DIGITÁLNA TOPOLÓGIA

Vojtech Jankovič, Ľudovít Niepel,
Katedra aplikovanej matematiky MFF UK, Univerzita Komenského
Mlynská dolina, 842 15 Bratislava, Slovensko.

0. ÚVOD.

Digitálna topológia sa objavila ako alternatívny modelovací prístup v počítačovej grafike len nedávno [Kauf91]. Jej podstata spočíva v predpoklade, že ľubovoľný objekt možno poskladať z konečného počtu menších trojrozmerných primitívov známych rozmerov a zloženia (=objemovej kvality alebo hustoty) Hlavným dôvodom pre takýto prístup sú digitálne údaje produkované rozličnými meracími a skanovacími zariadeniami (CT, NMR, PET, ultrazvuk) a ľahšia a rýchlejšia počítačová manipulácia a vizualizácia popisovaných objektov. V poslednom období sa začínajú objavovať aj nové hardwarové architektúry plne zohľadňujúce tento spôsob modelovania trojrozmerných scén.

Aj napriek tomu, že digitálna topológia je v podstate, rozšírenie rastrovej 2D grafiky do 3D, nemožno známe pojmy a algoritmy jednoducho zovšeobecniť. Ďalšia dimenzia totiž prináša nielen zvýšenie zložitosti, ale aj množstvo nových problémov.

1. d-ROZMERNÁ TESELÁCIA.

d-rozmerná teselácia (d = 2,3) je rozdelenie (rasterizácia) spojitého d-rozmerného priestoru do množiny (systému) malých dotýkajúcich sa d-rozmerných blokov. V našich úvahách budeme predpokladať, že tieto bloky sú vždy d-rozmerné mnohosteny (polytopy) - objekty, ktorých povrch pozostáva výhradne z lineárnych uzavretých m-rozmerných elementov ($0 \leq m < n$). V 2D priestore sú takýmito mnohostenmi rovinné polygóny, ktorých povrch pozostáva z vrcholov (0D) a hrán - úsečiek (1D). V 3D priestore sú takýmito mnohostenmi 3-rozmerné

(priestorové) mnohosteny, ktorých povrch pozostáva z vrcholov (0D), hrán - úsečiek (1D) a stien - rovinných polygónov (2D). Okrem toho v reálnych aplikáciách nesie každý polytop ešte číselnú informáciu, ktorá charakterizuje pôvodnú kvalitu priestoru (hmoty) v zodpovedajúcej oblasti (hustota látky, intenzita) [JaDu92].

Definícia 1.:

Nech C^d je ohraničená d-rozmerná podmnožina Euklidovského priestoru E^d ($d = 2, 3$). Nech γ je zobrazenie z C^d do konečnej množiny d-rozmerných polytopov $\mathcal{P}$ tak, že

(i) $\bigcup_i P_i \supseteq C^d$, $P_i \cap C^d \neq \emptyset$, $P_i \in \mathcal{P}$;

(ii) dimenzia prieniku dvoch rôznych polytopov P_i, $P_j \in \mathcal{P}$ je menšia ako d ($\dim(P_i \cap P_j) < d$).

Potom množina $\mathcal{P}$ sa nazýva d-rozmerná teselácia C^d (tiež d-rozmerná mriežka) a γ sa nazýva teselačná funkcia.

Maximálna podmnožina $\mathcal{P}^* \subseteq \mathcal{P}$, pre ktorú $\bigcup_i P_i \subset C^d$, $P_i \in \mathcal{P}^*$ a žiaden jej prvok neobsahuje časť hranice uzáveru $\overline{C}^d$ sa nazýva vnútrom teselácie $\mathcal{P}$; množina $\mathcal{P} - \mathcal{P}^*$ sa nazýva obalom teselácie.

Definícia 2.:

Teselácia $\mathcal{P}$ sa nazýva ideálna, ak splňuje nasledujúce vlastnosti:

(iii) Každý polytop $P_i \in \mathcal{P}$ je konvexný;

(iv) dimenzia prieniku dvoch rôznych polytopov P_i, $P_j \in \mathcal{P}$ je buď 0 alebo d-1 ($\dim(P_i \cap P_j) = 0 \vee \dim(P_i \cap P_j) = d-1$).

Ideálnu d-rozmernú teseláciu nazývame d-rozmerný diskrétny priestor (dD diskrétny priestor).

Definícia 3.:

Nech $\mathcal{P}$ je ideálna d-rozmerná teselácia. Nech I je ľubovoľná číselná množina ($I \subseteq \mathbb{R}$) a F: $\mathcal{P} \rightarrow I$ je funkcia, ktorá priradí každému polytopu P_i teselácie $\mathcal{P}$ číslo z množiny I. Potom hovoríme, že teselácia $\mathcal{P}$ je volumetrická a množinu I nazývame množinou intenzít (hustôt) a funkciu F funkciou intenzity (hustoty) teselácie $\mathcal{P}$.

V dvojrozmernom prípade prvok množiny $\mathcal{P}$ sa nazýva obrazový prvok (picture element = pixel) a v trojrozmernom prípade sa prvok množiny $\mathcal{P}$ 3D nazýva objemový prvok (volume element = voxel).

2. TROJROZMERNÁ DISKRÉTNA SCÉNA.

2. 1. Základné pojmy

Ak zavedieme vhodné označenie a usporiadanie základných prvkov teselácie - voxlov (súradnicový systém) a každý prvok budeme považovať za nedeliteľný a priamo adresovateľný, vybudujeme trojrozmerný diskrétny priestor (3DD priestor). Ak priradíme každému voxlu hustotu, vybudujeme trojrozmernú sikrétnu scénu (3DD scéna).

Definícia 4.:

3D diskrétny priestor je ideálna teselácia pravouhlého rovnobežnostena (kvádra) $\langle 0,X\rangle\times\langle 0,Y\rangle\times\langle 0,Z\rangle \subseteq E^3$, ktorý je rozdelený trojicou n(x), n(y) a n(z) navzájom na seba kolmých systémov rovnobežných rovín (x = const., y = const., z = const.) na 3D pole pravouhlých šesťstenov (kvádrov). Každý kváder je repezentovaný usporiadanou trojicou $v = (x,y,z) \in \mathbb{N}^3$, kde $1 \le x \le n(x)-1$, $1 \le y \le n(y)-1$, $1 \le z \le n(z)-1$.

Definícia 5.:

3DD scéna je usporiadaná dvojica $\Sigma = (\mathcal{P},F)$, kde

$\mathcal{P} = \{v(x,y,z) | 1\le x\le X \wedge 1\le y\le Y \wedge 1\le z\le Z\}$ je oblasť Σ (množina voxlov) a

F: $\mathcal{P}$-->I (funkcia hustoty) je zobrazenie z $\mathcal{P}$ do množiny hustôt (objemov) I. Zvyčajne je I podmnožina Z, prípadne N. Ak $I=\{0,1\}$, hovoríme o 3D binárnej scéne.

2. 2. Identifikácia objektu záujmu.

Na základe funkcie hustoty možno rozlišovať rôzne skupiny voxlov. Táto procedúra sa nazýva segmentácia a zvyčajne jej výsledkom je 3D binárna scéna ($I = \{0,1\}$). Voxle 3D binárnej scény s hodnotou 1 tvoria objekt O a voxle s hodnotou 0 tvoria

pozadie O'. Vzhľadom na vlastnosti segmentácie platí: $\mathcal{P} = O \cup O'$ a $O \cap O' = \emptyset$.

Najpoužívanejšou segmentačnou technikou v medicínskych aplikáciách je prahovanie - určenie prahovacích hodnôt (intervalov), ktoré jednoznačne určujú, ktorým voxlom bude priradená hodnota 1 a ktorým hodnota 0 [Jank91].

V ďalšom budeme predpokladať, že máme 3D binárnu scénu, v ktorej O - množina voxlov s hodnotou 1, zodpovedá skúmanemu objektu záujmu a ostatné patria pozadiu O' a majú hodnotu 0.

Na ďalšie spracovanie (popis, transformácie a zobrazovanie) objektu O si zavedieme niekoľko nasledovných pojmov.

Definícia 6.:

Nech $v = (x,y,z)$ a $v' = (x',y',z')$ sú dva voxle 3DD scény Σ. Potom

voxle v a v' sú 6-susedné, ak majú spoločnú stenu, t. j. $d_6(v,v') := |x-x'|+|y-y'|+|z-z'| = 1$;

voxle v a v' sú 18-susedné, ak majú spoločnú stenu alebo hranu, t. j. $0 < d_6(v,v') \le 2$;

voxle v a v' sú 26-susedné, ak majú spoločnú stenu, hranu alebo vrchol, t. j. $0 < d_6(v,v') \le 3$, resp. $d_{26}(v,v') := \max(|x-x'|,|y-y'|,|z-z'|) = 1$.

Množinu všetkých 6-susedov voxla v označujeme $N^*_6(v)$, resp. $N_6(v) = N^*_6(v) \cup \{v\}$ a nazývame rýdze diskrétne 6-okolie, resp. 6-okolie voxla v.

Množinu všetkých 18-susedov voxla v označujeme $N^*_{18}(v)$, resp. $N_{18}(v) = N^*_{18}(v) \cup \{v\}$ a nazývame rýdze diskrétne 18-okolie, resp. 18-okolie voxla v.

Množinu všetkých 26-susedov voxla v označujeme $N^*_{26}(v)$, resp. $N_{26}(v) = N^*_{26}(v) \cup \{v\}$ a nazývame rýdze diskrétne 26-okolie, resp. 26-okolie voxla v.

Vzhľadom na korektný popis 3D binárnej scény, je vhodné zaviesť odlišné typy susednosti: m pre objekt O a n pre pozadie O'. Najčastejšie používané dvojice (m,n) sú (6,26), (26,6), (6,18), (18,6) [KoRo89].

3D binárnu scénu Σ s objektom záujmu O, pozadím O'a m-susednosťou pre O a n-susednosťou pre O' označujeme (Σ, m, n, O).

Definícia 7.:

6-cesta (18-cesta, 26-cesta) je taká postupnosť voxlov v_o, v_1, ..., v_i, ..., v_n scény Σ, že v_i je 6-susedný (18-susedný, 26-susedný) s v_{i-1}, pre $1 \le i \le n$.

Objekt O je 6-súvislý (18-súvislý, 26-súvislý), ak pre ľubovoľné dva voxle v a v' z O existuje 6-cesta (18-cesta, 26-cesta), ktorá začína vo v a končí vo v' a celá leží v O.

2. 3. Dátové štruktúry.

Jedným zo závažných problémov pri spracovaní 3DD scény je jej úsporné uloženie a efektívna manipulácia. Tento problém je možné vyriešiť buď vytvorením "šikovných" dátových štruktúr alebo výrazným rozšírením pamäťového potenciálu počítača. Z pochopiteľných dôvodov sa bližšie zaoberáme prvou variantou.

Bolo navrhnutých niekoľko metód - dátových štruktúr: 3D matica digitálnych plátov pre manipuláciu s celou scénou (vhodné pre výkonné hardwarové konfigurácie s veľkou kapacitou RAM), stromové hierarchické štruktúry (oktálne a binárne stromy, marginálne indexovanie) [Srih81], prípadne štruktúry popisujúce len systém povrchových voxelov, resp. ich stien (stromy susedností, orientované obrysy) [ArFr81], [KoRo89], [MoRo81], [RoKo91], [Srih81], [Udup83].

2. 3. 1. Oktálny strom.

Na základe požiadaviek na maximálnu úsporu miesta pri uchovávaní binárnej diskrétnej scény, bola navrhnutá stromová dátová štruktúra, založená na rekurzívnom delení objektu na osem rovnako veľkých podobjektov. Delenie objektu pokračuje až pokým sa nedosiahne homogenita, t. j. nový poddobjekt buď celý patrí objektu záujmu O alebo celý patrí pozadiu O' (pozri [JaRu92]). Okrem úspory miesta, poskytuje oktálny strom aj efektívny algoritmus na odstránenie neviditeľných častí. Vzhľadom na hierarchiu v strome sa pri vizualizácii navštevujú len tie jeho

vetvy, ktoré reprezentujú potenciálne viditeľné časti objektu.

Jedným zo základných problémov pri vytváraní oktálneho stromu je rýchly prepis 3DD binárnej scény (z trojrozmernej matice voxlov do stromovej štruktúry). Pretože nie je možné uchovať v pamäti všetky rezy súčasne, prepis sa uskutočňuje postupne, po jednotlivých plátoch:

Input: Množina N binárnych digitálnych plátov.
Output: Oktálny strom
Data structures: Pomocný zasobník na dočasné uchovávanie stromov (mimo RAM).

```
Algorithm Slice_to_Oct-tree
1. for i= 2 to N step 2 do
   a. make_tree(i-1,i);
   b. k:=2;
   c. while (k<i) do
      c1. k:= 2*k;
      c2. if i = 0 (mod k) then load_tree and merge_trees;
      end while;
   d. save_tree;
   end for;
end.
```

- Funkcia make_tree(i,j) vytvorí oktálny strom z pôvodných rezov i a j a vracia jeho adresu (smerník).
- Procedúra load_tree presunie aktuálny strom zo zásobníka do pamäti a zníži počítadlo stromov v zásobníku o 1.
- Procedúra save_tree presunie aktuálny strom z pamäti do zásobníka zvýši počítadlo stromov v zásobníku o 1.
- Funkcia merge_trees spojí dva oktálne stromy do jedného a vrati jeho adresu (smerník).

INPUT: Smerníky a dva oktálne stromy;
OUTPUT: Smerník na novovytvorený oktálny strom;

```
Function merge_trees;

1. if ptr1 = ptr2 then ptr_new:=ptr1;
2. else if ptr1 = Zero and ptr2 = Nonzero then ptr_new:=ptrA;
3.       else if ptr1 = Nonzero and ptr2 = Zero then ptr_new:=ptrB;
4.             else
               a. ptr_new.1:=blend_tree(ptr1.1,ptr1.5);
               b. ptr_new.2:=blend_tree(ptr1.2,ptr1.6);
               c. ptr_new.3:=blend_tree(ptr1.3,ptr1.7);
               d. ptr_new.4:=blend_tree(ptr1.4,ptr1.8);
               e. ptr_new.5:=blend_tree(ptr2.1,ptr2.5);
               f. ptr_new.6:=blend_tree(ptr2.2,ptr2.6);
               g. ptr_new.7:=blend_tree(ptr2.3,ptr2.7);
               h. ptr_new.8:=blend_tree(ptr2.4,ptr2.8);
5. return ptr_new;
end.
```

Pamôťová náročnosť:

Maximálna pamôťová náročnosť algoritmu je zhora ohraničená uložením dvoch digitálnych plátov a jedného čiastočného oktálneho stromu alebo troma čiastočnými stromami (vo funkcii make_tree) a zásobníka, ktorý vyžaduje uloženie maximálne $\lceil \log_2 N \rceil - 1$ dočasných oktálnych stromov súčasne (N je rozmer digitálneho plátu).

3. 3D OBJEKT V 3D DIGITÁLNOM PRIESTORE.

3. 1. Klasifikácia voxlov.

Topologická klasifikácia jednotlivých voxlov alebo ich zoskupení je základom pre nájdenie topologických charakteristík a skeletu celého 3DD objektu. Vhodnou formou klasifikácie je vyhodnotenie malého okolia voxla 3x3x3 alebo 5x5x5. Pretože však, ako bolo uvedené vyššie, používame viacero typov susedností, exituje aj niekoľko odlišných klasifikátorov. Uvedieme tie najzákladnejšie:

Definícia 8. [Srih81]:
V $(\Sigma,6,26,O)$ voxel $v \in O$ nazývame
- m-hraničný voxel, ak nemá 6-suseda z O v smere

$m \in (x,-x,y,-y,z,-z)$;
- k-vrstvový voxel, ak je aj k-kraničný aj -k-hraničný
$k \in (x,y,z,)$;
- (s,t)-hraničný voxel ak je aj s-hraničný and t-hraničný a $|s| \neq |t$
- koncový voxel, ak má len jedného 6-suseda z O.

Definícia 9. [MaBe91]:
V $(\Sigma,26,6,O)$ sa klasifikuje voxel na základe hodnôt dvoch parametrov súvislosti C a C^*, ktoré sú definované nasledovne:

C^* = počet 26 komponentov $[O \cap N_{26}(x)] \setminus (x)$, ktoré sú 26-susedné s x.

C = počet 6 komponentov $[O \cap N_{26}(x)]$, ktoré sú 6-susedné s x.

Potom voxel môžme podľa C^* a C klasifikovať nasledovne:

Type 1 - interior voxel:	$C = 0$
Type 2 - isolated voxel:	$C = 1,\ C^* = 0$
Type 3 - edge voxel:	$C = 1,\ C^* = 1$
Type 4 - curve voxel:	$C = 1,\ C^* \geq 2$
(Type 4a - curve voxel:	$C = 1,\ C^* = 2)$
(Type 4b - curves junction:	$C = 1,\ C^* > 2)$
Type 5 - surface voxel:	$C \geq 2,\ C^* \geq 1$
(Type 5a - surface voxel:	$C = 2,\ C^* = 1)$
(Type 5b - surface & curve junction:	$C = 2,\ C^* \geq 2)$
(Type 5c - surfaces junction:	$C > 2,\ C^* = 1)$
(Type 5d - surfaces & curve junction:	$C > 2,\ C^* \geq 2)$

Vo všeobecnom prípade (bez ohľadu na typ susednosti) máme definovaný len jednoduchý povrchový voxel:

Definícia 10. [MoRo81]:
Voxel $x \in O$ sa nazýva jednoduchým povrchovým voxlom, ak
(i) $O \cap N_{26}(x)$ má práve jeden komponent susedný s x (v zmysle O); označme ho A_x.
(ii) $O' \cap N_{26}(x)$ má práve dva komponenty susedné s x (v zmysle O'). označme ich B_x a C_x
(iii) Pre každé $y \in O$ susedné s x (v zmysle O), je y susedné (v zmysle O') s nejakým voxlom z B_x a s nejakým voxlom z C_x.

3. 2. Digitálne plochy.

Na praktickú manipuláciu s 3D digitálnym objektom (vizualizácia, meranie) je dôležité mať vhodne popísaný jeho povrch. V súčasnosti sa používajú dva odlišné prístupy: voxlový povrch a stenový povrch. Prvý popisuje povrch ako vrstvu voxlov a je vhodnejší na skeletovanie, zatiaľ čo druhý prístup popisujúci povrch ako množinu stien voxlov (resp. množinu usporiadaných dvojíc stenovo susedných voxlov) je vhodnejší na vizualizáciu (surface rendering) a ďalšiu manipuláciu s objektom [ChHe85].

Ďalším z dôvodov pre dobrú definíciu povrchu digitálneho objektu je potreba spojenia dát vytvorených odlišnými modalitami: vektorový (spojitý) popis objektu a rastrový (diskrétny) popis objektu [Kauf91].

3. 2. 1. Voxlový povrch.

Definícia 11.:
Nech O je 3D object. Voxel $v \in O$ patrí povrchu O, ak existuje voxel u, ktorý je susedný s v (v zmysle O) a $u \notin O$. Množinu všetkých takých voxlov v nazývame voxlový povrch O a označujeme S(O). Množina všetkých voxlov $w \in O - S(O)$ nazývame voxlové vnútro O a označujeme I(O).
Objekt O môžeme vyjadriť ako zjednotenie S(O) a I(O):

$$O = S(O) \cup I(O) \text{ and } S(O) \cap I(O) = \emptyset.$$

V 3DD scéne $(\Sigma, 26, 6, O)$ jednoduchý voxlový povrch je definovaný nasledovne:

Definícia 12. [MaBe91]:
Množina voxlov sa nazýva jednoduchý voxlový povrch, ak je triedou ekvivalencie nasledujúcej relácie:

Nech x je povrchový voxel (type 5). Hovoríme, že dva povrchové voxle x a y sú v relácii, ak existuje 26-cesta $x_o, x_1, \ldots, x_i, \ldots, x_n$ s $x_o = x$ a $x_n = y$ taká, že pre $i \in$

$\langle 0, \ldots, n-1 \rangle$:

$$B_{x_i} \cap B_{x_{i+1}} \neq \emptyset \text{ a } C_{x_i} \cap C_{x_{i+1}} \neq \emptyset$$

$$\text{alebo } B_{x_i} \cap C_{x_{i+1}} \neq \emptyset \text{ a } C_{x_i} \cap B_{x_{i+1}} \neq \emptyset.$$

Hlavný teoretický výsledok v takto definovanom povrchu 3D digitálneho objektu je 3DC analóg Jordanovej vety. Táto platí pre všeobecný prípad súvislosti a je založená na definícii jednoduchého povrchového voxla a pojme orientovateľnosti povrchového voxla:

Definícia 13.:
Jednoduchá uzavretá plocha je súvislá množina pozostávajúca výlučne z orientovateľných jednoduchých povrchových voxlov. Jednoduchý povrchový voxel x je orientovateľný ak $\forall y \in A_x$ je tiež jednoduchý povrchový voxel a $O \cap N_{124}(x)$ má práve dva komponenty.

Veta 1. (3DC analóg Jordanovej vety) [MoRo81]:
Jednoduchá uzavretá plocha O rozdeľuje doplnok O' na práve jednu dutinu a jeden (vonkajší) komponent pozadia.

2. 2. 2. Stenový povrch.

Ako bolo uvedené vyššie stenový povrch je vhodnejší na efektívny popis hranice 3D digitálneho objektu. Hlavným výsledkom, vhodným pre naše aplikácie, je zlepšený algoritmus popisu povrchu (border tracking algorithm), ktorý je založený na pojme silnej hranovej súvislosti stien voxlov [ArFr81], [RoKo91]:

Definícia 14.:
Stena (face) je spoločná plocha dvoch 6-susedných voxlov.

Definícia 15.:
Nech P je stena voxla $v \in O$. Stena $Q \in O$ je hranovo susedná s P, ak P a Q majú spoločnú hranu. (P môže mať až 12 hranovo

susedných susedov).

Definícia 16.:

Nech P je stena voxla v∈O. Stena Q∈O je silne hranovo susedná s P, ak Q je hranovo susedná s P a je stenou voxla O, ktorý je 6-susedný s v pomocou postupnosti m≤3 voxlov z O. (P má práve štyroch silne hranovo susedných susedov).

Definícia 17.:

Dve steny P, Q objektu O sú (silne) stenovo súvislé, ak existuje postupnosť stien $P = P_1, P_2, \ldots, P_m = Q$ objektu O taká, že P_i je (silne) hranovo susednedá s P_{i+1} $1 \le i \le m$. Maximálna množina (silne) hranovo súvislých stien v objekte O sa nazýva (regulárny) povrch.

Tvrdenie 1:

Pre ľubovoľný regulárny povrch R objketu O, existuje 6-súvislý komponent L objektu O a 18-súvislý komponent M pozadia O' také, že R je množina stien, ktoré sú spoločnými stenami voxlov v L a v M.

Tvrdenie 2.

Pre ľubovoľný 6-súvislý komponent L objektu O a ľubovoľný 18-súvislý komponent M pozadia O' také, že existuje voxel v L a voxel v M, ktoré sú 6-susené, množina stien, ktoré sú spoločnými stenami voxlov z L a voxlov z M tvorí regulárny povrch O.

ALGORITMUS POPISU POVRCHU.

Algoritmus je založený na definícii pojmu silnej hranovej susednosti stien, ktorá dáva možnosť popísať povrch 3D digitálneho objektu pomocou orientovaného grafu, ktorého vrcholy reprezentujú povrchové steny a hrany reprezentujú silne hranovo susedné steny. Pretože každá povrchová stena silne hranovo susedí práve so štyrmi ďalšími stenami, bude mať každý vrchol v zodpovedajúcom orientovanom grafe vstupný stupeň i výstupný stupeň rovný dvom [ArFr81].

Tento algoritmus sa používa pri popise povrchu 3D objektu a jeho následnom vizuálnom spracovaní. Niekoľko tónovacích techník na pseudo 3D zobrazenie takto popísaného povrchu je detailne

popísaných v [ChHe85].

INPUT: Binárna scéna S a povrchová stena f_0.
OUTPUT: Zoznam L stien toho komponentu hranice, ktorý obsahuje f_0.
DATA STRUCTURES: Front Q obsahujúci steny na spracovanie a zoznam M označených stien.

```
Algorithm BD.
1. queue f0 and put two copies of f0 in M;
2. while Q is not empty
     a. remove a stena f from Q;
     b. find f_l, for 1≤l≤2, such that fTf_l;
     c. output f;
     d. for l = 1 to 2 do
          d1. if f_l∈M then delete f_l from M;
          d2. else queue f_l and put f_l in M;
          end for;
     end while;
end BD.
```

3. 3. Topologické Invarianty.

Topologické invarianty sú charakteristiky, ktoré zostávajú nezmenené pri akejkoľvek regulárnej transformácii definovanej v 3D priestore. Popíšeme tie najdôležitejšie a odvodíme ich vzájomné vzťahy [Srih81].

Uvažujme 3DD scénu $(\Sigma, 6, 26, O)$ a priraďme objektu O nasledovné charakteristiky:

K - počet 6-súvislých komponentov objektu O;

C - počet uzavretých dutín(holes) = počet 26-súvislých komponentov pozadia O' mínus 1;

T - počet tunelov (genus, handle).

Čísla K a C možno dostať spočítaním zodpovedajúcich množín voxlov.

Číslo T možno vypočítať $T = \sum_{i=1}^{K+C} T_i$, kde T_i je počet tunelov v i-tom povrch (pozri nižšie).

Povrch 3D objektu je reprezentovaný sieťou (neorientovaným grafom), ktorý je zložený z V vrcholov, E stien a F stien. Pre takýto povrch platí Eulerova formula:

$$V - E + F = 2 - 2T$$

Možno ľahko ukázať, že povrch O pozostáva z K + C komponentov. Potom pre každý komponent povrchu platí

$$V_i - E_i + F_i = 2 - 2T_i, \text{ kde } i = 1, 2, \ldots, K+C$$

Definujme pre celý objekt O

- Eulerovu charakteristiku χ

$$\chi = K - T + C$$

- parameter súvislosti ζ

$$\zeta = \sum_{i=1}^{K+C} (2 - 2T_i)$$

Potom dostávame

$$\zeta = \sum_{i=1}^{K+C} (2 - 2T_i) = 2 \sum_{i=1}^{K+C} 1 - \sum_{i=1}^{K+C} 2T_i = 2*(K+C) - 2T =$$

$$\zeta = 2*(K-T+C) = 2\chi$$

a

$$\zeta = \sum_{i=1}^{K+C} (2 - 2T_i) = \sum_{i=1}^{K+C} (V_i - E_i + F_i) = \sum_{i=1}^{K+C} V_i - \sum_{i=1}^{K+C} E_i + F$$

Pre počet všetkých vrcholov V a počet všetkých hrán E objektu O platí

$\sum_{i=1}^{K+C} V_i \geq V$ a $\sum_{i=1}^{K+C} E_i \geq E$, pretože ako vrchol tak aj hrane môžu patriť viacerým komponentom povrchu (vrchol najviac štyrom a hrana najviac dvom).

Pre celkový počet stien platí

$\sum_{i=1}^{K+C} F_i = F$, pretože jedna stena patrí práve jednému komponentu povrchu.

3. 4. Skelet trojrozmeného digitálneho objektu.

Výsledkom skeletovania 3D objektu je získanie (pod)objektu, ktorý má výrazne nižší počet voxlov, ale pri zachovaní základných topologických vlastností dobre charakterizuje pôvodný

objekt. Za dobrý skelet 3D digitálneho objektu sa považuje jeho madiálna os (medial axis):

Definícia 18. [Srih81]:
Voxel $v \in O$ patrí do mediálnej osi M(O), ak

$$\text{card } \{u : (u,v) = \min_{v \in B(O)} d(v,w)\} > 1,$$

kde d je 3D metrika, card A je kardinalite množiny A a B(O) je voxelový povrch O.

Avšak selekcia tých voxlov, ktoré splňujú predošlú definíciu nevedie vo všeobecnosti k uspokojivému výsledku, ale vytvára tzv. mediálnu plochu o šírke jedného voxla. Preto ak požadujeme skelet vo forme digitálnej krivky, treba selekciu vykonať znova. Takto dvojkrokovo pracuje aj väčšina navrhnutých algoritmov.

Definícia 19.:
Voxel $x \in O$ sa nazýva jednoduchý ak jeho odstránenie nezmení topológiu objektu.

V $(\Sigma, 26, 6, O)$ pre jednoduchý voxel platí:

Tvrdenie 3. [MaBe91]:
Voxel $x \in O$ je jednoduchý vtedy a len vtedy, ak

a) x je edge voxel (typ 3);

b) $NH[O \cap N_{26}(x)] = NH[O \cap N^*_{26}(x)]$;

c) $NH[(O' \cap N^*_{26}(x)) \cup \{x\}] = NH[O' \cap N^*_{26}(x)]$,

kde NH(A) značí počet dutín v A.

Tvrdenie 4.:
Ak odstránenie voxla $x \in O$ nespôsobí zmenu súvislosti $N^*_{26}(x)$, potom nespôsobí zmenu súvislosti ani v celom objekte O.
Ľahko možno nahliadnuť, že opačné tvrdenie neplatí.

Vo všeobecnosti je skeletovací algoritmus založený na princípe postupného odstraňovania jednoduchých bodov, t. j. bodov, ktorých odstránenie nespôsobuje zmenu topologických vlastností. Avšak len jednoduché odstránenie takýchto voxlov by mohlo viesť k "scvrknutiu" objektu do jedného voxla. Preto musíme na testovanie odstránitelnosti voxla musíme použiť ďašie dodatočné kritériá. Dve prechádzajúce tvrdenia zaručujú, že na klasifikáciu voxla je potrebné prezrieť len jeho malé okolie.

Ďalším problémom je potreba paralelizácie algoritmu, nakoľko skeletovaný objekt je zvyčajne veľmi rozsiahly.

Nasledujúci skeletovací algoritmus nájde skelet 3D digitálneho objektu v (Σ,6,26,0) scéne v dvoch krokoch. Prvý krok , v ktorom nájde mediálnu plochu je založený na kritŕiách A a B a druhý krok, ktorý nájde mediálnu os je založený na kritérii B [Srih81]:

A: voxel v je jednoduchý, m-hraničný a nie je k-vrstvový;

B: voxel v jednoduchý, (s,t)-hraničný a nie je koncový.

INPUT: Objekt S v 3D binárnej scéne;

OUTPUT: Mediálna plocha S, 3D objektu S.

```
Algorithm S1 (Converts object S to medial povrch S')
begin
     d:=[x,-x,y,-y,z,-z]; (direction vector)
     repeat
          D:=Ø;
          for i:=1 to 6 do
          begin
               m:=d(i); (the ith component of d)
               k:=|m|;  (the absolute value of m)
               in parallel for each v in S do
               begin
                    if(v satisfies A) then include v in D;
               end;
               S:=S-D;
          end;
     until (D=Ø);
     S':=S;
end;
```


INPUT: Mediálna plocha S' v 3D binárnej scéne;

OUTPUT: Mediálna os S'' 3D objektu S.

```
Algorithm S2 (Converts medial povrch S' to medial axes S")
begin
     d':=[(x,y),(-x,-y),(x,-y),(-x,y),etc.]; (direction vector)
     repeat
          D:=Ø;
          for i:= 1 to 12 do
          begin
               (s,t):=d'(i);
               in parallel for each v in S' do
               begin
                    if (v satisfies B) then include v in D;
               end;
               S':=S'-D;
          end;
     until (D=Ø);
     S":=S';
end;
```

Ako je uvedené v [TsFu81], tento algoritmus nemožno paralelizovať. Nové kritériá, ktoré umožňujú paralelizáciu (ak uvažujeme 26-súvislosť pre objekt) je:

C1: voxel v je jednoduchý a má najmenej dvoch 26-susedov z 0;

C2: voxel v je odstrániteľný v zodpovedajúcej kontrolnej rovine.

Zodpovedajúce kontrolné roviny, napríklad pre smer +x, sú obe roviny v ktorých ležia súčasne voxle x-1, x and x+1. Podobne pre ostatných päť smerov.

Voxel je odstrániteľný v kontrolnej rovine, ak jeho odstránenie nespôsobí rozpojenie zvyšného objektu v okolí 3x3 v rovine a počet voxlov patriacich objektu v tomto okolí nie je menší ako 2.

Použijúc tieto kritériá, paralelný algoritmus vyzerá nasledovne:

```
Algorithm T (Converts object S to medial povrch S')
begin
     d:=[x,-x,y,-y,z,-z]; (direction vector)
     repeat
          D:=Ø;
          for i:=1 to 6 do
          begin
               m:=d(i); (the ith component of d)
               in parallel for each v in S do
               begin
                    if (v satisfies C1 and C2)
                    then include v in D;
               end;
               S:=S-D;
          end;
     until (D=Ø);
     S':=S;
end;
```

Ak požadujeme vytvorenie mediálnej osi, použijeme opôtovne na mediálnu plochu ten istý algoritmus, s tým rozdielom, kritérium C2 modifikujeme tak, že pripustíme jedného 8-suseda, ak je tento v opačnom smere.

4. ZÁVER.

Predložený príspevok podáva základy relatívne nového odvetvia počítačovej grafiky, digitálnej topológie. Zavedením základných pojmov a niektorých algoritmov zhrňujeme najdôležitejšie výsledky, ktoré sa doteraz dosiahli. Čitateľa, ktorý má hlbší záujem o študovanú problematiku odkazujeme na uvedenú literatúru, prípadne priamo na autorov, ktorí s radosťou privítajú možnosť diskusie a výmeny informácie.

5. LITERATÚRA.

[ArFr81] - Artzy, E. - Frieder, G. - Herman, G. T.: The Theory, design, implementation and Evaluation of a Three-Dimensional Boundary Detection Algorithm, Computer Graphics and Image Processing, Vol. 15, 1981, pp. 1-24.

[ChHe85] - Chen, L. S. - Herman, G. T. - Reynolds, R. - Udupa, J.: Surface Shading in the Cuberille Environment, IEEE Computer Graphics and Applications, December 1985, pp. 33-43.

[Jank91] Jankovič, V.: 3D Reconstruction of Medical Objects from Serial Cross Sections (Interslice Interpolation), in: Proceedings of the Seventh Spring School on Computer Graphics and its Applications, pp. 67-77, Bratislava, CSFR, 1991.

[Jank92] - Jankovič, V.: Volumetrization - Basic Definitions and Data Structures, Proceedings of the Eight Spring School on Computer Graphics, Bratislava, CSFR, 1992.

[JaDu92] - Jankovič, V. - Ďurikovič, R.: Volumetrization - Classification and Thinning of Digital objects, Proceedings of the Eight Spring School on Computer Graphics, Bratislava, CSFR, 1992.

[JaRu92] - Jankovič, V. - Ružický, E.: Počítačové spracovanie a vizualizácia medicínskych údajov, Zimná škola z Počítačovej grafiky, Plzeň, December 1992.

[Kauf91] - Kaufman, A.: State of the Art in Volume Visualization, in Eurographics'91, State of the Art Reports, S. Coquillart (Ed.), Vienna, 1991, pp. 173-186.

[KoRo89] - Kong, T. Y. - Rosenfeld, A.: Digital Topology: Introduction and Survey, Computer Vision, Graphics and Image Processing, Vol. 48, 1989, pp. 357-393.

[MaBe91] - Malandain, G. - Bertrand, G. - Ayache, N.: Topological Classification in Digital Space,

[MoRo81] - Morgenthaler, D. G. - Rosenfeld, A.: Surfaces in Three-Dimensional Digital Images, Information and Control, Vol. 51, No. 3, December 1981, pp. 264-279.

[RoKo91] - Rosenfeld, A. - Kong, T. Y. - Wu, A. Y.: Digital Surfaces, Computer Vision, Graphics and Image Processing: Graphical Models and Image Processing, Vol. 53, No. 4, July 1991, pp. 305-312.

[Srih81] - Srihari, S. N.: Representation of Three-Dimensional Digital Images, ACM Computing Surveys, Vol. 13, No. 4, December 1981.

[TsFu81] - Tsao, Y. F. - Fu, K. S.: A Parallel Thinning Algorithm for 3-D Pictures, Computer Graphics and Image Processing, Vol. 17, 1981, pp. 315-331.