

Marcin Janaszewski^{1,2}, Laurent Babout¹

¹Katedra Informatyki Stosowanej, Politechnika Łódzka.

²Katedra Systemów Ekspertowych i Sztucznej Inteligencji,
Wyższa Szkoła Informatyki w Łodzi

PRZEGLĄD NAJPOPULARNIEJSZYCH KONCEPCJI SZKIELETYZACJI OBRAZÓW 3D

Streszczenie – Szkieletyzacja obrazów to technika upraszczająca wizualną reprezentację obiektu do zbioru połączonych, cienkich (grubości jednego piksela) łuków i odcinków. W przypadku obrazów 3D na szkielet mogą składać się dodatkowo powierzchnie. Na przestrzeni kilkudziesięciu lat badań nad metodami szkieletyzacji wypracowano wiele rozwiązań, które można zaliczyć do trzech podstawowych kategorii: algorytmy pocieniania, algorytmy oparte o diagram Voronoi oraz koncepcje budowane na bazie transformacji odległości. Wielość rozwiązań spowodowana jest tym, że do dnia dzisiejszego nie zbudowano jeszcze algorytmu szkieletyzacji, który byłby uniwersalny, zwracający zadowalające wyniki dla wszystkich typów obrazów. Stworzone dotychczas rozwiązania posiadają istotne wady, co ogranicza ich zastosowanie do pewnych typów obrazów. Niniejszy artykuł charakteryzuje przedstawicieli trzech wspomnianych wyżej klas algorytmów szkieletyzacji obrazów 3D. Ponadto autorzy zamieścili dyskusję wad i zalet proponowanych rozwiązań.

1 Wstęp

Szkieletyzacja zwykle należy do algorytmów wstępnego przetwarzania obrazu realizowanego w celu ilościowego opisu cech obiektu. Reprezentacja obiektu w postaci szkieletu zachowuje szereg cech kształtu oraz strukturę topologiczną oryginalnego obiektu. Zamiast badać pełną reprezentację zwykle łatwiej i szybciej realizowane jest badanie szkieletu. Przykładowo, jeżeli stosujemy algorytm szkieletyzacji, który zachowuje topologie obiektu to zadanie lokalizacji otworów w obiekcie z pewnością będzie prostsze do wykonania na szkielecie niż na pełnej wizualnej reprezentacji. Algorytmy szkieletyzacji stosowane są do budowania reprezentacji i rozpoznawania obiektów w następujących dziedzinach: rekonstrukcja obrazów np. [11], grafika komputerowa np. [26], rozpoznawanie znaków np. [20, 33], przetwarzanie obrazów np. [21], analiza obrazów medycznych np. [19, 27]. Wielu naukowców realizowało rozpoznawanie obiektów poprzez dopasowanie struktur reprezentowanych przez drzewa i grafy oparte na szkieletach tych obiektów [9, 29].

Niestety pojęcie szkieletu obiektu nie posiada ścisłej definicji. Naukowcy budując algorytmy szkieletyzacji kierują się jedynie zbiorem postulatów, które generowany szkielet powinien spełniać. Rezultatem takiej sytuacji jest ogromna liczba publikacji dotyczących nowych algorytmów szkieletyzacji obiektów. Większość z proponowanych algorytmów wykorzystuje różne definicje i parametry oraz jest testowana na niewielkiej liczbie wybranych obrazów. Ponadto autorzy realizują tylko jakościową ocenę rezultatów działania swoich algorytmów. Co gorsza na podstawie tej oceny wyciągają niesłuszne ogólne wnioski co do właściwości proponowanych rozwiązań np. [34]. Kolejna wada wielu proponowanych metod to dwuetapowa albo nawet trójetapowa generacja szkieletu: pierwszy etap to wstępne przetwarzanie obrazu w celu dostosowania obiektu do realizacji właściwego - drugiego etapu generowania szkieletu i trzeci etap polegający na zastosowaniu wyszukanych algorytmów poprawiania rezultatu drugiego etapu tak aby możliwa była pozytywna ocena ostatecznego szkieletu obiektu.

Konsekwencją powyższej sytuacji są problemy z wyborem algorytmu szkieletyzacji dla konkretnej klasy obrazów. Ponadto zgodnie z wiedzą autorów nie ma biblioteki przetwarzania obrazów 3D, która udostępniłaby implementacje najważniejszych algorytmów szkieletyzacji. Taka biblioteka umożliwiłaby szybkie wybranie odpowiedniego algorytmu dla konkretnego zastosowania.

W niniejszym artykule autorzy charakteryzują najważniejszych przedstawicieli trzech głównych klas algorytmów szkieletyzacji mając nadzieję, że ta charakterystyka pomoże w wyborze odpowiedniego algorytmu szkieletyzacji dla konkretnej klasy obiektów.

2 Podstawowe pojęcia. Koncepcja szkieletu obiektu

Zanim wprowadzone zostanie pojęcie szkieletu należy zdefiniować kilka podstawowych pojęć dotyczących obrazów 2D i obrazów 3D reprezentujących obiekty posiadające objętość. Obraz 2D definiowany jest jako zbiór pikseli – najmniejszych, niepodzielnych jednostkowych kwadratów obrazu. Każdy piksel opisany jest przez trójkę: (x, y, v) gdzie (x, y) reprezentuje lokalizację piksela natomiast v w rozważanych w tym artykule obrazach reprezentuje przynależność piksela. Jeżeli $v = 0$ to piksel należy do tła, $v = 1$ oznacza, że piksele należy do obiektu. Analogicznie definiowany jest obraz 3D reprezentujący obiekt posiadający objętość jako zbiór wokseli – jednostkowych sześciątów, z których każdy opisany jest następującą czwórką: (x, y, z, v) , gdzie (x, y, z) definiuje lokalizację woksela natomiast v identyfikuje jego przynależność ($v = 0$ woksel należy do tła, $v = 1$ woksel należy do obiektu). W przypadku obrazu 2D piksel może mieć maksymalnie 8 sąsiednich pikseli. Przy czym te, z którymi dzieli krawędź nazywane są jego e-sąsiadami nato-

miast te, z którymi dzieli wierzchołek nazywane są jego v-sąsiadami. W przypadku obrazów 3D woksel a może mieć aż 26 sąsiadów. Te woksele, które dzielą z a bok nazywane są jego f-sąsiadami, te które dzielą z a krawędź nazywane są e-sąsiadami natomiast te, które dzielą z nim wierzchołek nazywane są jego v-sąsiadami.

Ponieważ niektóre pojęcia mają taką samą definicję dla obrazów 3D i 2D to w celu uproszczenia zapisu w dalszej części artykułu pojęcie punkt będzie oznaczało zarówno piksel jak i woksel.

Punkt obiektu nazwany jest *punktem wewnętrznym* tego obiektu, jeżeli wszyscy jego sąsiedzi należą do tego obiektu. Punkt obiektu nazwany jest *punktem brzegowym* tego obiektu, jeżeli posiada przynajmniej jednego sąsiada należącego do tła. Punkt należący do tła nazwany jest *punktem zewnętrznym*, jeżeli wszyscy jego sąsiedzi należą do tła.

Niech $X \subset R^n, n=2,3$ będzie zbiorem punktów obrazu. Kulą o środku w punkcie x i promieniu $r \in R_+ \cup \{0\}$ nazywamy zbiór $S_r(x) = \{p \in X : d(x, p) \leq r\}$ gdzie d jest metryką zdefiniowaną na zbiorze X .

Pojęcie szkieletu dla obrazów 2D wprowadził H. Blum [6] definiując go jako rezultat transformaty osi środkowej MAT (ang. *Medial Axis Transformation*). Według tej definicji punkt wewnętrzny p obiektu O należy do *szkieletu* O (osi środkowej O) wtedy i tylko wtedy, gdy istnieje kula $S(p) \subset O$ taka, że nie istnieje żadna inna kula $S \subset O$ taka, że $S(p) \subset S$. Kulę $S(p)$ spełniającą warunek opisany w poprzednim zdaniu nazywamy *maksymalną kulą*.

Dla obrazów 2D, gdzie piksele, opisane za pomocą dwóch współrzędnych położenia stanowią zbiór dyskretny zwykle wykorzystuje się metrykę miejską zdefiniowaną następująco:

$$\forall (x_1, y_1), (x_2, y_2) \in O : d((x_1, y_1), (x_2, y_2)) = |x_2 - x_1| + |y_2 - y_1| \quad (1)$$

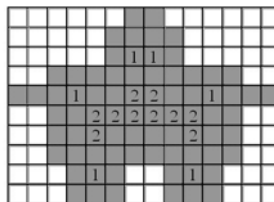
Gdzie: $(x_1, y_1), (x_2, y_2)$ to współrzędne reprezentujące piksele natomiast O reprezentuje obraz.

Na rys. 1 zaprezentowano fragment obrazu 2D. Każdy mały prostokąt reprezentuje piksel obrazu. Liczba wewnątrz prostokąta oznacza odległość piksela od piksela oznaczonego 0 zgodnie z metryką miejską. Dodatkowo kolorem szarym oznaczono kulę w środku w pikselu 0 i promieniu 3. Na rys. 2 zaprezentowano obiekt wyróżniony kolorem szarym oraz jego szkielet wyznaczony według definicji z wykorzystaniem metryki miejskiej. Piksel należący do szkieletu wyróżniony jest za pomocą cyfry reprezentującej promień maksymalnej kuli zawartej w obiekcie o środku w tym pikselu. Jak widać na rys. 2 szkielet wyznaczany z definicji nie

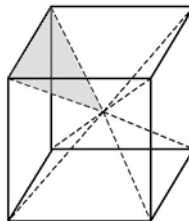
posiada podstawowych – pożądanymi właściwościami szkieletu: nie zachowuje topologii obiektu oraz nie jest grubości jednego piksela.

6	5	4	3	4	5	6
5	4	3	2	3	4	5
4	3	2	1	2	3	4
3	2	1	0	1	2	3
4	3	2	1	2	3	4
5	4	3	2	3	4	5
6	5	4	3	4	5	6

Rys. 1. Przykładowy obiekt wyróżniony kolorem szarym oraz jego szkielet złożony z pikseli zawierających cyfrę. Cyfra wewnątrz piksela oznacza promień maksymalnej kuli zawartej w obiekcie o środku w tym pikselu.



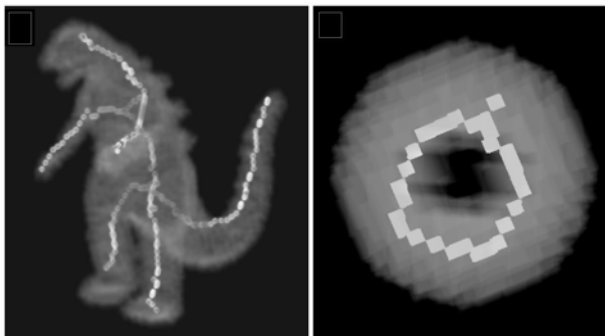
Rys. 2. Kula o promieniu 3 i środku w pikselu 0 wyróżniona kolorem szarym. Wartości liczbowe określają odległość piksela od środka kuli 0.



Rys. 3. Powierzchnie środkowe kostki sześciennej wyznaczone przez linie przerwane i krawędzie kostki. Jedna z powierzchni została wyróżniona kolorem szarym.

W przypadku obiektów 3D definiowana jest powierzchnia środkowa jako zbiór wokseli będących centrami maksymalnych kul. Na rys. 3 za-

prezentowano powierzchnie środkowe dla kostki sześcienniej. Każda krawędź kostki wraz z dwoma wychodzącymi z jej końców liniami przerywanymi tworzy powierzchnię środkową. Stąd dla kostki sześcienniej mamy 12 takich powierzchni.



Rys. 4. Przykładowe obiekty 3D oraz ich szkielety wygenerowane algorytmem pocieniania [7]

Ze względu na prostotę reprezentacji w wielu aplikacjach dotyczących obrazów 3D bardziej pożądanym jest szkielet w postaci krzywych i łuków. Taka reprezentacja obiektu jest zwana *liniami środkowymi* (ang. centerline) lub inaczej *szkieletem krzywoliniowym* (ang. curve-skeleton) [28] i stanowi uproszczoną, jednowymiarową reprezentację powierzchni środkowych. Na Rys. 4 zaprezentowano przykładowe szkielety krzywoliniowe obiektów 3D, wygenerowane za pomocą algorytmu pocieniania. Rys. 4 b) reprezentuje kulę z otworem. Dla obu obiektów otrzymano szkielet, który nie spełnia postulatu centryczności (rozdział **Błąd! Nie zdefiniowano zakładki.**). Natomiast w obu przypadkach szkielety zachowują topologie obiektu (postulat zachowania topologii rozdział **Błąd! Nie zdefiniowano zakładki.**).

3 Postulaty dotyczące szkieletyzacji

Postulaty dotyczące właściwości generowanego szkieletu krzywoliniowego różnią się w zależności od aplikacji. Istnieje jednak pewien zbiór poświadczonych właściwości, które postulowane są w większości publikacji niezależnie od zastosowania. Do tych właściwości należą: zachowanie topologii obiektu (ang. topology preserving), centryczność (ang. centeredness), połączeniowość (ang. connectivity), odporność na zakłócenia (ang. robust), grubość jednego piksela (ang. thin)

Zachowanie topologii – Dwa obiekty mają tę samą topologię, jeżeli składają się z jednakowej liczby rozłącznych komponentów, mają jednakową liczbę otworów i wgłębień. Przy czym otwór dla obiektów 3D nie stanowi zbioru pikseli. W literaturze znana jest jedynie definicja określająca występowanie otworu w obiekcie. Zgodnie z tą definicją obiekt posiada otwór wtedy i tylko wtedy, gdy istnieje zamknięty ciąg wokseli, którego nie można przekształcić z wykorzystaniem prostych transformacji do jednego wokseli. Zgodnie z definicją, kula i sfera nie posiadają otworu natomiast torus posiada jeden otwór. Szczegóły dotyczące definicji wspomnianych prostych transformacji można znaleźć w [13]. Wgłębienie definiowane jest jako zbiór połączonych wokseli tła otoczonych wokselami obiektu (pusta przestrzeń wewnątrz obiektu). Porównanie topologii obiektu i jego szkieletu krzywoliniowego wymaga modyfikacji powyższej definicji gdyż szkielet krzywoliniowy nie ma wgłębień. Dlatego na potrzeby szkieletyzacji podaje się następującą definicję zachowania topologii: szkielet krzywoliniowy zachowuje topologię obiektu oryginalnego O jeżeli ma taką samą liczbę rozłącznych komponentów i przynajmniej tyle samo pętli, co otworów i wgłębień w O .

Centryczność – szkielet krzywoliniowy powinien geometrycznie znajdować się w centrum obiektu lub jak najbliższej tego centrum względem brzegu tego obiektu.

Połączeniowość – Jeżeli rozpatrujemy szkielet jednego rozłącznego komponentu to ze względu na zachowanie topologii szkielet powinien stanowić jeden komponent połączonych wokseli. Inaczej mówiąc wewnątrz gałęzi szkieletu nie powinno być przerw. Oznacza to, że każdy woksel szkieletu z wyjątkiem wokseli kończących gałęzie powinien mieć przynajmniej dwóch sąsiadów należących do tego szkieletu. Ta cecha jest trudna do osiągnięcia, gdy brzeg obiektu jest bardzo nieregularny. Dla niektórych algorytmów szkieletyzacji trzeba stosować specjalne, wyszukane algorytmy poprawiania wygenerowanego szkieletu w celu osiągnięcia połączeniowości np. [34].

Oporność na zakłócenia – szkielet obiektu powinien być odporny na zakłócenia brzegu obiektu. Oznacza to że szkielet obiektu, którego brzeg jest zaszumiony powinien jak najmniej różnić się od szkieletu tego samego obiektu, którego brzeg nie jest zaszumiony. Właściwość ta jest szczególnie pożądana dla zadań rozpoznawania obiektów na podstawie ich szkieletów. Często dobre efekty w spełnieniu tego postulatu można uzyskać poprzez obcinanie szkieletu (ang. pruning) polegające na eliminacji drobnych gałęzi, które nie reprezentują najważniejszych właściwości geometrycznych rozpoznawanego obiektu np. [2].

Grubość jednego piksela – szkielet krzywoliniowy powinien być jednowymiarowy stąd postuluje się jego grubość o rozmiarze jednego piksela we wszystkich kierunkach, z wyjątkiem wokseli, w których łączą się gałęzie (woksele łączące). Woksele łączące mają więcej niż dwóch

sąsiadów, stąd grubość szkieletu badana w tych wokselach jest większa od 1.

Dodatkowo istnieją pewne postulaty dotyczące algorytmów generujących szkielety 3D. Ponieważ algorytmy te przetwarzają zwykle duże struktury danych, których rozmiar liczony jest w dziesiątkach i setkach megabajtów to podczas projektowania tych algorytmów powinien być położony szczególny nacisk na jak najefektywniejsze wykorzystanie zasobów komputera. Dotyczy to zarówno mocy obliczeniowej jak i pamięci operacyjnej. Ponadto w niektórych aplikacjach np. [34] wymaga się aby szkielet krzywoliniowy reprezentowany był w postaci grafu, gdzie każdy wksel łączący oraz wksel końcowy gałęzi szkieletu stanowi węzeł grafu a każdy ciąg wokseli szkieletu, nie zawierający wokseli łączących i końcowych, spinający dwa woksela łączące lub wksel łączący z końcowym tworzy gałąź grafu.

4 Algorytmy szkieletyzacji oparte na pocienianiu obiektu

Algorytmy oparte na pocienianiu tworzą szkielet poprzez iteracyjne usuwanie wokseli obiektu. W każdej iteracji analizują brzeg aktualnego obiektu – rezultatu przetwarzania z poprzedniej iteracji w celu usunięcia tzw. punktów prostych (ang. simple points), tzn. takich, których usunięcie nie powoduje zmiany topologii obiektu np. [32]. Algorytm kończy swoje działanie, kiedy otrzymany na pewnej iteracji obiekt nie ma już wokseli prostych. Testowanie czy wksel jest prosty realizowane jest z wykorzystaniem zbioru masek zwanych szablonami (ang. templates), które zwykle mają rozmiar 3x3x3 woksela. Według [18] algorytmy szkieletyzacji oparte na pocienianiu można podzielić ze względu na dwa kryteria:

1 - sposób organizacji iteracji,

2 - wybór typów punktów prostych przeznaczonych do usunięcia.

Podział ze względu na kryterium pierwsze daje trzy typy algorytmów:

- Algorytmy, które nie dzielą iteracji na poditeracje [14, 15]. W celu zachowania topologii algorytmy tego typu wyszukują punkty proste wykorzystując jednocześnie dwie maski o rozmiarach 5x5x5 i 3x3x3.
- Algorytmy te wykorzystują maski o rozmiarze 3x3x3 w celu zbadania sąsiedztwa punktów brzegowych obiektu. Każda iteracja algorytmu dzielona jest na poditeracje. W każdej poditeracji realizowane jest poszukiwanie punktów przeznaczonych do usunięcia w jednym unikalnym kierunku. Wszystkie poditeracje wykonywane są równolegle. Algorytmy tego typu nazwane są *kierunkowymi*. Ponieważ dla obrazów 3D wyróżniamy 6 podstawowych kierunków to większość algorytmów kierunkowych wykorzystuje 6 poditeracji. Spotykane

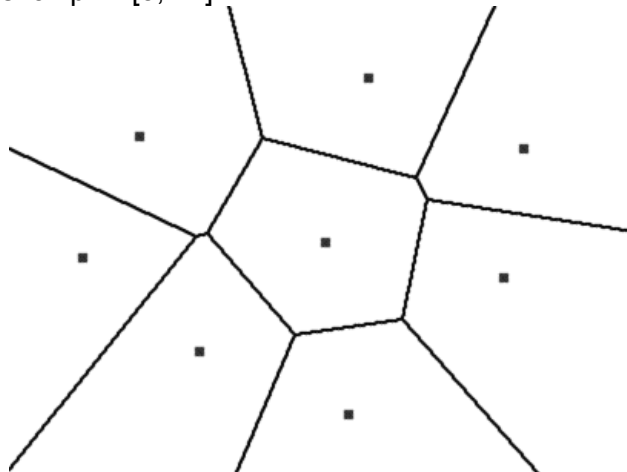
są również rozwiązania, w których stosowanych jest 12 kierunków poszukiwań [18].

Trzecie podejście to algorytmy sekwencyjne dzielące obiekt na kilka rozłącznych podzbiorów. W danej iteracji algorytm poszukuje punktów do usunięcia tylko na brzegu jednego z wyznaczonych podzbiorów. Tego typu algorytmy wykorzystują 8 podzbiorów, co zapewnia zachowanie topologii np. [4, 24].

Drugie kryterium dzieli algorytmy pocieniania na takie, które kasują wszystkie proste punkty określonego typu niebędące punktami końcowymi np. [4] i takie, które wykorzystują zbiór specjalnych masek w celu identyfikacji punktów przeznaczonych do skasowania np. [14, 15, 18, 24].

5 Algorytmy szkieletyzacji oparte na diagramach Voronoi

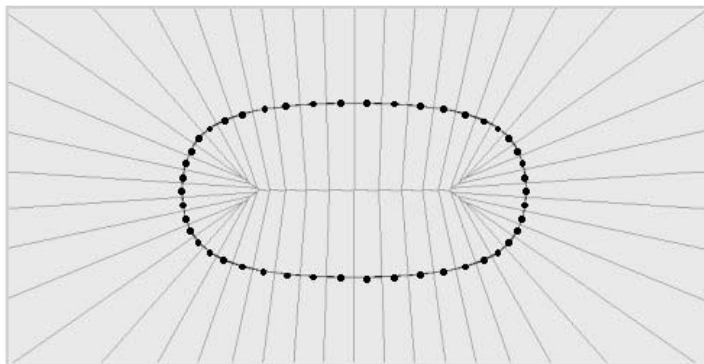
Diagram Voronoi dla zbioru punktów generujących to podział przestrzeni na podzbiory w ten sposób, że każdy podzbiór zawiera dokładnie jeden punkt generujący p i wszystkie punkty, dla których p jest najbliższy spośród wszystkich punktów generujących. Rys. 5 prezentuje osiem punktów generacyjnych na płaszczyźnie oraz granice diagramu Voronoi dla tych punktów. Więcej informacji na temat diagramu Voronoi można znaleźć np. w [3, 22]



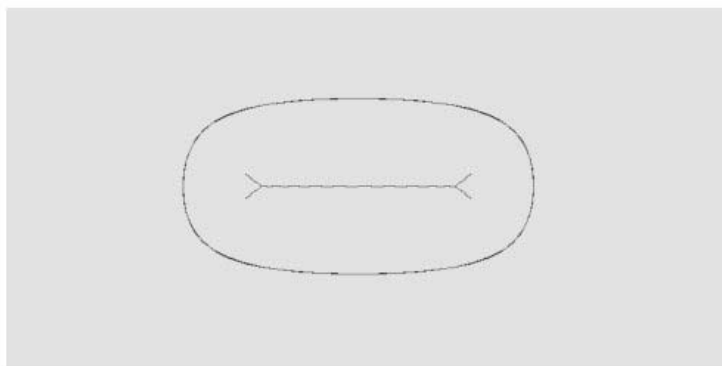
Rys. 5. Diagram Voronoi dla ośmiu punktów generacyjnych.

Szkielet w oparciu o diagram Voronoi generowany jest w dwóch etapach. Pierwszy etap polega na wyborze punktów generujących na brzegu obiektu i wygenerowaniu dla tych punktów diagramu Voronoi. Drugi etap polega na usuwaniu punktów z gałęzi diagramu Voronoi tak

aby pozostały tylko punkty należące do osi środkowych. Na rys. 6 zaprezentowano diagram Voronoi dla obiektu owalnego 2D, którego granica oraz punkty generujące na tej granicy zostały zaznaczone kolorem czarnym. Na rys. 7 pokazano efekt działania drugiego etapu generacji szkieletu w oparciu o diagram Voronoi dający ostateczny wynik.



Rys. 6. Diagram Voronoi dla obiektu, którego granica oraz punkty generujące na tej granicy zaznaczono kolorem czarnym.



Rys. 7. Szkielet wyznaczony w oparciu o diagram Voronoi zaprezentowany na rys. 6

W przypadku obiektów 3D wyznaczanie diagramu Voronoi wymaga wielokrotnie większych zasobów komputera niż dla obiektów 2D. Zwykle obrazy 3D wykorzystywane w praktyce takie jak CT lub MRI zawierają bardzo dużą liczbę wokseli oraz szum, co powoduje, że wyznaczenie Diagramu Voronoi wymaga wielu punktów generujących (dziesiątki lub

setki razy więcej niż w przypadku obiektów 2D). Ponadto wygenerowany diagram jest bardzo gęsty, co dodatkowo utrudnia i wydłuża czas jego obcinania do osi środkowych. Propozycje wyznaczania Diagramu Voronoi dla obiektów 3D można znaleźć np. w [1, 4]. Opis algorytmów realizujących redukcję Diagramu Voronoi do osi środkowych dla obiektów 3D zaprezentowano w [10, 17].

6 Algorytmy szkieletyzacji oparte o transformatę odległości

Transformata odległości definiowana jest dla każdego punktu wewnętrznego p obiektu O w przestrzeni 3D jako odległość p od brzegu O . W praktyce zamiast wyliczania odległości stosuje się jej aproksymacje z wykorzystaniem tzw. algorytmu kodowania wokseli.

Zgodnie z opisem zaprezentowanym w [34] algorytm ten jest rekursywną propagacją i przypisywaniem wokselom obiektu wartości całkowitych rozpoczynając od wokseli brzegowych obiektu. Podczas propagacji wykorzystuje się specjalne schematy kodowania opisane za pomocą trzech liczb całkowitych n_f , n_e , n_v ($n_f < n_e < n_v$). Najpierw wszystkie woksele obiektu otrzymują kod równy umownie nieskończoności. Następnie propagacja startuje od wokseli brzegowych obiektu, którym przypisywany jest kod 0. Dalej dla każdego wokseli brzegowego jego każdy f-sąsiad, e-sąsiad i v-sąsiad należący do obiektu otrzymuje odpowiednio kod n_f , n_e , n_v . W i -tej iteracji przetwarzany jest każdy wksel, którego kod był modyfikowany w iteracji $i-1$. Załóżmy, że w $i-1$ iteracji wokselowi p przypisano kod n . Wtedy w i -tej iteracji każdy jego f-sąsiad, e-sąsiad i v-sąsiad należący do obiektu otrzymuje odpowiednio kod $n + n_f$, $n + n_e$, $n + n_v$, pod warunkiem, że nowy kod jest mniejszy od aktualnie przypisanego. Algorytm kodowania wokseli zatrzymuje się, gdy w kolejnej iteracji nie ma już wokseli przeznaczonych do przetwarzania. Wybór schematu kodowania zależy od zastosowania. Najczęściej używane schematy to $n_f = 1$, $n_e = 2$, $n_v = 3$ (w skrócie 1-2-3), 2-3-5; 3-4-5. Jeżeli wokselami startowymi są woksele brzegu obiektu wynik transformaty przeprowadzonej według powyższego schematu nazywamy polem BS (ang. BS-field). Maksima lokalne oraz woksele siodłowe pola BS, dalej zwane wokselami grzbietowymi, leżą w centrum obiektu. Stąd te woksele stają się potencjalnymi kandydatami na elementy szkieletu. Aby otrzymać cienki i nieprzerwany szkielet należy zastosować dodatkowe przetwarzanie, w szczególności eliminację części wokseli stanowiących maksima lokalne pola BS. Stąd algorytmy te zwykle działają według czterech następujących kroków:

- generacja pola BS,
- identyfikacja wokseli grzbietowych,
- selekcja wokseli grzbietowych,
- łączenie wyselekcjonowanych wokseli.

Identyfikacja wokseli grzbietowych realizowana jest na wiele różnych sposobów. Najpopularniejsze podejścia to: pocienianie na bazie pola BS np. [8, 12, 23, 30], analiza gradientu np. [5], progowanie dwusiecznej kąta np. [16], kurczenie obiektu zgodnie z gradientem transformaty odległości np. [25]. Selekcja wokseli grzbietowych realizowana jest najczęściej poprzez pocienianie np. [30] lub algorytm wyznaczania najkrótszej ścieżki np. [34]. Łączenie wyselekcjonowanych wokseli realizowane jest przez wyznaczanie najkrótszej ścieżki np. [31] lub minimalnego drzewa spinającego lub innych algorytmów operujących na grafach.

W niektórych podejściach oprócz pola BS wykorzystuje się pole generowane podobnie jak pole BS z tym, że występuje jeden szczególnie wybrany woxsel startowy [34].

7 Posumowanie

Niniejszy artykuł prezentuje trzy główne kategorie algorytmów szkieletyzacji obrazów 3D. Analiza literatury dotyczącej szkieletyzacji obrazów 3D pokazuje, że nie zbudowano jeszcze algorytmu uniwersalnego, który dla dowolnego obiektu 3D generowałby zadowalający szkielet, biorąc pod uwagę każde z kryteriów zaprezentowane w rozdziale **Błąd! Nie zdefiniowano zakładki..** Mnogość rozwiązań w tej dziedzinie spowodowane jest szerokim zastosowaniem algorytmów szkieletyzacji, które wielokrotnie budowane są pod kątem wybranego typu obrazów. W większości algorytmów szkieletyzacji można wyróżnić dwa etapy. Pierwszy to tzw. rdzeń, który zbudowany jest w oparciu o jedną z wyżej opisanych koncepcji. Drugi etap, silnie zależny od zastosowania, to poprawa szkieletu wygenerowanego na etapie pierwszym.

Algorytmy należące do każdej z opisywanych w niniejszym artykule klas mają swoje wady i zalety. Do zalet algorytmów pocieniania należy generacja cienkiego, spełniającego postulat połączeniowości (patrz rozdział 0) szkieletu zachowującego topologię obiektu ale niekoniecznie spełniającego postulat centryczności Rys. 4. Niestety złożoność obliczeniowa pozostawia wiele do życzenia. Algorytmy oparte o Diagram Voronoi zwracają cienki, zachowujący połączeniowość szkielet ale ze względu na złożoność obliczeniową ich zastosowanie sprowadza się do obiektów opisanych za pomocą wielokątów lub małych obiektów opisanych za pomocą wokseli. Bezsprzecznie najszybsze są algorytmy oparte o transformację odległości. Wadą tego podejścia jest jednak generacja szkieletu, który może nie zachowywać topologii oraz nie spełnia postulatu połączeniowości, szczególnie dla obiektów szerokich, cienkich z wieloma otworami. Aby uzyskać połączeniowość trzeba stosować dodatkowe wyszukane algorytmy przetwarzania, co wydłuża czas generacji szkieletu.

Wady i zalety prezentowanych rozwiązań powodują, iż odpowiedni algorytm szkieletyzacji dobiera się do zastosowania i typu przetwarzanego obrazu. Stąd pożądane byłoby stworzenie narzędzia, które wykorzystując metody inteligencji obliczeniowej dobierałoby najlepszy algorytm szkieletyzacji dla konkretnego obrazu. Autorzy analizując bogatą literaturę dotyczącą szkieletyzacji obiektów 3D nie napotkali publikacji prezentującej takie narzędzie.

Literatura

- [1] Attali D., Montanvert A.: Semicontinuous skeletons of 2D and 3D shapes, presented at 2nd International Workshop on Visual Form, Singapore, 1994.
- [2] Bai X., Latecki L., Liu J.: Skeleton Pruning by Contour Partitioning with Discrete Curve Evolution, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 29, pp. 449--462, 2007.
- [3] Berg M., Kreveld M., Overmars M., Schwarzkopf O.: *Geometria obliczeniowa, algorytmy i zastosowania*: WNT, 2007.
- [4] Bertrand G., Aktouf Z.: A three-dimensional thinning algorithm using subfields, *Proceedings of the SPIE The International Society for Optical Engineering*, vol. 2356, pp. 113-24, 1995.
- [5] Bitter I., Kaufman A. E., Sato M.: Penalized-distance volumetric skeleton algorithm, *IEEE Transactions on Visualization and Computer Graphics*, vol. 7, pp. 195-206.
- [6] Blum H.: Transformation for Extracting New Descriptions of Shape, in *Models for the Perception of Speech and Visual Form*, 1967, pp. 362--380.
- [7] Cornea N., Silver D., Min P.: Curve-skeleton applications, <http://www.caip.rutgers.edu/~cornea/CurveSkelApp/>, 2007.
- [8] Couprie M., Zrou R.: Discrete bisector function and Euclidean skeleton.
- [9] Di Ruberto C.: Recognition of shapes by attributed skeletal graphs, *Pattern Recognition*, vol. 37, pp. 21-23, 2004.
- [10] Etzion M., Rappoport A.: Computing Voronoi skeletons of a 3-D polyhedron by space subdivision, *Computational Geometry*, vol. 21, pp. 87-120, 2002.
- [11] Ferley E., Cani M.-P., Attali D.: Skeletal Reconstruction of Branching Shapes, *Computer Graphics Forum*, vol. 16, pp. 283-293, 1997.
- [12] Gagvani N., Silver D.: Parameter-controlled volume thinning, *Graphical Models and Image Processing*, vol. 61, pp. 149-64.

- [13] Kong T. Y.: A digital fundamental group, *Computer Graphics*, vol. 13, pp. 159-166, 1989.
- [14] Ma C. M.: A 3D fully parallel thinning algorithm for generating medial faces, *Pattern Recognition Letters*, vol. 16, pp. 83–87, 1995.
- [15] Ma C. M., Sonka M.: A fully parallel 3D thinning algorithm and its applications, *Computer Vision and Image Understanding*, vol. 64, pp. 420–433, 1996.
- [16] Malandain G., Fernandez Vidal S.: Euclidean skeletons, *Image and Vision Computing*, vol. 16, pp. 317-27.
- [17] Naf M., Szekely G., Kikinis R., Shenton M. E., Kubler O.: 3D Voronoi skeletons and their usage for the characterization and recognition of 3D organ shape, *Computer Vision and Image Understanding*, vol. 66, pp. 147-161, 1997.
- [18] Palágyi K., Kuba A.: A parallel 3D 12-subiteration thinning algorithm, *Graphical Models and Image Processing*, vol. 61, pp. 199-221, 1999.
- [19] Palágyi K., Sorantin E., Balogh E., Kubal A., Halmai C., Erdohelyi B., Hausegger K.: A sequential 3D thinning algorithm and its medical applications, presented at *Information-Processing-in-Medical-Imaging.-17th-International-Conference*, 2001.
- [20] Pervouchine V., Leedham G, Melikhov K.: Handwritten character skeletonisation for forensic document analysis, in *Proceedings of the 2005 ACM symposium on Applied computing*. Santa Fe, New Mexico: ACM Press, 2005, pp. 754-758.
- [21] Pizer S. M., Gerig G., Joshi S., Aylward S. R.: Multiscale medial shape-based analysis of image objects, *Proceedings of the IEEE*, vol. 91, pp. 1670-9.
- [22] Preparata F. P., Shamos M. I.: *Computational Geometry*. New York: Springer-Verlag, 1985.
- [23] Pudney C.: Distance-based skeletonization of 3D images.
- [24] Saha P. K., Chaudhuri B. B., Majumder D. D.: A new shape preserving parallel thinning algorithm for 3D digital images, *Pattern Recognition*, vol. 30, pp. 1935-55, 1997.
- [25] Schirmacher H., Zockler M., Stalling D., Hege H. C.: Boundary surface shrinking—a continuous approach to 3D center line extraction.
- [26] Shamir A., Shaham A.: Skeleton based solid representation with topology preservation, *Graphical Models*, vol. 68, pp. 307-21.
- [27] Sorantin E., Halmai C., Erdohelyi B., Palágyi K., Nyúl L. G., Ollé L. K., Geiger B., Lindbichler F., et al.: Spiral-CT-based assessment of

- tracheal stenoses using 3-D-skeletonization, IEEE Transactions on Medical Imaging, vol. 21, pp. 263-273, 2002.
- [28] Svensson S., Nystrom I., Sanniti di Baja G.: Curve skeletonization of surface-like objects in 3D images guided by voxel classification, Pattern Recognition Letters, vol. 23, pp. 1419-26, 2002.
- [29] Torsello A., Hancock E. R.: A Skeletal Measure of 2D Shape Similarity, Computer Vision and Image Understanding, vol. 95, pp. 1 - 29, 2004.
- [30] Tran S., Shih L.: Efficient 3D binary image skeletonization, presented at Computational Systems Bioinformatics Conference Workshops, 2005.
- [31] Wade L., Parent R. E.: Automated generation of control skeletons for use in animation, Visual Computer, vol. 18, pp. 97-110.
- [32] Wang T., Basu A.: A note on a fully parallel 3D thinning algorithm and its applications, Pattern Recognition Letters, vol. 28, pp. 501--506, 2007.
- [33] Whichello A. P., Hong Y.: Reconstruction of character skeletons using Gabor filter features, Electronics Letters, vol. 31, pp. 1911-1912, 1995.
- [34] Zhou Y., Toga A. W.: Efficient skeletonization of volumetric objects, IEEE Transactions on visualization and computer graphics, vol. 5, pp. 196-209, 1999.

REVIEW OF POPULAR SKELETON CONCEPTION OF 3D IMAGES

Summary – The skeleton of images is the technique simplifying visual representation of object to collection joined, thin (thickness of one pixel) arcs and segments. In case of 3D images, the skeleton can contains surfaces too. Over some dozen years of the skeleton methods investigations, many solutions were worked out. It was possible to accept them to three basic categories: thinings algorithms, algorithms leaning on Voronoi diagram, as well as conceptions built on base of distance transformation .