inpainting surface holes*. Int. Conference on Image Processing, 2003.
- [18] Xiao C., Zheng W., Miao Y. *et al.*: *A unified method for appearance and geometry completion of point set surfaces*. The Visual Computer, 2007, 433–443.
- [19] Janaszewski M., Couprie M., Babout L., *Hole filling in 3D volumetric objects*. Pattern Recognition, 43(10), 2010, 3548–3559.