

Szegedi Tudományegyetem
Természettudományi Kar

Palágyi Kálmán

Vékonyító algoritmusok 3D képekre

Doktori értekezés

Szeged

2000.

Tartalom

Előszó	1
1. A váz és a vázkijelölés	2
1.1. A folytonos váz és tulajdonságai	2
1.2. A digitális topológia	6
1.2.1. Alapfogalmak	6
1.2.2. A topológia megőrzése	10
1.3. Vázkijelölő stratégiák	19
1.3.1. Távolság–transzformáción alapuló módszerek	19
1.3.2. A front–terjedés modellezése, a vékonyítás	26
1.3.3. A Voronoi–váz	30
1.4. A váz meghatározása és a matematikai morfológia	34
1.4.1. Alapműveletek	34
1.4.2. A morfológiai vékonyítás	41
1.4.3. A morfológiai váz	43
1.4.4. Morfológiai műveletek maszkillesztéssel	45
2. A vékonyítás és módszerei	51
2.1. Irány–szekvenciális módszerek	54
2.1.1. Tsao és Fu 6 al–iterációs algoritmus	56
2.1.2. Gong és Bertrand 6 al–iterációs algoritmus	58
2.1.3. Mukherjee, Das és Chatterjee 6 al–iterációs algoritmus	59
2.1.4. Lee, Kashyap és Chu 6 al–iterációs algoritmus	60
2.1.5. Bertrand 6 al–iterációs algoritmus	62
2.1.6. Egy 6 al–iterációs algoritmus	61
2.1.7. Egy 8 al–iterációs algoritmus	65
2.1.8. Egy 12 al–iterációs algoritmus	69
2.2. Almező–szekvenciális módszerek	73
2.2.1. Bertrand és Aktouf 8–almezős algoritmus	75
2.2.2. Saha, Chaudhuri és Majumder 8–almezős algoritmus	75
2.3. Teljesen párhuzamos módszerek	76
2.3.1. Ma algoritmus	78
2.3.2. Ma és Sonka algoritmus	80
2.3.3. Ma és Sonka algoritmusának módosítása	82
2.3.4. Manzanera, Bernard, Pretêux és Longuet algoritmus	88
2.4. Hibrid stratégia	90
2.4.1. Egy hibrid algoritmus	91
3. A zsugorítás	95
3.1. Egy 6 al–iterációs párhuzamos 3D zsugorító algoritmus	97
4. Elő– és utófeldolgozás	100
4.1. A vékonyítandó objektumok simítása	102
4.2. A vékonyítással kapott váz tisztítása	104
5. A 3D vékonyító algoritmusaink eredményei	108
6. A 3D váz alkalmazásai	123
6.1. Vérerek és légutak átmérőjének mérése	125
Irodalom	128
Összefoglaló	141
Summary	144
Köszönetnyilvánítás	147

Előszó

A váz egy gyakran alkalmazott régió–alapú alakjellemző, melynek a diszkrét objektumokra történő meghatározására többféle módszer ismert. A jelen dolgozat kulcsproblémája a vékonyítással történő vázkijelölés 3–dimenziós (a továbbiakban 3D) objektumokra.

Az értekezés 1. fejezetében ismertetjük a váz tulajdonságait, a digitális topológia alapfogalmait, a topológia megőrzésével kapcsolatos eredményeket, a fő vázkijelölő technikákat, a váz és a vékonyítás leírását a matematikai morfológia eszközeivel, valamint a morfológiai és a lokális és párhuzamos képműveletek szemléletes megadását és hatékony implementációját lehetővé tevő maszkillesztést. A dolgozat fő eredményei a 2. fejezetben találhatóak, ahol nem csak az általunk javasolt vékonyító algoritmusokat ismertetjük, hanem rendszerezzük is a 3D vékonyító technikákat és bemutatjuk az általunk ismert valamennyi párhuzamos vékonyító algoritmust is. A 3. fejezet tárgya a zsugorítás, mely művelet egyrészt azért fontos, mivel rokon a vékonyítással, másrészt pedig alkalmazható a vékonyítással kapott váz utófeldolgozásakor. A 4. fejezetben tárgyaljuk a vázkijelölés elő- és utófeldolgozását, az 5. fejezetben pedig a vékonyító algoritmusaink hatását illusztráljuk néhány “mesterséges” és “természetes” 3D bináris objektumra. Végezetül, a 6. fejezetben a 3D váz eddig felmerült alkalmazásait tekintjük át.

Terjedelmi okok miatt a dolgozatban nem bizonyítjuk a vékonyító algoritmusaink topológiai korrektségét: csak kimondjuk, hogy valamennyi eljárásunk topológia–megőrző és hivatkozunk azokra a közleményekre, amelyek tartalmazzák a részletes bizonyításokat. (A dolgozatban csak a topológia–megőrző párhuzamos redukciókra adott elégséges feltételünket, a tetszőleges morfológiai, valamint a lokális és párhuzamos képművelet maszkillesztéssel történő megadhatóságát és a — még nem közölt — 3D zsugorító algoritmusunk topológiai tulajdonságait bizonyítjuk.)

A dolgozatban tárgyalt és érintett területekről nem, vagy csak alig található magyar nyelvű szakirodalom, így nem támaszkodhattunk elfogadott szóhasználatra. Ezért a magyarra “fordított” szakkifejezések első előfordulásai mellett megadtuk azok angol eredetijét is. Így — remélhetőleg — egy kevésbé szerencsésen megválasztott magyar kifejezés sem lesz értelemzavaró.

A tudományos közleményekben az irodalmi hivatkozásoknak több módja is elfogadott. Tekintettel a hivatkozott közlemények nagy számára, a hivatkozás a szerzők nevének és a megjelenés évének megadásával történt.

Az adott terjedelmi korlátok mellett eredményeinket egy olyan áttekintő műben kívántuk bemutatni, ami a szűkebb témában kevésbé járatosak számára is olvasható. Az értekezést 80 ábra (köztük 30 egész oldalas) illusztrálja.

1. A váz és a vázkijelölés

Az 1.1. alfejezetben a váz definícióját és tulajdonságait ismertetjük euklidészi terekre. Diszkrét terek esetében a folytonos objektumoknak diszkrét bináris objektumok felelnek meg, ennél fogva a folytonos váznak csak egy közelítését tudjuk meghatározni. A diszkrét váz kijelölésére adott módszereknek az alábbi két követelményre — a geometriaira és a topológiaira — kell tekintettel lenniük (Székely, 1996):

- A váz a geometriailag korrekt pozíción (az objektum “közepén”) helyezkedjen el és legyen invariáns a lineáris transzformációkra.
- A kijelölt váz legyen topológikusan ekvivalens a kiindulási objektummal.

A topológiai kritérium teljesítése vizsgálatához szükséges, hogy részletesebben foglalkozunk a 3D digitális topológiával. Az 1.2. alfejezetben ismertetjük az alapfogalmakat és áttekintjük a topológia–megőrzés terén elért eredményeket.

A három alapvető vázkijelölő módszer bemutatásának szenteltük az 1.3. alfejezetet, míg az 1.4. alfejezetben a matematikai morfológia eszközeivel leírt vázkijelölést tárgyaljuk.

1.1. A folytonos váz és tulajdonságai

A \mathbb{R}^d d -dimenziós euklidészi térben az $\mathcal{O} \subset \mathbb{R}^d$ ponthalmaz *objektum*, ha \mathcal{O} nyílt halmaz¹.

A váz fogalma először Blum tanulmányaiban jelent meg (Blum, 1964; Blum, 1967) mint a *középtengely–transzformáció* (*Medial Axis Transform, MAT*), vagy más elnevezéssel a *szimmetriatengely–transzformáció* (*Symmetry Axis Transform, SAT*) eredménye. A középtengely–transzformáció az objektum minden egyes pontjára megkeresi a hozzá legközelebbre eső határponto(ka)t. Ha az eljárás egy belső pontra egynél több legközelebbi határpontot talál, akkor azt a vázhoz tartozónak, vázpontnak minősíti. A vázat Blum egy szemléletes hasonlattal, a *préritűz terjedésével* illusztrálta: Ha a vizsgált objektum határának minden pontját egyidejűleg meggyújtjuk és feltételezzük, hogy a tűzfrontok minden irányba egyenletes sebességgel terjednek, akkor a váz azokból a pontokból áll, ahol az objektum belsejében a tüzek találkoznak, kioltják egymást (1.1. ábra).

Ugyancsak Blum javasolta a váz *távolság–transzformációval* (*Distance Transform*) történő meghatározását. Az objektum minden pontjára kiszámítandó a tőle legközelebbre lévő határponttól vett távolsága, így megkapjuk a *távolságtérképet* (*distance map*). A vázat a távolságtérkép lokális maximumhelyei alkotják.

A vázra formális definíciót Calabi adott (Calabi, 1965): 2D-ben egy objektum vázát azon pontok alkotják, melyek középpontjai az objektumba írt maximális nyílt körlapoknak. Ez a meghatározás általánosítható magasabb dimenziókra is: d D-ben a nyílt körlapokat d -dimenziós nyílt hipergömbökkel kell helyettesíteni. Folytonos 3D vázra az 1.2. ábra mutat példát.

¹Serra a nyílt halmazokat részesíti előnyben, de megengedi azt is, hogy a váz definíciójában zárt halmazok szerepeljenek (Serra, 1982).

Az d -dimenziós valós tér valamely \mathcal{O} objektumának $S(\mathcal{O}) \subseteq \mathcal{O}$ váza a következőképpen adható meg:

Az r -sugarú ($r \geq 0$, $r \in \mathbb{R}$), $P \in \mathbb{R}^d$ középpontú nyílt hipergömb pontjai:

$$\mathcal{B}(P, r) = \{ Q \mid Q \in \mathbb{R}^d, D(P - Q) < r \},$$

ahol “ D ” az euklidészi távolságot jelöli.

Az \mathcal{O} objektumba beírt nyílt hipergömbök halmaza:

$$\mathcal{IB}(\mathcal{O}) = \{ \mathcal{B}(P, r) \mid \mathcal{B}(P, r) \cap \mathcal{O} = \mathcal{B}(P, r) \}.$$

Az \mathcal{O} objektumba beírt maximális nyílt hipergömbök halmaza:

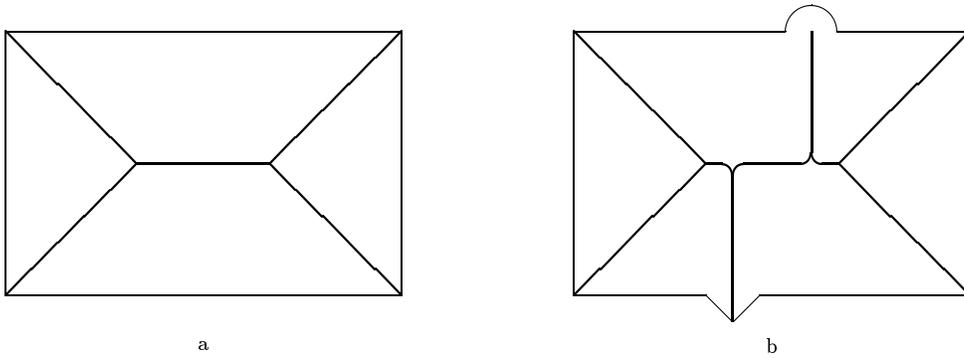
$$\mathcal{MIB}(\mathcal{O}) = \{ \mathcal{B}(P, r) \mid \mathcal{B}(P, r) \in \mathcal{IB}(\mathcal{O}), \forall \mathcal{B}(P', r') \in \mathcal{IB}(\mathcal{O}) : \mathcal{B}(P, r) \not\subseteq \mathcal{B}(P', r') \}.$$

A fentiek után a váz:

$$S(\mathcal{O}) = \{ P \mid \exists r \in \mathbb{R} : \mathcal{B}(P, r) \in \mathcal{MIB}(\mathcal{O}) \}.$$

A vázpontokhoz tartozó beírt maximális nyílt hipergömbök az objektum egy lefedőrendszerét alkotják (Serra, 1982), vagyis

$$\mathcal{O} = \bigcup \{ \mathcal{B}(P, r) \mid \mathcal{B}(P, r) \in \mathcal{MIB}(\mathcal{O}) \}.$$



1.1. ábra Példák 2D vázra. Eredeti objektumokként egy szabályos téglalap (a) és annak egy “kismértékben” megváltoztatott példánya (b) szerepel. Az objektumok határait vékony, a vázat vastagított vonalakkal rajzoltuk meg. A “közel egyforma” objektumok váza jelentős eltérést mutat. A (b) objektum vázán a két “hosszú” függőleges vonal akkor is felbukkan, ha a határkitüremkedések tetszőlegesen kicsik.

A prérítűz-analógia és a maximális nyílt hipergömbök középpontjaival definiált vázfogalmak ekvivalenciáját Calabi és Hartnett bizonyította be (Calabi, Hartnett, 1968). A váz egyértelműen létezik, így a vázkijelölés mint egy

$$S : 2^{\mathbb{R}^d} \rightarrow 2^{\mathbb{R}^d}$$

megfeleltetés (ahol $2^{\mathbb{R}^d}$ az \mathbb{R}^d ponthalmaz hatványhalmazát jelöli) egyben egy leképezés is.

- A váz második lényeges tulajdonsága az, hogy reprezentálja az eredeti objektum topológiai struktúráját: Összefüggő objektum váza összefüggő, üreget vagy lyukat tartalmazó objektumra a váz is üreges illetve lyukas⁴.
- A váz harmadik jellemző tulajdonsága az, hogy “vékony”, azaz vonal–szegmensekből, görbékből (3D-ben még felszín– és palást–szegmensekből) áll, amelyek alkalmasak szimbólikus reprezentációra. Mivel a váz az eredeti kiterjedt objektumnál jóval kevesebb pontból áll, így a vázokkal reprezentált objektumokra drasztikusan csökken a kiértékelendő adatmennyiség.

A fenti tulajdonságok miatt a váz számos területen alkalmazható, pl. az *alakfelismerésben* (*pattern recognition*) (Bow, 1992) vagy a képillesztés/képregisztráció (*image matching/image registration*) problémájának (Brown, 1992; van den Elsen, Pol, Viergever, 1993; Tanács, Palágyi, Kuba, 1998; Tanács, Palágyi, Kuba, 1999) megoldásakor is.

Nem hallgatható el a váz mint *régió-alapú alakjellemező* érzékenysége az objektum határának “kismértékű” megváltozására, még akkor sem, ha ez a stabilitási probléma a *határ-alapú jellemzőknél* is jelentkezik. Az 1.1a. ábrán egy szabályos téglalap, az 1.1b. ábrán pedig egy “kishibás” téglalap váza látható. Az objektumok “kismértékű” különbözősége a vázak “jelentős” eltérését vonja maga után. A váznak ez a hátrányos tulajdonsága az objektum határának *simításával* (*smoothing, filtering*) mint előfeldolgozással és/vagy a váz *tisztításával* (*pruning*) mint utófeldolgozó lépéssel orvosolható.

A folytonos váz jellemzését a reprodukálhatóság kérdésével zárjuk. Meghatározható-e egyértelműen az eredeti kiterjedt objektum a váz ismeretében?

A válasz nemleges. Pl., ha a váz 3D-ben egy l -hosszúságú egyenes szakasz, akkor az eredeti objektum kapszula-alakú is lehet: egy henger az alapjához és a tetejéhez illesztett félgömbökkel, ahol a henger magassága l . A henger (és a félgömbök) r sugarát nem tudjuk, mivel tetszőleges $r \geq 0$ esetén ugyanazt a vázat kapjuk. Kapszulán kívül lehet még az eredeti objektum olyan is, mint egy rögbilabda. Egy másik példa: ha a 3D váz egy r sugarú körvonal, akkor csak annyit mondhatunk, hogy az eredeti objektum tórusz-szerű, hasonlít egy amerikai fánkra. Tudjuk még, hogy a fánk lyukas, ahol a lyuk s (legszűkebb) sugarára $0 < s \leq r$ teljesül, de többet nem. Még az sem biztos, hogy a fánk vastagsága állandó.

Az \mathcal{O} objektum jobb jellemzése érdekében a váz mint *belső váz* (*endoskeleton*) mellett a *külső váz* (*exoskeleton*) (vagyis az $\mathbb{R}^d \setminus (\mathcal{O}) \cup \delta\mathcal{O}$) nyílt halmaz váza) meghatározását is javasolják. A reprodukálhatóságot a kétféle váz ismerete sem garantálja. Egyrészt konvex objektumok esetén a külső váz üres, másrészt egy konkáv objektumra sem biztos, hogy a kétféle váz “felgyújtásakor” a tűzfrontok az objektum határán oltják ki egymást.

Korábban már megállapítottuk, hogy a vázkijelölés egy leképezés. Mivel különböző objektumoknak megegyezhet a vázuk, így a vázkijelölés nem *injektív* — és ezáltal nem is *bijektív* — leképezés. A váz tehát csak akkor alkalmas az objektumok kódolására, ha minden egyes vázpontra (mint egy beírt maximális nyílt hipergömb középpontjára) ismert a hozzá tartozó hipergömb sugara is.

⁴Objektumként zárt halmazt és a váz definíciójakor zárt hipergömböket feltételezve, az objektum és a váz topológiai ekvivalenciája nem teljesül pl. arra az objektumra, amit kettő, egymást (egy pontban) érintő d -dimenziós zárt hipergömb alkot. A váz ekkor csak a két hipergömb középpontjaiból áll, így ennek az összefüggő objektumnak a váza nem összefüggő.

1.2. A digitális topológia

A *digitális topológia* (*digital topology*) a bináris digitális képek topológiai tulajdonságaival foglalkozik. Számos képművelet pl. az összefüggő komponensek címkézése és számlálása, a határkövetés, a kontúrfeltöltés, vagy a (jelen dolgozat tárgyát képező) vékonyítás elméleti megalapozását a topológia biztosítja.

A digitális topológia alapfogalmait a szakirodalom nem jelöli és nem is definiálja egységesen. A jelen dolgozatban Kong és Rosenfeld áttekintő cikkét (Kong, Rosenfeld, 1989) tekintettük irányadónak.

1.2.1. Alapfogalmak

Jelölje \mathbb{Z}^2 a 2D sík, és \mathbb{Z}^3 a 3D tér egész koordinátájú pontjait.

Legyen $x = (x_1, x_2)$ és $y = (y_1, y_2)$ két pont \mathbb{Z}^2 -ben. A \mathbb{Z}^2 -ben leggyakrabban használt *szomszédsági* (*adjacency*) relációkat az $\|x - y\| = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2}$ (euklidészi) távolság segítségével adjuk meg. Az x és az y pontok *4-szomszédosak*, ha $\|x - y\| \leq 1$, *8-szomszédosak*, ha $\|x - y\| \leq \sqrt{2}$. Jelölje $N_4(x)$ és $N_8(x)$ az x -szel 4-, illetve 8-szomszédos pontok \mathbb{Z}^2 -beli pontok halmazát (1.3. ábra).

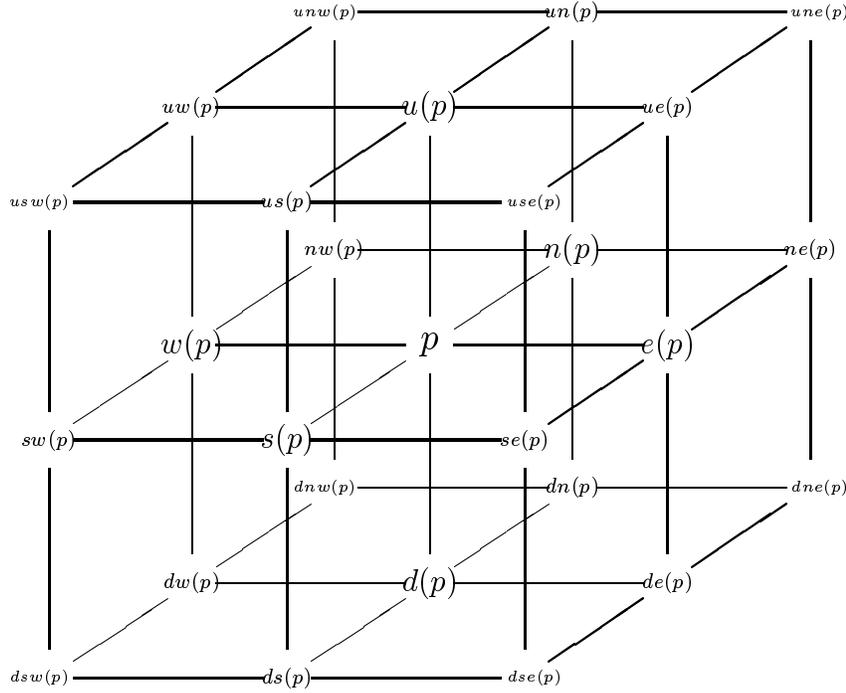


1.3. ábra A leggyakoribb szomszédsági relációk \mathbb{Z}^2 -ben. Az x ponttal 4-szomszédos (a) és 8-szomszédos (b) pontokat “ y ” szimbólumok jelölik. Megjegyzendő, hogy $x \in N_4(x)$ és $x \in N_8(x)$.

Legyenek $p = (p_1, p_2, p_3)$ és $q = (q_1, q_2, q_3)$ \mathbb{Z}^3 -beli pontok és tekintsük a $\|p - q\| = \sqrt{\sum_{i=1}^3 (p_i - q_i)^2}$ távolságot. A p és a q pontok *6-szomszédosak*, ha $\|p - q\| \leq 1$, *18-szomszédosak*, ha $\|p - q\| \leq \sqrt{2}$, *26-szomszédosak*, ha $\|p - q\| \leq \sqrt{3}$. Jelölje $N_j(p)$ ($j = 6, 18, 26$) a p -vel j -szomszédos \mathbb{Z}^3 -beli pontok halmazát, melyek az 1.4. ábra jelölései mellett a következő elemekből állnak:

$$\begin{aligned} N_6(p) &= \{p\} \cup \{u(p), n(p), e(p), s(p), w(p), d(p)\}, \\ N_{18}(p) &= N_6(p) \cup \\ &\quad \{un(p), ue(p), us(p), uw(p), ne(p), se(p), \\ &\quad sw(p), nw(p), dn(p), de(p), ds(p), dw(p)\}, \\ N_{26}(p) &= N_{18}(p) \cup \\ &\quad \{une(p), use(p), usw(p), unw(p), dne(p), dse(p), dsw(p), dnw(p)\}. \end{aligned}$$

Az továbbiakban pontokon mindig \mathbb{Z}^2 - vagy \mathbb{Z}^3 -beli elemeket értünk — az adott kép dimenziójától függően. A p pont j -szomszédos ($j = 4, 6, 8, 18, 26$) a nem-üres S pont-halmazzal, ha van olyan $s \in S$, hogy $s \in N_j(p)$. A különböző pontokból álló $\langle s_0, s_1, \dots, s_n \rangle$



1.4. ábra A p pont $3 \times 3 \times 3$ -as környezetébe eső pontok jelölése.

sorozat $n \geq 0$ hosszú j -út ($j = 4, 6, 8, 18, 26$) az s_0 pontból s_n -be az S ponthalmazban, ha a sorozat minden pontja S -beli és minden i -re ($1 \leq i \leq n$) s_i és s_{i-1} j -szomszédosak. (Megjegyezzük, hogy $\langle s_0 \rangle$ egy 0 -hosszú j -út.) Az $s_1 \in S$ és az $s_2 \in S$ pontok j -összefüggők ($j = 4, 6, 8, 18, 26$) az S ponthalmazban, ha létezik j -út s_1 és s_2 között S -ben. Az S ponthalmaz j -összefüggő ($j = 4, 6, 8, 18, 26$) az S' ponthalmazban ($S' \supseteq S$), ha S bármely két pontja j -összefüggő S' -ben. Könnyű belátni, hogy a j -összefüggőségi reláció — vagyis a reflexív és szimmetrikus j -szomszédosági reláció tranzitív lezártja — ekvivalenciareláció valamennyi j -re ($j = 4, 6, 8, 18, 26$). (A nulla hosszúságú út, egy út megfordítottja és két út konkatenáltja egyaránt út.) A j -összefüggőségi reláció tehát egy osztályozását adja meg egy tetszőleges ponthalmaznak, ahol az ekvivalenciaosztályokat, j -összefüggő komponenseknek vagy j -komponenseknek (j -component) nevezzük.

Egy 2D és egy 3D bináris digitális kép (a továbbiakban kép) \mathbb{Z}^2 -t, illetve \mathbb{Z}^3 -t képezi le a $\{0, 1\}$ halmazba, vagyis minden pont intenzitása/világosságkódja 0 vagy 1 . A \mathcal{P} képet a $\mathcal{P} = (V, m, n, B)$ rendezett négyessel írjuk le (Kong, Rosenfeld, 1989; Kong, Roscoe, Rosenfeld, 1992), ahol:

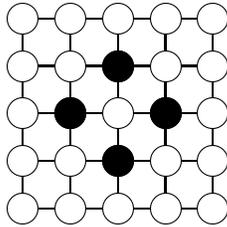
- V a diszkrét képpontok halmaza. A továbbiakban feltételezzük, hogy $V = \mathbb{Z}^2$ 2D kép esetén és $V = \mathbb{Z}^3$ 3D-ben, vagyis csak ortogonális rácsokon értelmezett képekkel foglalkozunk.
- $B \subseteq V$ a fekete pontok halmaza, mely pontokhoz “1” értéket rendelünk. $V \setminus B$ a fehér pontok halmaza, melynek elemei “0” értékűek.
- m a B -hez, a fekete pontokhoz rendelt szomszédosági reláció: $m = 4, 8$, ha $V = \mathbb{Z}^2$, és $m = 6, 18, 26$, ha $V = \mathbb{Z}^3$.
- n a $V \setminus B$ -re, a fehér pontokra érvényes szomszédosági reláció: $n = 4, 8$ 2D-ben és $n = 6, 18, 26$ 3D kép esetén.

A $\mathcal{P} = (V, m, n, B)$ képet röviden (m, n) képnek is nevezzük. A \mathcal{P} kép *véges*, ha B véges. A véges képek reprezentálhatók egy (véges) bináris tömbbel, ahol valamennyi tömbön kívüli pont értéke “0”.

Az m -összefüggőségi (ekvivalencia)reláció a fekete pontok halmazán létrehoz egy osztályozást. Az egy ekvivalenciaosztályba eső fekete pontok halmaza az adott kép egy m -komponense, más elnevezésekkel *fekete komponense* vagy *objektuma*. Hasonlóképpen: az n -összefüggőségi reláció a fehér pontokat n -komponensekre vagy *fehér komponensekre* osztja fel. Véges képen egyetlen végtelen fehér komponens található, amit *háttérnek* (*background*) nevezünk. A véges fehér komponens neve *üreg* (*cavity*).

A $\mathcal{P} = (V, m, n, B)$ képen a $p \in B$ fekete pont *határpont* (*border-point*), ha n -szomszédos legalább egy fehér ponttal, azaz: $N_n(p) \setminus B \neq \emptyset$. A $p \in B$ fekete pont *izolált* (*isolated*), ha $(N_m(p) \setminus \{p\}) \cap B = \emptyset$ (vagyis a $\{p\}$ halmaz egy egyelemű objektum).

A $\mathcal{P} = (V, m, n, B)$ képen található objektumok és üregek száma függ a szomszédsági relációk (n és m) megválasztásától. Az 1.5. ábrán látható 2D képen $(m, n) = (4, 8)$ esetén négy objektum található rajta (mindegyik egy-egy izolált pontot tartalmaz) és nincs ürege. Ha $(m, n) = (8, 4)$, akkor egy objektum és egy üreg van a képen.



1.5. ábra 2D példa összefüggőségi paradoxonokra. A “o” szimbólumok a fehér, a “•”-ek a fekete pontokat jelölik.

Az 1.5. ábra egyben példa az *összefüggőségi paradoxonokra* is (Kong, Rosenfeld, 1989). Ha $(m, n) = (8, 8)$, akkor egyetlen objektumunk van — ami megfelel egy egyszerű zárt görbének (Jordan-görbe) — és nincs a képen üreg, minden fehér pont a háttérhez tartozik. Ez esetben nem teljesül a *Jordan-görbe tulajdonság*, miszerint a Jordan-görbék a komplementerüket kettő — egy külső és egy belső — komponensre osztják (Herman, 1992). Ha $(4, 4)$ képet tételezünk fel, akkor a négy darab objektum képes a fehér pontokat két komponenssé szétválasztani (a háttérre és egy üregre), vagyis különálló objektumok Jordan-görbéként viselkednek. Az összefüggőségi paradoxonok miatt különböző szomszédsági relációkat szoktak választani a fekete és a fehér pontokra: 2D-ben $(m, n) = (8, 4)$, 3D-ben pedig $(m, n) = (26, 6)$ a leggyakoribb.

Megjegyezzük, hogy javasoltak azonos szomszédsági relációkat is a fehér és a fekete pontokra, de ekkor a fekete pontok halmaza nem lehet tetszőleges. A 2D *jól-képzett halmazok* (*well-composed sets*) (Latecki, Eckhardt, Rosenfeld, 1995) nem tartalmaznak egymással 8-szomszédos, de nem 4-szomszédos elemeket és a 3D *jól-képzett képek* (*well-composed pictures*) (Latecki, 1997) nem tartalmaznak egymással 26- vagy 18-szomszédos, de nem 6-szomszédos fekete pontokat. A fenti megszorításoknak eleget tevő képosztályokra teljesül a Jordan-görbe tulajdonság és az Euler-karakterisztika is számítható lokálisan.

A topológia 3D-ben bonyolultabb, mint 2D esetén, mivel 3D-ben a fehér pontok *lyukakat* (*hole*, *tunnel*) is alkothatnak⁵. Egy lyuk — ami például a tórusznak van — más, mint az üreg (Lee, Poston, Rosenfeld, 1991; Lee, Poston, Rosenfeld, 1993). A lyuk egzakt matematikai meghatározása (Kong, 1989) körülményes, de esetünkben nincs is rá szükség. A 3D digitális képek topológiája leírható a lyukak számának ismeretében, ami pedig — az általunk feltételezett 3D (26,6)) képekre — lokálisan meghatározható a lyuk definíciója nélkül is (Lobregt, Verbeek, Groen, 1980; Lee, Kashyap, Chu, 1994). 3D-ben fontos topológiai jellemző az *Euler-karakterisztika*, melyet a következő globális formula ad meg a \mathcal{P} képre:

$$\chi(\mathcal{P}) = \mathcal{O}(\mathcal{P}) - \mathcal{H}(\mathcal{P}) + \mathcal{C}(\mathcal{P}),$$

ahol $\mathcal{O}(\mathcal{P})$ az objektumok, $\mathcal{H}(\mathcal{P})$ a lyukak, $\mathcal{C}(\mathcal{P})$ pedig az üregek száma a \mathcal{P} képen⁶.

⁵Érdemes megjegyezni, hogy a szakirodalom szóhasználata félreértésekre adhat alkalmat. A véges fehér komponensekre 2D-ben a “*hole*”, 3D-ben pedig a “*cavity*” elnevezések terjedtek el, amelyekre a jelen dolgozatban egységesen az “*üreg*” kifejezést használjuk. A szakirodalomban a 3D-ben alkalmazott “*hole*” fogalomnak — amit mi “*lyuk*”-nak fordítottunk — nincs 2D megfelelője.

⁶Megjegyezzük, hogy a \mathcal{P} 2D kép Euler-karakterisztikáját a $\chi(\mathcal{P}) = \mathcal{O}(\mathcal{P}) - \mathcal{C}(\mathcal{P})$ formula adja meg, ahol $\mathcal{O}(\mathcal{P})$ az objektumok és $\mathcal{C}(\mathcal{P})$ az üregek számát jelöli ugyanúgy, mint a 3D formulában.

1.2.2. A topológia megőrzése

Egy \mathcal{T} (egyváltozós) képművelet a $\mathcal{P} = (V, m, n, B)$ kiindulási képre a $\mathcal{P}_{\mathcal{T}} = (V, m, n, \mathcal{T}(B))$ képet eredményezi.

A váz kijelölésekor a kiindulási képről töröljük (fehérré változtatjuk) azokat a fekete pontokat, amelyek nem tartoznak az objektum(ok) vázához. A vázkijelölés tehát egy olyan képművelettel írható le, ami a fekete pontokat fehérré változtathatja, de a fehéret nem színezheti át feketévé. Az ilyen tulajdonságú képműveleteket redukciónak nevezük. Formálisan: a \mathcal{T} képművelet *redukció* (*reduction*), ha tetszőleges kép B halmazára $\mathcal{T}(B) \subseteq B$.

Egy másik kitüntetett képművelet-osztályba azok a műveletek tartoznak, amelyek csak fehér pontokat változtathatnak meg. A \mathcal{T} képművelet *addíció* (*addition*), ha tetszőleges B halmazra $B \subseteq \mathcal{T}(B)$. (Addícióra példa a dilatáció, amit az 1.4.1. pontban mutatunk be.)

A vázkijelölés topológiai kritériuma megkívánja a kiindulási és az eredménykép topológiai ekvivalenciáját, vagyis hogy az alkalmazott képművelet *topológia-megőrző* (*topology preserving*) redukció legyen.

Egy képművelet akkor topológia-megőrző redukció, ha a kiindulási kép egyetlen objektumát sem vágja szét (kettő vagy több objektummá); egyetlen objektumot sem tüntet el teljesen (nem törli annak valamennyi pontját); a kiindulási kép egyetlen üregét sem olvasztja össze más üreggel vagy a háttérrel; nem hoz létre olyan üreget, ami nem volt az eredeti képen (Kong, 1995). 3D-ben egy topológia-megőrző redukciónak továbbá nem szabad a kiindulási kép egyetlen lyukát sem megszüntetnie vagy más lyukkal összevonnia; vagy vadonatúj lyukat létrehoznia. A fenti kritériumoknak teljesülniük kell minden lehetséges kiindulási képre. Az 1.6. ábra egy 2D, az 1.7. ábra pedig egy 3D példát ad olyan redukcióra, ami nem őrzi meg a topológiát.

A topológia-megőrző redukcióhoz hasonlóan megfogalmazhatjuk a *topológia-megőrző addícióval* szembeni követelményeket is. (Ez esetben objektumok összeolvasztása, lyukak és alagutak “feltöltése” minősül nemkívánatosnak.)

A topológia-megőrzés vizsgálatában kulcsfogalom az *egyszerű pont* (*simple point*) (Morgenthaler, 1981). A $p \in B$ pont egyszerű a (V, m, n, B) képen, ha a $(V, m, n, B \setminus \{p\})$ képet előállító képművelet topológia-megőrző redukció. Más szavakkal: azok a pontok egyszerűek, amelyeknek színe (fehér vagy fekete mivolta) nincs hatással az adott kép topológiájára.

Az “egyszerűség” fehér pontokra is megfogalmazható: A $p \in (V \setminus B)$ pont egyszerű a (V, m, n, B) képen, ha a $(V, m, n, B \cup \{p\})$ képet előállító képművelet topológia-megőrző addíció.

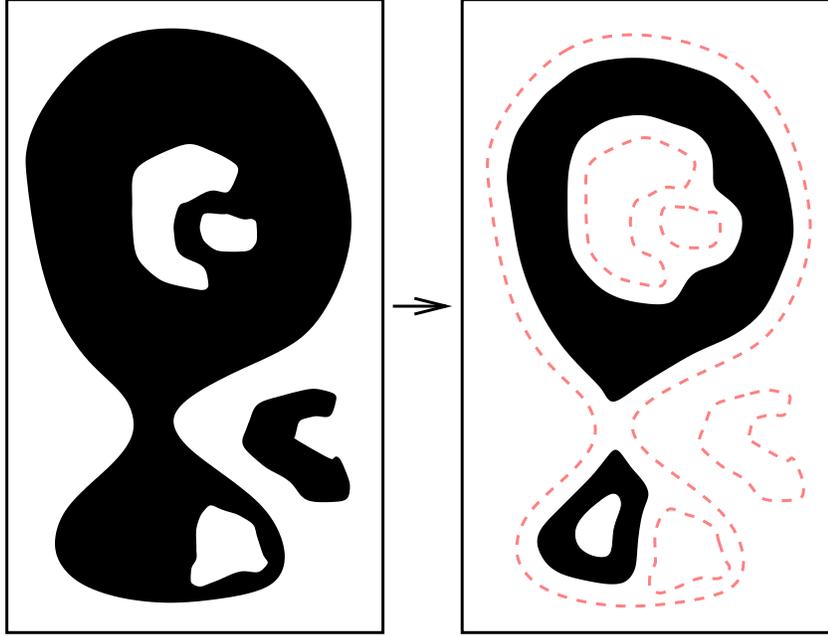
Az egyszerű pontra számos kritériumot állítottak fel. Ezek közül egyet ismertetünk 2D képekre és hármát 3D képekre.

1.1. KRITÉRIUM (Rosenfeld, 1970)

Legyen $p \in B$ egy (fekete) pont a $\mathcal{P} = (\mathbb{Z}^2, m, n, B)$ 2D képen, ahol $(m, n) = (4, 8)$ vagy $(8, 4)$. Tegyük fel, hogy $B \cap (N_m(p) \setminus \{p\}) \neq \emptyset$ (vagyis a p pont nem izolált) és $N_n(p) \setminus B \neq \emptyset$ (vagyis p határpont). Ekkor a következő állítások egymással ekvivalensek:

1. p egyszerű pont a $\mathcal{P} = (\mathbb{Z}^2, m, n, B)$ képen.
2. p m -szomszédos az $N_8(p) \cap (B \setminus \{p\})$ ponthalmaz pontosan egy m -komponensével.
3. p n -szomszédos az $N_8(p) \setminus B$ ponthalmaz pontosan egy n -komponensével.

Az 1.1. Kritérium szerint egy pont egyszerű volta eldönthető lokálisan, vagyis az adott pont 3×3 -as környezete alapján.



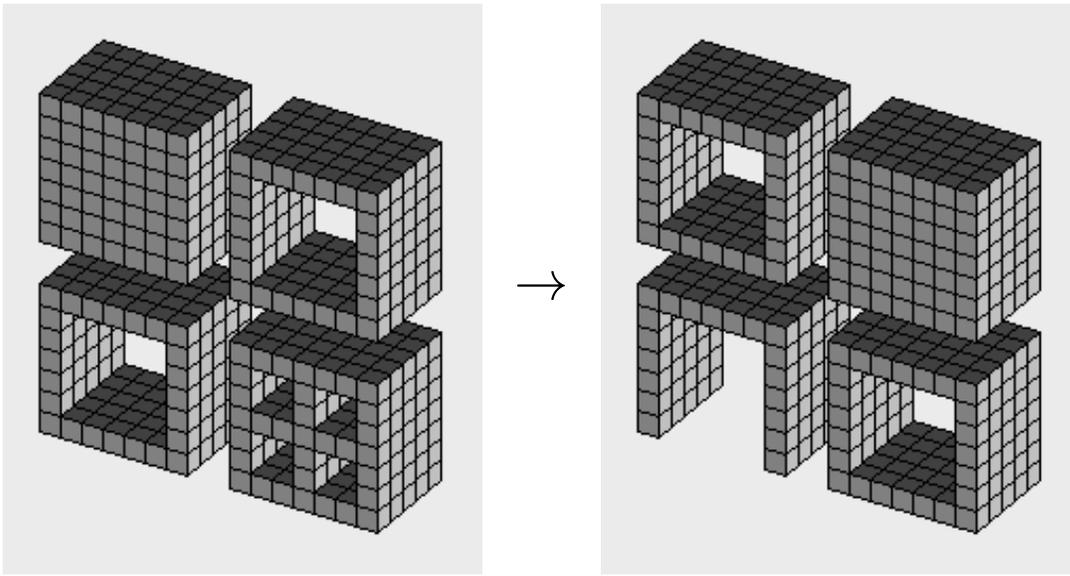
1.6. ábra Példa olyan 2D redukcióra, ami nem őrzi meg a topológiát. A kiindulási kép kettő objektuma közül az egyik három üreget tartalmaz. (Az eredménykép szaggatott vonalai csak az eredeti komponensek határvonalait jelzik, valójában nem szerepelnek a képen.) A képművelet a következők miatt nem őrzi meg a topológiát: a nagyobbik objektumot kettő objektummá vágta szét, a kisebbik objektumot teljesen eltüntette, (egy vadonatúj) üreget hozott létre, kettő üreget összeolvasztott, a harmadik üreget pedig megszüntette (összeolvasztotta a háttérrel).

1.2. KRITÉRIUM (Morgenthaler, 1981; Kong and Rosenfeld, 1989; Ma, 1994)

Legyen $p \in B$ egy (fekete) pont a $\mathcal{P} = (\mathbb{Z}^3, m, n, B)$ 3D képen, ahol $(m, n) = (6, 26)$ vagy $(26, 6)$. A p pont akkor és csakis akkor egyszerű, ha az alábbi feltételek teljesülnek:

1. p m -szomszédos az $N_{26}(p) \cap (B \setminus \{p\})$ ponthalmaznak pontosan egy m -komponensével.
2. p n -szomszédos az $N_{26}(p) \setminus B$ ponthalmaznak pontosan egy n -komponensével.
3. $\chi((\mathbb{Z}^3, m, n, B \cap N_{26}(p))) = \chi((\mathbb{Z}^3, m, n, (B \setminus \{p\}) \cap N_{26}(p)))$
(vagyis nem függ p "színétől" annak a képnek az Euler-karakterisztikája, amit \mathcal{P} -ből úgy kapunk, hogy töröljük a kitüntetett p pont $3 \times 3 \times 3$ -as környezetén kívül eső fekete pontokat).

Az 1.2. Kritérium érvényes a $(6, 18)$ és a $(18, 6)$ képekre is (Kong, 1986). A $(26, 6)$ és a $(18, 6)$ képek esetén az 1. és a 3. feltételek együttes teljesülése a 2. feltételt is maga után vonja (Tsao, Fu, 1982). A $(6, 26)$ és a $(6, 18)$ képekre a 2. és a 3. feltételekből következik az 1. (Kong, 1986). Az 1.2. Kritérium szerint az "egyszerűség" 3D-ben is



1.7. ábra Példa olyan 3D képműveletre, ami nem őrzi meg a topológiát. Ezúttal csak a lyukra (mely fogalomnak nincs megfelelője 2D-ben) vonatkozó anomáliákat érzékeltettük, mivel az objektumok és az üregek tekintetében nincs különbség a 2D és a 3D képek között. A bal oldali kiindulási kép négy objektuma közül három “lyukas”. A jobb oldali képet eredményező redukció nem őrzi meg a topológiát, mivel olyan lyukat hozott létre, ami nem szerepelt az eredeti képen, lyukat szüntetett meg és lyukakat olvasztott össze. (A képművelet nem tiszta redukció, mivel fehér pontokat is megváltoztatott.)

lokális tulajdonság: a felsorolt (különböző m és n relációkat feltételező) képek esetén eldönthető a kérdéses pont $3 \times 3 \times 3$ -as környezete alapján.

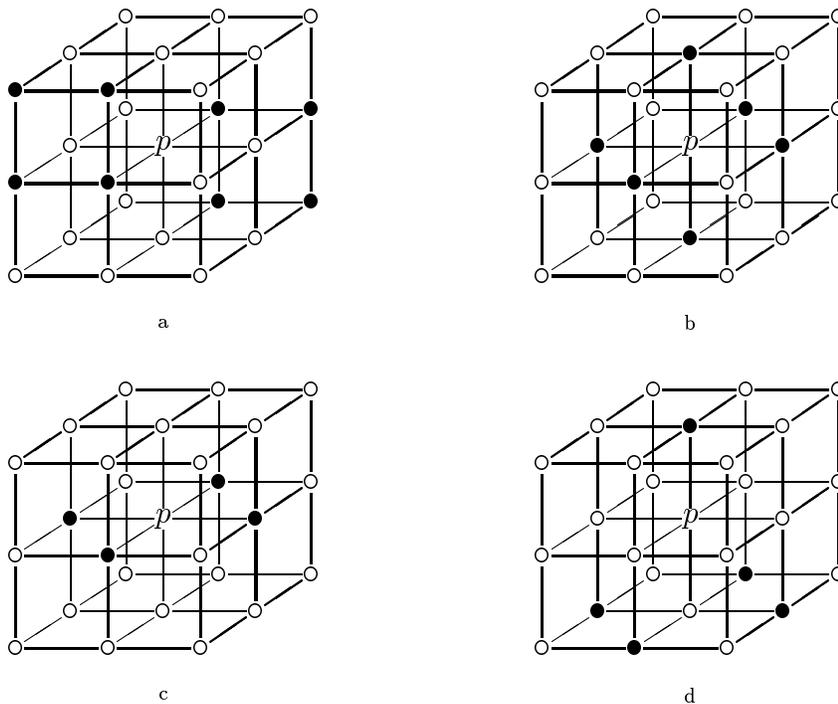
1.3. KRITÉRIUM (Malandain, Bertrand, 1992; Saha, Chaudhuri, 1994)

A p fekete pont akkor és csakis akkor egyszerű a $\mathcal{P} = (\mathbb{Z}^3, 26, 6, B)$ képen, ha az alábbi feltételek mind teljesülnek:

1. A $N_{26}(p) \cap (B \setminus \{p\})$ ponthalmaz pontosan egy 26-komponenst tartalmaz.
2. $(\mathbb{Z}^3 \setminus B) \cap N_6(p) \neq \emptyset$ (vagyis p 6-szomszédos legalább egy fehér ponttal).
3. A $(\mathbb{Z}^3 \setminus B) \cap N_6(p)$ ponthalmaz bármely két pontja 6-összefüggő $(\mathbb{Z}^3 \setminus B) \cap N_{18}(p)$ -ben (vagyis p fehér 6-szomszédai 6-úttal összeköthetők a p fehér 18-szomszédai alkotta ponthalmazban).

A $(26, 6)$ képekre érvényes 1.3. Kritérium szerint az “egyszerűség” az Euler-karakterisztika kiszámítása nélkül is eldönthető.

Az 1.8. ábrán példákat láthatunk olyan pontokra, amelyek nem egyszerűek $(26, 6)$ -os képeken. Az ábra (a) konfigurációja az 1.2. és az 1.3. Kritériumoknak egyaránt az 1. pontját sérti meg: A p pont törlése azzal a veszéllyel jár, hogy objektumot vágunk szét, ami nem megengedett topológia-megőrző képműveletknél. A (b) konfiguráció az 1.2. és az 1.3. Kritériumok 2. pontját nem elégíti ki: A p pont törlése egy egyelemű üreget hoz létre. A (c) konfiguráció esetén a 3D-re ismertetett kritériumok 3. pontja nem teljesül (míg az első kettő igen): a p -t törő képművelet nem topológia-megőrző, mivel egy lyukat hoz létre. A (b) és a (c) konfigurációk rámutatnak arra, hogy a kritériumok első feltételének teljesülése még nem elegendő az “egyszerűséghez”, így egy



1.8. ábra Példák a $(26, 6)$ képek nem-egyszerű pontjaira. (A “o” szimbólumok fehér, a “•”-ek fekete pontokat jelölnek.)

korai (szekvenciális) 3D vékonyító algoritmus (Srihari, Udupa, Yau, 1979) nem topológia-megőrző. Az 1.8. ábra (d) konfigurációja arra ad példát, hogy egymagában az 1.2. Kritérium 3. pontja sem elegendő. Ha p fekete, akkor a minikép egyetlen objektumot tartalmaz, nincs rajta sem üreg, sem lyuk, így az Euler-karakterisztika $1 - 0 + 0 = 1$. Ha p -t fehérré változtatjuk, akkor kettő objektumot kapunk, továbbra sincs üreg a képen, de egy lyuk is keletkezik. Az Euler-karakterisztika $2 - 1 + 0 = 1$, vagyis változatlan maradt. Ennélfogva a csak az Euler-karakterisztika változatlanságát figyelő korai vázkijelölő eljárás (Lobregt, Verbeek, Groen, 1980) sem topológia-megőrző.

A digitális képek pontjait a \mathbb{Z}^2 , illetve a \mathbb{Z}^3 halmazoknak, vagyis ortogonális rácsok rácspontjainak feleltettük meg (és az 1.5–1.5. ábrákon is ennek megfelelően ábrázoltuk őket). Egy képpontnak — a rácsponos megközelítéssel ekvivalens módon — megfeleltethetünk egy egységnégyzetet (2D-ben), illetve egy egységkockát (3D-ben). A $p \in \mathbb{Z}^2$ pontot helyettesíthetjük azzal a négyzettel — elnevezése *pixel* (*picture element*) —, amelynek “1” hosszúságú oldalai párhuzamosak a koordinátatengelyekkel és középpontja egybeesik p -vel. Hasonlóképpen: a $q \in \mathbb{Z}^3$ ponthoz hozzárendelhetjük azt az “1” élhosszúságú kockát — elnevezése *voxel* (*volume element*) —, amelynek élei párhuzamosak a koordinátatengelyekkel és középpontja egybeesik q -val.

Kong az egyszerű pontnak egy olyan jellemzését dolgozta ki, ami az egységnégyzetes, illetve az egységkockás reprezentációt használja (Kong, 1995). Kritériuma ismertetése előtt néhány további fogalmat kell bevezetni.

Egy pixel *határa* annak négy oldala (a végpontjaikkal együtt). Egy voxel *határát* pedig annak hat oldala alkotja (az éleket és a csúcsokat is beleértve). A \mathcal{P} 2D (3D) kép egy p fekete pixelének (voxelének) *csatolt halmaza* (*attachment set*) p határának azon pontjaiból áll, amelyek a \mathcal{P} kép legalább egy másik fekete pixelének (voxelének) határával közösek.

1.4. KRITÉRIUM (Kong, 1995)

A $\mathcal{P}(8, 4)$ vagy $(26, 6)$ kép egy p fekete pixele vagy voxele akkor és csakis akkor egyszerű, ha p csatolt halmaza és annak komplementere (vagyis a p pont határának azon elemei, amelyek nem tartoznak a p csatolt halmazához) egyszerre összefüggő és nem üres.

Az 1.4. Kritériumban szereplő összefüggőséget úgy kell érteni, hogy egy halmaz összefüggő, ha bármely két pontja összeköthető olyan görbével (pl. törtvonallal), melynek minden pontja az adott halmazba esik. Kong tehát a diszkrét képek topológiai jellemzéséhez egy folytonos modellt alkotott.

A csatolt halmazokra és az 1.4. Kritérium alkalmazására az 1.9. ábrával adunk példákat. Az 1.9. ábra (a) konfigurációján p egyszerű. A (c) konfiguráción nem egyszerű p , mivel a csatolt halmaza nem összefüggő. Az (e) konfiguráció szintén nem egyszerű pont p , mivel a csatolt halmazának a komplementere nem összefüggő.

A fenti 2D vagy a 3D képek fekete pontjairól tehát — azok 3×3 -as, illetve $3 \times 3 \times 3$ -as környezetük alapján — lokálisan eldönthető, hogy egyszerűek-e vagy sem. Térjünk ki arra is, hogy milyen áron.

2D-ben egyszerű az “egyszerűség” eldöntése, mivel egy pont 3×3 -as környezetében mindössze $2^8 = 256$ -féle konfiguráció fordul elő. Valamennyi konfiguráció “egyszerűsége-re” a választ tárolhatjuk egy előre kitöltött (explicit) 256-elemű bináris tömbben, más szóhasználattal: keresőtáblában (*look-up table*). Ha ez a tömb rendelkezésre áll, akkor egy adott pont 8-szomszédaiból képzett 8-bites bináris számmal kell csak azt megcímezni.

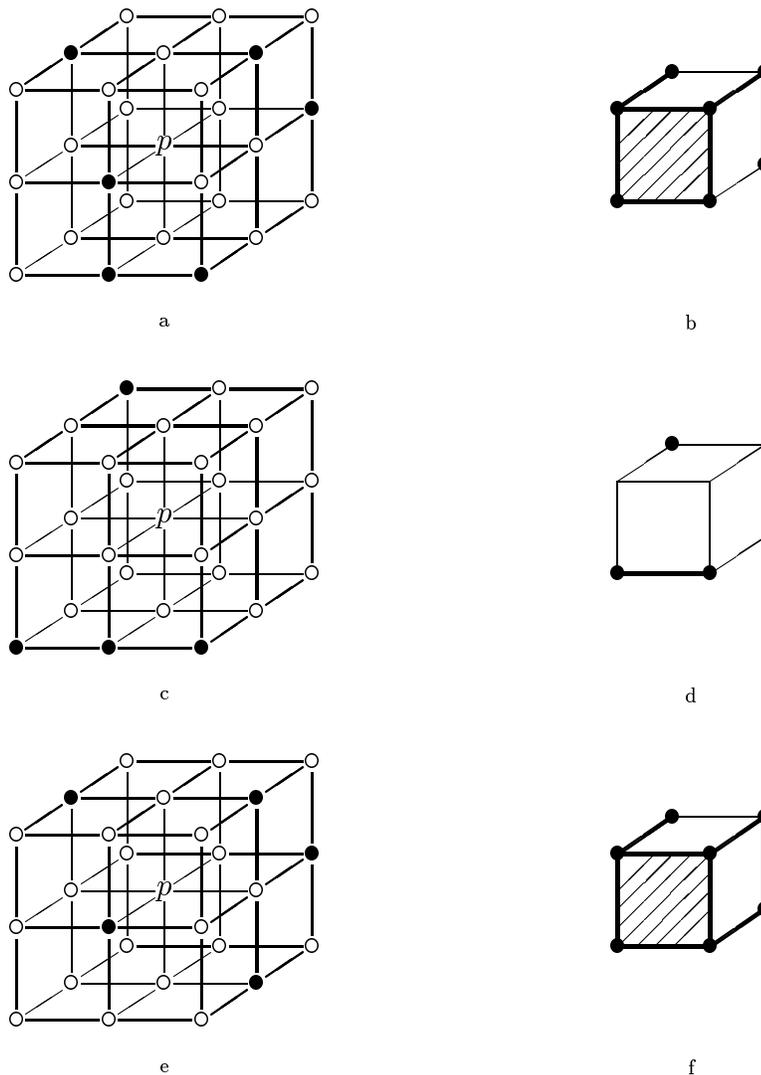
A 3D képek esetében a fenti út nehezen járható, vagy egyáltalán nem alkalmazható. Egy pont $3 \times 3 \times 3$ -as környezetében túl sok, $2^{26} = 67\,108\,864$ -féle konfiguráció fordul elő. Egy 2^{26} -elemű bináris tömb tárigénye 8 Mbyte. Nem csak az összes lehetséges konfigurációk száma nagy, hanem az egyszerű pontokat rejtőké is: 25 985 144-féle egyszerű pont létezik a $(26, 6)$ vagy a $(6, 26)$ képeken (Lee, Kashyap, Chu, 1994).

A fentiek miatt a tisztán keresőtáblás megoldás helyett más módszereket célszerű (vagy kell) alkalmazni. Az egyik megközelítés olyan algoritmusok kidolgozása, melyek az egyetlen nagy (globális) keresőtábla helyett kezelhető méretű táblát vagy táblákat igényelnek (Saha, Chaudhuri, 1994; Latecki, Ma, 1996; Saha, Chaudhuri, 1996). Lee, Kashyap és Chu objektumcímkézéssel és Euler-táblázat használatával oldották meg a problémát (Lee, Kashyap, Chu, 1994), míg Malandain és Bertrand az általuk javasolt címkézési eljárással (táblázat alkalmazása nélkül) döntötték el az “egyszerűséget” (Malandain, Bertrand, 1992). Bertrand egy egyszerű, könnyen implementálható Boole-feltételrendszert javasolt (Bertrand, 1996).

Az egyszerű pont jellemzése után két képművelet-osztály kerül bevezetésre. Egy képművelet *párhuzamos* (*parallel*), ha valamennyi képpont új értéke egyidejűleg számítható ki, vagyis egyetlen pont új értéke sem függ más pontok új értékétől. Ilyen képművelet például a pontonkénti negáció. Egy képműveletet *lokális* (*local*), ha egy képpont új értéke az adott pontnak csak egy szűk, a képmérettől (a fekete pontokat befoglaló tömb méretétől) független környezetétől függ. Lokális képműveletre számos példa található a jelen dolgozatban — többek között az ismertetésre kerülő valamennyi vékonyító algoritmus is lokális képműveletek sorozatából áll. Megjegyezzük, hogy egy lokális képművelet nem szükségképpen párhuzamos is egyben. A távolságtérképeket előállító algoritmusok — amelyekről szó lesz az 1.3.1. pontban — (nem-bináris képekre adott) lokális, de nem párhuzamos képműveletek.

Az egyszerű pont fogalma segítségével könnyen eldönthető, hogy egyetlen pont törlése mint redukció topológia-megőrző képművelet-e vagy sem. Több egyszerű pontot egyide-

júleg törlő párhuzamos redukció viszont nem biztos, hogy topológia-megőrző képművelet. Például egy 2D képen egy 2×2 pontból álló objektum valamennyi pontja egyszerű. Párhuzamos törlésük esetén viszont egy objektum teljesen eltűnik a képről, ami nem megengedett a topológia-megőrző redukciónál.



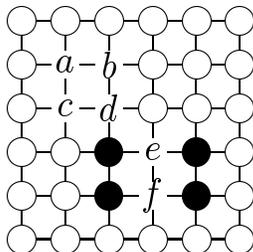
1.9. ábra Példák 3D csatolt halmazokra. Az (a), a (c) és az (e) konfigurációk a p fekete pont $3 \times 3 \times 3$ -as környezetét ábrázolják rácspontos megközelítés szerint. (A “o” szimbólumok fehér, a “•”-ek a fekete pontokat jelölnék.) Az (a), a (c) és az (e) konfigurációk p pontjainak rendre a (b), a (d) és az (f) csatolt halmazok felelnek meg. (A csatolt halmazokhoz tartozó lapokat sátrózás, az éleket vastagított vonalak, a csúcsokat pedig “•”-ek jelölik.) A három példa közül p csak az (a) konfiguráción egyszerű a (26, 6) képeken.

A párhuzamos redukciók topológia-megőrzésére 2D-ben Rosenfeld és Ronse fogalmazott meg elegendő feltételeket (Rosenfeld, 1975; Ronse, 1986; Ronse, 1988), amelyek ismertetésétől eltekintünk. 3D-ben C.M. Ma ért el eredményeket (Ma, 1993; Ma, 1994), melyek bemutatásához szükségünk van az egyszerű ponthalmaz és a minimálisan nem egyszerű ponthalmaz fogalmának bevezetésére.

Fekete pontok egy $D = \{d_1, \dots, d_k\}$ halmaza *egyszerű halmaz* (simple set) a $\mathcal{P} = (V, m, n, B)$ képen, ha a halmaz elemei olyan $\langle d_{i_1}, \dots, d_{i_k} \rangle$ sorozatba rendezhetők, ahol minden egyes d_{i_j} ($j = 1, \dots, k$) pont egyszerű a $\{d_{i_1}, \dots, d_{i_{j-1}}\}$ ponthalmaz törlése után

(Kong, 1993). (Az üres halmaz legyen — definíció szerint — egyszerű halmaz.) A fenti definícióban szereplő sorozatot *egyszerű sorozatnak* (*simple sequence*) nevezzük.

Egyszerű és nem egyszerű halmazokra az 1.10. ábra mutat példát. Az ábrán látható 2D (8, 4) képen az $\{a, b, c, d\}$ halmaz egyszerű, mivel pl. az $\langle a, b, c, d \rangle$ vagy a $\langle d, b, a, c \rangle$ sorozat egyszerű. Megjegyezzük, hogy ugyanakkor pl. a $\langle c, a, d, b \rangle$ sorozat nem egyszerű. Nem egyszerű halmazra példa $\{e, f\}$, mivel sem $\langle e, f \rangle$, sem $\langle f, e \rangle$ nem egyszerű sorozat, jóllehet e és f is egyszerű pont.



1.10. ábra 2D példa egyszerű és nem egyszerű halmazokra (8, 4) képeken. A “o” szimbólumok a fehér pontokatnak felelnek meg, míg a “•” szimbólumokkal és az a – f betűkkel jelölt pontok feketék. Az $\{a, b, c, d\}$ halmaz egyszerű, mivel pl. az $\langle a, b, c, d \rangle$ sorozat egyszerű, míg az $\{e, f\}$ halmaz nem egyszerű.

Nyilvánvaló, hogy a $\mathcal{P} = (V, m, n, B)$ képből a $D = \{d_1, \dots, d_k\}$ egyszerű halmazt törölő, vagyis a $\mathcal{P}_{\mathcal{T}} = (V, m, n, B \setminus D)$ képet eredményező \mathcal{T} redukció topológia-megőrző, mivel az helyettesíthető olyan képműveletek egymás utáni végrehajtásával, mely műveletek csak egy-egy egyszerű pontot törölnek. Az ilyen műveletek (definíció szerint) topológia-megőrzők és azok kompozíciója is az. Formálisan: $\mathcal{T} = \mathcal{T}_1 \cdot \dots \cdot \mathcal{T}_k$, ahol a \mathcal{T}_j képművelet a d_{i_j} (egyszerű) pontot törli a $(V, m, n, B \setminus \{d_{i_1}, \dots, d_{i_{j-1}}\})$ képről ($j = 1, \dots, k$).

A fentiekből adódóan a következő kritérium fogalmazható meg:

1.5. KRITÉRIUM (Kong, 1993)

Egy redukció topológia-megőrző, ha minden lehetséges képre az általa egyidejűleg törölt pontok halmaza egyszerű.

Minden lehetséges képre lehetetlen tesztelni egy képműveletet, de erre szerencsére nincs is szükség a minimálisan nem egyszerű halmazok tulajdonságai miatt.

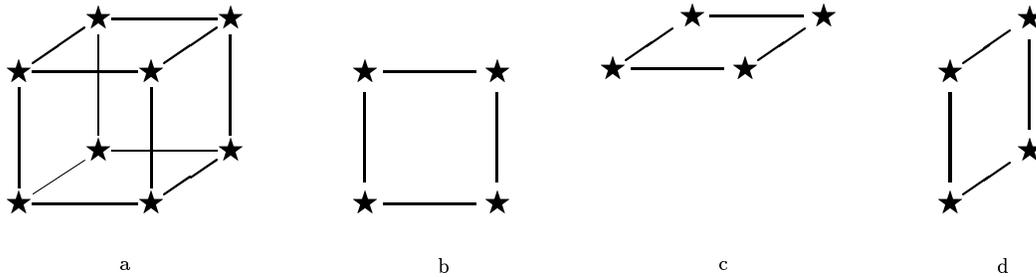
Az $S \subseteq B$ halmaz *minimálisan nem egyszerű* (*minimal non-simple*) a $\mathcal{P} = (V, m, n, B)$ képen, ha nem egyszerű, de bármely $Q \subset S$ valódi részhalmaza már egyszerű. (Az 1.10. ábrán látható (8, 4) képen minimálisan nem egyszerű halmazra példa $\{c, d\}$ vagy $\{e, f\}$.)

Az 1.5. Kritérium a minimálisan nem egyszerű halmazok segítségével a következőképpen írható át: Egy redukció topológia-megőrző, ha egyetlen képről sem töröl minimálisan nem egyszerű halmazt.

A fenti átfogalmazás azért célszerű, mert 2D-ben Ronse megmutatta, hogy a minimálisan nem egyszerű halmazok mérete korlátos: belefoglalhatók egy 2×2 -es pixel-négyzetbe (Ronse, 1988); C.M. Ma pedig bebizonyította, hogy 3D-ben a minimálisan nem egyszerű halmazokat egy $2 \times 2 \times 2$ -es voxel-kocka tartalmazza (Ma, 94). Bináris képeken 16-féle 2×2 -es négyzet és 256-féle $2 \times 2 \times 2$ -es kocka fordulhat elő, így csak “kisszámú” konfiguráció felelhet meg minimálisan nem egyszerű halmaznak.

A (26, 6)-os képek topológia-megőrző redukcióira adott elegendő feltételek ismertetése előtt két fogalmat még be kell vezetni. A \mathbb{Z}^3 ponthalmazon *egység-rácsnégyzetet* (*unit*

lattice square) alkotnak a $2 \times 2 \times 1$ -es, a $2 \times 1 \times 2$ -es vagy az $1 \times 2 \times 2$ -es pont-konfigurációk (1.11. ábra). A \mathbb{Z}^3 ponthalmazon a $2 \times 2 \times 2$ -es konfiguráció neve *egység-rácskočka* (*unit lattice cube*) (1.11. ábra).



1.11. ábra Egység-rácskočka (a) és az egység-rácsnégyzet háromféle megjelenési formája (b)–(d), ahol a (rács)pontokat “★” szimbólumok jelölik. Megállapítható, hogy egy egység-rácskočka minden egyes oldala egy-egy egység-rácsnégyzet. Az egység-rácskočka pontjai egymással 26-szomszédosak, míg az egység-rácsnégyzet pontjai 18-szomszédosak.

1.6. KRITÉRIUM (Ma, 1994)

Egy 3D párhuzamos redukció topológia-megőrző a $(26, 6)$ képekre, ha az alábbi feltételek mindegyikét kielégíti:

1. Csak egyszerű pontokat töröl.
2. Ha egy egység-rácsnégyzet p és q két fekete sarkát törli, akkor a $\{p, q\}$ halmaz egyszerű.
3. Ha egy egység-rácsnégyzet p , q és r három fekete sarkát törli, akkor a $\{p, q, r\}$ halmaz egyszerű.
4. Ha egy egység-rácsnégyzet p , q , r és s (mind a) négy fekete sarkát törli, akkor a $\{p, q, r, s\}$ halmaz egyszerű.
5. Egyetlen egység-rácskočka által tartalmazott objektumot sem töröl teljesen.

C.M. Ma nem csak az 1.6. Kritériumot fogalmazta meg, hanem elegendő feltételeket is adott a topológia-megőrzésre $(6, 26)$, $(18, 6)$ és $(6, 18)$ 3D képek esetében is (Ma, 1994). Ezen eredmények közzlése előtt publikált 3D vékonyító algoritmusokra a topológia-megőrzést nem bizonyították, a szerzők csak a törlésre kijelölt pontok egyszerű voltát igazolták.

Speciális párhuzamos redukciókra (mint amilyenek a 2.2. pontban ismertetésre kerülő 6 al-iterációs határszekvenciális vékonyító algoritmusok) más elegendő feltételeket is közöltek (Bertrand, 1995/1; Ma, 1996). Hangsúlyozandó, hogy a topológia-megőrzésre adott különböző feltételek csak elegendőek, de nem szükségesek és egymással nem is egyenértékűek. Pl. a 2.1.6. pontban bemutatásra kerülő algoritmus (Palágyi, Kuba, 1998/1) teljesíti az 1.6. Kritériumot, de nem elégíti ki C.M. Ma egy másik feltételrendszerét (Ma, 1996). Az alábbiakban egy új kritériumot ismertetünk, ami olyan feltételeket tartalmaz, melyek kielégítése — bizonyos algoritmusokra — egyszerűbben belátható, mint az 1.6. Kritérium feltételeinek teljesülése.

1.7. TÉTEL (Palágyi, Kuba, 1998/1; Palágyi, Kuba, 1999/2)

Legyen \mathcal{T} egy párhuzamos redukció, $\mathcal{P} = (\mathbb{Z}^3, 26, 6, B)$ pedig egy tetszőleges 3D kép. Legyen továbbá $p \in B$ és $Q \subseteq (N_{18}(p) \setminus \{p\}) \cap B$ tetszőleges olyan fekete pont és ponthalmaz, amelyeket töröl a \mathcal{T} képművelet. A \mathcal{T} képművelet topológia-megőrző, ha teljesíti az alábbi két feltételt:

1. p egyszerű a $(\mathbb{Z}^3, 26, 6, B \setminus Q)$ képen.
2. \mathcal{T} egyetlen egység-rácskocka által tartalmazott objektumot sem töröl teljesen.

BIZONYÍTÁS

Megmutatjuk, hogy a tétel feltételeinek teljesülése maga után vonja az 1.6. Kritérium valamennyi feltételének a teljesülését is.

A tétel 2. feltétele megegyezik az 1.6. Kritérium 5. feltételével, így csak azt kell belátni, hogy a tétel 1. feltételéből következnek az 1.6. Kritérium 1–4. feltételei.

Az 1.6. Kritérium 1. feltétele teljesül $Q = \emptyset$ választása esetén, tehát csak a 2.–4. feltételek vizsgálata maradt hátra.

Vegyük észre, hogy $N_{18}(p)$ tartalmazza mindazokat az egység-rácsnégyzeteket, amelyekben p mint csúcs előfordulhat.

Legyen a és b két fekete sarka egy egység-rácsnégyzetnek a $\mathcal{P} = (\mathbb{Z}^3, 26, 6, B)$ képen és törölje \mathcal{T} mind a kettő pontot. A tétel 1. feltétele szerint $a = p$ és $Q = \emptyset$ választásával a egyszerű pont a $(\mathbb{Z}^3, 26, 6, B \setminus Q)$ képen, továbbá $b = p$ és $Q = \{a\}$ esetben a b pont is az, így az $\langle a, b \rangle$ sorozat egyszerű a \mathcal{P} képen. Tehát $\{a, b\}$ halmaz egyszerű, így teljesül az 1.6. Kritérium 2. feltétele.

Legyen a , b és c három fekete sarka egy egység-rácsnégyzetnek a \mathcal{P} képen és törölje \mathcal{T} mindhárom pontot. A fentiekben beláttuk, hogy az a és b pontokra teljesül az 1.6. Kritérium 2. feltétele, vagyis az $\{a, b\}$ halmaz egyszerű a \mathcal{P} képen. A tétel 1. feltétele $c = p$ és $Q = \{a, b\}$ választásakor garantálja a c pont egyszerű voltát a $(\mathbb{Z}^3, 26, 6, B \setminus Q)$ képen. Ennélfogva az $\{a, b, c\}$ halmaz is egyszerű a \mathcal{P} képen, vagyis teljesül az 1.6. Kritérium 3. feltétele.

Legyen végül a , b , c és d négy fekete sarka egy egység-rácsnégyzetnek a \mathcal{P} képen és törölje \mathcal{T} mind négy pontot. Ha a tétel 1. feltételét $d = p$ -re és $Q = \{a, b, c\}$ -re alkalmazzuk, akkor a d pont egyszerű a $(\mathbb{Z}^3, 26, 6, B \setminus Q)$ képen. Beláttuk, hogy teljesül az 1.6. Kritérium 3. feltétele, vagyis az $\{a, b, c\}$ halmaz egyszerű a \mathcal{P} képen. Következésképpen az $\{a, b, c, d\}$ halmaz is egyszerű, azaz teljesül az 1.6. Kritérium 4. feltétele is.

Beláttuk, hogy mindazon \mathcal{T} párhuzamos redukciók, amelyeket kielégítik a tétel feltételeit, azok átmennek az 1.6. Kritérium szűrőjén is, vagyis megőrzik a topológiát. ■

1.3. Vázkielölő stratégiák

Diszkrét objektumok vázának meghatározására több módszert javasoltak. Ezek közül a három legfontosabb technikát a távolság–transzformán alapulót (1.3.1.), a front–terjedést modellező vékonyítást (1.3.2.) és a Voronoi–diagramok határpontokból történő generálását (1.3.3.) mutatjuk be. Nem tárgyaljuk azokat a módszereket, amelyek csak speciális osztályokhoz tartozó objektumok vázának analitikus kivonására fejlesztettek ki (Kirkpatrick, 1979; Osher, Sethian, 1988; Sethian, 1990).

1.3.1. Távolság–transzformáción alapuló módszerek

A váz *távolság–transzformációval* (*distance transformation*) történő meghatározása három lépésből áll:

1. A határpontok kivonása.
2. A *távolságtérkép* (*distance map*) előállítása, ahol a jellemző pontok a határpontok.
3. A vázpontok, vagyis a lokális maximumok, “hegygerincek” (*ridge*) megkeresése a távolságtérképen.

Diszkrét objektumok határa többféleképpen is megadható. Pl. 2D–ben a képpontok egységnégyzetes reprezentációja mellett álljon a határ azon fehér pontokból, amelyek élen vagy csúcson osztoznak fekete képpontokkal.

Egy d –dimenziós bináris képet reprezentáló d –dimenziós bináris tömbhöz rendelt távolságtérkép egy vele megegyező méretű d –dimenziós valós tömb, mely minden egyes képpontra megadja a hozzá legközelebb lévő *jellemző ponttól* (*feature point*) — esetünkben határponttól — vett távolságát. Távolságtérkép előállításához tehát a jellemző pontok mellett egy távolságfogalomra is szükség van.

A leggyakoribb távolságok az euklidészi távolság és a szomszédsági relációkon alapuló diszkrét távolságok (Rosenfeld, Pfaltz, 1968; Marchand–Maillet, Sharaiha, 2000). Legyen $x = (x_1, x_2)$ és $y = (y_1, y_2)$ két pont \mathbf{Z}^2 –ben és legyenek $p = (p_1, p_2, p_3)$ és $q = (q_1, q_2, q_3)$ \mathbf{Z}^3 –beli pontok.

$$D_i^2(x, y) \quad : \quad \text{az } x \text{ és az } y \text{ pontok közötti legrövidebb } i\text{-út hossza } \mathbf{Z}^2\text{-ben} \\ (i \in \{4, 8\}) ;$$

$$D_E^2(x, y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2} ;$$

$$D_j^3(p, q) \quad : \quad \text{a } p \text{ és a } q \text{ pontok közötti legrövidebb } j\text{-út hossza } \mathbf{Z}^3\text{-ben} \\ (j \in \{6, 18, 26\}) ;$$

$$D_E^3(p, q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + (p_3 - q_3)^2} .$$

Az ortogonális pontrácsokon az alkalmazott szomszédsági relációk nem adják jó közelítést az euklidészi távolságnak. A fenti távolságok viszonyát az alábbi egyenlőtlenségek fejezik ki (amelyek teljesülnek tetszőleges (x, y) és (p, q) pontpárokra):

$$D_8^2(x, y) \leq D_E^2(x, y) \leq D_4^2(x, y) ; \\ D_{26}^3(p, q) \leq D_{18}^3(p, q) \leq D_E^3(p, q) \leq D_6^3(p, q) .$$

Az euklidészi távolság jobb közelítése érdekében súlyozzák a pontokat összekötő legrövidebb 8-, illetve 26-utak éleit és a súlyok összege adja a $(d1, d2)$ -, illetve a $(d1, d2, d3)$ -*chamfer távolságot* (Borgefors, 1984; Borgefors, 1996).

Legyen az x és y pontok közötti legrövidebb 8-út $\langle x_0, x_1, \dots, x_n \rangle$ ($x_0 = x$, $x_n = y$). A $D_{ch}^{(d1, d2)}(x, y)$ $(d1, d2)$ -chamfer távolságnál az (x_{k-1}, x_k) él súlya legyen $d1$, ha x_{k-1} és x_k 4-szomszédosak, különben legyen $d2$ ($1 \leq k \leq n$).

A $D_{ch}^{(d1, d2, d3)}(p, q)$ $(d1, d2, d3)$ -chamfer távolság hasonlóképpen definiálható: Legyen a p és q pontok közötti legrövidebb 26-út $\langle p_0, p_1, \dots, p_m \rangle$ ($p_0 = p$, $p_m = q$). A (p_{k-1}, p_k) él súlya legyen $d1$, ha $p_{k-1} \in N_6(p_k)$, legyen $d2$, ha $p_{k-1} \in N_{18}(p_k) \setminus N_6(p_k)$ és legyen $d3$, ha $p_{k-1} \in N_{26}(p_k) \setminus N_{18}(p_k)$.

(Megjegyzendő, hogy a chamfer távolságok nem elégítik ki a matematikai távolságfogalmakkal szemben támasztott követelményeket tetszőleges súlyokra, pl. 2D-ben $d1 = 1$, $d2 = 3$ választásakor nem teljesül a háromszög-egyenlőtlenség sem.)

2D-ben a $d1 = 3$, $d2 = 4$ választás jól közelíti az euklidészi távolság 3-szorosát (mivel $\frac{d2}{d1} = \frac{4}{3} = 1,33 \dots \approx \sqrt{2} = 1,414 \dots$). 3D-ben a $d1 = 3$, $d2 = 4$ és $d3 = 5$ választással szokás az euklidészi távolság 3-szorosát közelíteni (mivel $\frac{d2}{d1} = \frac{4}{3} = 1,33 \dots \approx \sqrt{2} = 1,414 \dots$ és $\frac{d3}{d2} = \frac{5}{4} = 1,25 \dots \approx \sqrt{3} = 1,732 \dots$).

A következőkben a távolságtérképek megadását 3D-re mutatjuk be. A távolság-transzformáció bemenete legyen az $A = [a(i, j, k)]$ egy $n1 \times n2 \times n3$ -as bináris tömb, ahol:

$$a(i, j, k) = \begin{cases} 1, & \text{ha az } (i, j, k) \text{ pozíción jellemző pont található} \\ 0, & \text{különben} \end{cases}.$$

Tegyük fel, hogy A tartalmaz "1" elemet, vagyis a jellemző pontok halmaza nem üres.

A távolság-transzformáció eredménye a távolságtérkép, vagyis a $B = [b(i, j, k)]$, ugyancsak $n1 \times n2 \times n3$ -as (nemnegatív valós) tömb, ahol:

$$b(i, j, k) = \begin{cases} 0, & \text{ha az } a(i, j, k) = 1 \\ \min \{ \mathcal{D}((i, j, k), (i', j', k')) \mid a(i', j', k') = 1 \}, & \text{különben} \end{cases},$$

vagyis minden pozícióra a hozzá legközelebbi jellemző ponttól vett távolságát kell kiszámítani a fígyelembe vett \mathcal{D} -távolság szerint.

Borgefors megmutatta, hogy a chamfer távolságokon alapuló 3D távolság-transzformáció $\mathcal{O}(n1n2n3)$ időben meghatározható (Borgefors, 1984), és az időigény tetszőleges dimenzióban is arányos a távolságtérkép elemeinek számával. A távolságtérkép (tetszőleges dimenzió esetén) az alábbi három fázisú eljárással számítható ki:

1. inicializálás,
2. pásztázás előre (*forward scan*) és
3. pásztázás vissza (*backward scan*).

A chamfer távolságokon alapuló távolság-transzformációt 3D-ben a következő program írja le:

```

remark inicializálás
  for k=1 to n3 do
    for i=1 to n1 do
      for j=1 to n2 do
        if a(i,j,k)=1 then
          b(i,j,k)=0
        else
          b(i,j,k)=∞
remark pásztázás előre
  for k=1 to n3 do
    for i=1 to n1 do
      for j=1 to n2 do
        b(i,j,k)=min{
          b(i-1,j-1,k-1)+d3, b(i-1,j-1,k)+d2, b(i-1,j-1,k+1)+d3,
          b(i-1, j,k-1)+d2, b(i-1, j,k)+d1, b(i-1, j,k+1)+d2,
          b(i-1,j+1,k-1)+d3, b(i-1,j+1,k)+d2, b(i-1,j+1,k+1)+d3,
          b( i,j-1,k-1)+d2, b( i,j-1,k)+d1, b( i,j-1,k+1)+d2,
          b( i, j,k-1)+d1, b( i, j,k) }
remark pásztázás vissza
  for k=n3 downto 1 do
    for i=n1 downto 1 do
      for j=n2 downto 1 do
        b(i,j,k)=min{ b(i,j,k), b(i,j,k+1)+d1,
          b( i,j+1,k-1)+d2, b( i,j+1,k)+d1, b( i,j+1,k+1)+d2,
          b(i+1,j-1,k-1)+d3, b(i+1,j-1,k)+d2, b(i+1,j-1,k+1)+d3,
          b(i+1, j,k-1)+d2, b(i+1, j,k)+d1, b(i+1, j,k+1)+d2,
          b(i+1,j+1,k-1)+d3, b(i+1,j+1,k)+d2, b(i+1,j+1,k+1)+d3 }

```

A két bejárás (pásztázás) mindegyike lokális, de nem párhuzamos (tömb)művelet. A lokális függés környezetét és a súlyokat az 1.12. ábra mutatja. Távolság–transzformációra az 1.13. és az 1.14. ábrákkal adunk 2D példákat.

A B távolságtérkép minden elemét háromszor látogattuk meg és mindhárom esetben $\mathcal{O}(1)$ (konstans) időt igényel egy–egy elem új értékének meghatározása. Tehát a fenti program komplexitása valóban $\mathcal{O}(n_1 \cdot n_2 \cdot n_3)$.

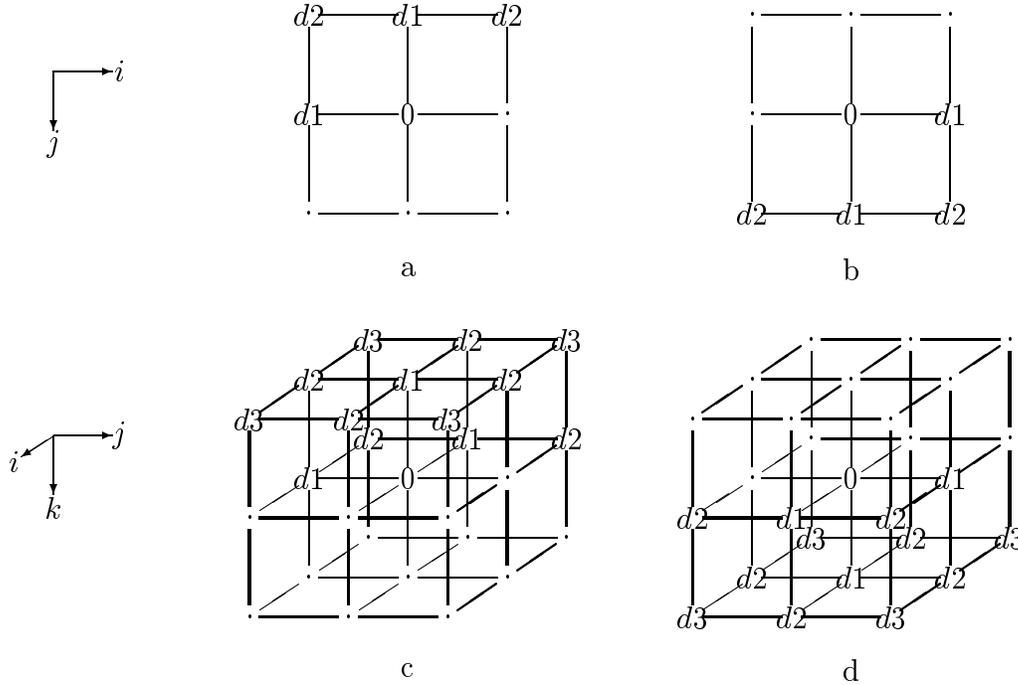
A szomszédsági reláción alapuló távolság–transzformációk megkaphatók alkalmas súlyozású chamfer távolságtérképek kiszámításával:

$$\begin{aligned}
D_4^2 & : d_1 = 1, d_2 = \infty; \\
D_8^2 & : d_1 = 1, d_2 = 1; \\
D_6^3 & : d_1 = 1, d_2 = \infty, d_3 = \infty; \\
D_{18}^3 & : d_1 = 1, d_2 = 1, d_3 = \infty; \\
D_{26}^3 & : d_1 = 1, d_2 = 1, d_3 = 1.
\end{aligned}$$

Egész súlyok mellett a távolságtérkép elemei is egészek. A fenti módszer célja, hogy az egymástól irracionális ($\sqrt{2}$ és $\sqrt{3}$) távolságra lévő rácspontok távolságát racionális számokkal közelítsük.

Danielsson hasonló módszert javasolt “közel–euklidészi” távolságtérképre (*almost Euclydean distance map*), ami csak a jellemző pontok néhány speciális konfigurációja esetén

nem ad pontos eredményt (Danielsson, 1980). Módszere vektor–komponensű (egész szám-párokat tároló) tömbbel dolgozik és az inicializálás után minden elemet négyszer látogat meg. Többen fejlesztettek ki módszereket *hibamentes euklidészi távolságtérkép* (*error-free Euclidean distance map*) előállítására (Yamada, 1984; Ragnemalm, 1990; Ragnemalm, 1992/1; Saito, Toriwaki, 1994; Breu, Gil, Kirkpatrick, Werman, 1995; Embrechts, Roose, 1996; Eggers, 1996; Eggers, 1998; Cuisenaire, Macq, 1999).



1.12. ábra Lokális környezetek és súlyok chamfer távolságtérképekhez. A figyelembe vett szomszédság 2D-ben a pásztázás előre fázisban (a) és a pásztázás vissza menet során (b). 3D-ben a pásztázás előre fázis súlyai (c), a pásztázás vissza menetben pedig (d). A fenti szomszédsági maszkokat középpontjukkal az éppen meglátogatott elemre kell helyezni és a súlyok által lefedett elemek súlyokkal növelt értékének minimumát kell képezni. (A “.”-tal jelölt pozíciók nem játszanak szerepet az adott fázis során.)

A határpontok (mint jellemző pontok) távolság–transzformációja után a távolságtérképen a “hegygerinceket”, a lokális maximumhelyeket kell megtalálni. Távolság–transzformációval történő vázkijelölésre számos algoritmust javasoltak (Dorst, 1986; Arcelli, Sanniti di Baja, 1992; Arcelli, Frucci, 1992; Saito, Toriwaki, 1994; Fernández–Vidal, Mandain, 1995; Saito, Toriwaki, 1995; Borgfors, Nyström, Sanniti di Baja, 1999; Svensson, Nyström, Borgfors, 1999; Gagvani, Silver, 1999).

A távolságtérképen a “hegygerincek”, vagyis a vázhoz tartozó pontok kijelölése nehéz és költséges feladat, különösen $d \geq 3$ dimenzióban. A lokális maximumhelyeket kereső módszerek a differenciálgeometria folytonos függvényekre vonatkozó eredményein alapulnak.

Legyen $f : \mathbb{R}^d \rightarrow \mathbb{R}$ egy folytonos függvény, amelyhez az alábbi ($d \times d$ méretű) Hesse-féle mátrix tartozik:

$$H(f) = \begin{pmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_d} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_d} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_d \partial x_1} & \frac{\partial^2 f}{\partial x_d \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_d^2} \end{pmatrix}.$$

Határozzuk meg a negatív Hesse-féle mátrix ($-H(f)$)

$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_d$$

sajátértékeit és az azokhoz tartozó

$$\vec{e}_1, \vec{e}_2, \dots, \vec{e}_d$$

sajátvektorokat, ahol

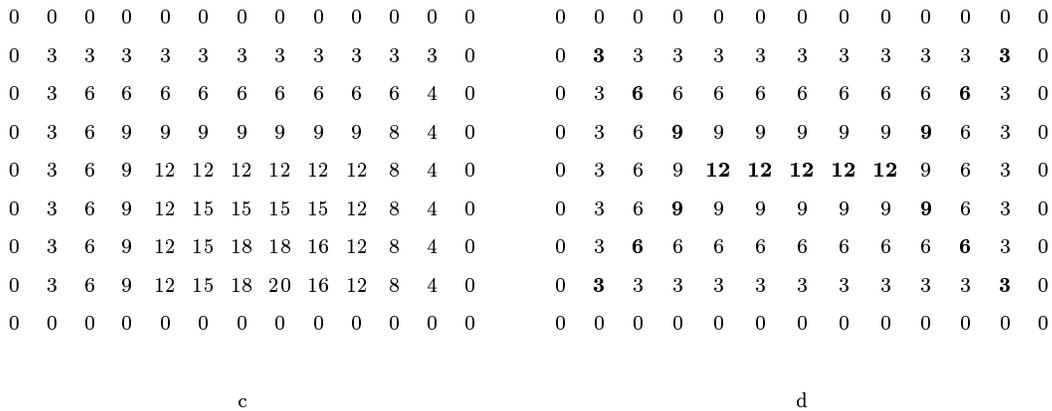
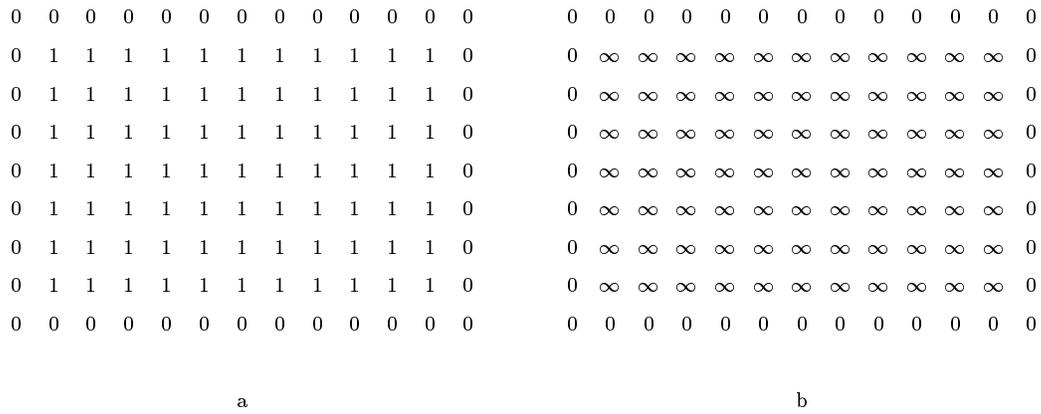
$$-H(f) \cdot \vec{e}_i = \lambda_i \cdot \vec{e}_i \quad (i = 1, 2, \dots, d).$$

A P pontban az f függvénynek $(d - m)$ -dimenziós ($m = 1, 2, \dots, d$) hegygerince található, ha f -nek lokális maximuma van a P pontban az $\vec{e}_1, \vec{e}_2, \dots, \vec{e}_m$ irányokban, vagyis a $-H(f)$ mátrix legnagyobb m darab sajátértékéhez tartozó sajátvektorainak mentén. *Ideális hegygerinc (ideal ridge)* esetén az f függvény csak a hegygerinc merőleges irányban változtatja meg az értékét és konstans a hegygerinc irányában. Ekkor a $-H(f)$ mátrix rangja m és $\lambda_j = 0$ ($j = m + 1, \dots, d$).

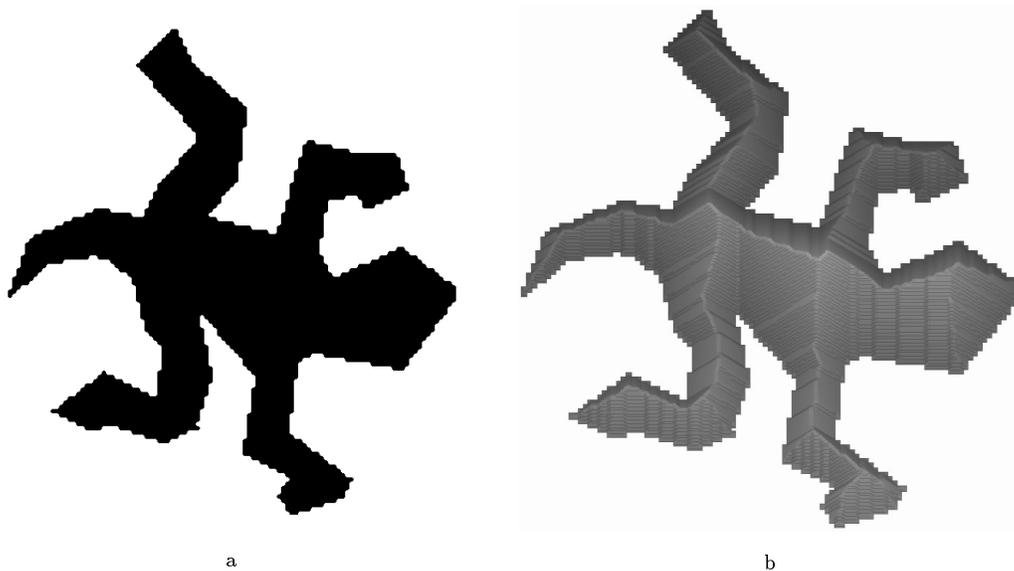
A gyakorlatban nem egy folytonos f függvényen, hanem egy diszkrét B távolságtérképen kell a hegygerinceket megtalálni. Diszkrét esetben nem várható el, hogy ideális hegygerincet találjunk. 2D-ben a távolságtérkép egy eleme hegygerincnek (vázpontnak) felel meg, ha abban a pontban egy “nagy” pozitív és egy “közel nulla” sajátértéke van. A “nagy” pozitív sajátértékhez tartozó sajátvektor adja meg a hegygerinc merőleges irányt. A Hesse-féle mátrix digitális közelítése véges differenciák kiszámításával történhet a távolságtérkép megfelelő (σ paraméterű) Gauss-szűrővel (Sonka, Hlaváč, Boyle, 1999) való simítását követően.

Arcelli és Sanniti di Baja egy egyszerűbb módszert javasolt súlyozott (pl. chamfer távolsággal számított) távolságtérképek lokális maximumhelyeinek a megtalálására (Arcelli, Sanniti di Baja, 1988). Egy pontot akkor tekintenek lokális maximumhelynek, ha a távolságtérkép számítása során (a fent vázolt pásztázási menetekben) onnan nem terjed tovább információ. Más szavakkal: a lokális maximumhelyek súlyokkal növelt értékei nem kisebbek a minimumképzésben szereplő többi elemnél.

A távolság-transzformáción alapuló eljárások nem garantálnak topológiailag korrekt vázat, ha a távolságtérképen “közel lapos” régiók találhatóak. Ugyanakkor a módszer fő erénye az, hogy hibamentes euklidészi távolságtérkép mellett geometriailag korrekt vázat eredményez.



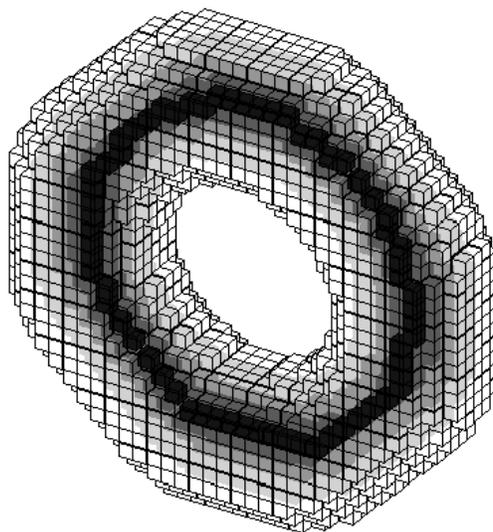
1.13. ábra 2D példa távolságtranszformációra a (3,4)-chamfer távolság szerint. Az eredeti (a) bináris kép egy 7×11 -es téglalap alakú objektumot tartalmaz. A távolságtérkép a határpontok kivonása és az inicializálás után (b). A tömb a pásztázás előre fázis végrehajtása után (c). A kiszámított távolságtérkép (a pásztázás vissza fázis után) (d), ahol a “hegygerinceket”, vagyis a vázhoz tartozó pozíciókat vastagított (félkövér) számokkal jelöltük.



1.14. ábra Példa 2D távolságtérképre. Az (a) kiindulási bináris képre a 4-szomszédság szerinti távolságtárszformációval a (b) távolságtérképet kaptuk. A távolságtérképnek a kiindulási objektumba eső részét egy 3D bináris objektumként ábrázoltuk az *emelés (lifting)* művelet (Udupa, 1993) segítségével. Az emelés az egészeket tartalmazó n D X tömbből az $(n + 1)$ D bináris Y tömböt állítja elő, ahol $Y[i_1, \dots, i_n, i_{n+1}] = 1$, ha $i_{n+1} \leq X[i_1, \dots, i_n]$, különben $Y[i_1, \dots, i_n, i_{n+1}] = 0$. (Az M.C. Escher tervezte tesztobjektum különlegessége az, hogy az alakzat és annak két elforgatottja példányaival “leparkettázható” a sík.)

1.3.2. A front-terjedés modellezése, a vékonyítás

Diszkrét objektumok vázának közelítésére gyakran használják a front-terjedést modellező iteratív eljárást, a vékonyítást. A front-terjedés a diszkrét térben diszkrét folyamat: egy időegység alatt a tűz frontja a tőle “egységnyi” távolságra lévő rácspontokat éri el. A vékonyítás egy iterációs lépése tehát a front-terjedés egy időegységének felel meg. Az eljárás akkor ér véget, ha már nincs “éghető anyag” az objektumon. A vékonyítás menetét az 1.15. ábrán mutatjuk be.



1.15. ábra Egy 3D (26, 6) kép tórusz-szerű objektumának vékonyítása. Az egymást követő iterációs lépések eredményeit egyre sötétebb árnyalatokkal jelentettük meg. A legsötétebb pontok adják a váz közelítését. (Egységkockás reprezentációt használtunk.)

A vékonyítás természeténél fogva párhuzamos. A prérítűz analógia szerint a határ valamennyi pontja egyidejűleg kap lángra, továbbá a tűzfrontok minden irányba egyenletes sebességgel terjednek. Megjegyezzük, hogy javasoltak szekvenciális vékonyító algoritmusokat is, melyek a kontúrkijelölés és a kontúrkövetés technikáját alkalmazva imitálják a külső réteg pontjainak egyidejű eltávolítását (Arcelli, 1981; Naccache, Shinghal, 1984; Kwok, 1988). A front-terjedés nem csak párhuzamos, hanem lokális is. Ezen második fontos tulajdonság szerint az, hogy egy objektumpont a “tűz áldozata” lesz-e az eljárás következő fázisában vagy sem, eldönthető a kérdéses pont egy — az objektum nagyságától független méretű — szűk környezete alapján.

A vékonyító algoritmusok egy iterációs lépése tehát egy olyan lokális és párhuzamos redukcióval adható meg, ami csak határpontokat töröl. A teljes eljárás ilyen képműveletek sorozatából áll. A vékonyítás menete az alábbi, teljesen általános sémával írható le:

```
repeat
  a “törölhető” határpontok eltávolítása
until nem történt változás
```

A `repeat`-ciklus magja a vékonyítás egy iterációs lépésének felel meg. Az egyes vékonyító eljárások az iterációs lépések megszervezésében és a “törölhető” pontokra adott feltételekben különböznek egymástól.

A vékonyítás a vázzal szemben támasztott topológiai kritérium teljesítésére helyezi a hangsúlyt, vagyis topológia-megőrző redukciós képműveletnek kell lennie. A másik, a geometriai kritériumnak viszont nem tud maradéktalanul eleget tenni, mivel az adott digitális térben (\mathbb{Z}^2 -ben vagy \mathbb{Z}^3 -ben) alkalmazott szomszédsági relációk által indukált “távolságok” nem közelítik “jól” az euklidészi távolságot. A vékonyításon és a távolság-transzformáción alapuló vázkijelölés egymást kiegészítő módszerek: ami az egyiknél erény, az a másiknál hiányosság (és viszont). Topológiailag és geometriailag egyaránt korrekt váz meghatározása lehetséges a fenti két módszer együttes alkalmazásával, ahol a távolság-transzformációval meghatározott vázat “horgonyként” kezeli az azt követő vékonyító eljárás (Arcelli, Cordella, Levialdi, 1981; Arcelli, Sanniti di Baja, 1992; Talbot, Vincent, 1992; Saito, Toriwaki, 1995; Ge, Stelts, Vining, 1996; Pudney, 1998; Svensson, Nyström, Borgefors, 1999). A kombinált módszernél a távolság-transzformáció ad egy, a geometriai kritériumnak eleget tevő, de nem összefüggő “félkész” vázat, amit a vékonyítás megőriz és kiegészít úgy, hogy az a topológiai feltételt is teljesítse. Ezt a megközelítést más szerzők — minden erénye ellenére — költségesnek és túlságosan heurisztikusnak minősítik (Fernández-Vidal, Malandain, 1995; Székely, 1996).

Sokféle topológia-megőrző lokális és párhuzamos redukció létezik, többségük nyilvánvalóan nem mutat hasonlóságot a front-terjedéssel. Vékonyító algoritmusok tervezésekor tehát nem csak arra kell ügyelni, hogy úgy töröljünk, hogy az eredetivel topológiailag ekvivalens képet kapjunk. Arra is figyelni kell, hogy mikor ne töröljünk, vagyis azok a pontok, ahol a tűzfrontok találkoznak, maradjanak meg akkor is, ha elhagyásuk megőrizné a topológiát. Ezeket a védett pontokat *végpontoknak* nevezzük.

A 3D vékonyító algoritmusokat az általuk kivont közelítő váz jellege szerint két osztályba soroljuk: vannak, amelyek a *középvonalra* vékonyítanak (*medial line thinning*), mások pedig *középfelületre* (*medial surface thinning*). Amennyiben nem alkalmazunk végpont-kritériumot, úgy a vékonyítás az objektum *topológiai magját* (*topological kernel*) határozza meg. Ez esetben a vékonyítás hatása ugyanaz, mint a csak redukcióval élő *zsugorításnak* (*shrinking*), amiről a 3. fejezetben lesz szó. Az 1.16. ábra mindhárom esetre példát mutat. Megjegyzendő, hogy az “igazi” 3D vázhoz a középfelület áll közelebb. A középvonalra vékonyítás elnyomja a tükrözéses szimmetriákat a tengelyesek javára, ami hasznos bizonyos objektumok (pl: orvosi képeken a vérerek, a légutak és bizonyos csontok) jellemzésekor.

A vonalvégpontok meghatározásában általános az egyetértés: a $\mathcal{P} = (V, m, n, B)$ képen a $p \in B$ pont akkor és csakis akkor *vonálvégpont*, ha az $(N_m(p) \cap B) \setminus \{p\}$ halmaz egyelemű, vagyis az adott pont csak egyetlen másik fekete ponttal m -szomszédos. Egyúttal megadjuk a vonalpontok és az elágazási pontok definícióját is: $p \in B$ akkor és csakis akkor *vonálpont*, ha p pontosan kettő (p -től különböző) fekete ponttal m -szomszédos; és akkor *elágazási pont*, ha több mint kettő fekete ponttal m -szomszédos.

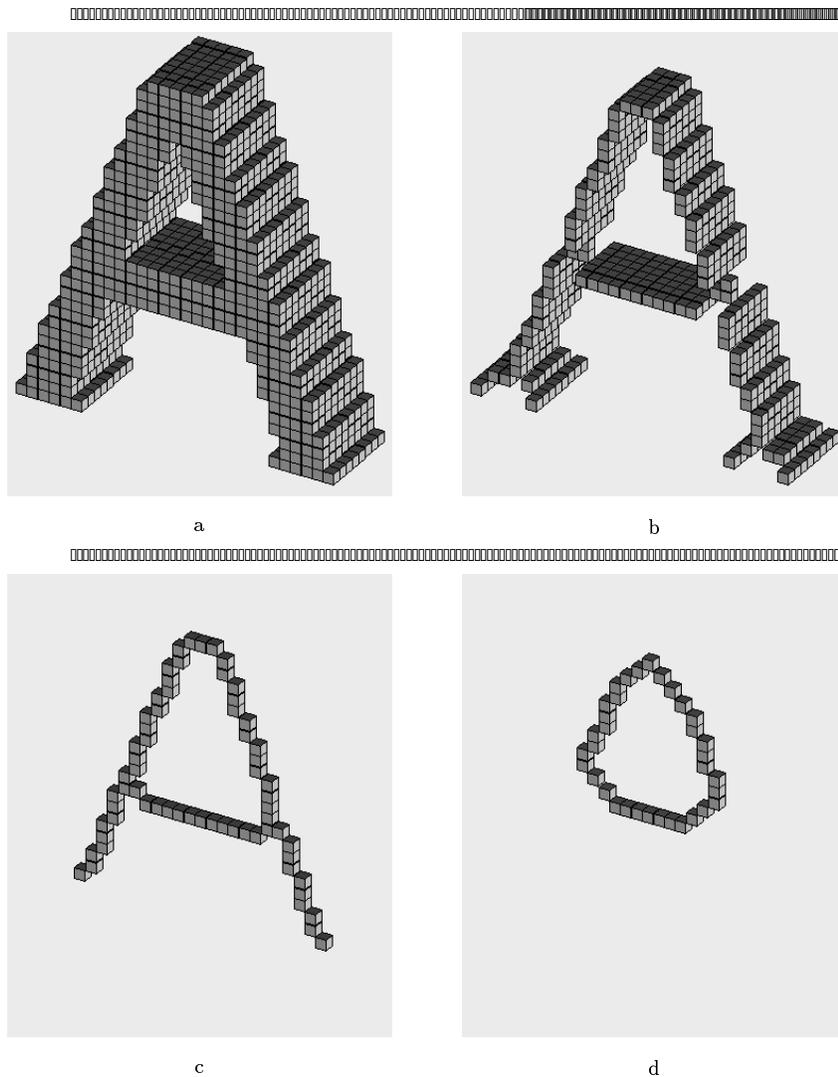
A *felszínvégpontok* esetében nincs konvenció. A különböző középfelületre vékonyító algoritmusok eltérő végpont-kritériumokkal dolgoznak, amelyekre a 2. fejezetben — az egyes eljárások ismertetésekor — térünk ki.

Az objektumok iteratív “hámozására” számos eljárást javasoltak, így szükségessé vált olyan feltétel megfogalmazása, aminek teljesítése esetén kijár a “vékonyító algoritmus” minősítés. Rosenfeld állított fel 2D esetén ilyen kritériumot (Rosenfeld, 1975), melyet Tsao és Fu adaptált a 3D középvonalra vékonyító algoritmusokra (Tsao, Fu, 1981):

1.8. KRITÉRIUM (Rosenfeld, 1975; Tsao, Fu, 1981)

Egy redukciós képművelet vékonyítás, ha eleget tesz az alábbi feltételeknek:

1. Az objektumok és azok komplementere összefüggőségén nem változtat, vagyis nem vág szét objektumot kettő vagy több részre (és ugyanez áll a komplementerükre is).
2. Izolált pontokat és görbéket változatlanul hagy.
3. Megváltoztatja a 2×2 -es négyzet-alakú (2D), illetve a $2 \times 2 \times 2$ -es kocka-alakú (3D) objektumot.



1.16. ábra Példa különböző végpont-kritériumokkal történő vékonyításra. Az eredeti (26, 6) kép objektuma egységkockás reprezentáció szerint (a), annak középfelület (b), középvonalai (c) és topológikus magja (d).

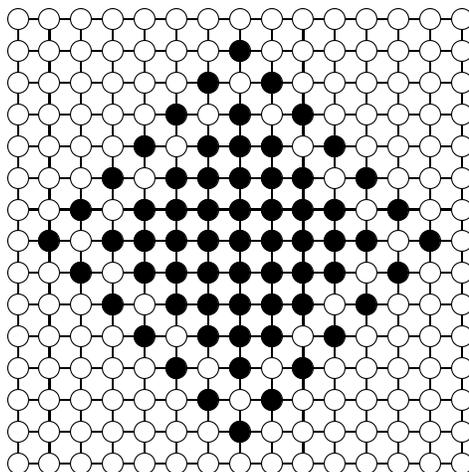
Az 1.8. Kritérium 1. feltétele 2D-ben egyenértékű a topológia-megőrzéssel, 3D-ben viszont nem zárja ki lyuk létrehozását. Ennélfogva, célszerűbb a topológia-megőrzést megkövetelni.

A 2. feltétel arra állít fel korlátot, hogy mikor nem szabad folytatni tovább a törést. A feltétel szerint az izolált pontok és a görbék már “vékonyak”. A *görbe* az egy “*egy pixel/voxel vastagságú*” objektum, ami vonalpontokat, vonalvégpontokat és elágazási pontokat tartalmaz. Vonalpontok törlése nyilvánvalóan az 1. feltétel megsértésével jár, egy izolált pont eltávolítása pedig egy objektum teljes törlését eredményezi⁷. Az 1. fel-

⁷Az izolált pontokra vonatkozó kitétel — valószínűleg — a Deutsch iránti tisztelet jeléül került be, akinek a klasszikus algoritmus (Deutsch, 1968) törli az izolált pontokat is.

tételhez képest csak a vonalvégpontok törlésének tilalma jelent újdonságot. (A kritérium fogyatékosága, hogy csak a középvonalak meghatározására vonatkozik, a középfelszínre vékonyításra nem.)

A 3. feltétel — a két korábbival ellentétben — nem a törlést korlátozza, hanem a minél “vékonyabb” objektumok elérését próbálja elősegíteni. Hangsúlyozandó, hogy ez a heurisztikus feltétel nem a kis négyzet-, illetve kocka-alakú objektumrészletek előfordulását tiltja meg, hanem magábanálló objektumról szól. Ez annál is inkább indokolt, mert megadhatók nem vékonyítható kiterjedt objektumok is⁸. Az 1.17. ábrán látható egy olyan objektum (Suen, Wang, 1994), amin — $(8, 4)$ 2D képet feltételezve — nem található egyszerű pont. Hasonló példaobjektumokat — tetszőleges méretben — konstruálhatunk 3D-ben is.



1.17. ábra Példa kiterjedt, de nem-vékonyítható objektumra. A $(8, 4)$ kép egyetlen fekete pontja sem egyszerű. Bármelyik fekete pont törlése vadonatúj üreget hozna létre, (egy vagy kettő) üreget szüntetne meg vagy (kettő vagy három) üreget olvasztana össze.

⁸Megjegyzendő, hogy az egyetlen szomszédsági relációt alkalmazó jól-képzett halmazokra Latecki, Eckhardt és Rosenfeld definiálták az *egy pixel vastag halmaz* fogalmát és javasoltak egy olyan 2D vékonyító algoritmust, ami tetszőleges jól-képzett halmazból egy pixel vastag halmazt állít elő (Latecki, Eckhardt, Rosenfeld, 1995).

1.3.3. A Voronoi-váz

Topológiaiilag és geometriailag egyaránt korrekt váz meghatározására ígéretes a *Voronoi-diagramon* (Aurenhammer, 1985; Preparata, Shamos, 1990; Voss, 1993) alapuló módszer.

Először a 2D Voronoi-diagramot mutatjuk be. Legyen $G = \{g_1, \dots, g_t\}$ t (különböző) diszkrét pontot tartalmazó halmaz, ahol $g_i \in \mathbb{Z}^2$ ($i = 1, \dots, t$). A Voronoi-diagram 0-, 1- és 2-dimenziós elemekből áll, melyeket rendre *Voronoi-csúcsoknak*, *Voronoi-éleknek* és *Voronoi-régióknak* neveznek.

A pontosan t darab Voronoi-régió mindegyike a G generálómalmaz egy-egy eleméhez rendelhető:

$$R_i(G) = \{ p \mid \forall j : 1 \leq j \leq t, j \neq i \quad D(p, g_i) < D(p, g_j) \} \quad (i = 1, \dots, t),$$

ahol “ D ” az euklidészi távolságot jelöli.

Egy régióba azon pontok kerülnek, amelyek közelebb vannak a generálómalmaz egy adott eleméhez, mint az összes többihez. Könnyen belátható, hogy valamennyi $R_i(G)$ Voronoi-régió konvex (egy régió bármely kettő pontjára a kiválasztott pontokat összekötő szakasz egésze is a régióba esik), egyértelműen létezik és nem üres (mivel $g_i \in R_i(G)$). Bizonyos régiók korlátosak, míg mások területe végtelen is lehet.

A Voronoi-éleket pontosan kettő G -beli ponttól egyenlő távolságra és minden más generáloponttól távolabbra eső pontok alkotják:

$$E_{ij}(G) = \{ p \mid D(p, g_i) = D(p, g_j) \text{ és } \forall k : 1 \leq k \leq t, k \neq i, k \neq j \quad D(p, g_i) < D(p, g_k) \}$$

($1 \leq i \neq j \leq t$). Megjegyzendő, hogy bizonyos g_i és g_j pontokra E_{ij} üres is lehet, ugyanakkor bizonyos Voronoi-élek lehetnek végtelen hosszúságúak is. Belátható, hogy E_{ij} része a g_i és a g_j pontokat összekötő szakasz szakaszfelező merőleges egyenesének.

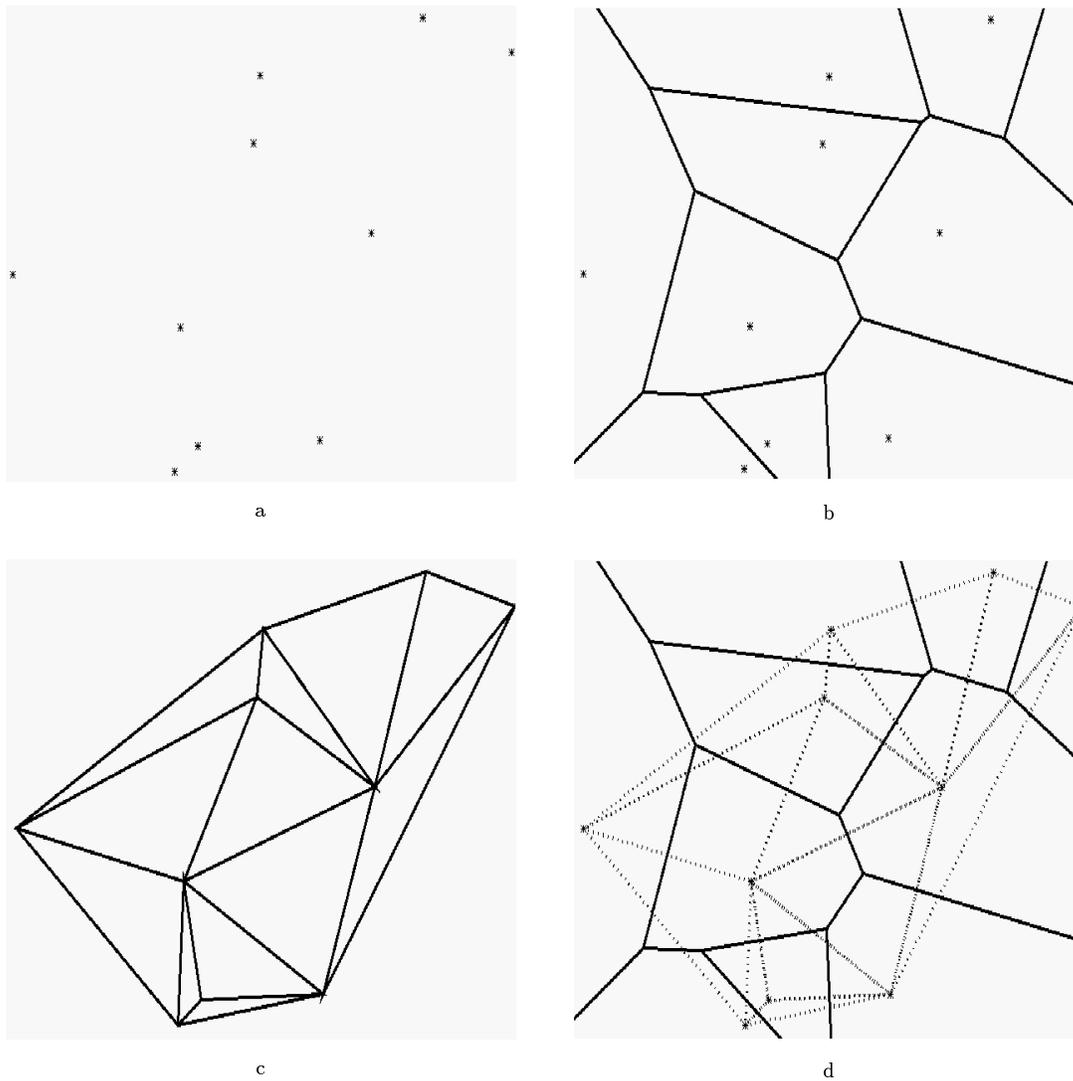
Egy Voronoi-csúcs (legalább) három G -beli ponttól egyenlő távolságra és a generálómalmaz minden más pontjától távolabbra esik:

$$V_{ijk}(G) = \{ p \mid D(p, g_i) = D(p, g_j) = D(p, g_k) \text{ és } \\ \forall l : 1 \leq l \leq t, l \neq i, l \neq j, l \neq k \quad D(p, g_i) \leq D(p, g_l) \}$$

($1 \leq i \neq j \neq k \leq t$). Megjegyezzük, hogy egyrészt bizonyos g_i , g_j és g_k pontokra nem létezik a $V_{ijk}(G)$ csúcs, másrészt fennállhat az is, hogy $V_{ijk} = V_{ijl}$ ($k \neq l$), mivel egy csúcstól háromnál több G -beli pont is eshet azonos (és minimális) távolságra.

Voronoi-diagramra az 1.18b. ábra ad példát, ahol a diagramot az 1.18a. ábrán látható (véletlenszerűen kiválasztott) 10-elemű generálómalmazra határoztuk meg.

A G generálómalazzal adott Voronoi-diagrammal szoros kapcsolatban áll G *Delaunay-felbontása* (Preparata, Shamos, 1990; Voss, 1993), mely G konvex burkát (*convex hull*) osztja fel háromszögekre (1.18c. ábra). Egy 2D diszkrét ponthalmaz konvex burka az a legkisebb területű konvex poligon, ami tartalmazza a ponthalmaz valamennyi pontját. (Megjegyezzük, hogy a konvex burok egyértelműen létezik tetszőleges ponthalmazra.) A 2D konvex burok meghatározására számos algoritmus létezik. A probléma bonyolultsága t pontból álló generálómalmaz esetén $\mathcal{O}(t \cdot \log_2 t)$, de az aszimptotikusan leggyorsabb eljárás időigénye $\mathcal{O}(t \cdot \log_2 \bar{t})$, ha \bar{t} csúcspont feszíti ki a konvex burkot (Graham, 1972; Jarvis, 1973).



1.18. ábra Példa Voronoi–diagramra és Delaunay–felbontásra. Egy 10–elemű ponthalmaz (a), annak Voronoi–diagramja (b), Delaunay–felbontása (c) és azok együtt es megjelenítése (d). A teljes 2D síkra kiterjedő Voronoi–diagramnak csak egy korlátos téglalapba eső része látható.

A Delaunay–felbontás a konvex burok egy speciális triangularizációja: a háromszögek köré írt körlapoknak nem belső pontja a generálópontja egyetlen pontja sem. Megjegyezzük, hogy a Delaunay–triangularizáció nem egyértelmű minden esetben, mivel egy (lehetőséges) felbontásban ugyanaz lehet a köréírt köre egyszerre több háromszögnek is. Az ilyen háromszögek olyan konvex poligont alkotnak, amelynek tetszőleges triangularizációjára teljesül a Delaunay–felbontás kritériuma. Az 1.19c. ábra példáján a Delaunay–felbontásban négyszögek is előfordulnak, melyeknek bármelyik átlója szerepelhet a triangularizációban.

A Delaunay–felbontásban is 0–, 1– és 2–dimenziós elemek szerepelnek, amelyeket rendre *Delaunay–csúcsoknak*, *Delaunay–éleknek* és *Delaunay–régióknak* neveznek. A Delaunay–csúcsok halmaza maga a G generálópontja, a Delaunay–éleket a felbontás háromszögeinek oldalai alkotják, a Delaunay–régiók pedig a háromszögek.

Az 1.18c. ábrán a Delaunay–felbontásra is szerepel példa. Az adott 10–elemű ponthalmaz 6 pontja szerepel a konvex burok csúcspontjaként és a felbontás 12 háromszöget

tartalmaz.

A Voronoi–diagram és a Delaunay–felbontás egymásnak *duálisa*: az egyik d –dimenziós ($d = 0, 1, 2$) elemeinek egy-egyértelmű módon megfeleltethetők a másik $(2 - d)$ –dimenziós elemei. A $V_{ijk}(G)$ (0D) Voronoi–csúcsnak az a (2D) Delaunay–háromszög felel meg, amelynek csúcsai g_i , g_j és g_k (a háromszög köré írt kör középpontja éppen $V_{ijk}(G)$). Az $E_{ij}(G)$ (1D) Voronoi–élhez a g_i és a g_j pontokat összekötő (1D) Delaunay–él tartozik. A $R_i(G)$ (2D) Voronoi–régiónak pedig a g_i (0D) Delaunay–csúcs felel meg. A két modell elemeinek kapcsolatát a 1.18d. ábra illusztrálja.

A 3D ponthalmazokra is megadható a Voronoi–diagram és a Delaunay–felbontás. Ekkor a Voronoi–régiók 3D konvex poliéderek lesznek, melyek lapjai, élei és csúcsai adják rendre a diagram 2D, 1D és 0D elemeit. A Delaunay–felbontásban a 3D elemek tetraéderek, melyek (2D) lapjai, (1D) élei és (0D) csúcsai alkotják a felbontás 2D, 1D és 0D elemeit. A dualitás ezúttal is fenáll: az egyik konstrukció d –dimenziós ($d = 0, 1, 2, 3$) elemeihez a másik $(3 - d)$ –dimenziós elemei tartoznak.

A 2D Voronoi–diagram meghatározására és a Delaunay–felbontásra számos módszert dolgoztak ki, melyek komplexitása t pontra 2D–ben $\mathcal{O}(t \cdot \log_2 t)$ és 3D–ben $\mathcal{O}(t^2)$ (Näf, 1996; Attali, Montanvert, 1997).

A Voronoi–diagram kapcsolatban áll a vázzal: ha a generálómalmaz pontjait egyenletes, minden határon túl finomodó mintavételezés szerint vesszük az objektum határáról, akkor a Voronoi–diagramnak az objektumba eső elemei a vázhoz konvergálnak (Boissonat, Kofakis, 1985; Schmitt, 1989; Brandt, Algazi, 1992).

Az 1.19b. ábra a Voronoi–vázra mutat példát. A példaobjektum (1.19a. ábra) — ezúttal is — egy szabályos téglalap, így a Voronoi–diagram összevethető az 1.1. ábrán látható folytonos vázzal.

A Voronoi–diagram (éppúgy, mint a Delaunay–felbontás) folytonos modell. Diszkrét objektumok esetén a diszkrét generálómalmaz kijelölése után modellt kell váltani: a folytonos Voronoi–diagramból kell kiválasztani azokat a folytonos elemeket, amelyek a vázhoz tartoznak. Ezért a Voronoi–diagramon alapuló vázkijelölést *félíg folytonos* (*semi-continuous*) megközelítésnek nevezik (Székely, 1996).

A Voronoi–váz meghatározásakor két kérdésre kell választ találni: mely pontokból álljon a generálómalmaz, és a Voronoi–diagram mely elemei tekintendők relevánsnak, a vázmodellhez tartozónak.

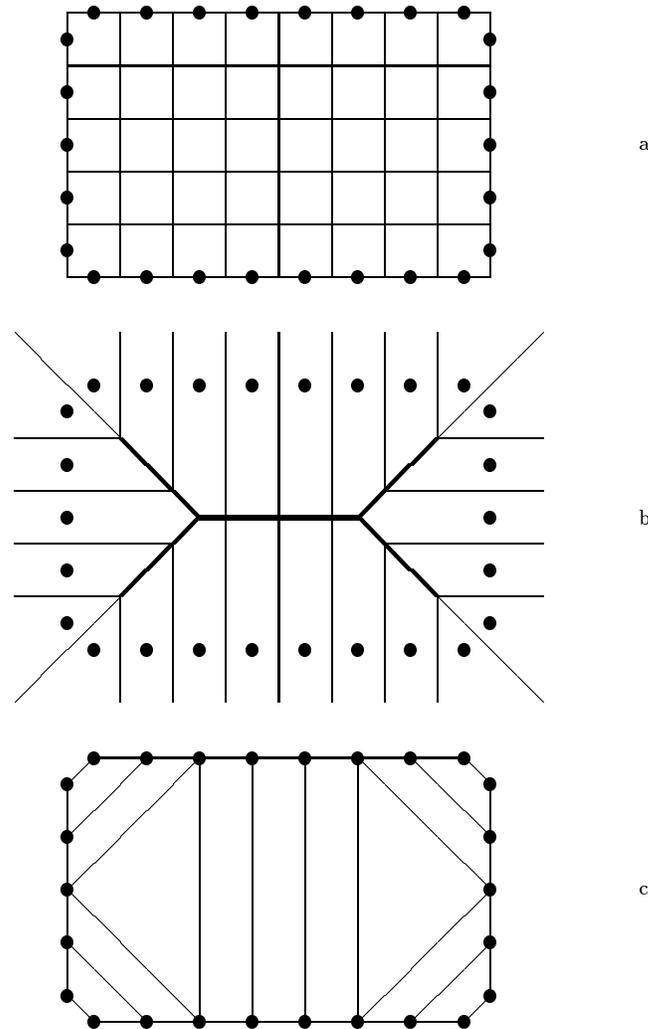
Az első kérdésre egy lehetséges válasz: egy 2D képen alkossák a generálómalmazt a fekete és fehér pixelek (egységnégyzetek) közös oldalainak felezőpontjai, 3D–ben pedig a fekete és fehér egységkockák közös lapjainak középpontjai (Székely, 1996; Näf, 1996; Näf et al., 1997).

A második probléma megválaszolására sokféle szubjektív és heurisztikus módszert tesz lehetővé az a tény, hogy a Voronoi–diagram (vagy a Delaunay–felbontás) összetett adatstruktúrákban tárolható. Ennélfogva a modellek mint folytonos 0D–, 1D–, \dots , d D–elemek halmazai különféle *fontossági mértékek* (*importance measure*) (Brandt, Algazi, 1992; Brandt, 1992; Ogniewicz, Ilg, 1992; Attali, Montanvert, 1997; Shaked, Bruckstein, 1998) alapján egyszerűsíthetők, tisztíthatók.

A Voronoi–diagramból geometriailag és topológiailag egyaránt korrekt váz vonható ki, aminek utófeldolgozására, szimbólikus leírására számos lehetőség adódik (Székely, 1996; Näf, 1996; Näf et al., 1997; Shaked, Bruckstein, 1998). A módszer hátránya az, hogy a folytonos modell feldolgozása nagymennyiségű, lebegőpontos aritmetikát igénylő számítást igényel. Összetett objektumokra a generálómalmaz pontjainak száma nagyon nagy lehet, pl. egy 3D mágneses rezonancia (MR) vizsgálatból szegmentált agy esetén

megközelítőleg félmillió pontból áll. Erre a generálómalmazra a Voronoi–diagram kiszámítása egy teljes napot igényel egy nagyteljesítményű (Sun Ultra1 Sparc 167 MHz) munkaállomáson (Näf et al., 1997).

Megjegyezzük, hogy nem csak a határpontok halmazával leírt 3D objektumok, hanem poliéderekkel közelített 3D testek Voronoi diagramjának vagy Delaunay–felbontásának meghatározására is léteznek módszerek (Reddy, Turkiyyah, 1995; Sheehy, Armstrong, Robinson, 1996; Sherbrooke, Patrikalakis, Brisson, 1996).



1.19. ábra Példa 2D Voronoi–vázra. Az eredeti objektum (a), annak Voronoi–diagramja (b) és Delaunay–felbontása (c). Az eredeti objektum egy szabályos 5×8 -as téglalap, ahol minden egyes objektumpontnak (“1” pixel) egy–egy egységnégyzet felel meg, míg a háttérpontokat (“0” pixelek) nem jelöltük. Az “1” és a “0” pixeleken osztozó egységnégyzet–élek felezőpontjai alkotják a generálómalmazt, mely a “•” szimbólumokkal jelölt pontokból áll. A téglalap Voronoi–diagramján a folytonos vázhoz tartozó Voronoi–éleket vastag vonalak jelölik. Könnyen belátható, hogy a mintavételezés finomításával a vázhoz tartozó élek egyre közelebb kerülnek a téglalap sarkaihoz, vagyis egyre pontosabb közelítést adják a váznak. A téglalap Delaunay–felbontásában a négyyszögeknek olyan Voronoi–csúcsok felelnek meg, amelyekbe négy él fut be.

1.4. A váz meghatározása és a matematikai morfológia

A *morfológia* (*morphology*) a biológiának az élőlények alakjával foglalkozó ága, míg a Matheron és Serra által megalapozott *matematikai morfológia* (*mathematical morphology*) a képanalízishez biztosít hasznos eszközöket (Matheron, 1975; Serra, 1982). A digitális képfeldolgozás számos problémája — beleértve a vékonyítást és a vázkijelölést — leírható morfológiai megközelítéssel (Schalkoff, 1989; Gonzales, Woods, 1992; Jähne, 1993; Serra, Soille, 1994; Glasbey, Horgan, 1995; Awcock, Thomas, 1996; Maragos, Schafer, Butt, 1996; Watt, Policarpo, 1998; Sonka, Hlaváč, Boyle, 1999).

A matematikai morfológia műveleteit halmazműveletekkel írják le, ahol halmazként elemi minták is szerepelnek. Megjegyzendő, hogy a morfológiai műveleteket kiterjesztették *szürkeárnyalatos* (*grey-scale*) képekre is (Gonzales, Woods, 1992; Baxes, 1994; Glasbey, Horgan, 1995; Watt, Policarpo, 1998).

A morfológiai vékonyítás (1.4.2.) és a morfológiai váz (1.4.3.) ismertetéséhez szükséges alpműveleteket az 1.4.1. pontban mutatjuk be. A morfológiai műveletek és a *maszkillesztés* (*template matching*) kapcsolatát az 1.4.4. pontban tárgyaljuk.

A morfológiai műveletek jelölése a szakirodalomban nem egységes. Dolgozatunkban a Gonzales és Woods által használt szimbólumok (Gonzales, Woods, 1992) szerepelnek.

1.4.1. Alpműveletek

A matematikai morfológia műveletei a d -dimenziós ($d \geq 1$) euklidészi tér tetszőleges részhalmazain értelmezettek. Számunkra csak a bináris képműveletek leírásában betöltött szerepük a fontos, így halmazon a továbbiakban a képpontok egy részhalmazát értjük. A jelen pontban azokat az alap- és segédműveleteket ismertetjük, amelyek a morfológiai váz meghatározásához és a vékonyítás megadásához szükségesek.

Legyenek X és Y a képpontok V halmazának részhalmazai. (2D kép esetén $V = \mathbb{Z}^2$, 3D-ben pedig $V = \mathbb{Z}^3$).

Az X ponthalmaz *komplementerét* (*complement*) jelölje X^c , ahol:

$$\begin{aligned} X^c &= \{ p \mid p \notin X \} \\ &= V \setminus X . \end{aligned}$$

Az X ponthalmaz a -val való *eltolása* (*translation*) a következőképpen definiált:

$$(X)_a = \{ x + a \mid x \in X \},$$

ahol $a \in V$ és “+” a komponensenkénti/koordinátánkénti összeadást jelöli.

Az Y ponthalmaznak (az origóra való) *tükrözését* (*reflection*) az alábbi formula adja meg:

$$\hat{Y} = \{ -y \mid y \in Y \},$$

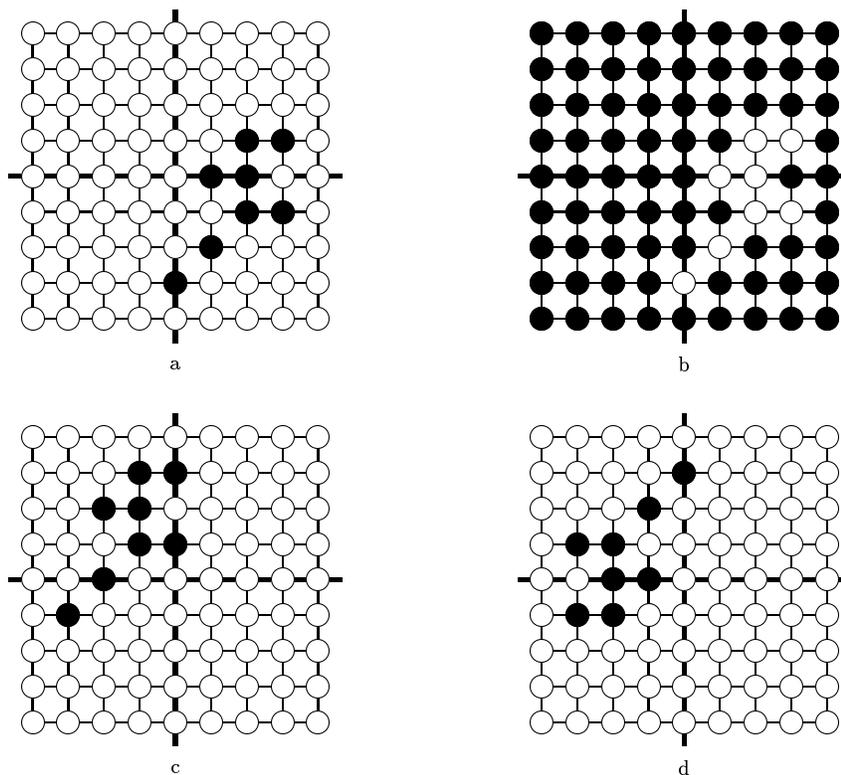
ahol a “-” a komponensenkénti “-1”-gyel való szorzást jelöli. Az Y ponthalmaz *szimmetrikus*, ha $Y = \hat{Y}$.

A fenti három segédműveletre az 1.20. ábra ad példákat.

Az X ponthalmaznak az Y -nal való *dilatációját* (*dilation*) a következő formula írja le:

$$\begin{aligned} X \oplus Y &= \{ a \mid (\hat{Y})_a \cap X \neq \emptyset \} \\ &= \{ a \mid [(\hat{Y})_a \cap X] \subseteq X \}, \end{aligned}$$

ahol az Y ponthalmazt *szerkesztőelemnek* (*structuring element*) nevezik. A szerkesztőelem hasonló szerepet játszik, mint a *konvolúciós maszk* (*convolution mask/kernel*) (Gonzales, Woods, 1992; Sonka, Hlaváč, Boyle, 1999): a szerkesztőelem koordinátarendszerének origóját eltoljuk V minden elemére és az adott elemet felvesszük a dilatált halmazba, ha a szerkesztőelemnek legalább egy pontja X -beli ponttal kerül fedésbe. A gyakorlatban Y -t lefedi az origó köré írt “kisméretű” (pl. 3- vagy 5-sugarú) körlap (2D-ben), illetve gömb (3D-ben). Könnyen belátható, hogy ha Y tartalmazza az origót, akkor tetszőleges Y -ra: $X \subseteq X \oplus Y$, vagyis a dilatációnak “hízlaló” hatása van.



1.20. ábra Példák a morfológiai segédműveletekre. Ax $X \subseteq \mathbb{Z}^2$ halmaz (a), ahol X elemeit “•”, a $\mathbb{Z}^2 \setminus X$ -be esőket pedig “o” szimbólumok jelölik. A vastag vonalak az origóban (a $(0, 0)$ pontban) metszik egymást és az ábrázolt 9×9 -es részleten kívül nincs X -beli pont. A három segédművelet eredménye: X^c (b), $(X)_a$ $a = (-3, 2)$ esetén (c) és \hat{X} (d).

A dilatáció duálisát az *eróziót* (*erosion*) az alábbi formula adja meg:

$$\begin{aligned} X \ominus Y &= \{ a \mid (Y)_a \subseteq X \} \\ &= (X^c \oplus \hat{Y})^c, \end{aligned}$$

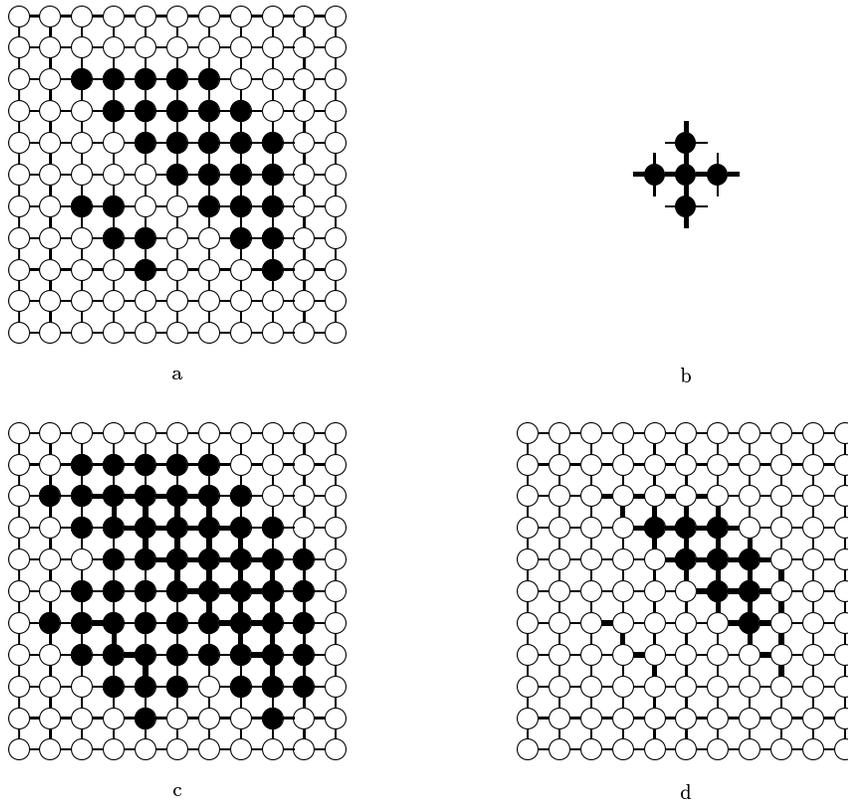
vagyis a V halmaz egy pontja akkor kerül be az erodált halmazba, ha a szerkesztőelem koordinátarendszerének origóját az adott pontba eltolva minden egyes Y -beli pont X -belivel kerül fedésbe. Könnyen belátható, hogy ha Y tartalmazza az origót, akkor tetszőleges Y -ra: $X \ominus Y \subseteq X$, vagyis az erózió “fogyaszt”.

A fenti alpműveleteket az 1.21. ábra illusztrálja. A 2D példákból az is kitűnik, hogy a dilatáció és az erózió mint bináris képműveletek nem őrzik meg a topológiát. A dilatáció üregek feltöltését és objektumok összeolvadását eredményezheti (3D-ben még lyukakat is megszüntethet), míg az erózió szétszakíthat vagy teljesen törölhet objektumokat, valamint összeolvaszthat üregeket egymással vagy a háttérrel.

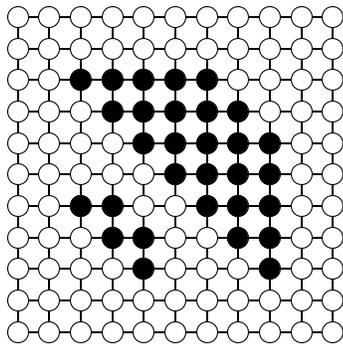
A morfológia következő alpműveletében az Y szerkesztőelemet két diszjunkt halmazzra bontjuk: $Y = Y_1 \cup Y_2$, ahol $Y_1 \cap Y_2 = \emptyset$. Az X ponthalmaznak az Y -nal való *hit-or-miss transzformációját* (*hit-or-miss transform*) az alábbi formula adja meg:

$$\begin{aligned} X \circledast Y &= (X \ominus Y_1) \cap (X^c \ominus Y_2) \\ &= (X \ominus Y_1) \setminus (X \oplus Y_2), \end{aligned}$$

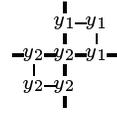
vagyis a hit-or-miss transzformált halmaznak olyan pont lesz eleme, amellyel a szerkesztőelem origóját eltolva az Y_1 által lefedett valamennyi pont X -be esik, ugyanakkor az Y_2 által lefedett pontok között nincs X -beli. A hit-or-miss transzformációra az 1.22. ábrával mutatunk példát.



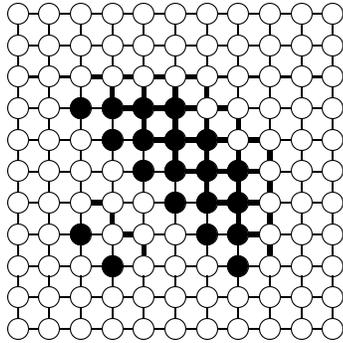
1.21. ábra Példák dilatációra és erózióra. Az $X \subseteq \mathbb{Z}^2$ halmaz vagy bináris kép (a), melynek elemeit illetve fekete pontjait “●” szimbólumok jelölik. Az Y (szimmetrikus) szerkesztőelem (b) elemeit “●”-ek jelölik és a vastag vonalak az origóban (a $(0, 0)$ koordinátájú pontban) metszik egymást. Az $X \oplus Y$ dilatáció (c) X hízlalását, az $X \ominus Y$ erózió (d) pedig X fogvását eredményezi. A 4-szomszédságnak megfelelő szerkesztőelem mellett a dilatáció feketévé változtatja mindazon pontokat, amelyeknek van fekete 4-szomszédja, az erózió pedig csak azokat a fekete pontokat hagyja meg, amelyeknek az összes 4-szomszédja fekete volt. (A (c) és a (d) képeken azok a vonaldarabok vastagok, amelyek 4-szomszédos X -beli pontokat kötnek össze.)



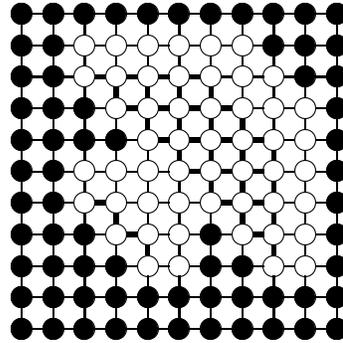
a



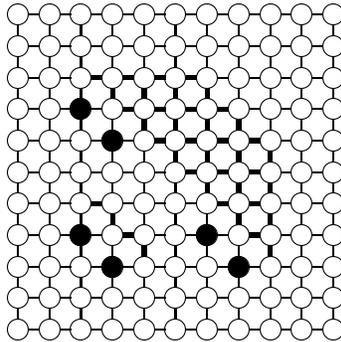
b



c



d



e

1.22. ábra Példa hit-or-miss transzformációra. Ax $X \subseteq \mathbb{Z}^2$ halmaz vagy bináris kép (a), melynek elemeit illetve fekete pontjait “•” szimbólumok jelölik. Az $Y = Y_1 \cup Y_2$ szerkesztőelem (b), ahol az Y_1 halmaz elemeit “ y_1 ”-ek, az Y_2 -belieket pedig “ y_2 ”-k jelölik és a vastag vonalak az origóban metszik egymást. A (c) képen látható $X \ominus Y_1$ eredménye, míg (d) mutatja az $X^c \ominus Y_2$ képet. A végeredmény, $X \circledast Y = (X \ominus Y_1) \cap (X^c \ominus Y_2)$ (e)-n látható. (A (c)-(e) képeken azok a vonaldarabok vastagok, amelyek 4-szomszédos X -beli pontokat kötnek össze.)

A morfológiai vázhoz szükségünk van a *nyitás* (*opening*) műveletre is. Az X pontthalmaznak az Y szerkesztőelemmel való nyitását a következő formula definiálja:

$$\begin{aligned} X \circ Y &= \bigcup \{ (Y)_a \mid (Y)_a \subset X \} \\ &= (X \ominus Y) \oplus Y, \end{aligned}$$

azaz a nyitás egy erózió és egy dilatáció egymás utáni végrehajtását jelenti.

A nyitás duálisa a *zárás* (*closing*), ahol a dilatációt követi az erózió:

$$\begin{aligned} X \bullet Y &= (X \oplus Y) \ominus Y \\ &= (X^c \circ \hat{Y})^c. \end{aligned}$$

A nyitás és a zárás a következő tulajdonságokkal bír:

1. $X \circ Y \subseteq X$ és $X \subseteq X \bullet Y$.
2. Ha $X_1 \subset X_2$, akkor $X_1 \circ Y \subseteq X_2 \circ Y$ és $X_1 \bullet Y \subseteq X_2 \bullet Y$.
3. $(X \circ Y) \circ Y = X \circ Y$ és $(X \bullet Y) \bullet Y = X \bullet Y$.

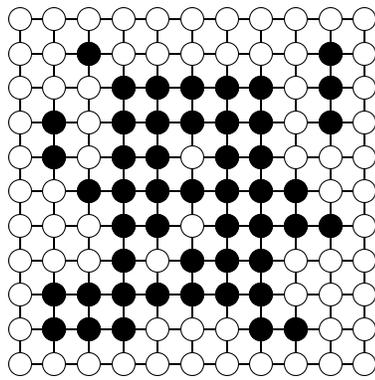
Az 1. tulajdonság szerint nyitás “fogyaszt”, a zárás pedig “hízlal”. A monotonitásról szól a 2. tulajdonság, míg a 3. tulajdonság azt mondja ki, hogy a nyitás és a zárás ismétlése hatástalan, azaz a műveletek idempotensek.

A fenti tulajdonságaik miatt a nyitás és a zárás szűrésre is használható. A *morfológiai szűrést* (*morphological filtering*) az

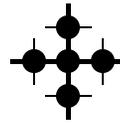
$$(X \circ Y) \bullet Y = (((X \ominus Y) \oplus Y) \oplus Y) \ominus Y$$

formula írja le, hatását az 1.23. ábrával szemléltetjük. Az ábra egyben a nyitásra és a zárássra is példát ad.

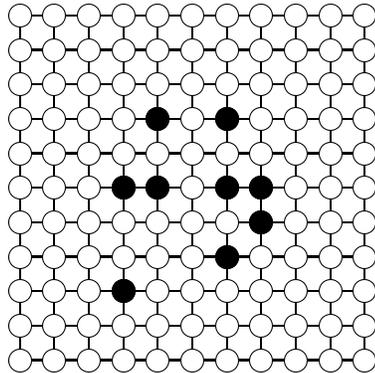
A morfológiai szűrő egy erózióval indul, ami a bináris képekről törli a “kisméretű külső” zajokat. A kezdeti erózió mellékhatásaként a “kisméretű” (zajnak minősülő) üregek megnőnek és a megőrzendő objektumok “fogynak”. Az eróziót egy dilatáció követi, ami visszaállítja az objektumok és az üregek eredeti méretét. (Mivel az erózió teljesen törölte a “kisméretű külső” zajokat, így azok végképp eltűntek.) A szűrésnek a nyitás fázisa ezzel befejeződött. A zárást indító újabb dilatáció kitölti a “kisméretű” (zajnak minősülő) üregeket. A túlhízlalt objektumok helyreállítása a zárást (és egyben a szűrést) befejező erózió feladata. A morfológiai szűrő hatását az alkalmazott Y szerkesztőelem megválasztása befolyásolja. A szerkesztőelem mérete (és alakja) határozza meg azt, hogy mely objektumok és üregek minősülnek “kisméretűnek”.



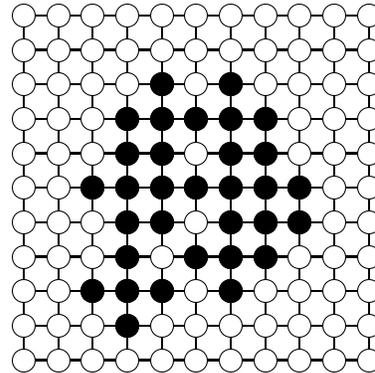
X



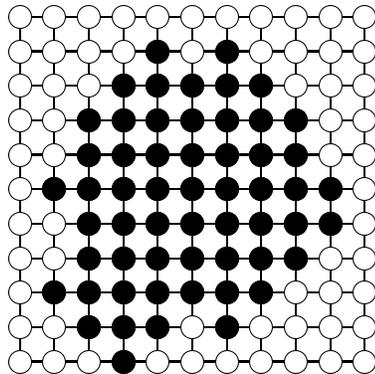
Y



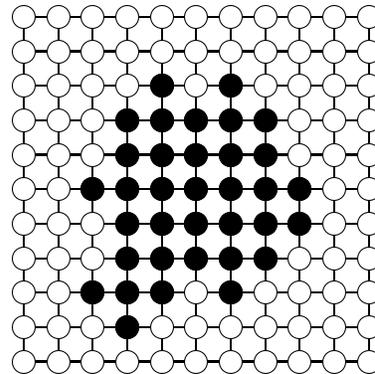
$X \ominus Y$



$(X \ominus Y) \oplus Y = X \circ Y$



$(X \circ Y) \oplus Y$



$((X \circ Y) \oplus Y) \ominus Y = (X \circ Y) \bullet Y$

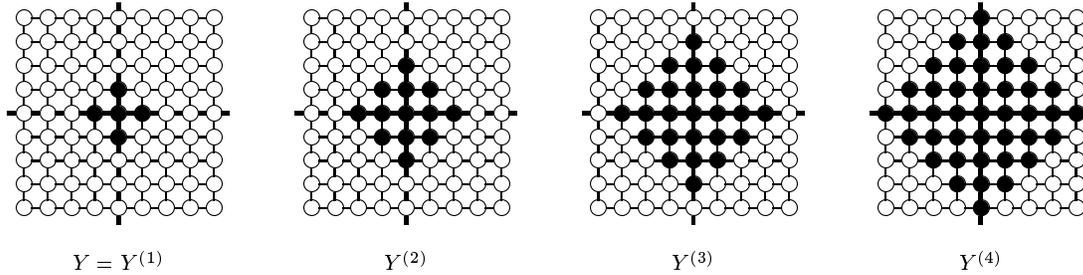
1.23. ábra Példa morfológiai szűrésre. Az Y szerkesztőelem ugyanaz, mint ami az 1.21b. ábrán látható, vagyis a 4–szomszédságnak felel meg. Az aktuális halmaz elemeit (vagy az aktuális kép fekete pontjait) “•” szimbólumok jelölik.

A morfológiai váz egyszerűbb tárgyalása érdekében bevezetjük az *iterált szerkesztőelem* fogalmát és az *iterált dilatació* valamint az *iterált erózió* műveleteket.

Legyen $Y \subset V$ egy tetszőleges szerkesztőelem ($V = \mathbb{Z}^2$ vagy \mathbb{Z}^3) és jelölje O_V az origót (ami $(0,0)$, ha $V = \mathbb{Z}^2$, és $(0,0,0)$, ha $V = \mathbb{Z}^3$). Az Y elem k -adik iteráltját jelölje $Y^{(k)}$, ahol:

$$Y^{(k)} = \begin{cases} \{O_V\} & , \text{ ha } k = 0 \\ Y^{(k-1)} \oplus Y & , \text{ ha } k \geq 1 \end{cases} .$$

Iterált szerkesztőelemre az 1.24. ábra ad példát.



1.24. ábra Példa iterált szerkesztőelemre. Az $Y \subseteq \mathbb{Z}^2$ elembe és annak iteráltjaiba eső pontokat “•” szimbólumok jelölik. A vastag vonalak az origóban metszik egymást.

Az X ponthalmaznak az Y szerkesztőelemmel való iterált dilatacióját a következő formula írja le:

$$X \oplus_k Y = \begin{cases} X & , \text{ ha } k = 0 \\ (X \oplus_{k-1} Y) \oplus Y & , \text{ ha } k > 0 \end{cases} .$$

Az iterált erózió hasonlóképpen definiálható:

$$X \ominus_k Y = \begin{cases} X & , \text{ ha } k = 0 \\ (X \ominus_{k-1} Y) \ominus Y & , \text{ ha } k > 0 \end{cases} .$$

Könnyen belátható, hogy az erózióra és a dilatacióra teljesülnek alábbi tulajdonságok tetszőleges X, Y és Z halmazok esetén:

$$\begin{aligned} (X \oplus Y) \oplus Z &= X \oplus (Y \oplus Z), \\ (X \ominus Y) \ominus Z &= X \ominus (Y \oplus Z). \end{aligned}$$

A dilatació és az erózió meghatározásából az alábbi két azonosság is következik:

$$\begin{aligned} X \oplus \{O_V\} &= X, \\ X \ominus \{O_V\} &= X. \end{aligned}$$

A fentiek alapján (pl. teljes indukcióval) könnyen beláthatók a következők:

$$\begin{aligned} X \oplus_k Y &= X \oplus Y^{(k)}, \\ X \ominus_k Y &= X \ominus Y^{(k)}. \end{aligned}$$

Megállapíthatjuk, hogy a “nagy méretű” szerkesztőelemekkel való dilatació és erózió kiváltható “kis méretűekkel” végrehajtott iterált műveletekkel.

1.4.2. A morfológiai vékonyítás

A vékonyítás a matematika morfológia eszközeivel is leírható (Serra, 1982; Schalkoff, 1989; Gonzales, Woods, 1992; Sonka, Hlaváč, Boyle, 1999; Watt, Policarpo, 1998), ahol az egy iterációs lépés során (az objektumok határáról) eltávolítandó pontokat a hit-or-miss transzformáció jelöli ki. Az X halmaznak az Y szerkesztőelemmel való vékonyítása formálisan:

$$\begin{aligned} X \otimes Y &= X \setminus (X \circledast Y) \\ &= X \cap (X \circledast Y)^c. \end{aligned}$$

Hangsúlyozandó, hogy a fenti formula csak a vékonyítás egy iterációs lépésére vonatkozik, amit addig kell ismételni, amíg $X \circledast Y \neq \emptyset$.

Ha a vékonyítás nem a 2.3. pontban ismertetésre kerülő teljesen párhuzamos, hanem a 2.1. pontban bemutatott irány-szekvenciális módszert követi, akkor egy iterációs lépés több fázisból (al-iterációból) áll, ahol az egymást követő fázisokhoz különböző szerkesztőelemek tartoznak. k -fázisú ($k > 1$) vékonyítás esetén egy k tagból álló szerkesztőelem-rendszert kell megadni:

$$\mathcal{Y} = \{Y_1, Y_2, \dots, Y_k\},$$

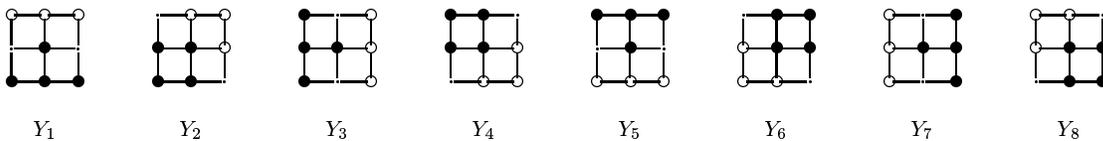
ahol Y_i az i -edik fázis során alkalmazandó szerkesztőelem ($1 \leq i \leq k$).

Az X halmaznak az \mathcal{Y} szerkesztőelem-rendszerrel történő vékonyításának egy iterációs lépését a következő formula írja le:

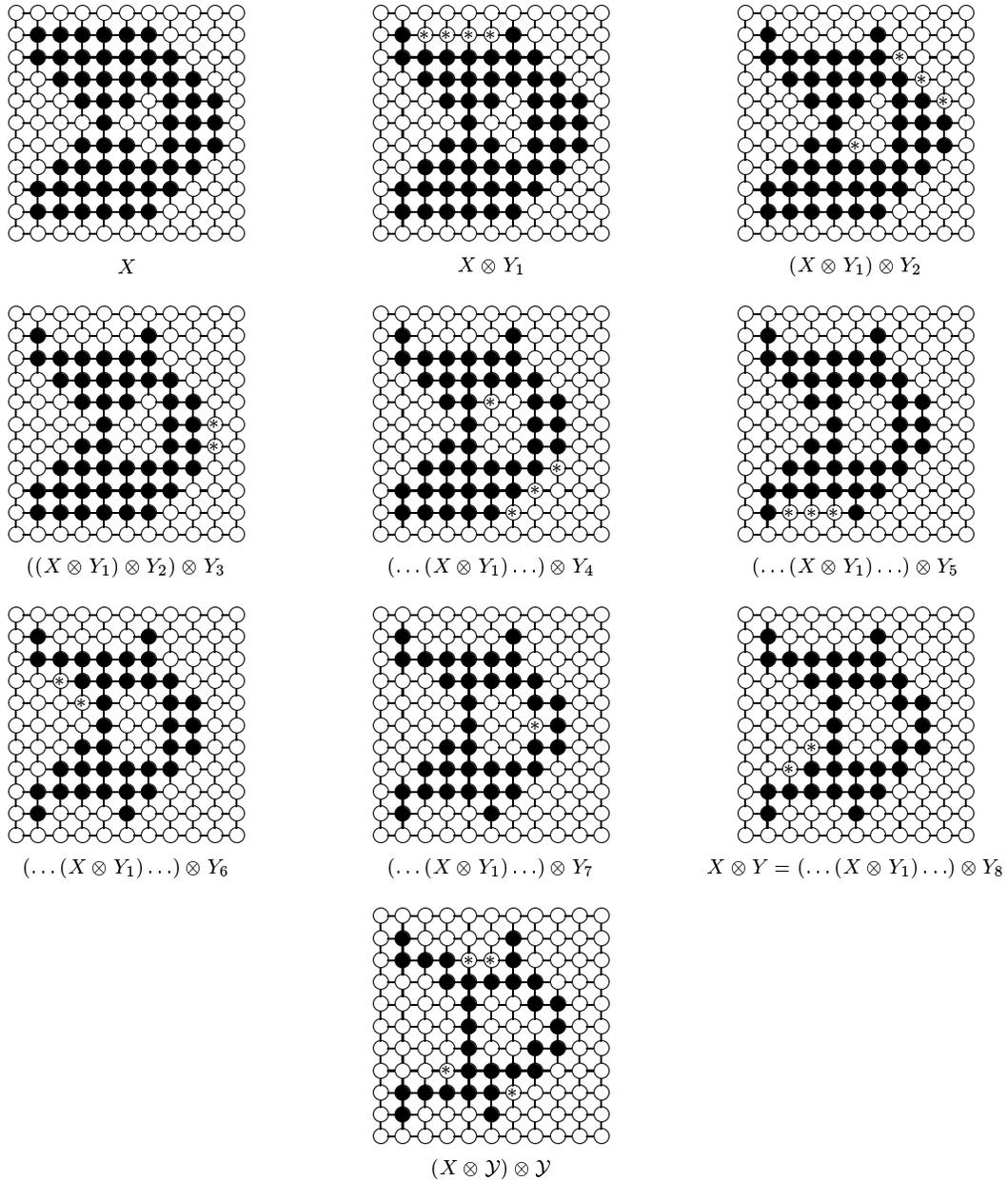
$$X \otimes \mathcal{Y} = ((\dots((X \otimes Y_1) \otimes Y_2) \dots) \otimes Y_k).$$

A morfológiai vékonyítást (pontosabban annak egy lépését) képműveletként is megfogalmazhatjuk: a $\mathcal{P} = (V, m, n, B)$ egy bináris képen az \mathcal{Y} szerkesztőelem-rendszerrel adott vékonyítás egy lépése a $\mathcal{P}' = (V, m, n, B \otimes \mathcal{Y})$ képet eredményezi.

A morfológiai vékonyítás kívánt hatása a szerkesztőelem-rendszer alkalmas megválasztásával érhető el. Az 1.25. ábrán látható 2D szerkesztőelem-rendszerrel (Gonzales, Woods, 1992) végrehajtott 8-fázisú vékonyításra az 1.26. ábra mutat példát. Megjegyezzük, hogy az eljárás megőrzi a topológiát a (4, 8) és a (8, 4) képeken, de csak a (4, 8) képekre szolgáltat “egy pont vastagságú” vonalszegmenseket.



1.25. ábra Az $\mathcal{Y} = \{Y_1, Y_2, \dots, Y_8\}$ szerkesztőelem-rendszer 2D vékonyításra. Az $Y_i = Y_{i1} \cup Y_{i2}$ szerkesztőelem a 8-fázisú vékonyítás i -edik fázisához tartozik ($1 \leq i \leq 8$). Az Y_{i1} -beli elemeket “•”, az Y_{i2} -be esőket pedig “o” szimbólumok jelölik. Az origó minden esetben a 3×3 -as környezet középpontja. Megjegyezzük, hogy az utolsó hat szerkesztőelem az első kettőnek elforgatott vátozata, ahol a forgatási szögek: 90° , 180° és 270° .



1.26. ábra Példa morfológiai vékonyításra. Az eljárás az 1.25. ábrán bemutatott $\mathcal{Y} = \{Y_1, \dots, Y_8\}$ szerkesztőelem-rendszerrel dolgozik. Az adott halmazba tartozó \mathbb{Z}^2 -beli pontoknak “•”, míg a halmazon kívülieknek “○” szimbólumok felelnek meg. Az adott fázisban törölt pontokat még a “*” szimbólummal is megjelöltük. A képsorozat első három sora a kiindulási halmazt és a vékonyítás első iterációs lépésének 8 fázisát mutatja be. A második iterációs lépésben kialakult végeredmény a negyedik sorban látható.

1.4.3. A morfológiai váz

A váz kifejezhető a matematikai morfológia műveleteivel is (Serra, 1982; Matheron, 1988; Gonzales, Woods, 1992; Awcock, Thomas, 1996). A d -dimenziós euklidészi tér tetszőleges X ponthalmazának vázát a halmazba beírt maximális d -dimenziós nyílt hipergömbök középpontjai alkotják.

Legyen $\delta > 0$ tetszőleges valós szám és legyen az Y_δ szerkesztőelem az origó középpontú, δ -sugarú (n -dimenziós) hipergömb, vagyis az 1.1. pontbeli jelöléssel: $Y_\delta = \mathcal{B}(O, \delta)$. Könnyen belátható, hogy $\mathcal{B}(O, k\delta) = Y_\delta^{(k)}$ ($0 \leq k$).

Az X ponthalmazba írható $k\delta$ -sugarú hipergömbök középpontjainak halmaza kifejezhető az erózióval: $X \ominus Y_\delta^{(k)}$. Azon X -beli pontok, amelyek $k\delta$ -sugarú beírt hipergömbök középpontjai, de nem középpontjai már a $(k+1)\delta$ -sugarú beírt hipergömböknek, a következő formulával adhatók meg:

$$(X \ominus Y_\delta^{(k)}) \setminus [(X \ominus Y_\delta^{(k+1)}) \oplus Y_\delta] = (X \ominus_k Y_\delta) \setminus [(X \ominus_k Y_\delta) \circ Y_\delta].$$

Az X halmaz $S(X)$ váza tehát a következő formulával írható le:

$$S(X) = \lim_{\delta \rightarrow +0} \bigcup_{k=0}^{\infty} \{(X \ominus_k Y_\delta) \setminus [(X \ominus_k Y_\delta) \circ Y_\delta]\}.$$

A maximális $k\delta$ -sugarú beírt hipergömbök középpontjai dilatációval $k\delta$ -sugarú hipergömbökké hízlalhatók, amelyek egyesítése az X halmaz egy lefedőrendszerét adja:

$$X = \lim_{\delta \rightarrow +0} \bigcup_{k=0}^{\infty} \{(X \ominus_k Y_\delta) \setminus [(X \ominus_k Y_\delta) \circ Y_\delta]\} \oplus_k Y_\delta.$$

Diszkrét terek (pl. \mathbb{Z}^2 és \mathbb{Z}^3 esetében) a morfológiai váz meghatározásához olyan Y (szimmetrikus) szerkesztőelemet szoktak feltételezni, ami az origó középpontú, 1-sugarú hipergömb egy közelítése.

A diszkrét morfológiai vázat a következő formula adja meg:

$$S(X) = \bigcup_{k=0}^{\infty} \{(X \ominus_k Y) \setminus [(X \ominus_k Y) \circ Y]\}.$$

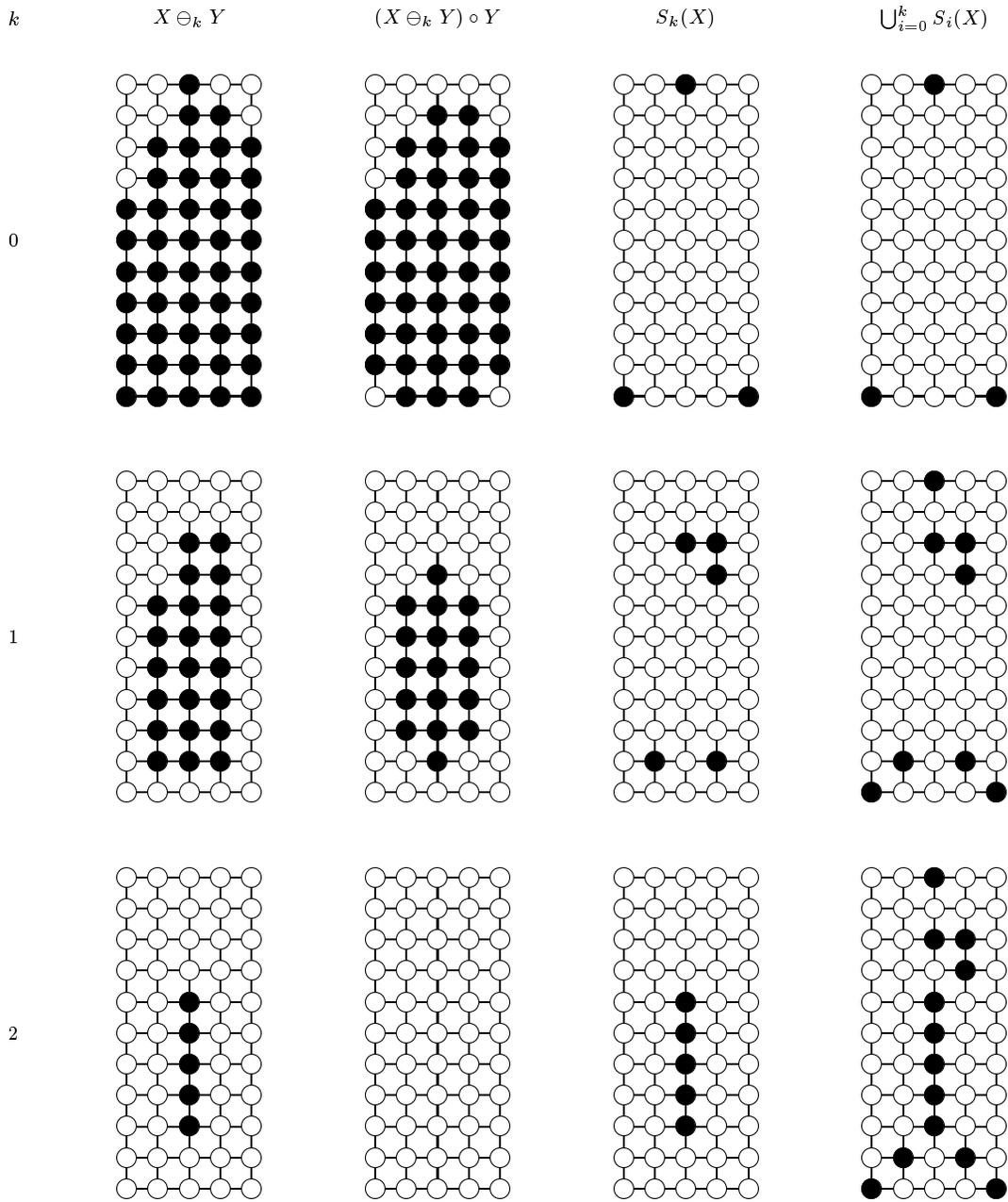
Véges X halmazra az $S(X)$ váz véges számú halmaz egyesítésével áll elő, vagyis:

$$S(X) = \bigcup_{k=0}^K \{(X \ominus_k Y) \setminus [(X \ominus_k Y) \circ Y]\},$$

ahol

$$K = \max\{k \mid (X \ominus_k Y) \neq \emptyset\}.$$

Morfológiai vázra az 1.27. ábrával mutatunk egy 2D példát. Megállapíthatjuk, hogy a példában szereplő objektum váza nem lesz összefüggő. Ez egyrészt abból ered, hogy diszkrét pontrácson értelmezett a kép, másrészt pedig a szerkesztőelem és annak iteráltjai nem közelítik jól a körlapokat.



1.27. ábra Példa morfológiai vázra. Az Y szerkesztőelem ugyanaz, mint ami az 1.21. ábrán látható, vagyis a 4-szomszédságnak felel meg. A bal felső képen látható a kiindulási $X = X \ominus_0 Y$ halmaz, melynek $S(X) = \bigcup_{i=0}^2 S_i(X)$ váza a jobb alsó képen található, ahol $S_i(X) = (X \ominus_i Y) \setminus [(X \ominus_i Y) \circ Y]$.

1.4.4. Morfológiai műveletek maszkillesztéssel

A matematikai morfológia eszközeivel számos képművelet leírható és tulajdonságaik is bizonyíthatók a segítségükkel. A morfológiai megközelítés hátránya, hogy közepesen bonyolult alkalmazások is a morfológiai alpműveletek tucatjainak vagy százainak egymásutáni végrehajtását igénylik.

Könnyen belátható, hogy a morfológiai alpműveletek (a dilatáció, az erózió és a hit-or-miss transzformáció) lokálisak és párhuzamosak, vagyis egyfelől, egy képpont új értéke csak a kérdéses pontnak az alkalmazott szerkesztőelem kiterjedésével megegyező méretű környezetétől függ, másrészt pedig valamennyi képpont egyidejűleg számítható. (Egy d -dimenziós szerkesztőelem kiterjedésén az origó középpontú, a szerkesztőelemet lefedő legszűkebb d -dimenziós hipergömb átmérőjét értjük.)

Lokális bináris képműveletek megadására gyakran alkalmazott módszer a *maszkillesztés* (*template matching*). A *maszk* (*mask, template*) egy bináris tömb (pl. 3×3 -as 2D-ben és $3 \times 3 \times 3$ -as 3D-ben). A maszkot középpontjával ráhelyezzük minden egyes képpontra és azon képpontok értékét változtatjuk meg (negáljuk), ahol valamennyi 0 maszkelem 0 (fehér) képponttal és valamennyi 1 maszkelem 1 (fekete) képponttal kerül fedésbe. Hangsúlyozandó, hogy a maszk “ráhelyezésekor” annak elforgatása vagy tükrözése nem engedélyezett. Az egyszerűség kedvéért a maszk illesztését 2D-re adjuk meg formálisan, ami 3D-re teljesen analóg módon terjeszthető ki.

Legyen $A = [a_{i,j}]$ egy $n_1 \times n_2$ méretű bináris tömb, ami egy (véges) 2D bináris képet tárol. (Elegendő a véges kép fekete pontjait “befoglaló” legszűkebb téglalapnak megfelelő képrészletet tárolni a bináris tömbben.) Legyen $M = [m_{u,v}]$ egy $(2k + 1) \times (2k + 1)$ -es maszk, ahol $u, v \in \{-k, \dots, -1, 0, 1, \dots, k\}$. Az A tömbön az M maszk illesztése az — ugyancsak — $n_1 \times n_2$ méretű $A * M = C = [c_{i,j}]$ tömböt eredményezi⁹, ahol:

$$c_{i,j} = a_{i,j} \diamond \bigwedge_{u=-k}^k \bigwedge_{v=-k}^k (m_{u,v} \equiv a_{i+u,j+v}).$$

(“ \diamond ” a *kizáró vagy* (*exclusive or, xor*), “ \wedge ” az *és* (*and*), “ \equiv ” pedig az *ekvivalencia* 2-változós Boole-műveletet jelöli.) Tegyük fel, hogy a fenti formulában $a_{i+u,j+v} = 0$, ha $i + u \notin \{1, \dots, n_1\}$ vagy $j + v \notin \{1, \dots, n_2\}$ (mivel a tömben tárolt képrészleten kívül csak fehér pontok találhatóak)¹⁰. Könnyen belátható, hogy a tömbelemek értéke valóban csak a maszk illeszkedése esetén változik meg, mivel tetszőleges $a \in \{0, 1\}$ esetén: $a \diamond 0 = a$ és $a \diamond 1 = \bar{a}$ (ahol “ $\bar{}$ ” a *negáció* (*not*) 1-változós Boole-műveletet jelöli).

Számos esetben a maszkokban a 0 és az 1 mellett a “.” *semleges* (*don't care*) elemek előfordulását is megengedik, amelyek a fehér és a fekete tömbelemekre (képpontokra) egyaránt illeszkednek. Semleges elemeket is tartalmazó maszk esetén $C = A * M$ a következő formula szerint számítandó:

$$c_{i,j} = a_{i,j} \diamond \bigwedge_{u=-k}^k \bigwedge_{v=-k}^k (m_{u,v} \simeq a_{i+u,j+v}),$$

⁹A diszkrét konvolúcióval (Gonzales, Woods, 1992; Sonka, Hlaváč, Boyle, 1999), való hasonlóság miatt választottuk a “*” szimbólumot a maszkillesztés műveleti jelének.

¹⁰A bináris tömbből való “kicímzés” két ok miatt célszerű: egyrészt így a maszkillesztés formulájában nem kell megkülönböztetnünk az eredménytömb “szélén” lévő elemeket, másrészt pedig nem kell kikötni, hogy a maszk mérete nem lehet nagyobb a bináris tömbénél.

ahol a “ \simeq ” 2-változós műveletet az alábbi formula írja le:

$$m \simeq a = \begin{cases} 1 & , \text{ ha } m = \cdot \\ m \equiv a & , \text{ ha } m \in \{0, 1\} \end{cases} .$$

A maszkillesztés és a morfológiai műveletek kapcsolatára a következő tétellel világítunk rá.

1.8. TÉTEL

A morfológiai műveletekkel leírt képműveletek (ahol a halmazokat a képek fekete pontjai alkotják és a szerkesztőelemek is az adott diszkrét rácson értelmezettek) megadhatók maszkillesztéssel és pontonkénti Boole-műveletekkel.

BIZONYÍTÁS

Megmutatjuk, hogy az 1.4.1. pontban bemutatott valamennyi morfológiai művelethez konstruálható olyan (semleges elemeket is tartalmazó) maszk, hogy a maszkillesztéssel (és a megfelelő pontonkénti Boole-művelet segítségével) leírt képművelet megfelel a kérdéses morfológiai műveletnek.

- Tekintsük először az $X \otimes Y$ hit-or-miss transzformációt, ahol $X \subset \mathbb{Z}^2$ vagy $X \subset \mathbb{Z}^3$. Feleltessük meg az X ponthalmaznak egy A bináris tömb fekete elemeit. Az $Y = Y_1 \cup Y_2$ szerkesztőelemhez konstruáljuk meg az M maszkot a következőképpen: Legyen a maszk mérete akkora, hogy lefedje Y valamennyi elemét. Az Y_1 -be eső pontokhoz rendeljünk fekete maszkelemeket, az Y_2 -beliekhez fehéreket, a maszk többi eleme pedig legyen semleges. (Az 1.25. ábrán bemutatott szerkesztőelemek maszkoknak is tekinthetők, ha a “•” pozíciókat fekete maszkelemeknek, a “o” pozíciókat pedig fehéreknek vesszük.)

A fenti konstrukció mellett a hit-or-miss transzformáció eredményét az $A \diamond (A * M)$ bináris tömb reprezentálja, ahol “ \diamond ” az elemenkénti *kizáró vagy* műveletet jelöli.

Ha az $Y = Y_1 \cup Y_2$ szerkesztőelem origója Y_1 -be esik, akkor az $X \otimes Y = X \setminus (X \otimes Y)$ vékonyítás eredménye (1.4.2.) éppen $A * M$, vagyis a maszkillesztés alkalmas eszköz a vékonyító algoritmusok leírására.

- Legyen az $Y = Y_1 \cup Y_2$ szerkesztőelem olyan, hogy $Y = Y_1$ és $Y_2 = \emptyset$. Ekkor:

$$X \otimes Y = (X \ominus Y_1) \setminus (X \oplus \hat{Y}_2) = X \ominus Y_1 = X \ominus Y ,$$

mivel tetszőleges Z halmazra $Z \oplus \emptyset = \emptyset$, vagyis az erózió kifejezhető a hit-or-miss transzformációval, ami a fentiek szerint megadható maszkillesztéssel (és az elemenkénti kizáró vagy művelettel).

- A dilatáció megadásához használjuk ki, hogy duális párja az erózióknak, azaz

$$X \oplus Y = (X^c \ominus \hat{Y})^c .$$

Ha $X \ominus Y$ eredményét az $A * M$ tömb reprezentálja, akkor az $X \oplus Y$ dilatációnak az

$$\overline{\overline{A * M}}$$

tömb felel meg, ahol a felülvonás az elemenkénti negációt jelöli, az M' maszkot pedig pedig M -ből kapjuk a középpontra való tükrözéssel.

- Az $(X)_a$ eltolás kifejezhető az $X \oplus Y$ dilatacióval is, ahol $Y = \{a\}$, vagyis az eltolás is megadható maszkillesztéssel.

Megjegyezzük, hogy az origótól “távoli” a eltolási értékre a szükséges maszk mérete is “nagy” lesz. Ekkor a -t érdemes felbontani az $a_1 + a_2 + \dots + a_p = a$ “közeli” vektorok összegére, mivel ekkor $(X)_a = (((\dots (X)_{a_1})_{a_2}) \dots)_{a_p}$.

Az elemi halmazműveleteknek: “-” (negáció, komplementerképzés), “ \cup ”, “ \cap ” és “ \setminus ” rendre megfelelnek a bináris tömbökön a pontonkénti “-” (“**not**”), “ \vee ” (“**or**”), “ \wedge ” (“**and**”) és “ $\overline{\wedge}$ ” (“**andnot**”) Boole-műveletek. Megjegyezzük, hogy a negáció megadható maszkillesztéssel is, ahol az (akár egyetlen elemet tartalmazó) maszk valamennyi eleme semleges.

Mivel az 1.4.2. és a 1.4.3. pontokban leírt morfológiai vékonyítás és a morfológiai váz megadásában is a fenti alpműveletek szerepelnek, így azokra az összetett műveletekre is érvényes a tétel állítása. ■

Az 1.8. Tétel bizonyítása szerint egyetlen maszkillesztéssel (és az elemenkénti logikai műveletek segítségével) leírható bármelyik morfológiai alpművelet. Ugyanakkor egyetlen maszkillesztés nem képes tetszőleges lokális műveletet megadni, ahol a $(2k+1) \times (2k+1)$ -es környezettől való függést egy $(2k+1)(2k+1)$ -változós “ f ” Boole-függvény írja le, vagyis az eredménytömb elemei az alábbi formula szerint számíthatók:

$$c_{i,j} = a_{i,j} \diamond f(a_{i-k,j-k}, \dots, a_{i,j}, \dots, a_{i+k,j+k}).$$

Vezessük be az *általánosított maszkillesztést*, ami nem egyetlen maszkkal, hanem egy

$$\mathcal{M} = \{M^1, M^2, \dots, M^s\}$$

s -elemű maszkrendszerrel dolgozik. Az egyszerűség kedvéért tegyük fel, hogy a maszkrendszer valamennyi $M^t = [m_{u,v}^t]$ eleme $(2k+1) \times (2k+1)$ -es ($1 \leq t \leq s$). (Ha nem így lenne, akkor a kisebb méretűeket egészítsük ki semleges komponensekkel.) Az A tömre az \mathcal{M} maszkrendszer illesztése az $A \ast \mathcal{M} = C = [c_{i,j}]$ $n_1 \times n_2$ -es tömböt eredményez, ahol:

$$c_{i,j} = a_{i,j} \diamond \bigvee_{t=1}^s \bigwedge_{u=-k}^k \bigwedge_{v=-k}^k (m_{u,v}^t \simeq a_{i+u,j+v}),$$

vagyis egy képpont értéke akkor változik meg, ha a maszkrendszer legalább egy eleme illeszkedik az adott pont környezetére.

Az általánosított maszkillesztés már alkalmas eszköz tetszőleges lokális művelet megadására.

1.9. TÉTEL (Palágyi, 1993)

Bármely lokális párhuzamos képművelet leírható általánosított maszkillesztéssel.

BIZONYÍTÁS

Tekintsük az $f(x_1, x_2, \dots, x_p)$ $p=(2k+1)(2k+1)$ -változós Boole-függvényt, ami egy tetszőleges lokális párhuzamos képművelethez tartozik. Bármely Boole-függvény, így f is megadható a *diszjunktív normálformula/normálforma* segítségével, vagyis:

$$f(x_1, x_2, \dots, x_p) = \bigvee_{f(\alpha_1, \alpha_2, \dots, \alpha_p)=1} (x_1^{\alpha_1} \wedge x_2^{\alpha_2} \wedge \dots \wedge x_p^{\alpha_p}),$$

ahol:

$$x^\alpha = (\alpha \simeq x) = \begin{cases} \bar{x} & , \text{ ha } \alpha = 0 \\ x & , \text{ ha } \alpha = 1 \\ 1 & , \text{ ha } \alpha = \cdot \end{cases} .$$

Megjegyezzük, hogy $\alpha \in \{0, 1\}$ esetén $(\alpha \simeq x) = (\alpha \equiv x)$.

Tegyük fel, hogy az f függvény az alábbi s darab bemeneti kombinációra ad igaz választ ($0 \leq s \leq 2^p$):

$$\begin{aligned} f(\alpha_1^1, \alpha_2^1, \dots, \alpha_p^1) &= 1 \\ f(\alpha_1^2, \alpha_2^2, \dots, \alpha_p^2) &= 1 \\ &\vdots \\ f(\alpha_1^s, \alpha_2^s, \dots, \alpha_p^s) &= 1 . \end{aligned}$$

Az f függvény diszjunktív normálformulája a következő alakban is felírható:

$$f(x_1, x_2, \dots, x_p) = \bigvee_{t=1}^s (x_1^{\alpha_1^t} \wedge x_2^{\alpha_2^t} \wedge \dots \wedge x_p^{\alpha_p^t}) = \bigvee_{t=1}^s \bigwedge_{q=1}^p (\alpha_q^t \simeq x_q) .$$

Az α_q^t értékeket tömbalakba rendezve az $M^t = [m_{u,v}^t]$ maszkot kapjuk ($1 \leq t \leq s$), amelyek egy s -elemű \mathcal{M} maszkrendszer alkotnak. Ekkor:

$$c_{i,j} = a_{i,j} \diamond f(a_{i-k,j-k}, \dots, a_{i,j}, \dots, a_{i+k,j+k}) = a_{i,j} \diamond \bigvee_{t=1}^s \bigwedge_{u=-k}^k \bigwedge_{v=-k}^k (m_{u,v}^t \simeq a_{i+u,j+v}) ,$$

vagyis tetszőleges lokális művelethez konstruálható olyan maszkrendszer, amivel az általánosított maszkillesztés éppen az adott művelettel ekvivalens. ■

Az 1.9. Tétel bizonyításában alkalmazott konstrukcióval kapott maszkrendszer elemszáma csökkenthető, ha minimalizált normálformulából indulunk ki. A normálformula két tagja (mintermje vagy elemi konjunkciója) összevonható abban az esetben, ha csak egy változó — mondjuk a q -adik — kitevőjében különböznek egymástól:

$$\begin{aligned} (x_1^{\alpha_1} \wedge \dots \wedge x_{q-1}^{\alpha_{q-1}} \wedge x_q^0 \wedge x_{q+1}^{\alpha_{q+1}} \wedge \dots \wedge x_p^{\alpha_p}) &\vee \\ (x_1^{\alpha_1} \wedge \dots \wedge x_{q-1}^{\alpha_{q-1}} \wedge x_q^1 \wedge x_{q+1}^{\alpha_{q+1}} \wedge \dots \wedge x_p^{\alpha_p}) &= \\ (x_1^{\alpha_1} \wedge \dots \wedge x_{q-1}^{\alpha_{q-1}} \wedge \bar{x}_q \wedge x_{q+1}^{\alpha_{q+1}} \wedge \dots \wedge x_p^{\alpha_p}) &\vee \\ (x_1^{\alpha_1} \wedge \dots \wedge x_{q-1}^{\alpha_{q-1}} \wedge x_q \wedge x_{q+1}^{\alpha_{q+1}} \wedge \dots \wedge x_p^{\alpha_p}) &= \\ (x_1^{\alpha_1} \wedge \dots \wedge x_{q-1}^{\alpha_{q-1}} \wedge 1 \wedge x_{q+1}^{\alpha_{q+1}} \wedge \dots \wedge x_p^{\alpha_p}) &= \\ (x_1^{\alpha_1} \wedge \dots \wedge x_{q-1}^{\alpha_{q-1}} \wedge x_{q+1}^{\alpha_{q+1}} \wedge \dots \wedge x_p^{\alpha_p}) & . \end{aligned}$$

Ekkor a fenti két mintermhez rendelt maszk helyett csak egyet kell alkalmazni, ahol a kiejtett változónak megfelelő pozícióra semleges elem kerül.

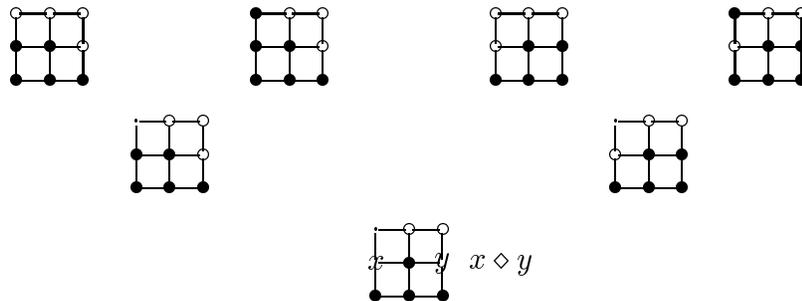
Az összevonás az “és” művelet kommutativitása mellett a következőkön alapul:

$$(\beta \wedge \bar{x}) \vee (\beta \wedge x) = \beta \wedge (\bar{x} \vee x) = \beta \wedge 1 = \beta ,$$

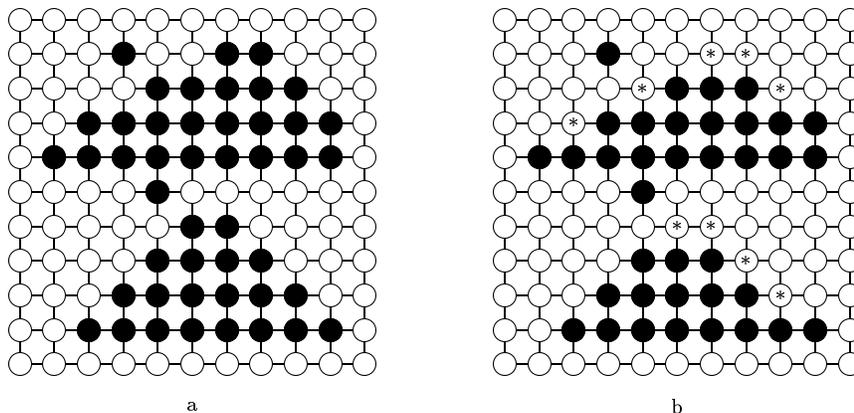
ahol x egy Boole-változó, β pedig egy tetszőleges logikai kifejezés.

Hatékonyabb optimalizálás érhető el, vagyis kevesebb maszkra van szükség, ha a maszkokat nem csak “vagy”-kapcsolat fűzi össze, hanem általánosabb (pl. “negáció”-t, “és”-t és “kizáró vagy”-ot is megengedő) formulát használunk. Számos esetben hasznos, ha maszkok a “0”, az “1” és a “.” konstansok mellett Boole-változókat is tartalmazhatnak és az ilyen maszkokhoz a Boole-változóik egy formuláját is csatoljuk. Az ilyen maszkok akkor illeszkednek, ha valamennyi konstansuk illeszkedik a lefedett környezetre és a csatolt formula igaz a változói által lefedett értékekre.

Bináris, semleges elemet és változót tartalmazó maszkokra az 1.28. ábra mutat példákat, a maszkillesztést pedig az 1.29. ábra illusztrálja.



1.28. ábra Példa maszkokra. Az első sor 4 (bit)maszkjával adott általánosított maszkillesztés ugyanazt a lokális operátort írja le, mint a második sor kettő (semleges elemet is megengedő) vagy a harmadik sor egyetlen (változókat és csatolt formulát tartalmazó) maszkja. (Az “1” maszkelemeket “•”, a “0”-kat pedig “o” szimbólumok jelölik.)



1.29. ábra Példa maszkillesztésre. Az eredeti (a) képre (bináris tömbre) az 1.28. ábra maszkjaival adott általánosított maszkillesztés a (b) képet (tömböt) eredményezi. A fekete pontoknak “•”, a fehéreknek “o” szimbólumok felelnek meg. A törölt pontokat megjelöltük a “*” szimbólummal.

Boole-függvények szimbólikus manipulációira (pl. az optimalizálásra) hatékony reprezentációt biztosítanak a *bináris döntési diagramok* (*Binary Decision Diagram, BDD*) (Akers, 1978), amelyeket digitális- és véges-állapotú rendszerek tervezésére és analízisére is alkalmaznak (Bryant, 1992). Az $f(x_1, x_2, \dots, x_p)$ Boole-függvény BDD-je egy olyan körmentes irányított gráf, ahol azok a szögponatok, amelyekből élek futnak ki az x_i ($i = 1, \dots, p$) változók tesztelésére szolgálnak, míg a többi csúcs a (0 vagy 1) függvényértékeknek felel meg, az élek pedig a 0 vagy 1 értékekkel címkézettek. A BDD mérete

drasztikusan csökkenthető a *rendezett és redukált BDD* (*ordered reduced BDD*) (Bryant, 1992) előállításával, amiből magkapható az adott Boole-függvényt kiértékelő hatékony (pl. C-nyelvű) kód (Robert, Malandain, 1998).

A rendezett és redukált BDD előállításához hasonló optimalizálást alkalmaz az a programrendszer, amivel a jelen dolgozatban bemutatásra kerülő eljárásokat megadtuk és teszteltük (Alb, Tanács, 1997; Tanács, Palágyi, 1998). A rendszer táblázatgeneráló modulja meghatározza az adott maszkrendszerhez tartozó Boole-függvényt, majd az $f(x_1, x_2, \dots, x_p)$ függvény igazságtáblájának megfelelően egy teljes bináris fát, ahol a 2^p darab levél tárolja a függvényértékeket, az i -edik ($i = 1, \dots, p$) szinten lévő szögpontok pedig az x_i változóhoz tartoznak. A “tömörítés” az alábbi két szabályt használja:

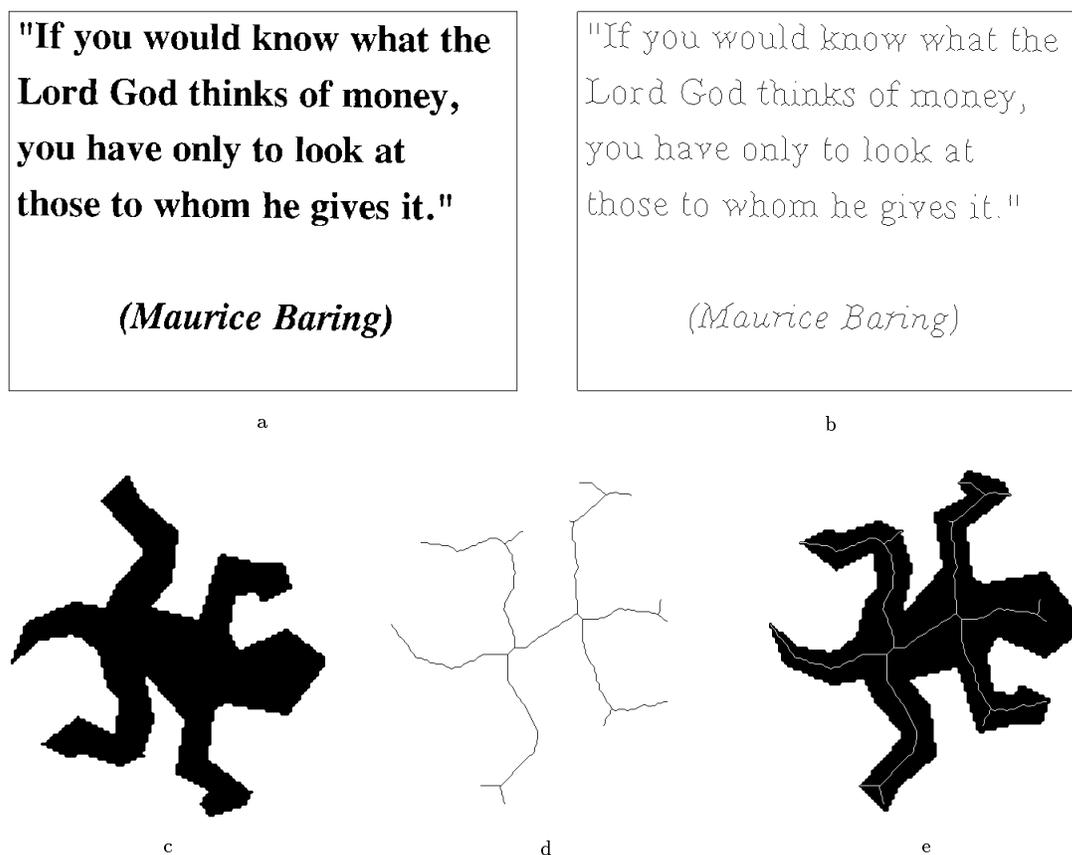
- Ha egy részfában minden levél értéke ugyanaz, akkor a részfát helyettesítse annak egyetlen levele.
- Ha egy szögpontnak megegyezik a két részfája, akkor azokat csak egy példányban tároljuk.

A rendszer végrehajtó modulja az így kapott redukált táblázatban keres — kiértékelés helyett. Megjegyzendő, hogy az optimalizálást a teljes igazságtábla (fa) felépítése közben sikerült megoldani. 3D-ben a leggyakrabban figyelt $3 \times 3 \times 3$ -as lokális környezethez általában egy 26-változós Boole-függvény tartozik, aminek még a redukált táblája is túl nagy lehet (és a redukált tábla kiszámítása is nagyon hosszú ideig tart). Ekkor a táblázatgeneráló modul a kiértékelendő Boole-formulát továbbítja a végrehajtó modulnak.

A végrehajtást tovább gyorsítja az, hogy a táblázatgeneráló modul informálja a végrehajtó modult az adott lokális párhuzamos képművelet jellegéről (addíció, redukció, vagy vegyes). Addíció vagy redukció esetén szükségtelen a “1”, illetve az “0” képpontokkal foglalkozni (a képpont adott környezetébe eső pontokat összegyűjteni és azokkal megcímezni a táblát vagy kiértékelni a formulát), mivel azon pontok értéke nem változhat meg. Ez különösen fontos a vékonyító algoritmusoknál, mivel az redukciók sorozatából áll. Az iteratív eljárás egy lépése a feldolgozás előrehaladtával fokozatosan gyorsul, mivel egyre csökken a képen az objektumpontok száma.

2. A vékonyítás és módszerei

A 2D vékonyítást mint előfeldolgozó technikát gyakran használják az alakfelismerő rendszerekben, pl. karakterek, ujjlenyomatok, vonalrajzok felismerése és elemzése során (Suen, Wang, 1994), vagy a raster-vektor konverzió előfeldolgozó lépéseként (Katona, Palágyi, Tóth, 1995). A vékonyítás célja egyrészt alkalmas képjellemzők kinyerése, másrészt pedig a kiértékelendő adatmennyiség csökkentése. 2D vékonyításra a 2.1. ábra ad példát, ahol Pratt és Kabir 2 al-iterációs párhuzamos algoritmusát (Pratt, Kabir, 1985) alkalmaztuk.



2.1. ábra Példák 2D vékonyításra. A vékonyított képen (b) a szöveg ugyanolyan jól olvasható, mint az első tesztképen (a). A második tesztképre (c) az alkalmazott vékonyító eljárás eredménye (d). Az egy pixel vastagságú középvonalak elhelyezkedése az objektumban (e). (A második objektum szerepelt már az 1.14. ábrán.)

A 2D vékonyítás — vagyis a kiterjedt bináris objektumok “egy pont vastagságú” középvonalukká redukálása — könnyű feladatnak tűnhet, mégis nagy kihívásnak bizonyult: egy 1994-ben megjelent, a 2D vékonyító stratégiákat áttekintő könyv több mint 300 publikációt említ e témában (Suen, Wang, 1994) és azóta is évente kb. 20 cikk jelenik meg 2D vékonyításról csak azokban a folyóiratokban, amelyeket Rosenfeld figyelemmel kísér¹¹. Míg 2D-ben számos vékonyító algoritmus létezik, addig 3D-ben csak néhány

¹¹A digitális képfeldolgozás tudományág egyik legtekintélyesebb személyisége Azriel Rosenfeld (Center for Automation Research, University of Maryland), aki a *Computer Vision and Image Understanding* folyóiratban évente közöl egy áttekintést az *Image Analysis and Computer Vision* területeken az előző évben publikált eredményekről.

eljárást közöltek. Ennek elsődleges oka az, hogy a topológia és annak megőrzése jóval bonyolultabb 3D-ben, mint 2D esetén.

A publikált 3D vékonyító algoritmusok száma rohamosan nőtt az utóbbi években, de — tudomásunk szerint — így is mindössze egy tucat — zömében párhuzamos — eljárást közöltek eddig más szerzők, melyek közül kilencet ismertetünk a jelen fejezetben. Nem mutatunk be két korai módszert (Srihari, Udupa, Yau, 1979; Lobregt, Verbeek, Groen, 1980), mivel azok nem őrzik meg a topológiát (egyedül csak az objektumok összefüggőségére, illetve az Euler-karakterisztika változatlanóságára ügyeltek) és nem tárgyaljuk a tetradekahedron pontrácsot feltételező algoritmust (Hafford, Preston, 1984) sem. A 2D vékonyító algoritmusok és technikák áttekintésére egyrészt helyhiány miatt nem vállalkozunk, másrészt pedig azért, mert azokra már elérhető több összefoglaló mű is (Smith, 1987; Lam, Lee, Suen, 1992; Arcelli, Sanniti di Baja, 1996; Hall, 1996; Fazekas, 1998/1; Fazekas, 1998/2; Marchand-Maillet, Sharaiha, 2000). Érdemes megemlíteni, hogy nincs tudomásunk a 3D vékonyító algoritmusokat áttekintő tanulmányról.

Más szerzők 3D algoritmusainak ismertetése mellett bemutatásra kerül négy olyan eljárás is, amelyeket Palágyi és Kuba dolgoztak ki. Az általunk javasolt algoritmusok teljesítik a vékonyítással szemben támasztott legfőbb követelményt, vagyis mind a négy bizonyítottan megőrzi a topológiát 3D (26,6) képekre.

Rosenfeld bebizonyította (Rosenfeld, 1970), hogy csupán a 3×3 -as környezetet figyelve, nem konstruálható olyan 2D topológia-megőrző lokális párhuzamos redukció, ami a zsugorítás problémáját megoldaná. Hasonló állítás teljesül 3D-ben a $3 \times 3 \times 3$ -as környezetre¹². Rosenfeld megállapítása fokozottan érvényes a vékonyításra, mivel a vékonyítás végpont-kritériummal nehezített zsugorítás. A fentiek komoly problémát jelentenek, hiszen 3D-ben egyrészt már a $3 \times 3 \times 3$ -as környezetet is nagy (bő 67 millió a lehetséges konfigurációk száma, a középpontot nem számítva), másrészt pedig az a lokális környezet már elegendő egy pont egyszerű voltának eldöntésére (mivel a figyelembe vett szomszédsági reláció-párok mellett az Euler-karakterisztika lokálisan számítható). A $3 \times 3 \times 3$ -as környezet elégtelensége miatt a 3D párhuzamos vékonyító algoritmusok tervezői két megközelítést javasoltak.

Az első módszert alkalmazók ragaszkodnak a $3 \times 3 \times 3$ -as környezethez és a fenti korlátozás hatálya alól az iterációs lépések megszervezésével bújnak ki. Erre két módszer született: az *irány-szekvenciális* (*directional, border-sequential*) (2.1.) és az *almező-szekvenciális* (*subfield-sequential*) (2.2.) technika. A 2.4. pontban bemutatjuk a fenti két stratégiát kombináló, Palágyi és Kuba által javasolt *hibrid* módszert is.

A második megközelítéssel élők nem bonyolítják el az iterációs lépéseket, inkább a $3 \times 3 \times 3$ -as környezetnél bővebb szomszédságot figyelnek. A 2.3. pontban a *teljesen párhuzamos* (*fully parallel*) megközelítést ismertetjük és bemutatjuk az általunk ismert három teljesen párhuzamos algoritmust. Az egyik eljárásról — tesztelése során — kiderítettük, hogy korrekcióra szorul. Ezt az algoritmust sikerült úgy módosítanunk, hogy topológiailag korrekt legyen.

Megjegyezendő, hogy párhuzamos vékonyításra 2D-ben is az irány-szekvenciális, az almező-szekvenciális és a teljesen párhuzamos technikák ismertek (Hall, 1996), de hibrid módszert követő 2D eljárás közléséről nem tudunk.

¹²Hangsúlyozandó, hogy az állítás párhuzamos redukciókra vonatkozik, ahol nem egyetlen pontot, hanem egy ponthalmazt törölünk egyidejűleg. Egyetlen pont topológia-megőrző törlése, vagyis az adott pont egyszerű volta a figyelembe vett szomszédsági relációk mellett eldönthető a 3×3 -as, illetve a $3 \times 3 \times 3$ -as környezet ismeretében.

A vékonyító eljárások ismertetésekor szembeötlő hiányosság, hogy nem közöljük az algoritmusok bonyolultságát, műveletigényét. A komplexitás megadása a vékonyítás jellege miatt nagyon nehéz probléma. Valamennyi vékonyító eljárás időigényét az egy iterációs lépés műveletigényének és a szükséges iterációs lépések számának a szorzata adja. Ha egy párhuzamos vékonyító eljárást szekvenciális számítógépen hajtunk végre, akkor egy lépés időigénye nemcsak attól függ, hogy mennyi objektumpont található a képen a feldolgozás adott fázisában, hanem attól is, hogy mi a határpontok és az objektumpontok számának az aránya. (Az időigényt ráadásul az is befolyásolja, hogy a határpontok milyen konfigurációkban fordulnak elő.) Ugyanakkor a szükséges iterációs lépések száma erősen függ a képtartalomtól. Az iterációs szám ugyanazon képre jelentősen eltérhet a különböző vékonyító eljárások esetében.

Valamennyi algoritmusunkat maszkokkal adott lokális és párhuzamos képműveletekből építettünk fel, ami hatékony Boole-implementációt tesz lehetővé. Tervezzük valamennyi ismert 3D vékonyító algoritmus Boole-implementációját, ahol a törlési feltételeket Boole-függvényekkel írjuk le és optimalizáljuk azokat. Ugyancsak terv a vékonyító eljárások összehasonlító elemzése alkalmasan választott tesztképekre és azok zajjal terhelt változataira is.

2.1. Irány–szekvenciális módszerek

Az irány–szekvenciális 3D vékonyító algoritmusoknál egy iterációs lépés tevékenysége több egymást követő *al-iteráció*ból áll. Az *al-iteráció* olyan különböző lokális és párhuzamos redukciókkal adható meg, amelyek csak a képpontok $3 \times 3 \times 3$ -as környezete alapján döntenek azok sorsáról és csak bizonyos típusú/irányú határpontokat törölhetnek.

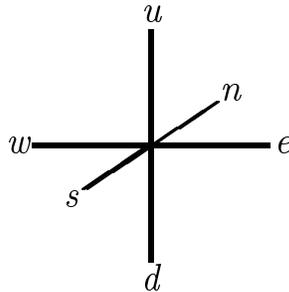
A k *al-iterációs* ($k \geq 2$) módszer tehát az alábbi általános sémával írható le:

```
repeat
  for  $i = 1$  to  $k$  do
    egyidejű törlése mindazon határpontoknak, amelyek
    kielégítik az  $i$ -edik irányhoz tartozó feltételt
until nem történt változás
```

A $(26, 6)$ képeken 6-féle határpont létezik, a 6-szomszédoknak megfelelően. A $\mathcal{P} = (\mathbb{Z}^3, 26, 6, B)$ képen a p pont *u-határpont*, ha $p \in B$ és $u(p) \in (V \setminus B)$ (1.4. ábra). Az *n-*, *e-*, *s-*, *w-* és *d-határpontok* hasonlóképpen definiálhatók.

A 3D irány–szekvenciális vékonyító algoritmusok általában egy iterációs lépést 6 *al-iteráció*ra bontanak fel, amelyek törlési irányait a 6-féle határponthoz rendelik (2.2. ábra).

Megjegyezzük, hogy az irány–szekvenciális algoritmusok eredménye függ az irányok bejárásától, az *al-iterációk* sorrendjétől. Az *al-iterációk* különféle permutációi különböző vékonyító algoritmusokhoz vezetnek.



2.2. ábra A 6-féle határponthoz rendelt törlési irányok, amelyek egybeesnek a 3D koordináta-rendszer tengelyeivel.

Az irány–szekvenciális algoritmusok fontos jellemzője az, hogy többféle törlési feltételt alkalmaznak: minden egyes *al-iteráció*hoz más és más törlési szabály tartozik. A törlési szabályok viszont egymásból könnyen származtathatók változóik permutálásával, az egyes változókhöz tartozó pozícióknak a vizsgált képpont koordinátáira történő tükrözésével és/vagy elforgatásával. Más szavakkal: az algoritmus tetszőleges két *al-iteráció*jának megfelelő \mathcal{T}_1 és \mathcal{T}_2 képműveletekhez található olyan \mathcal{G}_{12} geometriai transzformáció, hogy bármely \mathcal{P} képre:

$$\mathcal{T}_1(\mathcal{P}) = \mathcal{T}_2(\mathcal{G}_{12}(\mathcal{P})),$$

mégpedig \mathcal{G}_{12} pontosan az a transzformáció, ami a törlési irányok vektorait egymásba viszi.

Az irány–szekvenciális vékonyítás — mint valamennyi vékonyító technika — megőrzi a topológiát és az objektum határának körbejárása (a törlési irányokból) képes biztosítani

a “váz” pozíciójának korrektségét, azt, hogy a vékonyított objektum az eredeti “közepén” helyezkedjen el (± 1 voxel pontossággal). A módszer előnye még, hogy nem érzékeny az objektum pozíciójára (eltolás-invariáns), ugyanakkor nem garantált az, hogy teljesül a vékonyító algoritmusokkal szemben támasztott *gyengített izotrópia kritérium* (Manzanera et al., 1999/1), miszerint a vékonyítás és az elforgatás legyen felcserélhető legalább azokra a forgatási szögekre, amelyek a 90° többszörösei. Ez utóbbi (hátrányos) tulajdonság abból fakad, hogy a módszer érzékeny a törlési irányok bejárési sorrendjére. A fentiek szerint tehát az irány-szekvenciális technika kielégíti a vázkijelölés geometriai kritériumának első, de nem teljesíti maradéktalanul annak második felét.

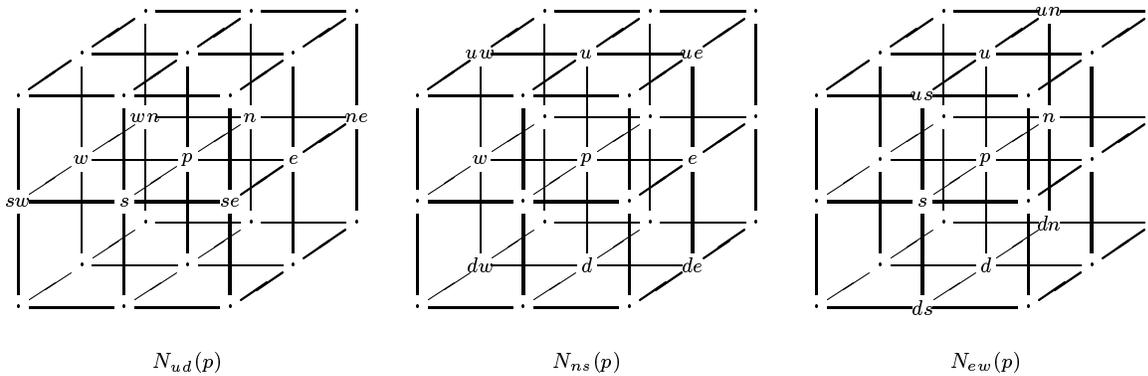
A következőkben nyolc irány-szekvenciális vékonyító algoritmust ismertetünk. Az első hat (2.1.1.–2.1.6. pont) a hagyományos hat al-iterációs megközelítéssel él. Közülük öt eljárást idegen szerzők javasoltak, a hatodik pedig a sajátunk. A 2.1.7. pontban a nyolc, a 2.1.8. pontban pedig a tizenkettő al-iterációs eljárást mutatjuk be.

2.1.1. Tsao és Fu 6 al-iterációs algoritmus

Tsao és Fu fejlesztették ki az első 6 al-iterációs 3D irány-szekvenciális vékonyító algoritmust (Tsao & Fu, 1981; Tsao & Fu, 1982). Eljárásuk egyaránt alkalmas a középfelszín és a középvonalak meghatározására. A középvonalak kivonására 2-fázisú módszert javasoltak: az első fázis során a kiterjedt bináris objektumokból a középfelszín vonják ki, majd a második fázisban (“gyengébb” törlési feltételek mellett) a középfelszín vékonyításával határozzák meg a középvonalakat. (Középfelszínre történő vékonyítás esetén a második fázis elmarad.)

Az algoritmust (26, 6) 3D képekre fejlesztették ki. A törlési feltételek a fekete pontok $3 \times 3 \times 3$ -as környezetét vizsgálják és a 2.2. ábrán bemutatott törlési irányok tartoznak a 6 al-iterációhoz.

A törölhető pontokat 2D *ellenőrző ablakok* (*checking window*) segítségével jelölik ki. A $3 \times 3 \times 3$ -as környezet három ellenőrző ablakát, vagyis az $N_{ud}(p)$, az $N_{ns}(p)$ és az $N_{ew}(p)$ halmazokat a 2.3. ábra mutatja be.



2.3. ábra A p pont $3 \times 3 \times 3$ -as környezetének ellenőrző ablakai. (Az egyes pontok elnevezései összhangban vannak az 1.4. ábra jelöléseivel, csak a “ p ” argumentumokat hagytuk el.) Az $N_{ud}(p)$ ablak síkja merőleges az u és a d törlési irányokra (2.2. ábra). Hasonlóképpen: $N_{ns}(p)$ és $N_{ew}(p)$ síkjai merőlegesek az n és az s , illetve az e és a w irányokra.

Az u törlési irányhoz tartozó al-iteráció során a “ p ” u -határpont akkor törölhető, ha az alábbi feltételek mindegyike teljesül:

1. “ p ” pontosan egy 26-objektummal 26-szomszédos $N_{26}(p)$ -ben.
2. “ p ” az $N_{ns}(p)$ és az $N_{ew}(p)$ halmazoknak csak egy-egy 26-objektumával 26-szomszédos.
3. Az $N_{ns}(p)$ és az $N_{ew}(p)$ halmazok legalább kettő-kettő fekete pontot tartalmaznak.

A fenti feltételek a középfelszín kivonása esetén érvényesek. A középvonalak meghatározásakor elegendő, ha “ p ” nem vonalvégpont, így a 3. kritérium elhagyható.

Az eljárás különlegessége, hogy a 3D-ben való törölhetőséghez 2D-s feltételeket fogalmaz meg és a d irányhoz is a fenti kritériumok tartoznak (tehát az implementáció során három feltételrendszert kell lekódolni).

Az algoritmus hátránya, hogy jóllehet a fenti feltételek a (26, 6)-egyszerű pontok egy valódi részhalmazát adják, a párhuzamos eljárás nem őrzi meg a topológiát, mivel az “egy voxel vékonyságú” ferde felszínszegmensek törlésével szétszakíthat vagy teljesen

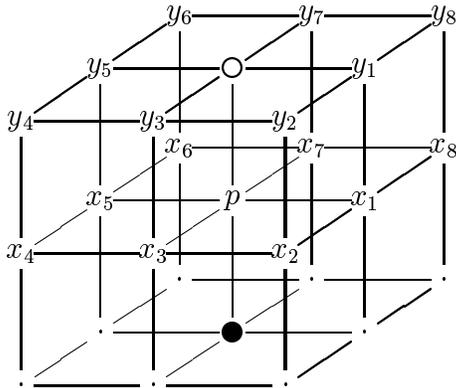
eltüntethet objektumokat (Lee, Kashyap, Chu, 1994). Emellett az eljárás nem képes bizonyos képrészletek maximális vékonyítására sem (Gerig et al., 1993).

Kritika tárgyát képezheti a középvonalak kinyerésére javasolt kétfázisú módszer is, miszerint az első fázis során a középfelszín határozzák meg, majd azt vékonyítják tovább a második fázisban. Zajos objektum esetén annak határán számos felszínvégpont is előfordulhat, amelyekből a középfelszín kivonása során szükségképpen nemkívánatos felszín-szegmensek keletkeznek, azokból pedig a második fázis parazita ágakat növeszt (Palágyi, Kuba, 1997/1). A középvonalak kivonására ígéretesebb a direkt módszer, amikor minden iterációs lépés törlési feltétele a vonalvégpontokat őrzi meg.

2.1.2. Gong és Bertrand 6 al-iterációs algoritmus

Gong és Bertrand egy olyan egyszerű 6 al-iterációs algoritmust javasolt középfelszín kivonására, ami a 3D vékonyító algoritmusokkal szemben támasztott valamennyi követelménynek eleget tesz (Gong, Bertrand, 1990).

A (26, 6) 3D képekre működő algoritmus az u irányhoz tartozó al-iteráció során törli mindazon “ p ” fekete pontokat, amelyek nem “felszínvégpontok” és teljesül rájuk az alábbi feltétel:



$$\begin{aligned} \forall i \quad i \in \{1, 3, 5, 7\} : \\ & (x_i = 0) \Rightarrow (y_i = 0), \\ \forall j \quad j \in \{2, 4, 6, 8\} : \\ & (x_{j-1} = x_j = x_{(j+1) \bmod 8} = 0) \Rightarrow (y_j = 0). \end{aligned}$$

A szerzők csak azt igazolták, hogy az algoritmusuk kizárólag (26,6)-egyszerű pontokat töröl. A párhuzamos eljárás topológia megőrzőnek bizonyult, mivel teljesíti a C.M. Ma által (hat évvel később publikált), a 6-aliterációs párhuzamos vékonyító algoritmusokra felállított kritériumokat (Ma, 1996).

Gong és Bertrand sajátos végpontkritériumot javasolt, mely megközelítés motivációját Lee, Kashyap és Chu homályosnak tartják (Lee, Kashyap, Chu, 1994). Valamely “ p ” fekete pont *nem-felszínvégpont*, ha $N_{26}(p)$ legalább nyolc fekete pontot tartalmaz, vagy legalább négyet, de ekkor “ p ” $3 \times 3 \times 3$ -as környezetének valamely $2 \times 2 \times 2$ -es részkockájában (a nyolc ilyen egység-rácskocka van) mindhárom p -vel 6-szomszédos pontnak feketének kell lennie.

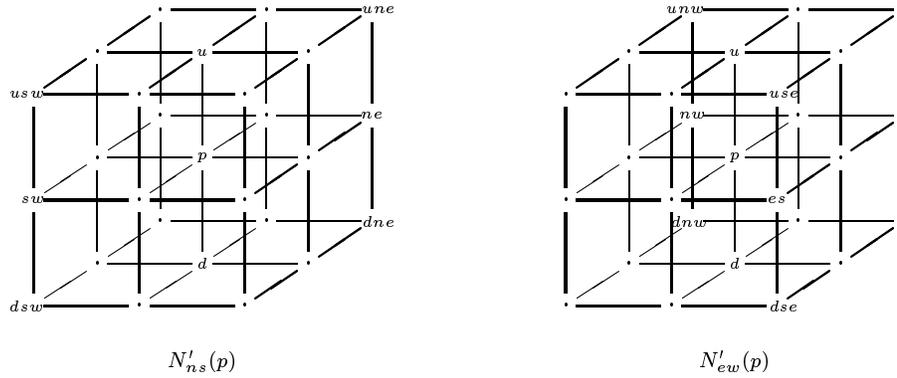
A fenti végpontkritérium helyett más feltételek is alkalmazhatók, így újabb középfelszín vagy közép vonalakat kivonó vékonyító algoritmusokhoz juthatunk. Gong és Bertrand algoritmusát Rolland, Chassery és Montanvert módosították (Rolland, Chassery, Montanvert, 1992). Eljárásukban a felszínvégpontok megőrzésére vonatkozó kritériumot a vonalvégpontok törlésének tilalmára változtatták és az így kapott, a közép vonalak meghatározására szolgáló algoritmus eredményét *drótkeret váznak* (*wireframe skeleton*) nevezték el¹³.

¹³A 3D felszín *drótváz* (*wireframe*) megjelenítése (Foley et al., 1991) nem hozható kapcsolatba a módosított algoritmus eredményével.

2.1.3. Mukherjee, Das és Chatterjee 6 al-iterációs algoritmus

Mukherjee, Das és Chatterjee a középfelzínre vékonyító algoritmus (Mukherjee, Das, Chatterjee, 1990) — hasonlóan a 2.1.1. pontban felvázolthoz — 3×3 -as 2D ellenőrző ablakokat használ.

Az u irányhoz rendelt al-iteráció törlési feltétele 2.3. ábrán látható $N_{ns}(p)$ és az $N_{ew}(p)$ ellenőrző ablakok mellett kettő további ablakot is figyel. Az $N'_{ns}(p)$ és az $N'_{ew}(p)$ ferde ablakokat rendre $N_{ns}(p)$ -nek és $N_{ew}(p)$ -nek az ud -koordinátatengely (2.2. ábra) körüli 45° -os elforgatásával kapjuk meg (2.4. ábra).



2.4. ábra A p pont $3 \times 3 \times 3$ -as környezetének kettő ferde ellenőrző ablaka. (Az egyes pontok elnevezései összhangban vannak az 1.4. ábra jelöléseivel, csak a “ p ” argumentumokat hagytuk el.) Az $N'_{ns}(p)$ és az $N'_{ew}(p)$ ablakok síkjai párhuzamosak az u és a d törlési irányokkal.

Az u irányhoz rendelt al-iteráció során a “ p ” u -határpont akkor törölhető, ha az alábbi feltételek mindegyike teljesül:

1. “ $d(p)$ ” fekete pont.
2. “ p ” az $N_{ns}(p)$ és az $N_{ew}(p)$ halmazoknak csak egy-egy 26-objektumával 26-szomszédos.
3. “ p ” az $N'_{ns}(p)$ és az $N'_{ew}(p)$ halmazoknak csak egy-egy 26-objektumával 26-szomszédos.

Ma és Sonka megmutatta, hogy az eljárás csak $(26,6)$ -egyszerű pontokat minősít törölhetőnek, ráadásul a párhuzamos törlés is topológia-megőrző a $(26,6)$ 3D képekre (Ma, Sonka, 1996).

2.1.4. Lee, Kashyap és Chu 6 al-iterációs algoritmus

Lee, Kashyap és Chu középfelszín vagy középvonalakot kijelölő algoritmus minden komponensében különleges (Lee, Kashyap, Chu, 1994). Az u törlési irányhoz tartozó al-iteráció a következő két fázisból áll:

1. Valamennyi $(26,6)$ -egyszerű u -határpont és "nem-végpont" megjelölése.
2. Szekvenciális (!) újraellenőrzés dönt a megjelölt pontok törléséről vagy meghagyásáról.

A $(26,6)$ -egyszerű pontokat a 1.3. Kritérium alapján ismerik fel, azaz a kérdéses pont $3 \times 3 \times 3$ -as környezetében (magát a pontot nem beleértve) egy fekete objektum található és a pont törlése nem változtatja meg a $3 \times 3 \times 3$ -as részképen az Euler-karakterisztikát.

A $3 \times 3 \times 3$ -as környezetbe eső fekete pontok 26-összefüggősége eldöntésére egy olyan címkéző eljárást javasoltak, ami egy speciális fa-struktúrát használ. A kettő mélységű fa gyökeréből nyolc él indul ki a $3 \times 3 \times 3$ -as környezet nyolc $(2 \times 2 \times 2)$ -es egység-rácskockájával címkézett szögpontokba, amelyekből a további élek az egyes rácskockák pontjaival címkézett nyolc csúcsba (a fa leveleibe) mutatnak. Az Euler-karakterisztika számítása a $3 \times 3 \times 3$ -as környezet egység-rácskockáinak pontjaiból képzett 8-bites számokkal megcímezhető keresőtáblákkal történik. A javasolt, meglehetősen bonyolult végpontkritérium szintén az egység-rácskockák vizsgálatával történik.

A szekvenciális újraellenőrzés során azok a megjelölt pontok törlődnek, amelyeknek (az esetleg már megváltoztatott) $3 \times 3 \times 3$ -as környezetében (magát a pontot nem beleértve) továbbra is csak egyetlen fekete objektum található.

A szerzők bizonyították algoritmusuk topológia-megőrző voltát és kemény, helyenként igazságtalan kritikával illették a korábban közölt módszereket és az egyszerű pontokra adott jellemzéseket¹⁴.

Lee, Kashyap és Chu algoritmusát alkalmazta Ge, Stelts és Vining (Ge, Stelts, Vining, 1996). Gyorsították a vizsgált pont $3 \times 3 \times 3$ -as környezetére adott objektumszámláló eljárást és a vékonyító módszert távolság-transzformációval kombinálták, ahol a távolság-transzformációval kapott vázpontok "horgonyként" szerepelnek a vékonyítás során.

¹⁴Lee, Kashyap és Chu hibásnak minősítette Malandain és Bertrand $(26,6)$ -egyszerű pontokra adott kritériumát (Malandain, Bertrand, 1992). A cikkre válaszolva a megkritizált szerzők rámutattak Lee, Kashyap és Chu érvelésének megalapozatlanságára (Bertrand, Malandain, 1995).

2.1.5. Bertrand 6 al-iterációs algoritmus

Bertrand algoritmus az ismert 3D vékonyító algoritmusok közül egyedülálló abban, hogy (6, 26) képeket vesz figyelembe (Bertrand, 1995/2).

Az eljárás — ugyanúgy, mint a 2.1.1.–2.1.4. pontokban vázoltak — a 2.2. ábra szerinti hat törlési irányt veszi figyelembe. A határpontok, a “nem-felszínvégpontok” és a (6, 26)–egyszerű pontok kijelölése egyaránt Boole-feltételekkel történik.

Az algoritmus bizonyítottan megőrzi a topológiát a (6, 26) képekre.

Az egyszerű pontok jellemzésére Bertrand kétféle *lokális topológiai számot* (*local topological number*) vezetett be (Bertrand, 1996), amelyeket egy-egy Boole-feltétellel írt le. A nem-felszínvégpontokat újabb két Boole-feltétellel adta meg. Robert és Malandain a fenti négy feltételre meghatározták a redukált bináris döntési diagramot (BDD, ld. 1.4.4.) és C-kódot is generáltak (Robert, Malandain, 1998).

2.1.6. Egy 6 al-iterációs algoritmus

A jelen pontban egy középvonalak és középfelszínek kivonására egyaránt alkalmas, (26,6) képekre kidolgozott vékonyító algoritmust ismertetünk (Palágyi, Kuba, 1997/3; Palágyi, Kuba, 1998/1).

Az eljárást az alábbi sémával írjuk le:

Input: a vékonyítandó (26,6) képet reprezentáló X bináris tömb.

Output: a vékonyított képet tároló Y bináris tömb.

6_al-iterációs_vékonyító_algoritmus(X, Y)

```
begin
  Y = X;
  repeat
    Y = törlés_u_irányból(Y);
    Y = törlés_d_irányból(Y);
    Y = törlés_n_irányból(Y);
    Y = törlés_s_irányból(Y);
    Y = törlés_e_irányból(Y);
    Y = törlés_w_irányból(Y);
  until nem történt változás;
end.
```

Az alkalmazott törlési irányok ugyanazok, mint amiket a konvencionális irány-szekvenciális 3D vékonyító algoritmusok használnak (2.2. ábra).

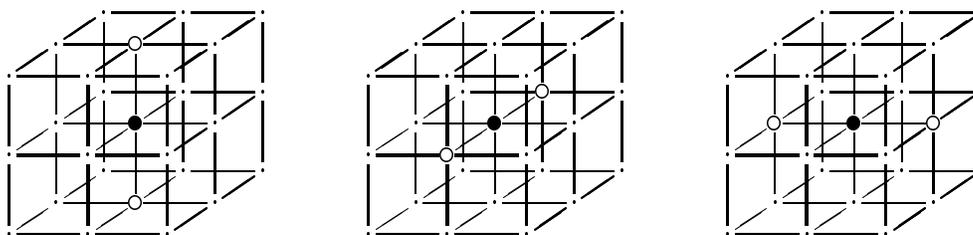
Az eljárás az alábbi végpontkritériumokat alkalmazza:

2.1. DEFINÍCIÓ

A $p \in B$ pont akkor és csakis akkor *vonalvégpont* a $(\mathbb{Z}^3, 26, 6, B)$ képen, ha az $(N_{26}(p) \cap B) \setminus \{p\}$ halmaz egyelemű, vagyis p csak egyetlen másik fekete ponttal 26-szomszédos.

2.2. DEFINÍCIÓ

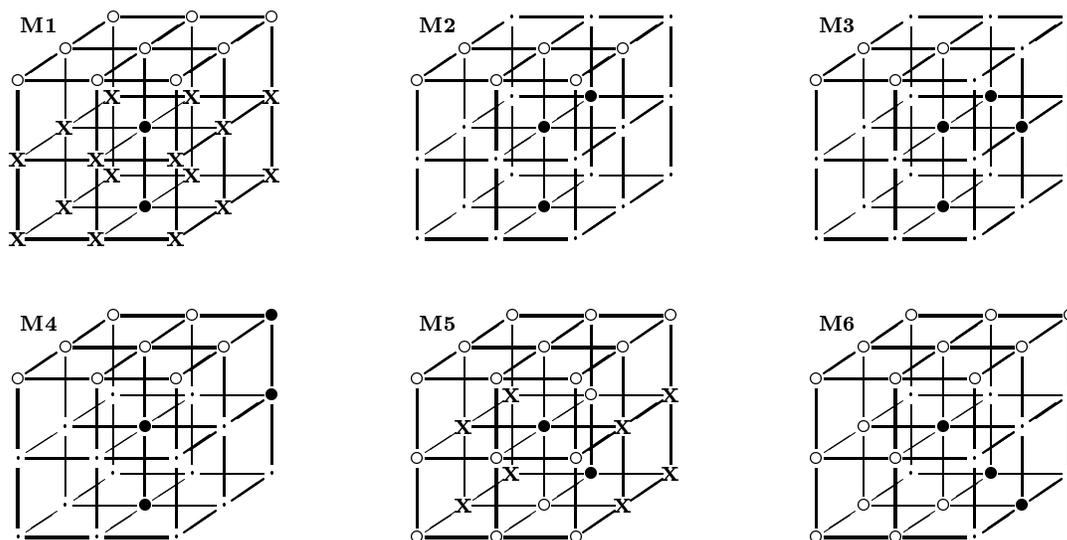
A $p \in B$ pont akkor és csakis akkor *felszínvégpont* a $(\mathbb{Z}^3, 26, 6, B)$ képen, ha a p ellentétes 6-szomszédjaiból alkotott 3 pontpár — $(u(p), d(p))$, $(n(p), s(p))$ és $(e(p), w(p))$ — közül legalább egy fehér pontokat tartalmaz. (Megjegyzendő, hogy valamennyi vonalvégpont egyben felszínvégpont is. A felszínvégpontok konfigurációi a 2.5. ábrán láthatók.)



2.5. ábra A 2.1. Definícióval jellemzett felszínvégpontokhoz tartozó három konfiguráció. Minden “•” pozíció fekete és minden “o” fehér. A “,” (“don’t care”) pozíciók színe tetszőleges.

A középvonalra vékonyító eljárásnak az u törlési irányhoz rendelt lokális párhuzamos redukcióját (törlés_u_irányból) a 2.6. ábra maszkjaival adjuk meg. Megjegyezzük, hogy az ábra hat alapmaszkjának a függőleges tengely körül 90° , 180° , 270° fokkal elforgatott változatai is a maszkkészlethez tartoznak. A fenti maszkokkal adott képművelet — az 1.4.4. pontban leírtak szerint — törli mindazon fekete pontokat, amelyekre a maszkkészlet legalább egy tagja illeszkedik.

A fennmaradó öt al-iterációhoz tartozó képművelet maszkjai megkaphatók az u törlési irányhoz rendelt maszkok megfelelő elforgatásával és/vagy tükrözésével.



2.6. ábra A középvonalas vékonyító eljárásnak az u törlési irányhoz tartozó alapmaszkjai. Az **M1** és az **M5** maszkoknál az “x”-szel jelölt pozíciók közül legalább egy “•” (fekete). Az **M1–M6** maszkoknak a függőleges tengely körül 90° , 180° , 270° fokkal elforgatott változatai is a maszkkészlethez tartoznak. Könnyen ellenőrizhető, hogy a készlet minden egyes eleme u -határpontra illeszkedik és nem töröl vonalvégpontot (2.1. Definíció).

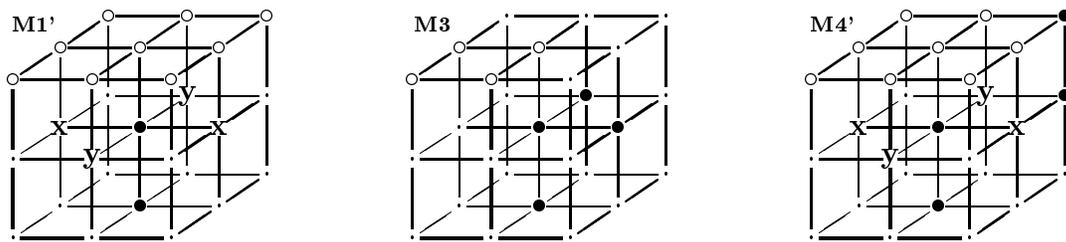
A középfelületre vékonyító algoritmust a középvonalat kivonóból a maszkkészletek módosításával terveztük meg. A középfelület meghatározó vékonyítás u törlési irányához tartozó maszkjait a 2.7. ábra mutatja be. A többi al-iteráció maszkjai megkaphatók az u törlési irányhoz rendelt elforgatásával és/vagy tükrözésével.

A fenti vékonyító eljárások topológiailag korrektek, vagyis megfogalmazható az alábbi tétel:

2.3. TÉTEL

A középvonalakat kivonó és a középfelület meghatározó 6 al-iterációs vékonyító eljárások megőrzik a topológiát a (26,6) képekre.

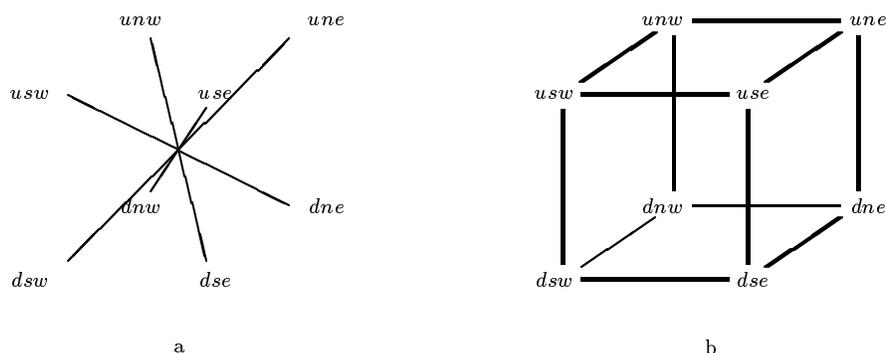
A fenti állítást részletesen bizonyítottuk — az 1.7. Tétel feltételeit alkalmazva — a középvonalakat kivonó eljárásra (Palágyi, Kuba, 1998/1). Könnyen belátható, hogy a középfelület meghatározó vékonyító algoritmus minden egyes képművelete kevesebb fekete pontot képes törölni, mint a középvonalat kivonó változat megfelelő lokális és párhuzamos redukciója. Ennélfogva a középfelületre vékonyító algoritmus is megőrzi a topológiát a (26,6) képekre.



2.7. ábra A középfelületre vékonyító eljárásnak az u törlési irányhoz tartozó alapmaszkjai. A maszkoknak a függőleges tengely körül 90° , 180° , 270° fokkal elforgatott változatai is a maszkkészlethez tartoznak. Az **M1'** és az **M4'** maszk a 2.6. ábra **M1** és **M4** maszkjainak módosításával keletkezett, míg az **M3** maszk változtatás nélkül került át. Az **M1'** és az **M4'** maszkoknál legalább egy "x" és egy "y" pozíció fekete. Könnyen ellenőrizhető, hogy a maszkkészlet minden egyes eleme u -határpontra illeszkedik és nem töröl felszínvégpontot (2.2. Definíció).

2.1.7. Egy 8 al-iterációs algoritmus

A 2.1.1–2.1.6. pontokban bemutatott valamennyi 3D vékonyító eljárás — a 2.2. ábrán bemutatott törlési irányokkal — 6 al-iterációt alkalmaz iterációs lépésenként. Most egy olyan középvonalak és középfelszínek meghatározására egyaránt alkalmas, (26,6) képekre kidolgozott vékonyító algoritmust ismertetünk, melynek egy iterációs lépése 8 al-iterációból áll (Palágyi, Kuba, 1999/1). A konvencionális 6 törlési irány megfeleltethető egy kocka 6 lapjának, ahol az egyes irányok egybeesnek a lapok (mint síkok) normálisaival, míg a 8 al-iterációhoz tartozó törlési irányok hozzárendelhetők egy kocka 8 csúcsához (pontosabban: az egyes csúcsokat és a kocka középpontját összekötő vektorokhoz). Az új törlési irányokat — amelyek jelölése: *usw*, *dne*, *use*, *dnw*, *une*, *dsw*, *unw*, *dse* — a 2.8. ábra szemlélteti.



2.8. ábra A törlési irányok modellezése. Míg a szokásos 6 irány egy $p \in \mathbb{Z}^3$ pont $N_6(p)$ halmazához tartozik, addig a 8 törlési irány (a) hozzárendelhető a $N_{26}(p) \setminus N_{18}(p)$ halmazhoz. A 8 törlési irányt megfeleltethetjük egy kocka 8 csúcsának is (b). A jelölés konzekvens, hiszen az *usw* csúcson az *u*, az *s* és a *w* irányokhoz tartozó lapok osztoznak.

Az eljárás az alábbi program szerint működik:

Input: a vékonyítandó (26,6) képet reprezentáló X bináris tömb.

Output: a vékonyított képet tároló Y bináris tömb.

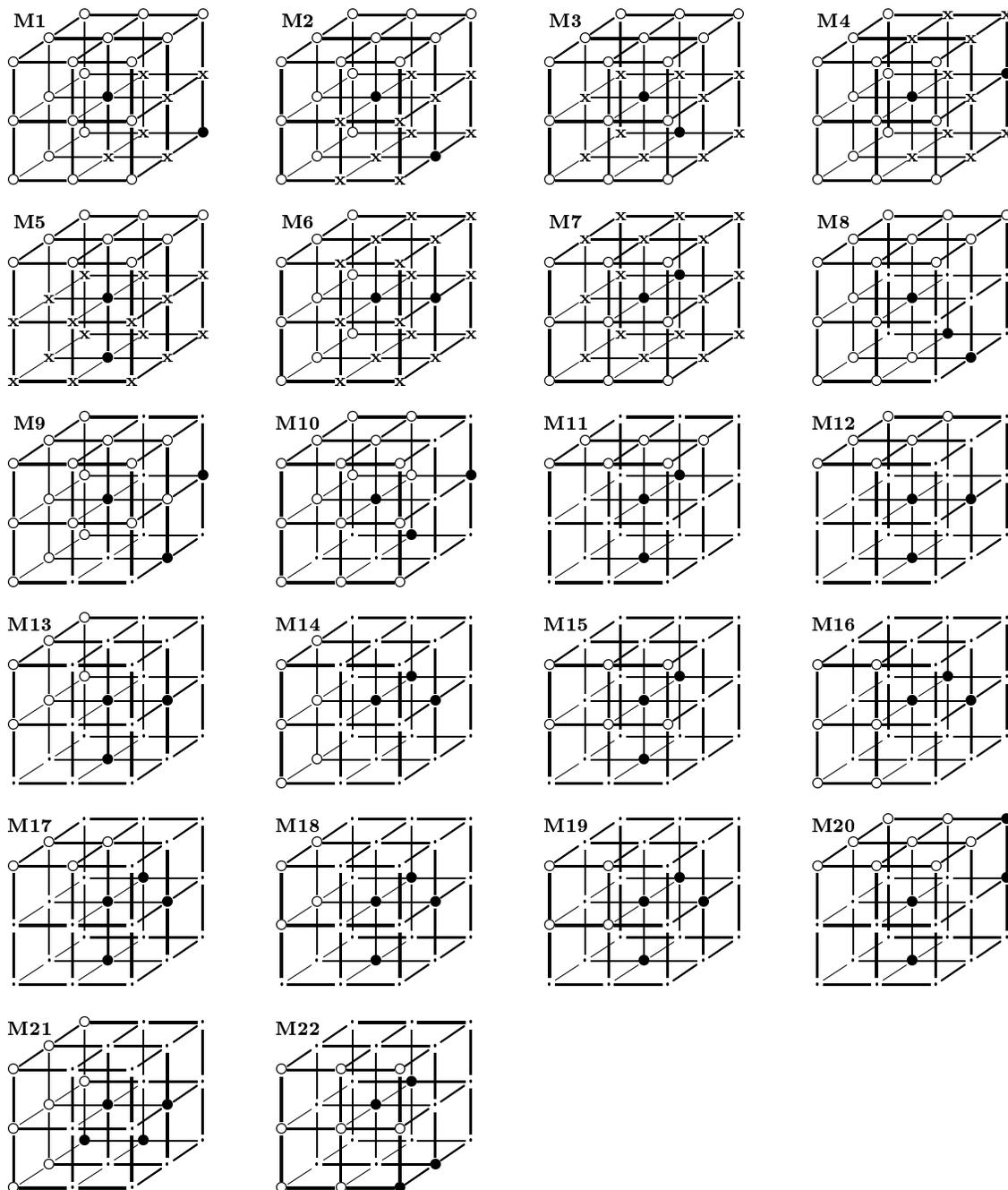
`8_al-iterációs_vékonyító_algoritmus(X, Y)`

```
begin
   $Y = X$ ;
  repeat
     $Y = \text{törlés\_usw\_irányból}(Y)$ ;
     $Y = \text{törlés\_dne\_irányból}(Y)$ ;
     $Y = \text{törlés\_use\_irányból}(Y)$ ;
     $Y = \text{törlés\_dnw\_irányból}(Y)$ ;
     $Y = \text{törlés\_une\_irányból}(Y)$ ;
     $Y = \text{törlés\_dsw\_irányból}(Y)$ ;
     $Y = \text{törlés\_unw\_irányból}(Y)$ ;
     $Y = \text{törlés\_dse\_irányból}(Y)$ ;
  until nem történt változás;
end.
```

Megjegyezzük, hogy a **repeat**-ciklus magjában szereplő párhuzamos redukciók sorrendjének minden egyes permutációja egy-egy újabb algoritmushoz vezet. A fent javasolt sorrend egyfajta szimmetriát mutat, ami biztosítja azt, hogy a vékonyítás eredménye valóban az objektumok közepén helyezkedjen el.

Az *usw* törlési irányhoz tartozó képművelet (törlés_*usw*_irányból) bizonyos *u*-, *s*- és *w*-határpontokat töröl, a *dne* irány párhuzamos redukciója (törlés_*dne*_irányból) pedig bizonyos *d*-, *n*- és *e*-határpontokat hámoz le az objektumokról, és így tovább.

A középvonalra vékonyító eljárásnak az *usw* törlési irányhoz rendelt lokális párhuzamos redukcióját a 2.9. ábra maszkjaival adjuk meg.

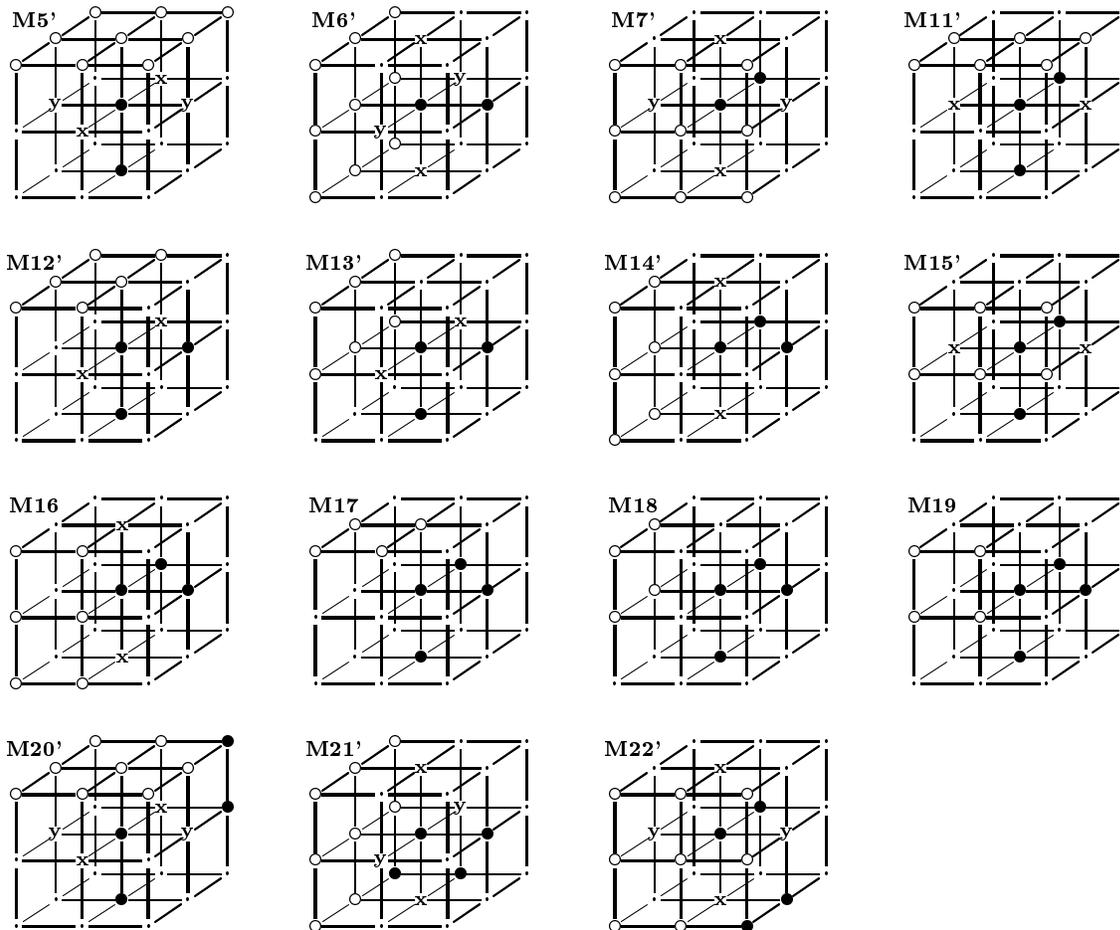


2.9. ábra A középvonalra vékonyító eljárás *usw* törlési irányához tartozó maszkok. A vonalvégpontok megőrzése érdekében az M1–M7 maszkoknál az “x” pozíciók közül legalább egy “•” (fekete).

A fenti maszkokkal adott képművelet — az 1.4.4. pontban leírtak szerint — törli mindazon fekete pontokat, amelyekre legalább egy maszk illeszkedik a maszkkészletből. Könnyen ellenőrizhető, hogy a maszkok csak u -, s - vagy w -határpontokra illeszkednek és nem törölnek vonalvégpontot (2.1. Definíció).

A fennmaradó hét al-iterációhoz tartozó képművelet maszkjai megkaphatók az usw törlési irányhoz rendelt maszkok megfelelő elforgatásával és/vagy tükrözésével.

A középfelületre vékonyító algoritmust a középvonalat kivonó eljárásból a maszkkészlet módosításával kaptuk. A középfelületre vékonyítás usw törlési irányához tartozó maszkjait a 2.10. ábra mutatja be.



2.10. ábra A középfelületre vékonyító eljárásnak a usw törlési irányhoz tartozó maszkjai. Az **M5'–M15'** és az **M20'–M22'** maszkok rendre a 2.9. ábra **M5–M15** és **M20–M22** maszkjainak módosításával álltak elő, míg az **M16–M19** maszkok változtatás nélkül kerültek át. A maszkokban legalább egy “**x**” és egy “**y**” pozíció fekete. Könnyen ellenőrizhető, hogy a maszkkészlet minden egyes eleme u -határpontra illeszkedik és nem töröl felszínvégpontot (2.2. Definíció).

Az ismertetett vékonyító eljárások topológiailag korrektek, amit az alábbi tétellel fogalmaztunk meg:

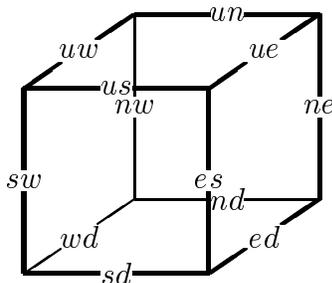
2.4. TÉTEL

A középvonalat kivonó és a középfelszint meghatározó 8 al-iterációs vékonyító eljárások megőrzik a topológiát a $(2\epsilon, 6)$ képekre.

A fenti állítást — az 1.7. Tétel feltételeinek segítségével — részletesen bizonyítottuk (Palágyi, Kuba, 1999/1).

2.1.8. Egy 12 al-iterációs algoritmus

A jelen pontban egy középvonalak és középfelzárk megdhatározására egyaránt alkalmas, (26,6) képekre kidolgozott vékonyító algoritmust ismertetünk (Palágyi, Kuba, 1997/1; Palágyi, Kuba, 1997/2; Palágyi, Kuba, 1999/2). Az eljárás különlegessége az, hogy 12 törlési irányt feltételez, vagyis egy iterációs lépésben 12 al-iterációt alkalmaz. A törlési irányok megfelelnek egy kocka 12 élének, pontosabban: az élék felezőpontjait a kocka középpontjával összekötő vektoroknak (2.11. ábra), vagy hozzárendelhetők egy $p \in \mathbb{Z}^3$ pont $N_{18}(p) \setminus N_6(p)$ halmazának 12 eleméhez.



2.11. ábra A figyelembe vett törlési irányok, amelyek egy kocka 12 élfelező pontját a kocka középpontjával összekötő vektorokhoz rendelhetők.

Az eljárást az alábbi séma írja le:

Input: a vékonyítandó (26,6) képet reprezentáló X bináris tömb.

Output: a vékonyított képet tároló Y bináris tömb.

12_al-iterációs_vékonyító_algoritmus(X, Y)

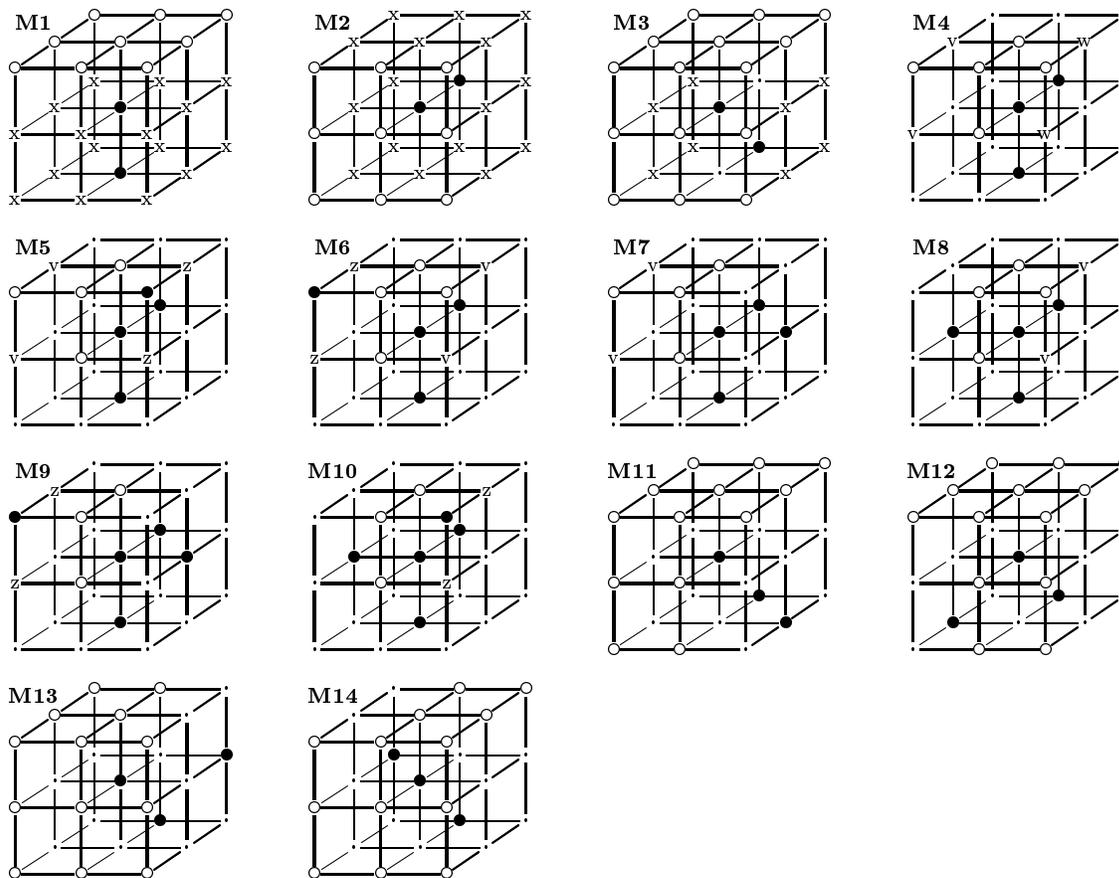
```
begin
  Y = X;
  repeat
    Y = törlés_us_irányból(Y);
    Y = törlés_ne_irányból(Y);
    Y = törlés_wd_irányból(Y);
    Y = törlés_es_irányból(Y);
    Y = törlés_uw_irányból(Y);
    Y = törlés_nd_irányból(Y);
    Y = törlés_sw_irányból(Y);
    Y = törlés_un_irányból(Y);
    Y = törlés_ed_irányból(Y);
    Y = törlés_nw_irányból(Y);
    Y = törlés_ue_irányból(Y);
    Y = törlés_sd_irányból(Y);
  until nem történt változás;
end.
```

Megjegyezzük, hogy a repeat-ciklus magját alkotó párhuzamos redukciók sorrendjét megváltoztatva egy újabb algoritmushoz jutunk. A fenti sorrend egyfajta szimmetriát

mutat, ami biztosítja azt, hogy a vékonyítás eredménye valóban az objektumok közepén helyezkedjen el. A 12 képművelet 4 olyan csoportból áll, amelyek mindegyike 3 olyan párhuzamos redukciót tartalmaz, amelyekhez rendelt kockaéleken megosztozik a kockának mind a 6 lapja.

Az us törlési irányhoz tartozó képművelet (törlés_us_irányból) bizonyos u - és s -határpontokat töröl, az ne irány párhuzamos redukciója (törlés_ne_irányból) pedig bizonyos n - és e -határpontokat hámoz le az objektumokról, és így tovább.

A középvonalra vékonyító eljárásnak az us törlési irányhoz rendelt lokális párhuzamos redukcióját a 2.12. ábra maszkjaival adjuk meg. Az eljárás maszkjai nem törölnek vonalvégpontokat (2.1. Definíció), mivel minden egyes maszk csak olyan fekete pontra illeszkehdhet, aminek legalább kettő fekete 26–szomszédja van.



2.12. ábra A középvonalra vékonyító eljárás us törlési irányához tartozó maszkok.

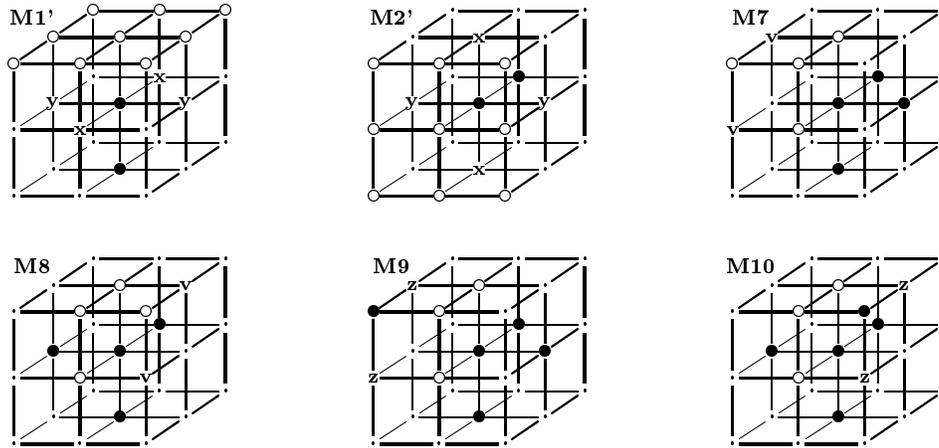
Jelölés: legalább egy “ x ” pozíció színe fekete, legalább egy “ v ” pozíció fehér, legalább egy “ w ” pozíció fehér és a két “ z ”-vel jelölt pozíció különböző színű (az egyik fekete, a másik pedig fehér).

A fennmaradó 11 al-iterációhoz tartozó képművelet maszkjai megkaphatók az us törlési irányhoz rendelt maszkok megfelelő elforgatásával és/vagy tükrözésével.

A középfelzínre vékonyító algoritmust a középvonalasból a maszkkészletek módosításával származtattuk. A középfelzínés vékonyítás us törlési irányához tartozó maszkjait a 2.13. ábra mutatja be.

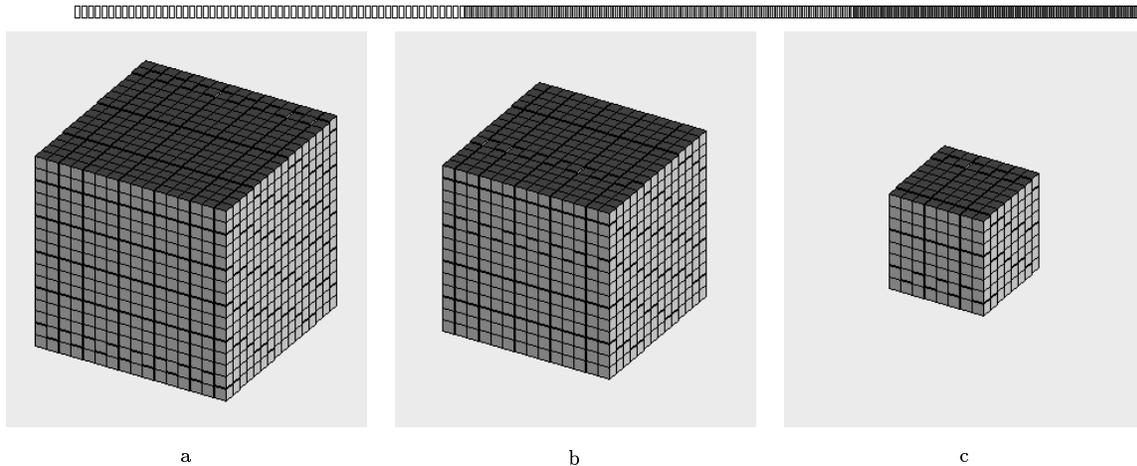
A 2.13. ábra maszkjai nem törölnek felszínvégpontokat (2.2. Definíció), amit az “ x ” és az “ y ” szimbólumok bevezetésével értünk el.

Úgy tűnhet, hogy a vékonyítás egy iterációs lépését nem érdemes 8 vagy 12 al-iterációra bontani, mivel akkor a konvencionális 6 al-iterációs algoritmusoknál lassabb eljárásokat kapunk. Ez nincs szükségképpen így. Igaz, hogy várhatóan egy iterációs lépés gyorsabb 6 al-iteráció esetén, de várhatóan több iterációra van akkor szükség. A 2.14. ábra példája azt mutatja, hogy bizonyos objektumokra a 8 és a 12 al-iterációs eljárások negyedannyi iterációs lépést igényelnek, mivel azok egy lépés során 4 “réteget” hámoznak le, míg a 6 al-iterációsok csak egyet.



2.13. ábra A középfelületre vékonyító eljárás *us* törlési irányához tartozó maszkok. Az **M1'–M2'** maszkok rendre a 2.12. ábra **M1–M2** maszkjainak módosításával álltak elő, míg az **M7–M10** maszkok változtatás nélkül kerültek át.

Jelölés: legalább egy “**x**” pozíció színe fekete, legalább egy “**y**” pozíció színe fekete, legalább egy “**v**” pozíció fehér és a két “**z**”-vel jelölt pozíció különböző színű (az egyik fekete, a másik pedig fehér).



2.14. ábra Irány-szekvenciális módszerek összehasonlítása. Az eredeti $16 \times 16 \times 16$ -os tömör kockát (a) a 6 al-iterációs algoritmus (2.1.6.) egy iterációs lépése $14 \times 14 \times 14$ -es kockára (b) vékonyítja le. A 8 al-iterációs eljárás (2.1.7.) és a 12 al-iterációs módszer (2.1.8.) egy iterációs lépése $8 \times 8 \times 8$ -as kockát (c) állít elő a kiindulási (a) objektumból.

Az ismertett vékonyító eljárások topológiailag korrektek, amit az alábbi tétel mond ki:

2.5. TÉTEL

A középvonalat kivonó és a középfelszint meghatározó 12 al-iterációs vékonyító eljárások megőrzik a topológiát a $(26,6)$ képekre.

A fenti állítást — az 1.7. Tétel feltételei teljesülését igazolva — részletesen bizonyítottuk (Palágyi, Kuba, 1999/2).

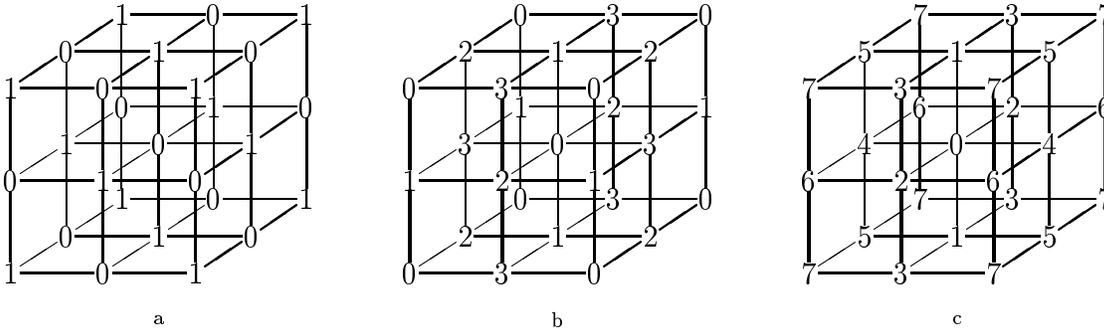
2.2. Almező–szekvenciális módszerek

A 2.1. pontban tárgyalt irány–szekvenciális vékonyító algoritmusok a topológia–megőrzés érdekében egy iterációs lépést olyan al–iterációkra bontanak, amelyek csak bizonyos típusú határpontokat törölhetnek. Az almező–szekvenciális eljárások az iterációs lépéseket más megközelítéssel szervezik meg. A diszkrét képrácsot $k \geq 2$ diszjunkt részhalmazra, almezőre osztják fel. A k al–iterációból álló iterációs lépés során az almezők felváltva aktivizálódnak és csak az aktív almezőbe eső, a törlési szabályokat kielégítő fekete pontok változnak meg.

Az irány–szekvenciális eljárásoknál mindegyik al–iterációhoz más és más törlési feltétel tartozik, míg az eddig közölt almező–szekvenciális vékonyító algoritmusok globális törlési szabályt alkalmaznak, vagyis minden egyes al–iteráció során ugyanaz a feltétel érvényes¹⁵. A k al–iterációs almező–szekvenciális eljárások az alábbi általános sémával írhatók le:

```
repeat
  for  $i = 1$  to  $k$  do
    egyidejű törlése mindazon fekete pontoknak, amelyek
    egyrészt az  $i$ -edik almezőbe esnek, másrészt kielégítik
    a (globális) törlési feltételt
until nem történt változás
```

3D-ben az egész koordinátájú rácspontok \mathbb{Z}^3 halmazára a 2.15. ábrával illusztrált 2-, 4- és 8-almezős felbontást javasolják.



2.15. ábra A \mathbb{Z}^3 halmaz felbontása 2 almezőre (a), 4 almezőre (b) és 8 almezőre (c). Az “ i ”-vel jelölt rácspontok az i -edik almezőhöz tartoznak ($i = 0, 1, \dots$).

A $\mathbb{Z}^3 = \bigcup_{i=0}^{k-1} S_i^k$ ($k \in \{2, 4, 8\}$) felbontások formálisan:

$$S_i^2 = \{ (x_1, x_2, x_3) \in \mathbb{Z}^3 \mid (x_1 + x_2 + x_3 \bmod 2) = i \},$$

$$S_i^4 = \{ (x_1, x_2, x_3) \in \mathbb{Z}^3 \mid (x_1 + 1 \bmod 2) \cdot [2 \cdot (x_2 \bmod 2) + (x_3 \bmod 2)] + (x_1 \bmod 2) \cdot [2 \cdot (x_2 + 1 \bmod 2) + (x_3 + 1 \bmod 2)] = i \},$$

$$S_i^8 = \{ (x_1, x_2, x_3) \in \mathbb{Z}^3 \mid 4 \cdot (x_1 \bmod 2) + 2 \cdot (x_2 \bmod 2) + (x_3 \bmod 2) = i \}.$$

¹⁵Személyes konzultáció során Christian Brechbühler (ETH, Zürich) egy olyan 2D vékonyító algoritmust mutatott be nekem, ami 4 almezőt alkalmaz és az egyes al–iterációkhoz eltérő törlési szabályokat rendel. Különleges módszerét eddig még nem közölte.

Megállapíthatjuk, hogy ha $p \in S_i^8$ akkor tetszőleges $q \in N_{26}(p) \setminus \{p\}$ pont esetén $q \notin S_i^8$ (a 2.14/c. ábrán csak a $3 \times 3 \times 3$ -as környezet középpontját címkézi 0). Az 1.6. Kritérium értelmében, tehát azok a 8-almezős vékonyító algoritmusok, amelyek csak $(26, 6)$ -egyszerű pontokat törölnek, megőrzik a topológiát a $(26, 6)$ képekre. Nyolc almező esetén tehát a topológia-megőrzés biztosítása triviális feladat. Nem véletlen tehát, hogy az ezidáig javasolt két almező-szekvenciális vékonyító eljárás — amelyeket a 2.2.1. és a 2.2.2. pontokban mutatunk be — nyolc almezőt használ (Bertrand, Aktouf, 1994; Saha, Chaudhuri, Majumder, 1997).

Az almező-szekvenciális vékonyítás is — hasonlóan a többi vékonyító technikához — megőrzi a topológiát és képes biztosítani azt, hogy a vékonyított objektum az eredetinek a “közepén” helyezkedjen el (± 1 voxel pontossággal). A módszer előnye még, hogy kevésbé érzékeny az objektum elforgatására, mint az irány-szekvenciális algoritmusok. Érzékeny viszont az objektum pozíciójára, vagyis nem eltolás-invariáns. A vékonyított kép egyenes vonalai gyakran “beremegnek”, hajlamosak kirajzolni az almezők mintázatát. Ez az egyenetlenség viszonylag könnyen korrigálható utófeldolgozással.

2.2.1. Bertrand és Aktouf 8–almezős algoritmus

A (26, 6) 3D képek vékonyítására javasolt eljárás (Bertrand, Aktouf, 1994) a 2.15/c. ábra szerint bontja 8–almezőre a \mathbb{Z}^3 halmazt. A topológia–megőrzés érdekében csak (26,6)–egyszerű pontokat jelölnek ki törlésre, ami az egyszerű pontok Bertrand által kidolgozott Boole–jellemzése szerint történik (Bertrand, 1996). Módszerük érdekessége, hogy az egyszerű pontokat, valamint a vonal– és a felszín–végpontokat is, ugyanazon két *topológiai számra* (*topological number*) felállított feltételek segítségével adják meg.

Az alkalmazott végpontkritériumtól függően az objektumok középvonalait vagy középfelszínét meghatározó vékonyító algoritmusokat kapunk. Mivel eljárásuk valamennyi egyszerű pontot kijelöli törlésre (nem csak az egyszerű pontok egy bizonyos részhalmazát), így a végpontkritérium elhagyása a (26, 6)–objektumok topológikus magját előállító zsugorító algoritmushoz vezet.

2.2.2. Saha, Chaudhuri és Majumder 8–almezős algoritmus

Saha, Chaudhuri és Majumder egy bonyolult szekvenciális vékonyító algoritmust javasoltak (Saha, Chaudhuri, Majumder, 1997), ahol egyrészt a "félkész" eredményre egy, a korábbiaktól eltérő törlési szabállyal dolgozó iterációs lépést kell alkalmazni, hogy a (26,6)–objektumok középvonalait vagy középfelszínét kapjuk eredményül, másrészt a módszer három pásztázó menetet igényel iterációs lépésenként. A három pásztázás során a három osztályba sorolt határpontokat jelölik meg, vagyis az algoritmus nem hajtható végre "helyben", azaz egyetlen egy bináris tömb használatával. A bonyolult törlési feltételek azt biztosítják, hogy az eljárás megőrzi a koordinátatengelyekkel párhuzamos "éleket", nem simítja, nem kerekíti le azokat. (Lokális törlési feltételek mellett — természetesen — nem konstruálható olyan vékonyító algoritmus, ami megőrizne tetszőleges irányú éleket.)

A szerzők eljárásuk párhuzamos, a 2.15c. ábra szerinti 8–almezős változatát is kidolgozták, ami a szekvenciális pásztázás helyett párhuzamos kijelölést, majd azt követően párhuzamos törlést alkalmaz.

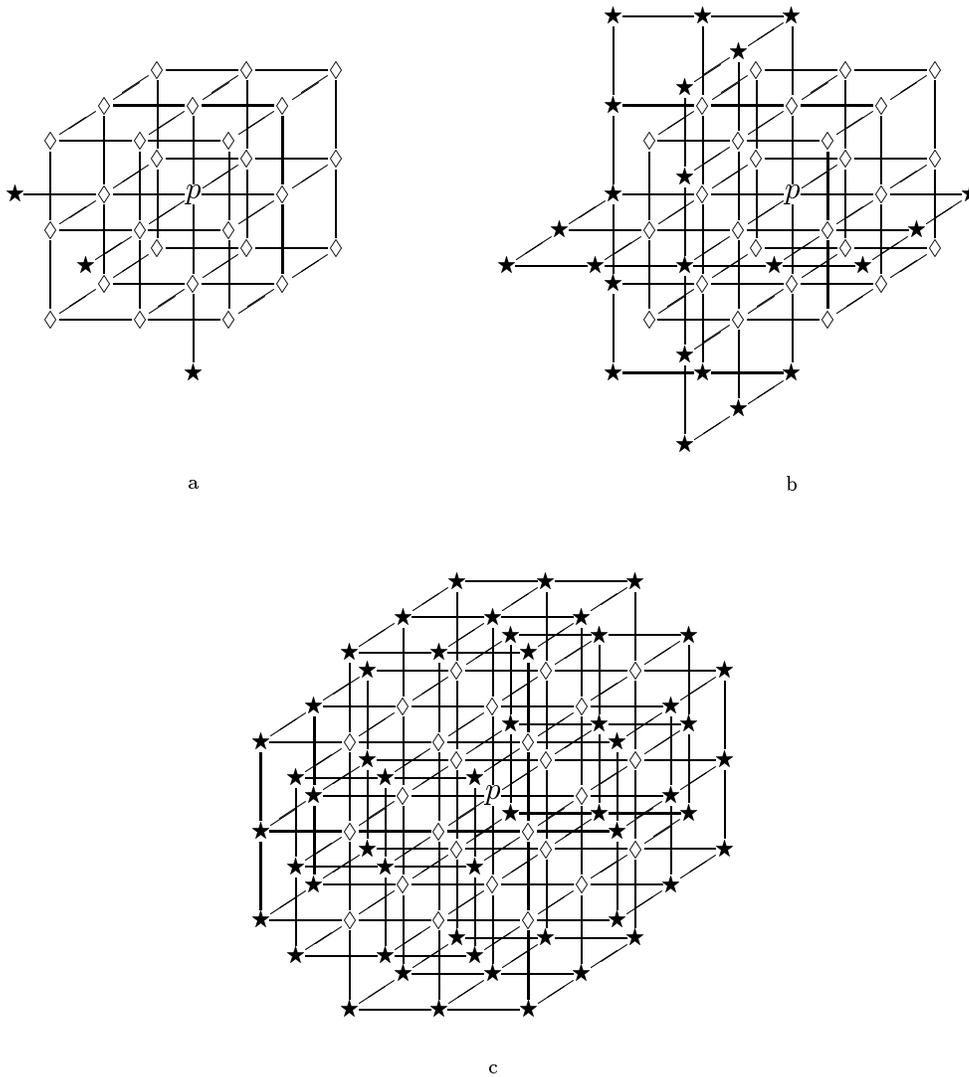
Mind a szekvenciális, mind a párhuzamos változat topológia–megőrző a (26, 6) képekre.

2.3. Teljesen párhuzamos módszerek

A teljesen párhuzamos vékonyító algoritmusok egy iterációs lépését egyetlen párhuzamos redukció írja le, vagyis nem alkalmaznak al-iterációkat. Az általunk ismert három teljesen párhuzamos vékonyító eljárás a topológia-megőrzés érdekében a fekete pontok 2.16. ábrán látható környezetét vizsgálja (Ma, 1995; Ma, Sonka, 1996, Manzanera et al., 1999/3).

A teljesen párhuzamos vékonyítás is — csakúgy, mint a másik kettő fő vékonyító technika — megőrzi a topológiát és képes biztosítani azt, hogy a vékonyított objektum az eredetinek a “közepén” helyezkedjen el (± 1 voxel pontossággal). Nem érzékeny az objektum pozíciójára sem, vagyis eltolás-invariáns. A módszer előnye még, hogy kevésbé érzékeny az objektum elforgatására, mint az irány-szekvenciális algoritmusok, de a gyengített izotrópia kritérium (Manzanera et al., 1999/1) teljesítésére ezúttal sincs garancia az asszimmetrikus lokális szomszédságot vizsgáló teljesen párhuzamos vékonyító algoritmusok esetében.

A teljesen párhuzamos megközelítéssel élő három algoritmust a 2.3.1., a 2.3.2. és a 2.3.4. pontokban mutatjuk be. A második eljárás tesztelése során kiderült, hogy bizonyos (26,6)-objektumokat “szétszakít”. A 2.3.3. pontban egy olyan módosított algoritmust mutatunk be, amelyik már topológia-megőrző.



2.16. ábra A teljesen párhuzamos vékonyító algoritmusok által figyelt környezetek. C.M. Ma algoritmus (Ma, 1995) csak három olyan pontot vizsgál, amit nem tartalmaz a vizsgált pont $3 \times 3 \times 3$ -as szomszédsága, viszont megkívánja a törlésre kijelölt pontok párhuzamos újraellenőrzését (a). C.M. Ma és Sonka eljárása (Ma, Sonka, 1996) egy olyan bővebb (de ugyancsak asszimmetrikus) környezetet vizsgál, ami a $3 \times 3 \times 3$ -as szomszédságon kívül még 23 további pontot is tartalmaz (b). A Manzanera, Bernard, Pretêux és Longuet algoritmus (Manzanera et al., 1999/3) által vizsgált szimmetrikus környezet 54 olyan pontot tartalmaz, ami eleme az $5 \times 5 \times 5$ -ös szomszédságnak, de nincs benne a $3 \times 3 \times 3$ -asban (c). (A vizsgált “ p ” pont $3 \times 3 \times 3$ -as szomszédságába eső pontokat a “ \diamond ”, az abból kilógókat pedig a “ \star ” szimbólumokkal jelöltük.)

2.3.1. Ma algoritmus

A teljesen párhuzamos vékonyító technikát 3D-ben C.M. Ma vezette be mint egy lehetséges módszert új generációs 3D párhuzamos vékonyító eljárások tervezésére (Ma, 1995)¹⁶.

A javasolt, középfelszín meghatározására alkalmas vékonyító algoritmus egy iterációs lépése két fázisból áll:

1. A “törlési feltételnek” eleget tevő fekete pontok egyidejű megjelölése.
2. A megjelölt pontok egyidejű ellenőrzése (melynek során eldől, hogy az adott pont véglegesen törlődik vagy továbbra is fekete marad).

A “ p ” pont eleget tesz a törlési feltételnek, ha az alábbiak mindegyike teljesül:

1. Nem “felszínvégpont”.
2. Illeszkedik rá a 2.17. ábrával adott maszkrendszer legalább egy tagja.
3. Ha $n(p) = 0$, akkor $s(s(p)) = 1$ (1.4. ábra).
4. Ha $e(p) = 0$, akkor $w(w(p)) = 1$.
5. Ha $u(p) = 0$, akkor $d(d(p)) = 1$.

Az algoritmus a 2.16a. ábra szerinti (nem-szimmetrikus) környezetet vizsgálja, ami tartalmazza a $3 \times 3 \times 3$ -as szomszédságból kilógó $s(s(p))$, $w(w(p))$ és $d(d(p))$ pontokat is.

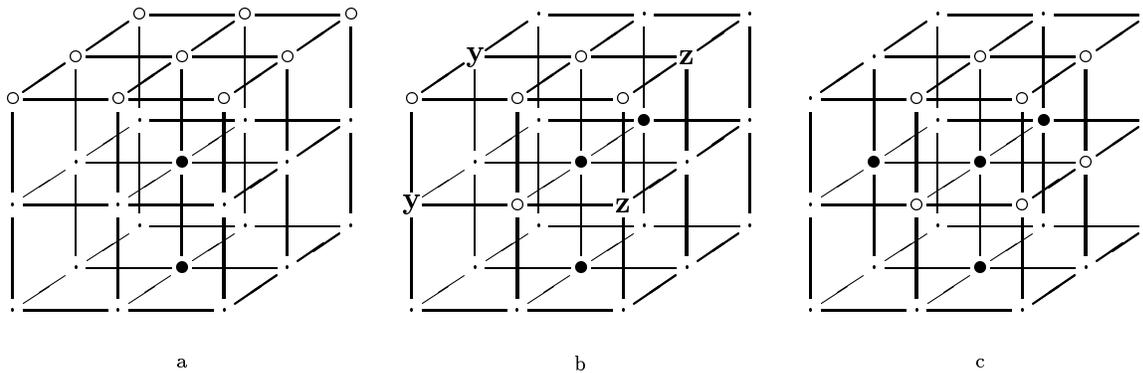
Az eljárás végpontkritériuma felszínvégpontnak minősíti a $\mathcal{P} = (\mathbb{Z}^3, 26, 6, B)$ képen a $p \in B$ pontot, ha az $N_{26}(p) \setminus \{p\}$ halmaz egyelemű (vagyis “ p ” vonalvégpont), vagy a $N_{ud}(p) \setminus \{p\}$, a $N_{ns}(p) \setminus \{p\}$ és a $N_{ew}(p) \setminus \{p\}$ halmazok (ellenőrző ablakok, 2.3. ábra) mindegyike egyelemű. (Megjegyzendő, hogy más végpontkritériumok alkalmazása újabb középfelszínre vagy középvonalra vékonyító algoritmusokhoz vezet.)

A párhuzamos újraellenőrzés (az iterációs lépések második fázisa) nem törli a “ q ” megjelölt pontot, ha az legkisebb indexösszegű pontja egy csupa megjelölt pontból álló egység-rácsnégyzetnek és “ q ” nem $(26, 6)$ -egyszerű pont a rácsnégyzet többi három pontjának törlése után (2.18. ábra).

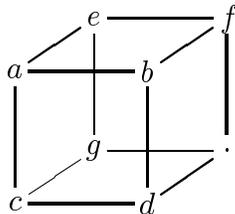
Az újraellenőrzés során tehát bizonyos törlésre megjelölt pont-konfigurációkra meg kell vizsgálni az egyszerűséget, amire több módszer is rendelkezésre áll (pl. az 1.2.2. pontban ismertetett kritériumok teljesülésének ellenőrzése). Megjegyzendő, hogy a $(26, 6)$ -egyszerűség lokálisan eldönthető az adott pont $3 \times 3 \times 3$ -as környezete alapján.

Az eljárás hátránya, hogy párhuzamos számítógéppel sem végezhető el “helyben” (mivel az iterációs lépések első fázisában meghatározandó jelzőbitek tárolására egy újabb bináris tömbre is szükség van), viszont a kétfázisú megoldás garantálja a topológia megőrzését a $(26, 6)$ képekre.

¹⁶Megjegyzendő, hogy 2D-ben már korábban is javasoltak teljesen párhuzamos vékonyító algoritmusokat (Suen, Wang, 1994; Hall, 1996).



2.17. ábra Az algoritmus alapmaszkjai. Mindhárom maszkon a “•” fekete, “o” fehér, a “.” pedig közömbös elemet jelöl. A (b) alapmaszkon a két “y” közül legalább az egyik fehér és a két “z” közül is legalább az egyiknek fehérnek kell lennie. Az eljárás teljes maszkrendszere tartalmazza az alapmaszkok összes tükrözöttjét és elforgatottját. (Könnyen belátható, hogy 6-féle (a)-típusú, 12-féle (b)-típusú és 8-féle (c)-típusú maszk képezhető.)



2.18. ábra A párhuzamos újraellenőrzés törlési kritériuma.

Tegyük fel, hogy az “a” pont törlésre megjelölt.

(a) Ha a “b”, a “c” és a “d” pontok is törlésre megjelöltek és “a” nem $(26, 6)$ -egyszerű a $\{b, c, d\}$ pont halmaz törlése után, akkor “a” törölhetőségét vissza kell vonni, vagyis meg kell őrizni fekete pontként.

(b) Ha a “b”, az “e” és az “f” pontok is törlésre megjelöltek és “a” nem $(26, 6)$ -egyszerű a $\{b, e, f\}$ pont halmaz törlése után, akkor az “a” pont fekete marad.

(c) Ha a “c”, az “e” és a “g” pontok is törlésre megjelöltek és “a” nem $(26, 6)$ -egyszerű a $\{c, e, g\}$ pont halmaz törlése után, akkor az “a” pont fekete marad.

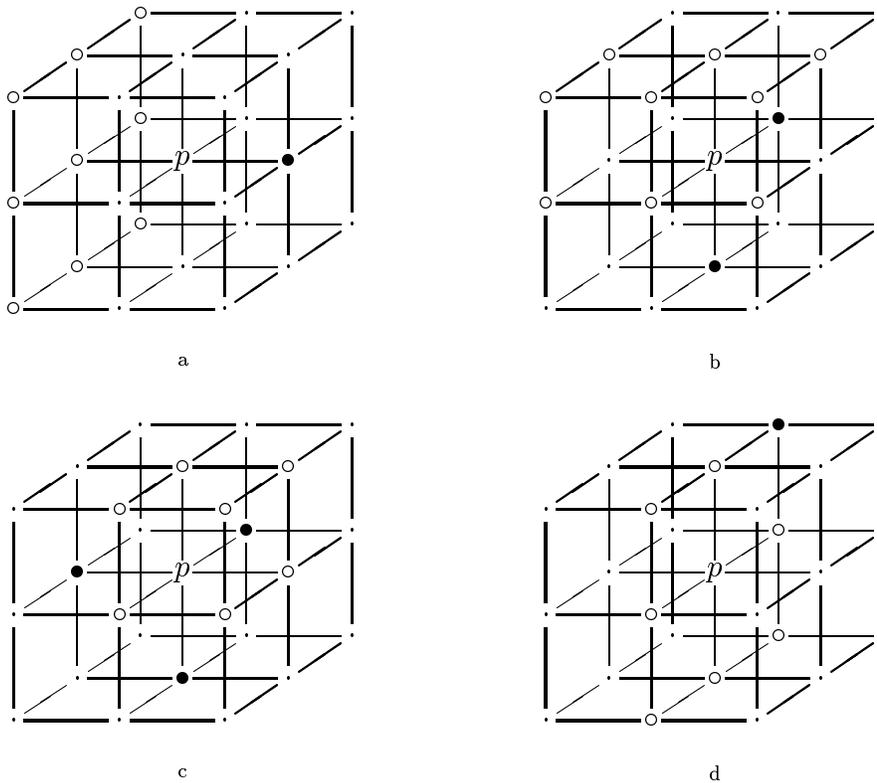
Minden más esetben a törlésre megjelölt “a” pont véglegesen törölhető.

2.3.2. Ma és Sonka algoritmus

Ma és Sonka a középvonalak meghatározására fejlesztett ki egy 3D teljesen párhuzamos vékonyító algoritmust (Ma, Sonka, 1996). Ez az eljárás “valóban” teljesen párhuzamos, mivel egy iterációs lépés egyetlen fázisból áll: nincs szükség újraellenőrzésre, mint az előző (a 2.3.1.) pontban bemutatott algoritmus esetében.

A (26, 6) képek vékonyítására javasolt eljárás törlési szabályai a 2.16b. ábrán látható (nem-szimmetrikus) lokális környezetet figyelik, mely a $3 \times 3 \times 3$ -as szomszedság 27 pontja mellett még további 23 pontot is tartalmaz.

A törlési feltételeket a szerzők négy maszkosztály egy-egy reprezentánsaival (2.19. ábra) és a további maszkok képzésére vonatkozó kiegészítő feltételekkel adták meg.



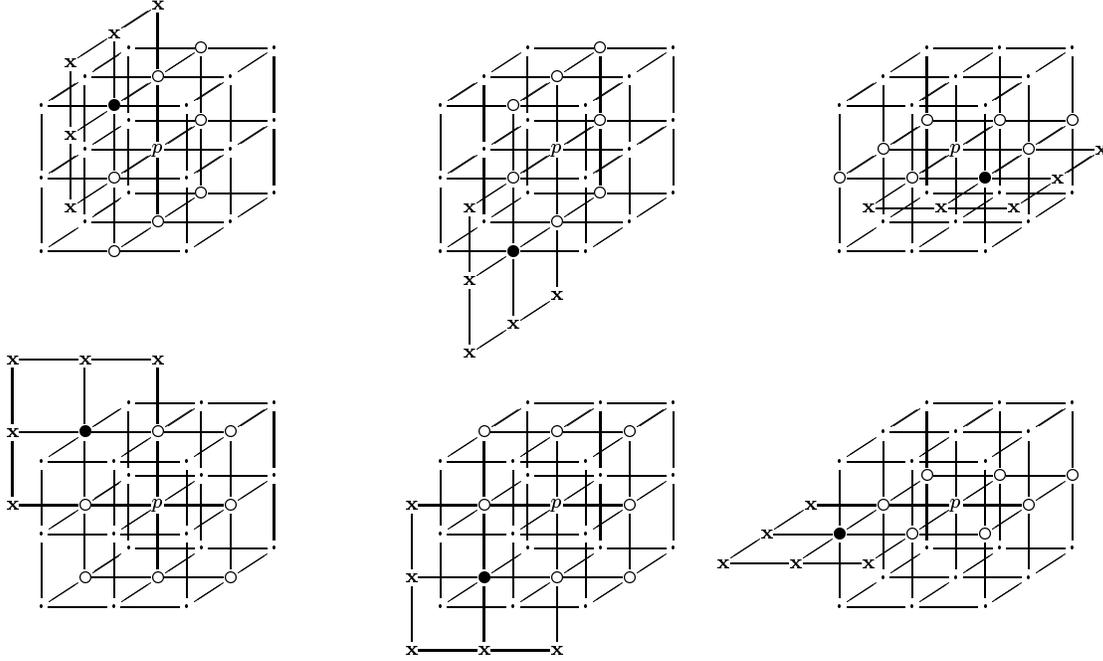
2.19. ábra Az algoritmus négy maszkosztályának reprezentáns elemei, ahol p fekete (\bullet). Az (a)–(d) maszkok minden elforgatottja és tükrözöttje is az adott maszkosztályba tartozik. Könnyen belátható, hogy az (a)–(c) maszkok csak (26,6)–egyszerű pontokra illeszkednek. Ez a tulajdonság nem teljesül automatikusan a (d) maszkra, így további feltétel az, hogy a (d) maszk illeszkedése esetén is csak (26,6)–egyszerű pontok törölhetők. Némely maszkhoz a p pont $3 \times 3 \times 3$ -as környezetén kívüli pontok is adódnak.

A 6, 12 és 8 elemből álló A, B és C maszkosztályok maszkjai rendre a 2.19. ábra (a), (b) és (c) maszkjából származnak elforgatásokkal és/vagy tükrözésekkel a következő szabály szerint:

- Ha a maszkban $s(p) = 1$, akkor $s(s(p)) = 1$.
- Ha a maszkban $w(p) = 1$, akkor $w(w(p)) = 1$.
- Ha a maszkban $d(p) = 1$, akkor $d(d(p)) = 1$.

Megállapíthatjuk, hogy az A,B és C maszkosztály elemei ugyanazt a környezetet figyelik, mint amivel az előző pontbeli (2.3.1.) algoritmus dolgozik és ami a 2.19a. ábrán látható.

A 2.19d. ábra mutatja be a D maszkosztály reprezentáns elemét. A maszkosztály 12 maszkja közül 6 különleges, mivel azok a $3 \times 3 \times 3$ -as környezetnél bővebb szomszédságot figyelnek. A különleges maszkokat (szöveges leírás helyett) a 2.20. ábrával mutatjuk be. Hangsúlyozandó, hogy a D maszkosztály valamennyi elemének $(26, 6)$ -egyszerű pontokra kell illeszkedniük.



2.20. ábra A 12 maszkból álló D maszkosztály 6 speciális, $p = 1$ középpontú, a $3 \times 3 \times 3$ -as szomszédságnál bővebb környezetet figyelő maszkja. A maszkok illeszkedését a $(26, 6)$ -egyszerű pontokra korlátozzák. Valamennyi maszk esetén az öt “x”-szel jelölt pozíció közül legalább egynek fekete ponttal kell találkozni.

Az algoritmus egy iterációs lépésében azon fekete pontok törlődnek, amelyek “nem végpontok” és amelyekre legalább egy maszk illeszkedik az A, a B, a C vagy a D maszkosztályból.

Az eljárás speciális végpontkritériummal dolgozik: A $\mathcal{P} = (\mathbb{Z}^3, 26, 6, B)$ képen a $p \in B$ pont végpont, ha a $N_{26}(p) \setminus \{p\}$ halmaz egyelemű (azaz p vonalvégpont), vagy pontosan kettő eleme van. A két fekete ponttal 26-szomszédos pont csak bizonyos esetekben lehet végpont. Tegyük fel, hogy a $q \in B$ és az $r \in B$ pontokra a $N_{26}(q) \setminus \{q\}$ és a $N_{26}(r) \setminus \{r\}$ halmazoknak pontosan kettő-kettő elemük van. Ekkor a következők teljesülnek:

- Ha $N_{26}(q) \setminus \{q\} = \{s(q), e(q)\}$ és $N_{26}(r) \setminus \{r\} = \{s(r), u(r)\}$, akkor q és r közül pontosan az egyik pont végpont.
- Ha $N_{26}(q) \setminus \{q\} = \{n(q), w(q)\}$ és $N_{26}(r) \setminus \{r\} = \{u(r), w(r)\}$, akkor q és r közül pontosan az egyik pont végpont.
- Ha $N_{26}(q) \setminus \{q\} = \{n(q), d(q)\}$ és $N_{26}(r) \setminus \{r\} = \{e(r), d(r)\}$, akkor q és r közül pontosan az egyik pont végpont.

A vonalvégpontok mellett azért minősülnek egyéb pontok is végpontnak, hogy a 6-összefüggő vonalrészletek ne szakadjanak szét.

2.3.3. Ma és Sonka algoritmusának módosítása

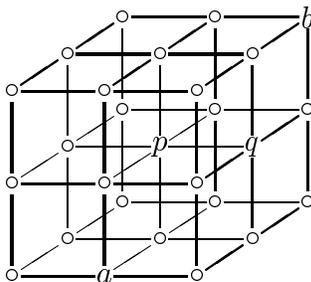
Ma és Sonka bebizonyította, hogy algoritmusuk topológia-megőrző (Ma, Sonka, 1996). Bizonyításukba — ami megmutatta, hogy a javasolt eljárás teljesíti az 1.6. Kritérium valamennyi feltételét — azonban becsúszott egy hiba.

Két egymással 6-szomszédos, törölhető p és q pont esetén osztályozták p -t aszerint, hogy az a q pontot törlő maszk fekete vagy semleges elemével kerül-e fedésbe. Ezen osztályozáson alapuló állításaik korrektek is abban az esetben, ha a maszkokban csak “0”, “1” és “.” (semleges) elemek szerepelnek.

A D osztály (2.20d. ábra) maszkjaira kikötötték, hogy azok csak egyszerű pontokra illeszkehetnek, viszont azokban a maszkokban is a nem “0” és nem “1” elemeket “.” szimbólumokkal jelölték, mintha azok is semlegesek lennének, holott az azokkal lefedett pontoknak (az egyszerűség miatt) nem megengedett tetszőleges kombinációja.

Bizonyításukban a D osztály maszkjainak “.”-jelű elemeit is “igazi semleges” pozícióknak tekintették, így a bizonyított tulajdonságok nem érvényesek az eredeti algoritmusra.

A 2.21. ábra példájával megmutatjuk, hogy az eredeti algoritmus szétszakíthat objektumokat, ennél fogva nem őrzik meg a topológiát valamennyi $(26,6)$ képre.



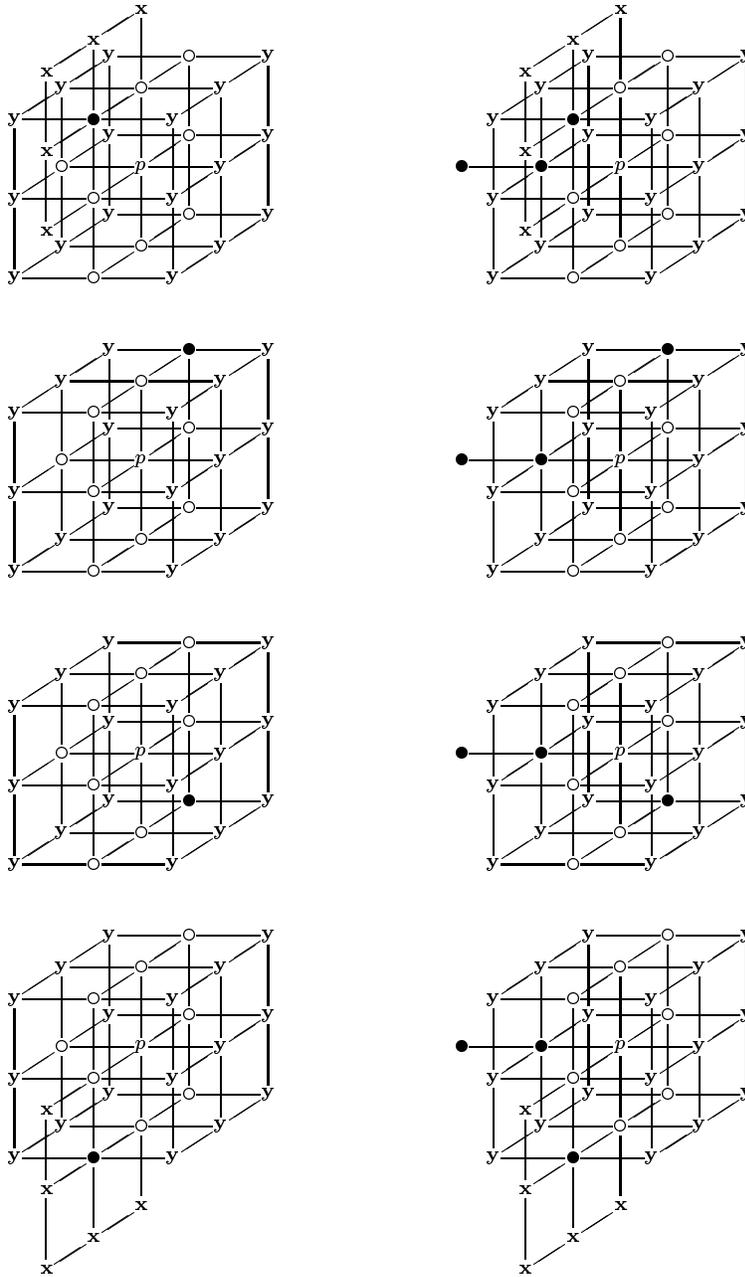
2.21. ábra Példa kritikus képre. Az ábra a $\mathcal{P} = (\mathbb{Z}^3, 26, 6, \{a, p, q, b\})$ kép azon $3 \times 3 \times 3$ -as részletét mutatja, amelyik tartalmazza az összes fekete pontot (mind a négyet). Az eredeti algoritmus törlési feltételei szerint p törölhető egy, az A maszkosztályba eső maszkkal (2.19a. ábra), q pedig törölhető egy, a D maszkosztályba eső maszkkal (2.20d. ábra). p és q egyidejű törlése viszont szétszakítja a kép egyetlen fekete komponensét, vagyis az eljárás nem topológia-megőrző.

Az algoritmus tehát módosításra szorul¹⁷. Az alábbiakban egy olyan módosító javaslatot ismertetünk, amit az motivált, hogy a D maszkosztályt egy olyan D'-vel helyettesítsünk, hogy az A, B, C és D' maszkosztályokra már korrekt legyen Ma és Sonka bizonyítása.

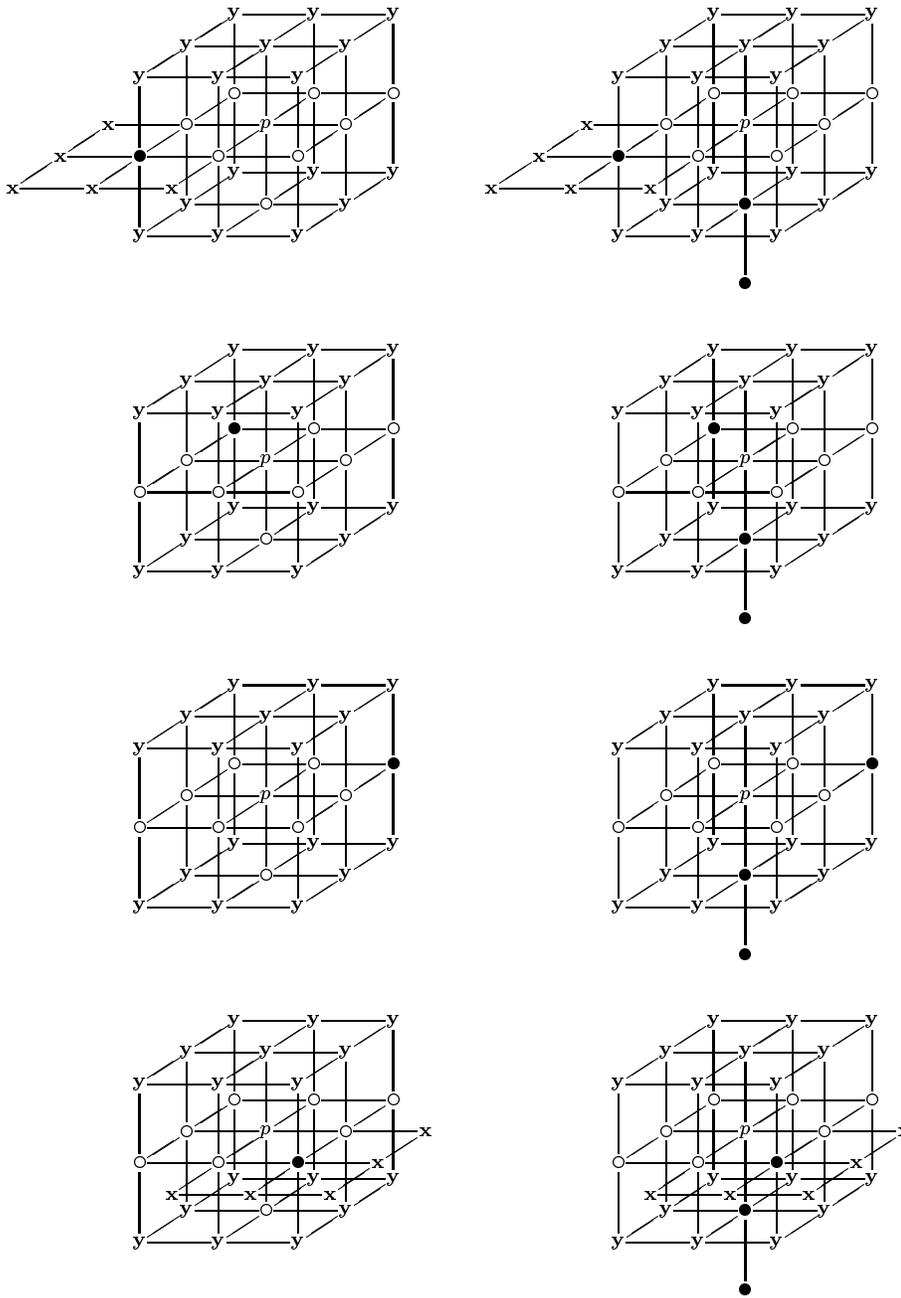
A D' maszkosztályban D minden maszkja helyett egy maszkpár szerepel, tehát az új osztály 12 helyett 24 maszkot tartalmaz.

Az alábbiakban, a 2.22a.–2.22c. ábrákon felsoroljuk a D' maszkosztály $p = 1$ középpontú maszkjait, ahol az y pozíciókat úgy kell kitölteni, hogy valamennyi maszk $(26,6)$ -egyszerű pontra illeszkegjen, továbbá a 2.20. ábra 6 speciális maszkját helyettesítő maszkoknál az x pozíciók közül legalább egynek a színe fekete. Hangsúlyozzuk, hogy az y pozíciók nem tekinthetők “.” (semleges) elemeknek.

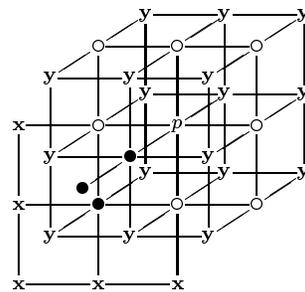
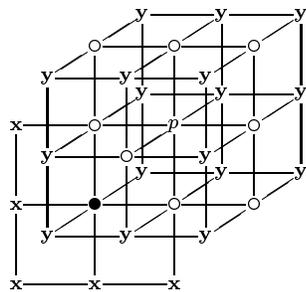
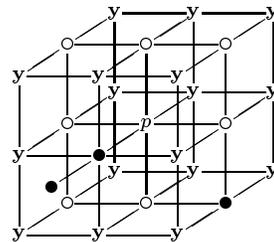
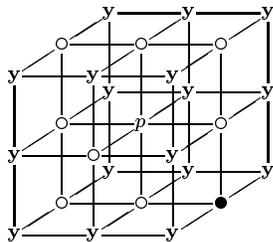
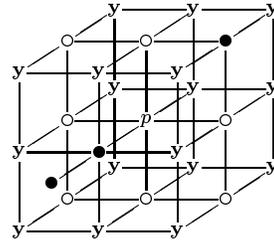
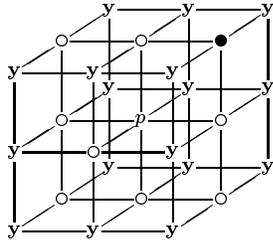
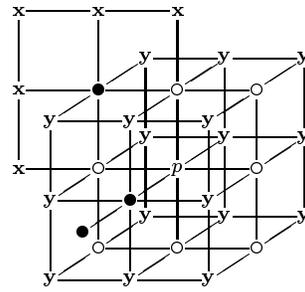
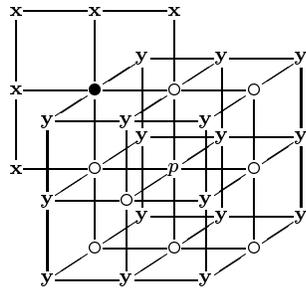
¹⁷Az eredeti algoritusról és a javasolt módosításról 1998-ban számos levelet váltottam Milan Sonkával (University of Iowa), aki elküldte nekem az eredeti algoritmus Cherg Min Ma által megírt C-kódját. Az nem szakította szét a 2.23. ábra objektumát, viszont másokat igen. A kód ellenőrzése során kiderült, hogy az implementáció eltér a cikkben leírtaktól, korábbi vagy későbbi algoritmusváltozatnak felel meg. Sajnos Cherg Min Ma még Milan Sonka számára is elérhetetlen, így vele még nem sikerült konzultálnom.



2.22/a. ábra A D' maszkosztály első négy maszkpárja.

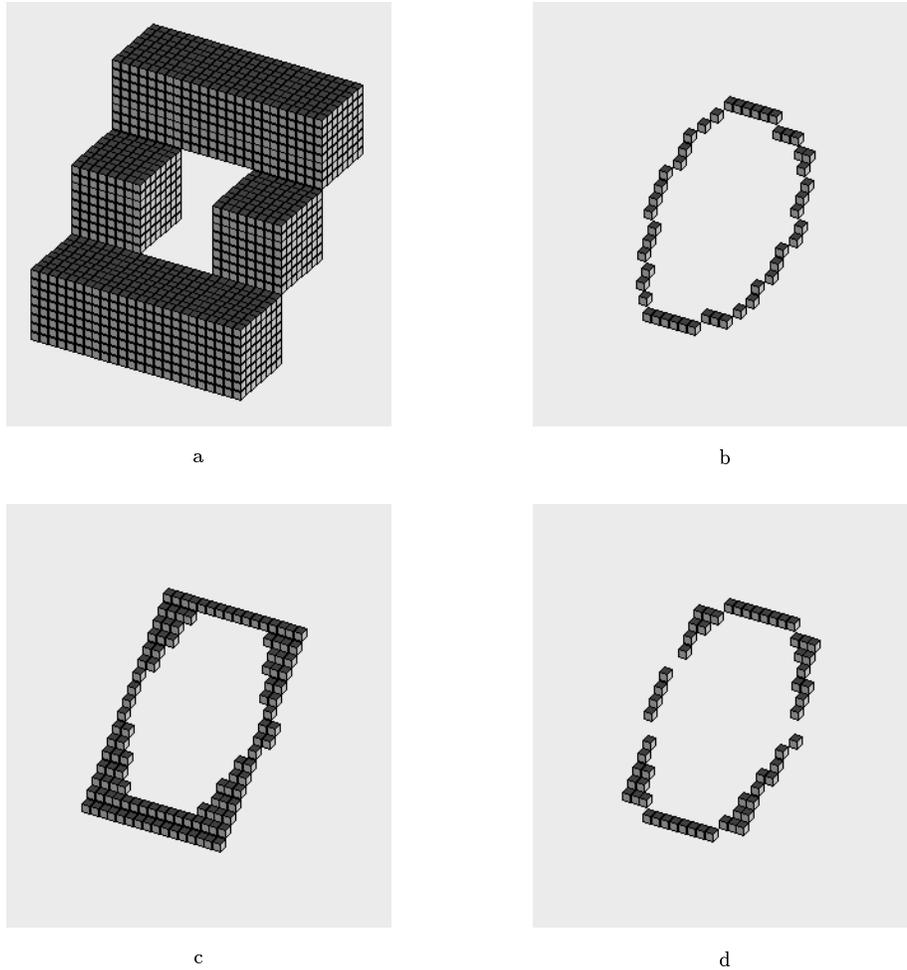


2.22/b. ábra A D' maszkosztály második négy maszkpárja.



2.22/c. ábra A D' maszkosztály harmadik (egyben utolsó) négy maszkpárja.

Az eredeti és a módosított algoritmus eredménye a 2.23. ábrán látható.



2.23. ábra Példa az eredeti és a módosított algoritmusra. A kiindulási $(26, 6)$ objektum (a) topológikusan ekvivalens a módosított algoritmus végeredményével (b). Az eredeti algoritmus az (a) objektumot néhány iterációs lépésben a (c) objektummá “hámozza”, amit a következő lépése háromfelé szakít szét (d).

Kimondjuk a módosított algoritmus legfontosabb tulajdonságát:

2.6. TÉTEL

Ma és Sonka algoritmusának általunk javasolt módosítása megőrzi a topológiát a $(26,6)$ képekre.

BIZONYÍTÁS¹⁸

Ma és Sonka korrekt módon bebizonyította, hogy ha a törlési feltételekre bizonyos tulajdonságok teljesülnek, akkor eljárásuk megőrzi a topológiát a $(26,6)$ képekre. Sajnos az általuk megfogalmazott egyik lemma (a cikkben Lemma 3.5.) nem teljesül az eredeti maszkokra.

¹⁸A módosított algoritmus topológiai korrektségére ugyanaz a hosszú bizonyítás szolgál, mint amit Ma és Sonka közölt. Helyhiány miatt azt nem másoljuk be a dolgozatba, csak a kritikus lemmával foglalkozunk.

A kérdéses lemma állítása: ha az egymással 6-szomszédos p és q fekete pontok közül mindkettő törölhető, akkor az alábbi két eset közül pontosan egy teljesül:

- A p pont törölhető q törlése után is (vagyis egy, a p fekete pontra illeszkedő maszkban q semleges maszkelemmel találkozik).
- A q pont törölhető p törlése után is (vagyis egy, a q -ra illeszkedő maszk p -vel fedésbe kerülő eleme semleges).

A 2.21. ábrával megmutattuk, hogy az eredeti maszkokra a fenti lemma nem teljesül, mivel p törléséhez szükséges az, hogy q fekete legyen, ugyanakkor q csak egy, a D osztályba eső maszkkal törölhető, amikor q egyszerű voltához p -nek szintén feketének kell lennie, dacára annak, hogy azt a maszkpozíciót ugyanúgy jelölték, mint a semleges elemeket.

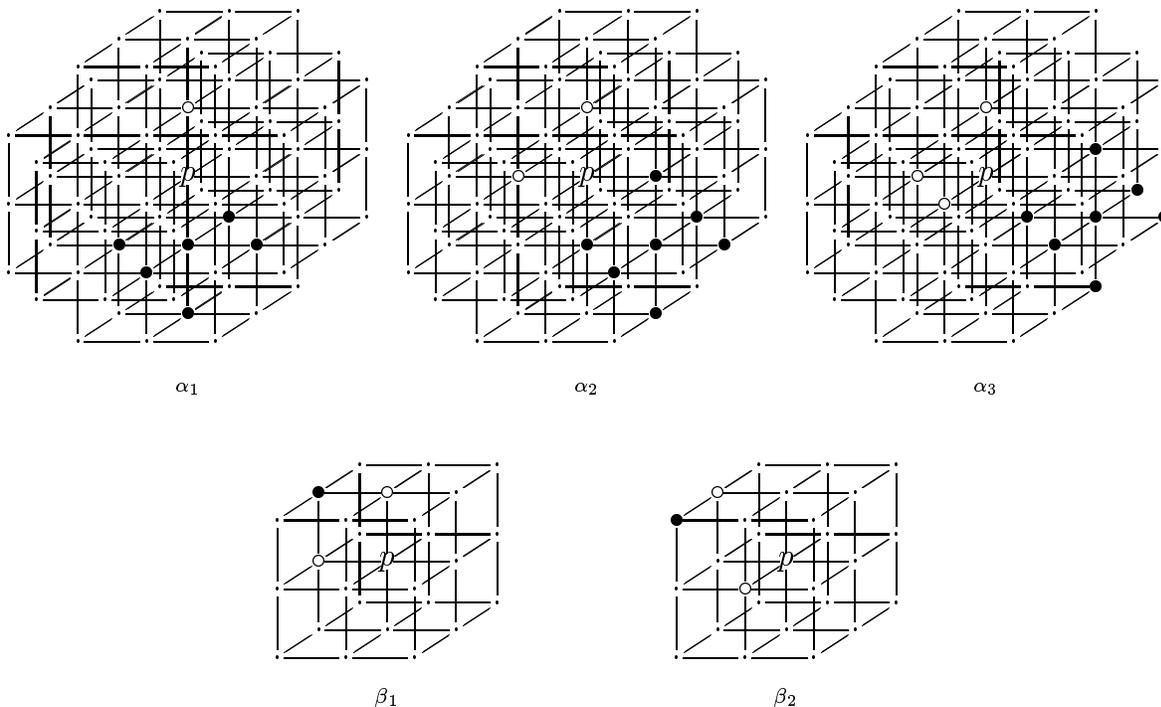
Ez a jelöléstechnikai hiba vezetett oda, hogy (a 2.21. ábrán) az egy rácsnégyzetbe eső $\{p, q\}$ ponthalmaz nem egyszerű, így az eredeti algoritmus megsérti a topológia-megőrzésre vonatkozó 1.6. Kritériumot.

Könnyen ellenőrizhető, hogy a kritikus lemma már teljesül a javasolt módosított algoritmusra. ■

2.3.4. Manzanera, Bernard, Pretêux és Longuet algoritmus

Manzanera, Bernard, Pretêux és Longuet középfelszín meghatározására fejlesztették ki 3D teljesen párhuzamos vékonyító algoritmusukat (Manzanera et al., 1999/3). Az eljárás törlési szabályai a 2.16c. ábrán látható (szimmetrikus) lokális környezetet figyelik, mely a $3 \times 3 \times 3$ -as szomszédság 27 pontján kívül még 54 pontot is tartalmaz az $5 \times 5 \times 5$ -ös szomszédságból.

Az algoritmus törlési szabályait a 2.24. ábra konfigurációi írják le: egy pont akkor törölhető, ha illeszkedik a környezetére az α_1 , az α_2 vagy az α_3 maszkosztályok valamelyik eleme, de nem illeszkedik rá a β_1 és a β_2 maszkosztályok egyetlen eleme sem.



2.24. ábra Az eljárás törlési szabályaihoz tartozó maszkosztályok reprezentáns elemei, alapmaszkjai. A maszkosztályok tartalmazzák az alapmaszkjaik valamennyi elforgatottját és tükrözöttjét, így az α_1 , az α_2 , az α_3 , a β_1 és a β_2 maszkosztályok rendre 6, 12, 8, 12 és 16 elemből állnak. A maszkok fekete pontjait “•”, fehér pontjait pedig “o” szimbólumok jelölik. A maszkok középpontjai, a “p”-vel jelölt maszkkelemek szintén feketék. Hangsúlyozandó, hogy az α_1 , az α_2 és az α_3 maszkosztályok elemeire történő illeszkedés engedélyezi, a β_1 és a β_2 osztályok maszkjaira való illeszkedés pedig tiltja a törlést. Megjegyzendő, hogy a β_1 és a β_2 maszkosztályok maszkjai csak a $3 \times 3 \times 3$ -as környezetbe eső pontokat vizsgálják, így azok reprezentáns elemeit a könnyebben áttekinthető kisebb környezettel adtuk meg.

A szerzők — háromféle képtípusra — három algoritmust adtak meg:

- A (6, 6) képekre javasolt eljárás az α_1 , valamint a β_1 és a β_2 maszkokkal dolgozik.
- A (6, 18)-as képek középfelszíne az α_1 és az α_2 , valamint a β_1 és a β_2 maszkokat veszi figyelembe.
- A (6, 26) képek esetén az α_1 , az α_2 és az α_3 , valamint a β_1 és a β_2 maszkok érvényesek.

Mindhárom eljárás megőrzi a topológiát az adott képtípusra, viszont bizonyos objektumokra (pl. páros oldalhosszúságú álló téglatestekre) csak “kettő voxel vékonyságú” felszínszegmenseket képesek előállítani. Az “egy voxel vékony” középfelzín kinyerése könnyen megoldható utófeldolgozással.

Megjegyezzük, hogy a szerzők módszerüket kidolgozták 1D, 2D és 3D objektumok vékonyítására, továbbá általánosították azt nD -re ($n > 3$) is (Manzanera et al., 1999/1; Manzanera et al., 1999/2).

2.4. Hibrid stratégia

A 2.1. és a 2.3. pontokban bemutatott irány-szekvenciális és almező-szekvenciális stratégiák kombinációjával is megoldható a vékonyítás (Palágyi, Kuba, 1998/2). A hibrid módszer egy iterációs lépése k_1 irány-szekvenciális és k_2 almező-szekvenciális al-iterációból áll a következő séma szerint:

```
repeat
  remark a  $k_1$  irány-szekvenciális al-iteráció
  for  $i = 1$  to  $k_1$  do
    egyidejű törlése mindazon fekete pontoknak, amelyek
    kielégítik az  $i$ -edik irányhoz tartozó feltételt
  remark a  $k_2$  almező-szekvenciális al-iteráció
  for  $j = 1$  to  $k_2$  do
    egyidejű törlése mindazon fekete pontoknak, amelyek
    a  $j$ -edik almezőbe esnek és teljesítik a minden
    egyik almezőre érvényes (globális) törlési feltételt
until nem történt változás
```

A 2.4.1. pontban egy olyan hibrid algoritmust mutatunk be, amelyik nyolc irányt és mindössze kettő almezőt feltételez (Palágyi, Kuba, 1998/2).

2.4.1. Egy hibrid algoritmus

A jelen pontban egy nyolc irányt és kettő almezőt alkalmazó hibrid algoritmust mutatunk be középvonalakat meghatározó vékonyításra (Palágyi, Kuba, 1998/2). Az eljárás ugyanazokat a törlési irányokat alkalmazza, mint amelyeket a 2.1.7. pontban ismertetett 8 al-iterációs vékonyító algoritmus (2.8. ábra). Az egész koordinátájú rácsponatok halmaza a 2.15a. ábra szerinti felbontás érvényes, vagyis $\mathbb{Z}^3 = S_1^2 \cup S_2^2$.

Az eljárást az alábbi séma írja le:

Input: a vékonyítandó (26,6) képet reprezentáló X bináris tömb.

Output: a vékonyított képet tároló Y bináris tömb.

```
hibrid_vékonyító_algoritmus( $X, Y$ )
begin
   $Y = X$ ;
  repeat
    remark irány-szekvenciális al-iterációk
     $Y = \text{törlés\_usw\_irányból}(Y)$ ;
     $Y = \text{törlés\_dne\_irányból}(Y)$ ;
     $Y = \text{törlés\_use\_irányból}(Y)$ ;
     $Y = \text{törlés\_dnw\_irányból}(Y)$ ;
     $Y = \text{törlés\_une\_irányból}(Y)$ ;
     $Y = \text{törlés\_dsw\_irányból}(Y)$ ;
     $Y = \text{törlés\_unw\_irányból}(Y)$ ;
     $Y = \text{törlés\_dse\_irányból}(Y)$ ;
    remark almező-szekvenciális al-iterációk
     $Y = \text{törlés\_}S_1^2\text{\_aktivizálásával}(Y)$ ;
     $Y = \text{törlés\_}S_2^2\text{\_aktivizálásával}(Y)$ ;
  until nem történt változás;
end.
```

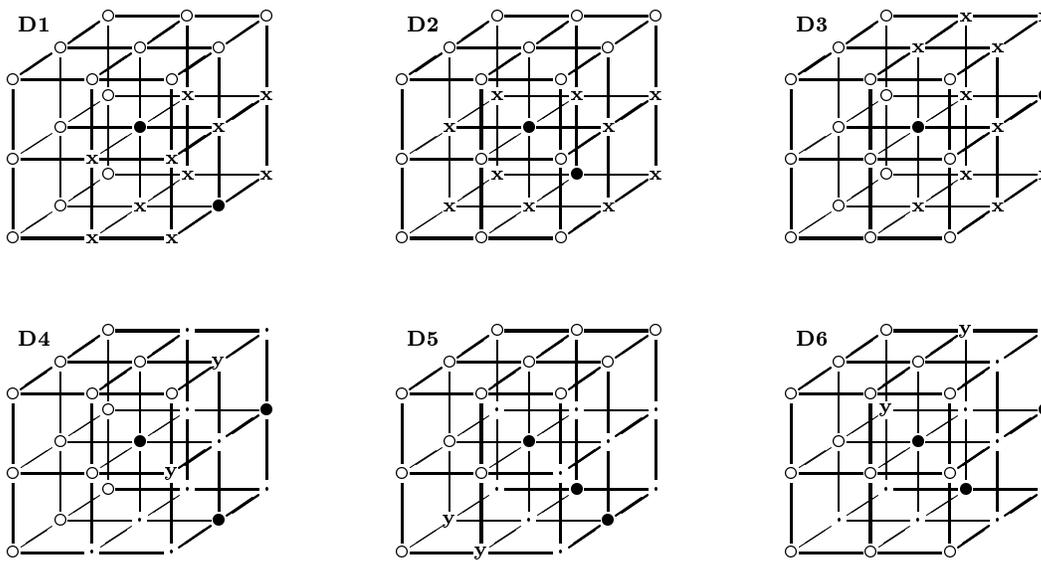
Az *usw* törlési irányhoz tartozó képművelet (*törlés_usw_irányból*) bizonyos *u*-, *s*- és *w*-határpontokat töröl, a *dne* irány párhuzamos redukciója (*törlés_dne_irányból*) pedig bizonyos *d*-, *n*- és *e*-határpontokat hámoz le az objektumokról, és így tovább.

Az eljárásnak az *usw* törlési irányhoz rendelt lokális párhuzamos redukcióját a 2.25. ábra maszkjaival adjuk meg.

Könnyen ellenőrizhető, hogy a 2.25. ábrán látható maszkkészlet minden egyes eleme csak *u*-, *s*- vagy *w*-határpontokra illeszkedik. A fenti maszkokkal adott képművelet — az 1.4.4. pontban leírtak szerint — törli mindazon fekete pontokat, amelyekre legalább egy maszk illeszkedik a maszkkészletből.

A fennmaradó hét al-iterációhoz tartozó képművelet maszkjai megkaphatók az *usw* törlési irányhoz rendelt maszkok megfelelő elforgatásával és/vagy tükrözésével.

Az eljárás almező-szekvenciális fázisának mindkettő al-iterációja ugyanazon törlési feltételt alkalmazza. Az almezős al-iterációkhoz rendelt lokális párhuzamos redukciót a 2.26. ábra maszkkészlete adja meg. Az ábrán csak az alapmaszkokat mutatjuk be, amelyekből az elforgatással és/vagy tükrözéssel megkapható maszkok szintén a maszkkészlethez tartoznak.



2.25. ábra A hibrid algoritmusnak az usw törlési irányhoz tartozó maszkjai. A vonalvégpontok megőrzése érdekében a **D1–D3** maszkoknál az “x” pozíciók közül legalább egy “•” (fekete). Az “y”-nal jelölt maszkpozíciók közül legalább egy “o” (fehér) a **D4–D6** maszkokon.

Könnnyen belátható, az eljárás valamennyi maszkja (ld. a 2.25. és a 2.26. ábra) csak nem-vonalvégpontra illeszkedik (2.1. Definíció), amit az “x” szimbólumok biztosítanak. Az “y” és a “z” szimbólumok felvétele garantálja, hogy a maszkok csak (26,6) egyszerű pontokat töröljenek.

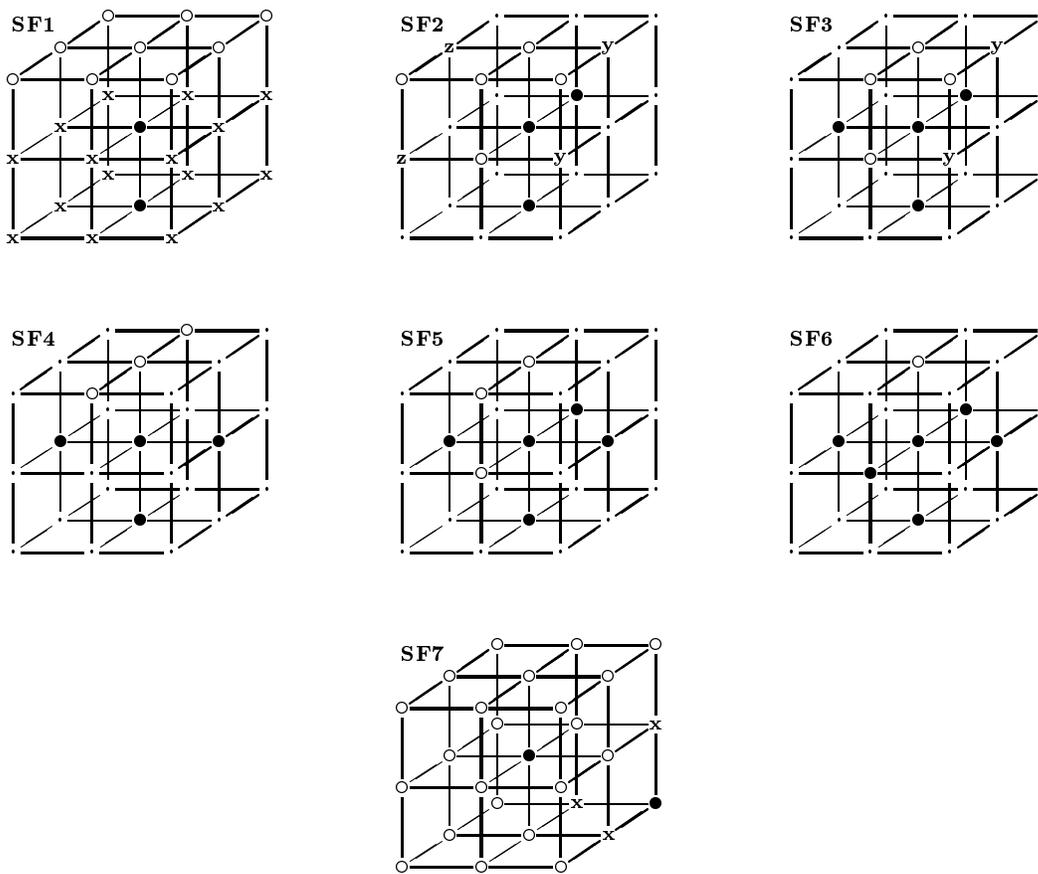
Az algoritmus működésének könnyebb megértését a 2.27. ábra segíti.

Az ismertetett vékonyító eljárás topológiailag korrekt, amit az alábbi tétel mond ki:

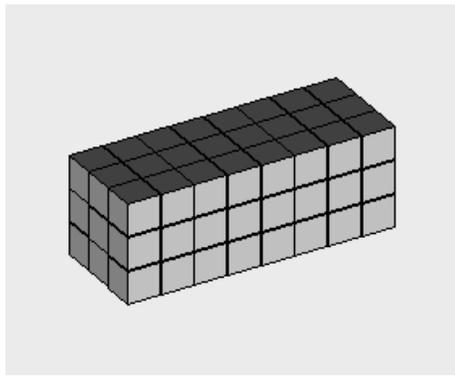
2.7. TÉTEL

A középvonalakat kivonó hibrid vékonyító eljárás megőrzi a topológiát a (26,6) képekre.

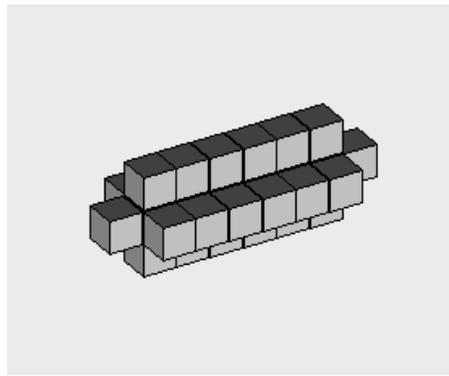
A fenti állítást — az 1.7. Tétel segítségével — részletesen bizonyítottuk (Palágyi, Kuba, 1998/2).



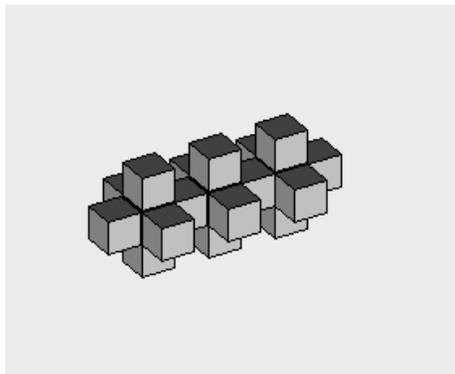
2.26. ábra A hibrid algoritmus almező-szekvenciális al-iterációihoz rendelt alapmaszkjai. Jelölés: az “x” pozíciók közül legalább egy “•” (fekete), az “y”-nal jelölt maszkpozíciók közül legalább egy “o” (fehér), és a “z” pozíciók közül is legalább egy “o” (fehér).



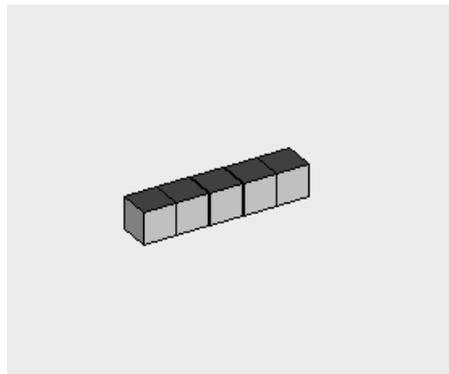
a



b



c



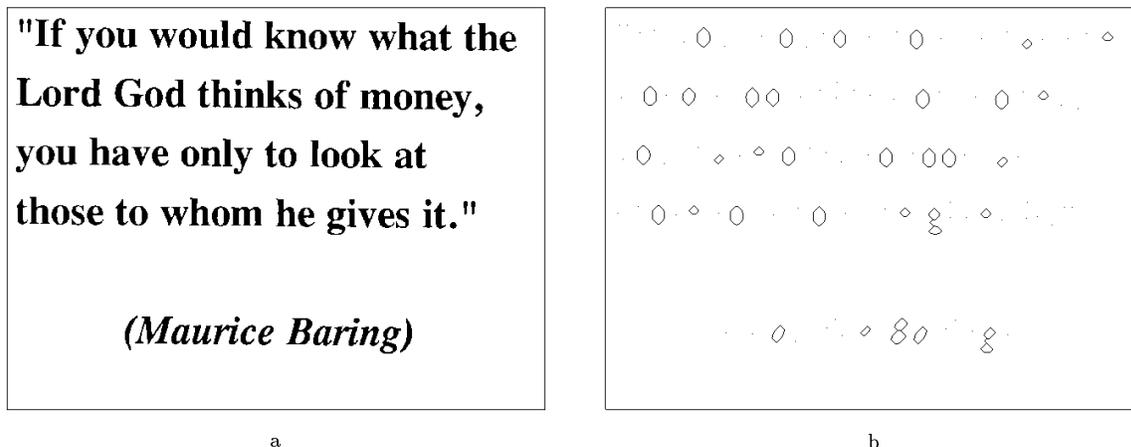
d

2.27. ábra A hibrid algoritmus egy lépésének hatása. Az (a) kiindulási objektum egy $3 \times 8 \times 3$ -as tömör téglaltest. A nyolc irány-szekvenciális al-iteráció végrehajtásával a (b) objektumot kaptuk. Az első almező-szekvenciális al-iteráció a (c), majd az iterációs lépést befejező második almező-szekvenciális al-iteráció a (d) objektumot eredményezi.

3. A zsugorítás

A *zsugorítás* (*shrinking*) mint képművelet meghatározása nem egyértelmű, mivel a szakirodalomban különböző célokra szolgáló és eltérő tulajdonságú eljárásokat is zsugorító algoritmusoknak neveznek (Hall, Kong, Rosenfeld, 1996).

Egyes szerzők a bináris objektum topológiai magját (1.3.2.) kivonó képműveletként értelmezik. A topológia-megőrző redukció üreget nem tartalmazó 2D objektumot valamint üreget és lyukat nem tartalmazó 3D objektumot egyetlen izolált pontra húz össze. Az olyan üreget tartalmazó 2D objektumból, aminek az üregébe nincs objektum “ágyazva” a zsugorítás egyszerű zárt görbét (Jordan-görbét) állít elő, amint azt a 3.1. ábra illusztrálja. A zsugorítás hatására 3D-ben a lyukas objektumból zárt görbét kapunk (1.15d. ábra), míg az üreges objektum Jordan-felzínne alakul át (Herman, 1992). A zsugorítás a vékonyítással rokon redukció, a vékonyítás végpont-kritériummal nehezített zsugorításnak is tekinthető. Zsugorító algoritmusokhoz a korrekt (egy pont vastagságú) középvonalakat/középfelzín eredményező vékonyító eljárások módosításával is eljuthatunk, ha enyhítünk azok törlési feltételein (nem ragaszkodunk a végpontok megőrzéséhez).



3.1 ábra Példa 2D zsugorításra. A kiindulási (a) kép objektumainak topológiai magját egy 2 al-iterációs irány-szekvenciális eljárással (Palágyi, 1993) határoztuk meg. A zsugorítás és a vékonyítás hatásának különbözősége kitűnik, ha a zsugorított (b) képet összevetjük a 2.1. ábrával.

Más szerzők — akik a zsugorítást az objektumok címkézésére vagy számlálására alkalmazzák — csak a fekete komponensek összefüggőségének megőrzésére ügyelnek, a fehér komponenseket (a háttér kivételével) pedig el szeretnék távolítani a képről. Ekkor nem a topológiai mag kivonása a cél, hanem az, hogy (lehetőleg) minden objektumot csak egy-egy izolált pont helyettesítsen. Ebben a felfogásban a zsugorítás nem őrzi meg a topológiát és nem tartozik a redukciós képműveletek közé sem, mivel fehér pontok is megváltozhatnak az üregek és/vagy a lyukak feltöltése érdekében.

A fenti két felfogás megkülönböztetése érdekében a *redukciós zsugorítás* és a *vegyes zsugorítás* elnevezésekkel élünk.

Zsugorításra 2D-ben számos lokális és párhuzamos eljárást javasoltak. Levialedi bináris minták számlálására feljlesztette ki vegyes zsugorító algoritmusát (Levialedi, 1972). Az

eljárás sok szempontból különleges: csak 2×2 -es (!) maszkokkal dolgozik, minden egyes objektumot a befoglaló téglalapja jobb-felső sarkába húz össze egyetlen izolált ponttá, és törli az izolált pontokat is. Így az objektumok száma csak úgy határozható meg, ha minden egyes iterációs lépés előtt megvizsgáljuk a képet, hogy tartalmaz-e izolált pontot. Az eljárás akkor terminál, ha a kép már nem tartalmaz egyetlen fekete pontot sem. Levialdi algoritmusát Shi és Ritter módosította úgy, hogy a befoglaló téglalap jobb-felső sarkába összehúzó eljárást kombinálta a bal-felső sarokba zsugorító változattal (Shi, Ritter, 1995). A módosított vegyes eljárás kevesebb iterációs lépést igényel, viszont a 8-összefüggőség megőrzése érdekében 2×3 -as maszkokat használ.

Kameswara Rao és munkatársai 2 al-iterációs vegyes eljárása (Kameswara Rao, Danielsson, Kruse, 1978) 3×3 -as törölő és kitöltő maszkokkal dolgozva képes egy-egy izolált pontra összehúzni az üreget tartalmazó objektumokat is. Hasonló eredményt szolgáltat Gökmen és Hall vegyes algoritmus (Gökmen, Hall, 1990), ami a \mathbb{Z}^2 pontjait sakktáblaszerűen osztja fel kettő almezőre. Az almező-szekvenciális eljárás is 3×3 -as törölő és kitöltő feltételeket használ.

Kifejlesztettek redukciós módszereket is. Pratt és Kabir (Pratt, Kabir, 1985) és Palágyi (Palágyi, 1993) 2 al-iterációs irány-szekvenciális eljárásokat fejlesztettek ki (8,4) képek topológia-megőrző zsugorítására. (Ez utóbbi algoritmus hatására a 3.1. ábra ad példát.) Létezik (4,8) képekre is olyan megoldás, ami 4 al-iterációt alkalmaz iterációs lépésként (Palágyi, 1993).

3D-ben csak két zsugorító eljárásról tudunk. Levialdi klasszikus, 2×2 -es maszkokat használó vegyes 2D algoritmusát (Levialdi, 1972) Arcelli és Levialdi terjesztette ki 3D-re (Arcelli, Levialdi, 1972). A 26-összefüggő objektumokat szét nem szakító módszer $2 \times 2 \times 2$ -es maszkokat használ. A másik eljárás redukciós típusú, a \mathbb{Z}^3 pontthalmazt a 2.15/a. ábra szerint bontja fel kettő almezőre (Hall, Küçük, 1992). Az almező-szekvenciális algoritmus $3 \times 3 \times 3$ -as maszkokkal törölve állítja elő a topológiai magot.

A 3.1. pontban a 6 al-iterációs, irány-szekvenciális párhuzamos, a (26,6) képekre kifejlesztett zsugorító algoritmusunkat ismertetjük.

3.1. Egy 6 al-iterációs párhuzamos 3D zsugorító algoritmus

A jelen pontban a $(26,6)$ képekre kifejlesztett redukciós zsugorító algoritmusunkat mutatjuk be. A 6 al-iterációs, irány-szekvenciális eljárást az alábbi séma írja le:

Input: a zsugorítandó $(26,6)$ képet reprezentáló X bináris tömb.

Output: az objektumok topológiai magjait tároló Y bináris tömb.

`6_al-iterációs_zsugorító_algoritmus(X, Y)`

```
begin
   $Y = X$ ;
  repeat
     $Y = \text{törlés\_u\_irányból}(Y)$ ;
     $Y = \text{törlés\_d\_irányból}(Y)$ ;
     $Y = \text{törlés\_n\_irányból}(Y)$ ;
     $Y = \text{törlés\_s\_irányból}(Y)$ ;
     $Y = \text{törlés\_e\_irányból}(Y)$ ;
     $Y = \text{törlés\_w\_irányból}(Y)$ ;
  until nem történt változás;
end.
```

Az alkalmazott törlési irányok ugyanazok, mint amiket általában az irány-szekvenciális 3D vékonyító algoritmusok használnak (2.2. ábra).

A `törlés_u_irányból` lokális párhuzamos redukció bizonyos u határpontokat töröl a 3.2. ábra maszkjai szerint. A fennmaradó öt al-iterációhoz tartozó képművelet maszkjai megkaphatók az u törlési irányhoz rendelt maszkok megfelelő elforgatásával és/vagy tükrözésével.

A következőkben bebizonyítjuk, hogy zsugorító algoritmusunk topológiailag korrekt.

3.1. TÉTEL

Az ismertett zsugorító algoritmus topológia-megőrző a $(26,6)$ képekre.

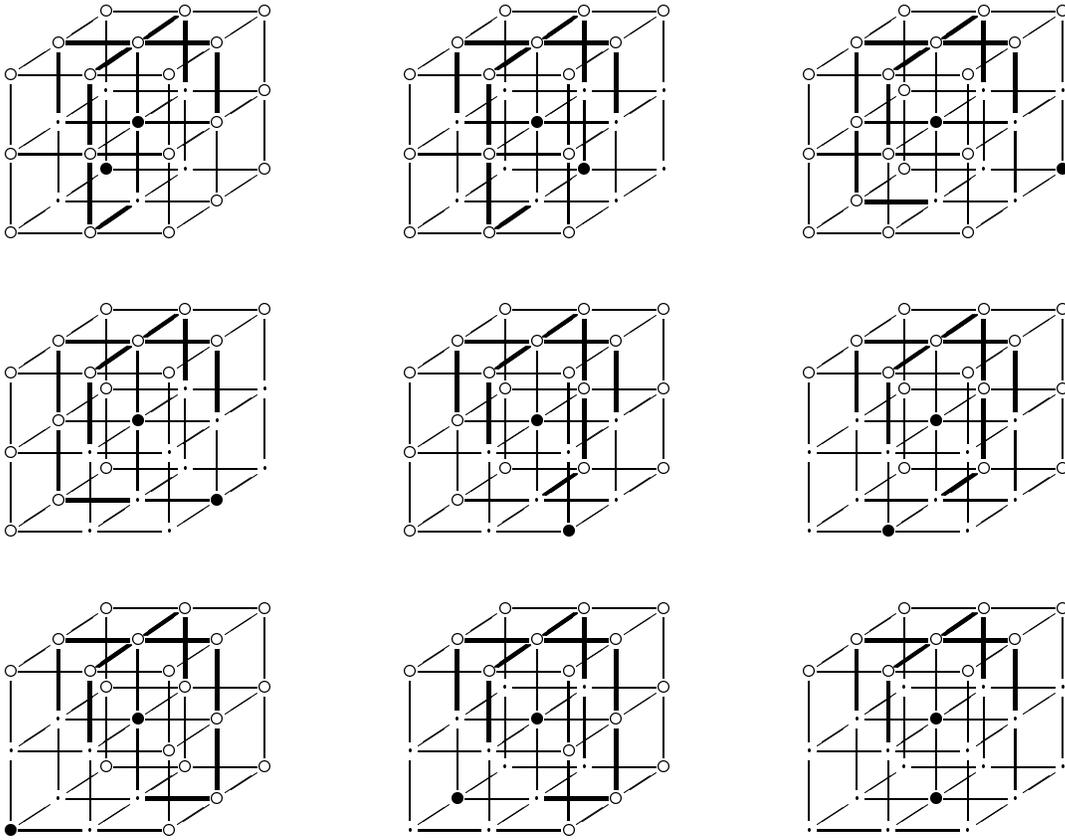
BIZONYÍTÁS

Megmutatjuk, hogy a 3.2. ábra maszkjaival megadott `törlés_u_irányból` párhuzamos redukcióra teljesülnek az 1.7. Tétel feltételei.

Osztályozzuk a maszkok elemeit:

- a *központi* elem a maszk középpontja,
- *fekete* az a maszkelem, amelyet a “•” szimbólum jelöl és nem központi elem,
- egy maszkelem *potenciálisan fekete*, ha a “.” szimbólum jelöli,
- *fehér* az a maszkelem, amelyet a “o” szimbólum jelöl.

1. Legyen $\mathcal{P} = (\mathbb{Z}^3, 26, 6, B)$ egy tetszőleges 3D kép. Legyen továbbá $p \in B$ egy tetszőleges olyan fekete pont és $Q \subseteq (N_{18}(p) \setminus \{p\}) \cap B$ egy tetszőleges olyan fekete ponthalmaz a \mathcal{P} képen, hogy p -t és Q minden egyes pontját törli a 3.2. ábra kilenc maszkjának valamelyike. Megmutatjuk, hogy ekkor p egyszerű a $(\mathbb{Z}^3, 26, 6, B \setminus Q)$ képen.



3.2. ábra A zsugorító eljárásnak az u törlési irányhoz tartozó maszkjai. A vastagított vonalak a topológiai megőrzéshez szükséges 6-utakat jelölik.

Tegyük fel, hogy $Q = \emptyset$. Ekkor azt kell belátnunk, hogy a 3.2. ábra valamennyi maszkja csak egyszerű pontokat töröl. A maszkokra illeszkedő p pontok egyszerű voltát az 1.3. Kritérium segítségével igazoljuk.

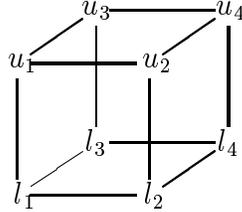
- (a) A $N_{26}(p) \cap (B \setminus \{p\})$ ponthalmaz pontosan egy 26-komponensből áll, mivel mindegyik maszk csak egyetlen fekete elemet tartalmaz, ami 26-szomszédos valamennyi potenciálisan fekete maszkelemmel.
- (b) $(\mathbb{Z}^3 \setminus B) \cap N_6(p) \neq \emptyset$, mivel a központi maszkelem felső (u) 6-szomszédja valamennyi maszkon fehér.
- (c) A 3.2. ábra vastagított vonalai 6-összefüggő utakat reprezentálnak a $(\mathbb{Z}^3 \setminus B) \cap N_6(p)$ ponthalmaz bármely két pontja között a $(\mathbb{Z}^3 \setminus B) \cap N_{18}(p)$ halmazban.

A fenti ($Q = \emptyset$) esetben p egyszerű volta nem függött a potenciálisan fekete maszkelemekkel fedésbe kerülő pontok színétől. Megállapíthatjuk továbbá, hogy a p pontot törölő maszk egyetlen fekete elemével fedésbe kerülő fekete pont nem törölődhet, mivel a maszkok fekete pontjai a maszk alsó 3×3 -as lapján találhatóak, ugyanakkor a felső 3×3 -as lap valamennyi eleme fehér.

Tegyük fel, hogy $Q \neq \emptyset$. Ekkor egyetlen $q \in Q$ képpont sem kerülhet fedésbe fekete elemmel, mivel akkor q nem lenne törölhető (ami ellentmond Q választásának). Ennélfogva Q pontjai csak potenciálisan fekete pontokkal kerülhetnek fedésbe. Ekkor (a $Q = \emptyset$ esetben bizonyítottak szerint) Q törlése nem változtat p egyszerű voltán.

2. Megmutatjuk, hogy a `törlés_u_irányból` képművelet egyetlen egység-rácskocka által tartalmazott objektumot sem töröl teljesen.

Legyen C egy olyan objektum, amit tartalmaz a 3.3. ábrán látható egység-rácskocka. A kocka az $U = \{u_1, u_2, u_3, u_4\}$ (a rácskocka felső lapja) és az $L = \{l_1, l_2, l_3, l_4\}$ (a rácskocka alsó lapja) halmazokból áll. Mivel a rácskocka tartalmazza C -t, így $C \subseteq U \cup L$.



3.3. ábra Egység-rácskocka, amelynek nyolc pontját az $U = \{u_1, u_2, u_3, u_4\}$ és az $L = \{l_1, l_2, l_3, l_4\}$ halmazok egyesítése adja.

Két esetet különböztetünk meg:

- (a) $C \cap U = \emptyset$.

Ekkor $C \cap L \neq \emptyset$, mivel minden komponens tartalmaz legalább egy pontot. Ez esetben $C \cap L$ egyetlen pontja sem törölhető, mivel a maszkokban a fekete elemek csak az alsó 3×3 -as lapon találhatóak. Ennélfogva C nem törölhető teljesen.

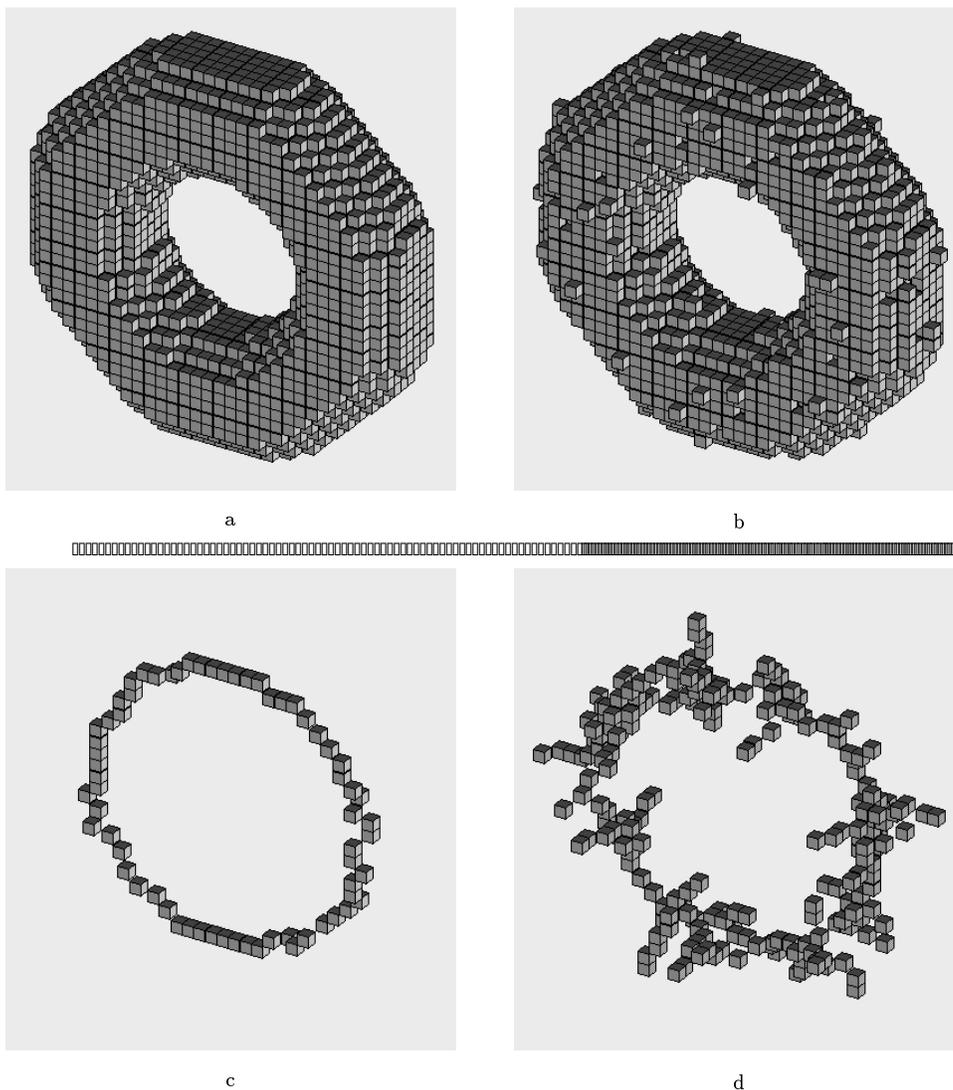
- (b) $C \cap U \neq \emptyset$.

Ekkor $C \cap U$ valamely eleme csak akkor törölhető, ha $C \cap L \neq \emptyset$, mivel a maszkokban csak az alsó lapon található fekete elem. Visszakaptuk az (a) esetet, amikor is C nem volt teljesen törölhető.

A fentiekben megmutattuk, hogy a `törlés_u_irányból` párhuzamos redukció kielégíti az 1.7. Tétel minkét feltételét. A (26,6) képek topológiájának megőrzése a többi öt al-iteráció esetén hasonlóképpen igazolható. A teljes zsugorító algoritmus topológia-megőrző, mivel topológia-megőrző képműveletek sorozatából áll. ■

4. Elő- és utófeldolgozás

A vékonyítással előállított vázközelítés — hasonlóan az 1.1. pontban bemutatott folytonos vázhoz (1.1b. ábra) — érzékeny az objektum határának egyenetlenségeire. A vékonyítás zajérzékenységét a 4.1. ábra szemlélteti, ahol a középvonalak kivonásakor *parazita* ágak, vonalszegmenssek keletkeztek.



4.1. ábra Példa a vékonyítás zajérzékenységére. Az eredeti objektum (a) bizonyos határpontjainak törlésével és bizonyos, a határpontokkal 6–szomszédos fehér pontok feketévé változtatásával kaptuk a zajos objektumot (b). A zajmentes objektum középvonala (c) és a zajterhelt változatra kapott eredmény (d).

A bináris objektumokat előállító képkalkító berendezések vagy a szegmentáló módszerek nem garantálják a határ “simaságát”, így a fenti probléma megoldására a vázkijelölést megelőző *simítás* (*smoothing, filtering*) mint előfeldolgozó lépés és/vagy a váz *tisztítása* (*pruning*) mint utófeldolgozás kínálkozik. Lee, Kashyap és Chu megemlíti egy

harmadik lehetőséget is, ami a vékonyító módszereknél alkalmazható: simítsunk a vékonyítás minden egyes iterációs lépése után¹⁹ (Lee, Kashyap, Chu, 1994).

Az előfeldolgozással, azaz a vékonyítandó objektumok simításával, zajszűréssel a 4.1. pontban foglalkozunk. Az utófeldolgozást, vagyis a vékonyítással kapott váz tisztítását a 4.2. pontban tárgyaljuk.

¹⁹Ilyen eljárást eddig senki sem dolgozott ki, még Lee, Kashyap és Chu sem. Ők azért nem, mivel az algoritmusuknak már eleve simító hatást tulajdonítanak.

4.1. A vékonyítandó objektumok simítása

A többszintű (*multi-level*) 2D digitális képek zajszűrésére, zajcsökkentésére számos olyan módszert javasoltak (Schalkoff, 1989; Gonzales, Woods, 1992), amelyek nem bizonyultak alkalmasnak bináris képekre²⁰. Helyettük leggyakrabban a morfológiai szűrést (1.4.1.) alkalmazzák, ami a határ simítása mellett törli a “kisméretű” objektumokat és kitölti a “kisméretű” üregeket is. A morfológiai szűrés hatását a 4.2. ábrával szemléltetjük.



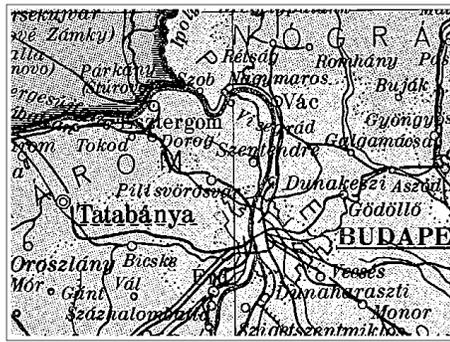
4.2. ábra Példa morfológiai szűrésre. Az (a) — 480×640 méretű — térképrészletre az eljárás a (b) képet eredményezte. A nem-törölt komponensek is megváltoztak, számos helységnév olvashatatlanná vált. (A szerkesztőelem választott mérete: 5×5 .)

A részletek jobb megőrzése elérhető a módszer finomításával, a *lágymorfológiai szűréssel* (*soft morphological filtering*) (Kuosmanen, Koskinen, Astola, 1992). Palágyi egy olyan morfológiai algoritmust javasolt, ami csak adott méretnél kisebb (vagy nagyobb, vagy két mérethatár közé eső) objektumokat távolít el a képről, a többit pedig változtatás nélkül megőrzi (Palágyi, 1993). Ez utóbbi, lokális és párhuzamos — 3D-re is adaptálható — képműveletekből építkező módszerre a 4.3. ábrával adunk példákat.

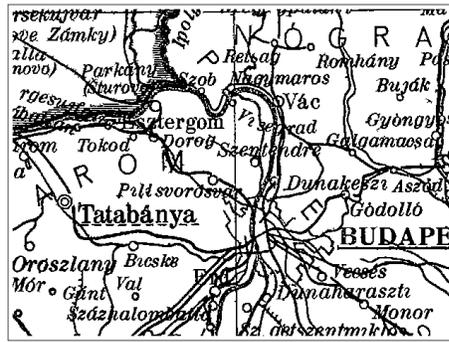
A morfológiai megközelítés mellett egyéb módszerek is léteznek, mint például Ray szekvenciális eljárása (Ray, 1988), mely különböző méretű (3×3 -tól 5×5 -ig terjedő) lokális környezetet vizsgálva dönt az objektumpontok megtartásáról vagy törléséről. Az eljárás hátránya, hogy korlátozott a zaj mérete.

Ali és Pavlidis kontúrkövetéses módszerével “gyanús” összeérések és szakadások szüntethetők meg (Ali, Pavlidis, 1982). További kontúrsimításon alapuló módszereket is kidolgoztak: Yu és Yan (Yu, Yan, 1997) az objektum *lánckóddal* (*chain code*) (Gonzales, Woods, 1992; Marchand–Maillet, Sharaiha, 2000) leírt határvonalát egyszerűsíti bizonyos kételemű vektorkonfigurációk egyetlen vektorral való helyettesítésével. Arcelli és Sanniti di Baja euklidészi távolságtranszformáción alapuló módszerére (Arcelli, Sanniti di Baja, 1998) Ragnemalm adott hatékony implementációt. (Ragnemalm, 1992/2).

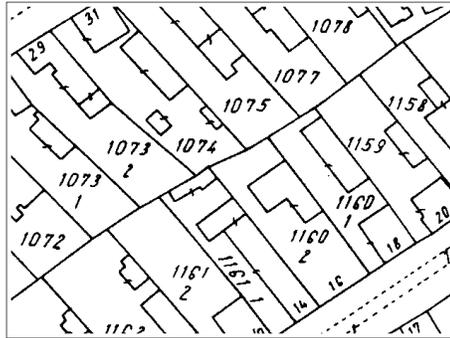
²⁰A többszintű képekre alkalmazott *mediánszűrés* (*median filtering*) hatékony módszer bináris képekre is (Marchand–Maillet, Sharaiha, 2000). Ráadásul bináris kép esetén — rendezés helyett — elegendő az objektumpontok számát összehasonlítani a vizsgált környezet elemszámának a felével.



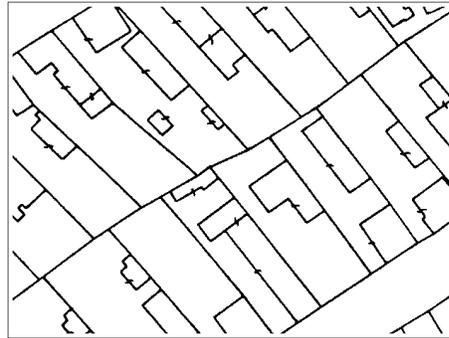
a



b



c



d

4.3. ábra Példák zajszűrésre. Az eljárás az (a) — 480×640 méretű — térképészletre a (b) képet eredményezte. Az eltávolítandó 8-összefüggő objektumok, zajok maximális méretének 4-et adtuk meg. (Az eljárást ugyanarra a képre hajtottuk végre, mint a 4.1. ábrán szemléltetett morfológiai szűrés. Ezúttal a nem-törölt komponensek változtatás nélkül megőrződtek, valamennyi helységnév olvasható maradt.) A második példánkon a (c) — 480×640 -es, egy kataszteri térkép részletét tartalmazó — képről a 10-nél kisebb vagy egyenlő méretű 8-összefüggő objektumokat távolítottuk el. Az eljárás a (b) képet eredményezte, melyen a helyrajzi számok és a szaggatott vonalak már nem szerepelnek.

4.2. A vékonyítással kapott váz tisztítása

A vékonyítás bemenete és kimenete egyaránt egy-egy bináris kép, ennél fogva a vékonyítással kivont váz utófeldolgozására, a parazita szegmensek eltávolítására nehéz fogódzót találni. Nem úgy, mint pl. a hierarchikus Voronoi–diagramok esetében, ahol a Voronoi–váz elemeire számos *fontossági mértéket* (*importance measure*) javasoltak (Meyer, 1989; Talbot, Vincent, 1992; Attali, Montanvert, 1994; Näf, 1996; Attali, Montanvert, 1997; Shaked, Bruckstein, 1998).

A középvonalakat kivonó vékonyítás eredményének simítására a *morfológiai tisztítást* (*morphological pruning*) (Gonzales, Woods, 1992) javasolják, ami egy adott konstansnál rövidebb vonalszegmenseket tekint zajnak, jöllehet a határ “kismértékű” egyenetlensége is eredményezhet “hosszú” parazita ágakat (1.1b. ábra). A morfológiai tisztítás összetett eljárás, amiben a 3. pontban tárgyalt zsugorításnak jut kulcsszerep.

A következőkben a morfológiai tisztítást ismertetjük az 1.4. pontban alkalmazott jelölésekkel.

Input:

- a tisztítandó képet reprezentáló A bináris tömb,
- az eltávolítandó parazita ágak d (maximális) hossza,
- a zsugorítás (mint speciális vékonyító operátor) egy lépését megadó X szerkesztőelem-rendszer, és
- a dilatáció Y szerkesztőeleme.

Segédtömb:

az A bináris tömbbel megegyező méretű C tömb.

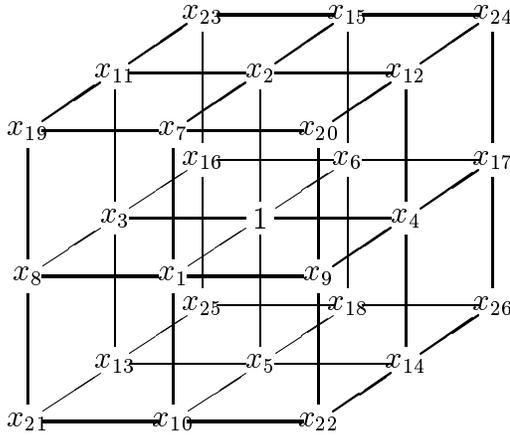
Output: a tisztított képet tároló B bináris tömb.

```
1.  pruning( $A, d, X, Y, B$ )
2.    begin
3.       $B = A$ ;
4.      for  $i = 1$  to  $d$  do
5.         $B = B \otimes X$ ;
6.      endfor
7.       $C = \text{endpoints}(B)$ ;
8.      for  $i = 1$  to  $d$  do
9.         $C = C \oplus Y$ ;
10.      $C = \text{and}(C, A)$ ;
11.     endfor
12.      $B = \text{or}(B, C)$ ;
13.     return( $B$ );
14.   end
```

A morfológiai tisztítás egy d -lépéses zsugorítással indul (4–6. utasítások), melynek során a vékonyított kép legfeljebb d hosszú vonalszegmensei nyomtalanul eltűnnek. Ugyanakkor a d -nél hosszabb, vagyis a zajnak nem minősülő ágak megmaradnak, de a hosszuk d -vel csökken. Megjegyzendő, hogy olyan X szerkesztőelem-rendszert kell alkalmazni, amelyik egyrészt törli a vonalvégpontokat, másrészt pedig megőrzi a topológiát.

Az eljárás a továbbiakban a zajnak nem minősülő ágakat állítja helyre a végpontokból növesztve.

A 7. utasítás végrehajtásával a B -ben tárolt zsugorított kép vonalvégpontjai a C tömbbe kerülnek. (C -ben a végpontok pozícióján állnak egyesek, minden más képpont értéke nulla.) Az “endpoints” lokális és párhuzamos képművelet egyszerűen megadható, pl. 3D-ben a következő maszkkal és csatolt formulával:



(6, 26) képekre:

$$\sum_{i=1}^6 x_i \neq 1,$$

(26, 6) képekre:

$$\sum_{i=1}^{26} x_i \neq 1.$$

(Mindazon “1” pontok “0”-ra változnak, amelyek 6-, illetve 26-szomszédai között csak nulla vagy egyél több “1” szerepel, vagyis csak a vonalvégpontok maradnak meg “1” értékűnek.)

A 8–11. utasítások d -szer hízlalják a C képet (alkalmasan választott Y szerkesztőelemmel végrehajtott) dilatációval. A kiindulási A képen nem szereplő részletek eltávolítására szolgál a pontonkénti konjunkciót jelentő “and” képművelet. A hízlalás és a visszanyesés váltakozó ismétlésével, vagyis a *felügyelt dilatációval* (*controlled dilation*) a C képtömbben előállnak a zajnak nem minősülő ágak hiányzó végei.

A B és a C képek pontonkénti diszjunkciója, vagyis a 12. utasítás “or” parancsa éppen a kívánt hatást eredményezi, vagyis a “rövid” ágak eltűnését és a “hosszúak” megmaradását.

A morfológiai tisztításra a 4.4. ábrával adunk egy 2D példát, ami rámutat az eljárás egy fogyatékoságára is. Ha egy megőrzendő “hosszú” vonalszegmens végétől $2d$ -nél rövidebb távolságra ágazik le egy “rövid” vonal, akkor az eltávolítandó ágot (részben vagy teljes egészében) rekonstruálja a felügyelt dilatáció. Az eljárásnak ezt a hibáját sok esetben orvosolja a morfológiai tisztítás ismételt, egyre növekvő d értékekre történő végrehajtása.

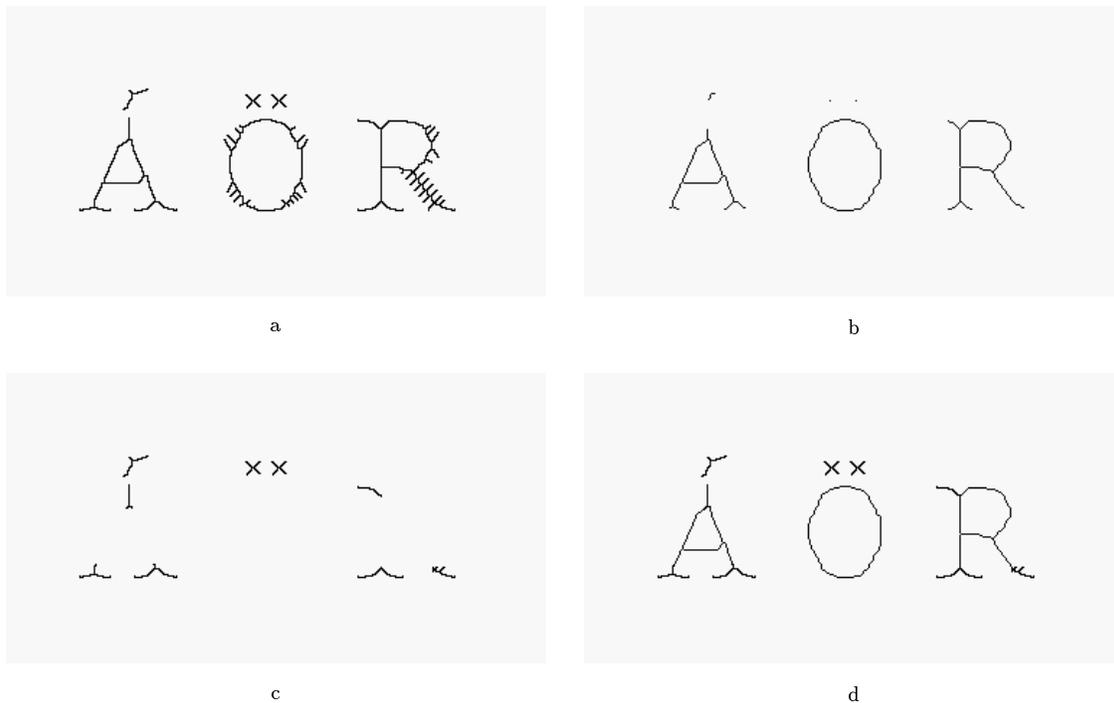
A módszer hatását erősen befolyásolja az, hogy milyen zsugorító módszert alkalmazunk. Egy ág ugyanis nem akkor tűnik el, ha a benne szereplő fekete pontok száma legfeljebb d , hanem akkor, ha őt törli a zsugorítás d lépésben. Az irány-szekvenciális zsugorító eljárások egy lépése az ágakból több pontot is törölhet. (3D-ben egy vonalvégpontnak legalább öt fehér 6-szomszédja van, így egyszerre legalább ötféle típusú határpontnak bizonyul. Ha egy végpont törlődik egy al-iteráció során, akkor valószínű,

hogy a szomszédját is törli ugyanazon iterációs lépés következő al-iterációja.) Ennélfogva a d -lépéses zsugorításnak alávett megőrzendő “hosszú” ágakat gyakran nem képes rekonstruálni a d -lépéses felügyelt dilatáció.

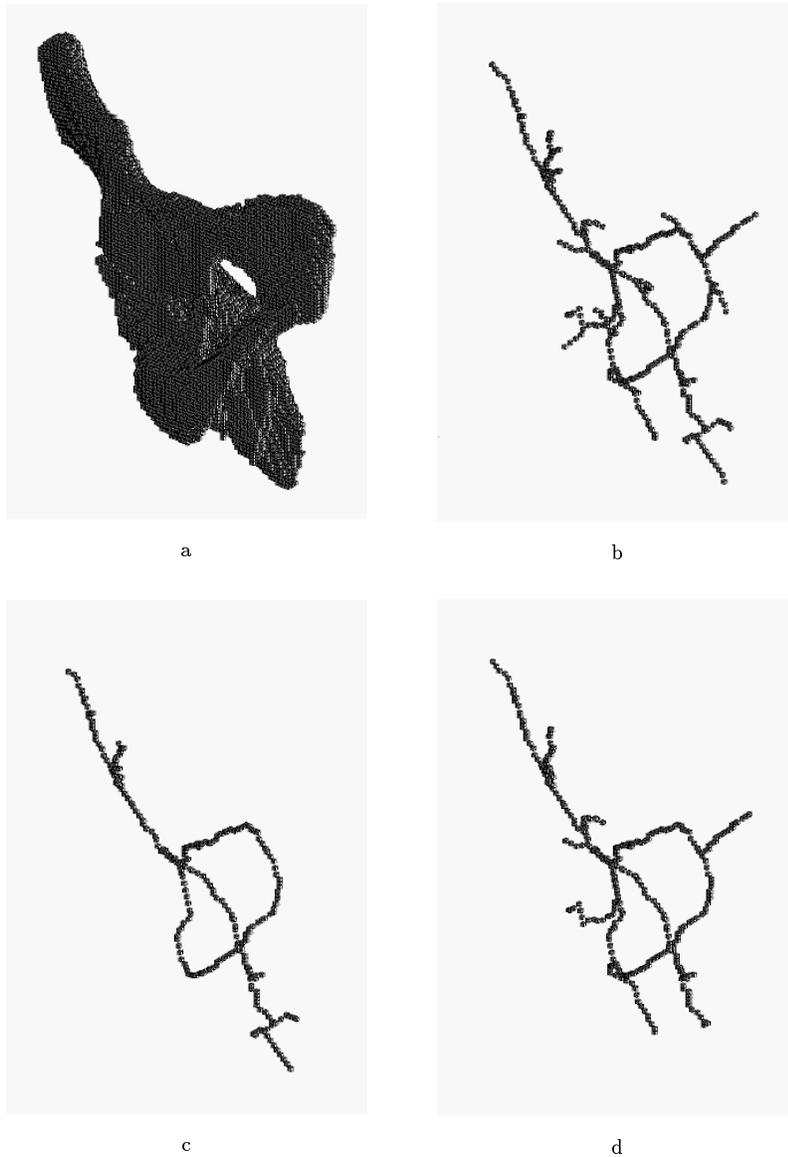
A fenti probléma megoldására két lehetőség kínálkozik. Az egyik módszer az, hogy d -nél több lépésben alkalmazzunk felügyelt dilatációt. (Pl. egy 6 al-iterációs, irány-szekvenciális 3D zsugorítást biztosan kompenzál a felügyelt dilatáció $6d$ lépése.) Ez a technika viszont egyrészt időigényes, másrészt továbbra sem ad korrekt mértéket az ágak hosszára. Ugyanis egy olyan ág, aminek az egymást követő pontjai 6-szomszédosak, hatszor rövidebbnek is bizonyulhat, mint egy olyan ág, ahol a szomszédos pontok 18- vagy 26-szomszédosági relációban állnak egymással.

A második lehetőség az, hogy a szakirodalomban (Gonzales, Woods, 1992) javasolt eljárásban a zsugorítás helyett — kihasználva, hogy vékonyított képpel dolgozunk — egy jóval egyszerűbb képműveletet alkalmazunk, ami csak a vonalvégpontokat törli. Ez a művelet hasonlít a fentebb bemutatott “endpoints” képműveletre. Az eltérés csak annyi, hogy a maszkhoz csatolt formulában “ $\neq 1$ ” helyett ekkor “ $= 1$ ” szerepel. A módosított eljárás hátránya az, hogy bizonyos elágazás-kombinációknál megakadhat, ráadásul az olyan elágazásokig törölt ágdarabokat sem képes helyreállítani.

Morfológiai tisztításra a 4.5. ábra mutat 3D példát.



4.4. ábra Példa morfológiai tisztításra. A kiindulási vékonyított (200×375 -ös) képről (a) a legfeljebb $d = 10$ hosszú ágakat kívántuk eltávolítani. A zsugorítás után a (b), a felügyelt dilatáció végrehajtásával pedig a (c) képet kaptuk. A végeredményt, a (d) képet a (b) és a (c) képek pontonkénti diszjunkciója adta. Az eljárás nem változtatta meg az “**Á**” betűt és a várt eredményt hozta a “**Ö**”-re is. Az “**R**” ferde szárát viszont nem volt képes tökéletesen megtisztítani, mivel annak közel a végéhez is csatlakoztak olyan parazita ágak, amelyeket a felügyelt dilatáció (részben vagy teljes mértékben) helyreállít. Az eljárás Palágyi (8,4) képekre javasolt, 2 al-iterációs zsugorító algoritmusával dolgozott (Palágyi, 1993) és a 8-szomszédoságnak megfelelő dilatációt alkalmazta.



4.5. ábra Példa 3D morfológiai tisztításra. Az eredeti (a) kép egy 3D CT vizsgálatból szegmentálással előállított csípőcsont-részletet tartalmaz. Az (a) képből kivont (b) középvonalakról leválogattuk a legfeljebb $d = 10$ hosszú parazita ágakat. A tisztítás az eredeti zsugorítási módszerrel a (c), a vonalvégpontok törlését alkalmazó, módosított eljárással pedig a (d) képet eredményezte. Az első esetben a 3.1. pontban ismertett zsugorító módszerünket használtuk.

5. A 3D vékonyító algoritmusaink eredményei

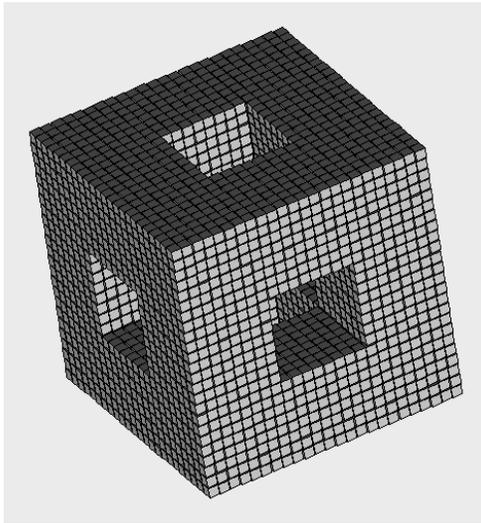
A jelen fejezetben az ismertetett 3D vékonyító algoritmusaink eredményeit mutatjuk be “mesterséges” és “természetes” képekre. Az ábrák a bináris képekre kifejlesztett megjelenítő rendszerünkkel (Oltyán, 1997) és a 3DVIEWNIX (MIPG, University of Pennsylvania, Philadelphia) orvosi képfeldolgozó rendszerrel (Udupa et al., 1994) készültek.

Hét algoritmusunkról mutatunk eredményeket, melyekhez a következő rövidítéseket rendeljük:

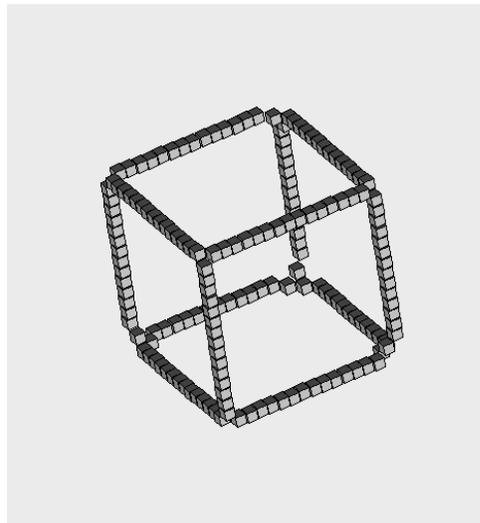
- **PK-6-C**
6 al-iterációs, irány-szekvenciális algoritmus középvonalak kivonására (2.1.6.)
(Palágyi, Kuba, 1997/3; Palágyi, Kuba, 1998/1),
- **PK-6-S**
6 al-iterációs, irány-szekvenciális algoritmus középfelszín kivonására (2.1.6.)
(Palágyi, Kuba, 1997/3; Palágyi, Kuba, 1998/1),
- **PK-8-C**
8 al-iterációs, irány-szekvenciális algoritmus középvonalak kivonására (2.1.7.)
(Palágyi, Kuba, 1999/1),
- **PK-8-S**
8 al-iterációs, irány-szekvenciális algoritmus középfelszín kivonására (2.1.7.)
(Palágyi, Kuba, 1999/1),
- **PK-12-C**
12 al-iterációs, irány-szekvenciális algoritmus középvonalak kivonására (2.1.8.)
(Palágyi, Kuba, 1997/1; Palágyi, Kuba, 1997/2; Palágyi, Kuba, 1999/2).
- **PK-12-S**
12 al-iterációs, irány-szekvenciális algoritmus középfelszín kivonására (2.1.8.)
(Palágyi, Kuba, 1999/2).
- **PK-H-C**
hibrid algoritmus középvonalak kivonására (2.4.1.)
(Palágyi, Kuba, 1998/2).

Az 5.1–5.7. ábrák szintetikus objektumokra mutatják be a vékonyító algoritmusaink hatását. Az 5.7. ábra azt illusztrálja, hogy a középvonalakat kivonó algoritmusaink képesek parazita ágaktól mentes eredményt produkálni zajos objektumra is. (Meg kell jegyezni, hogy minden egyes vázkijelölő technika érzékeny a zajra, az objektum határának egyenetlenségeire. Bizonyos mértékű zaj már nem tolerálható, így nem-kívánatos vonal- vagy felszín-szegmensek jelentkeznek a “vázon”.) Az 5.8–5.14. ábrákon természetes objektumokra láthatók a vékonyító algoritmusaink eredményei.

Topológia-megőrző algoritmusainkat még számos más objektumra is teszteltük — közel nyolcvan 3D bináris képet állítottunk elő vagy kaptunk másoktól —, de helyhiány miatt csak néhányukra tudjuk bemutatni algoritmusaink hatását.



eredeti objektum



PK-6-C

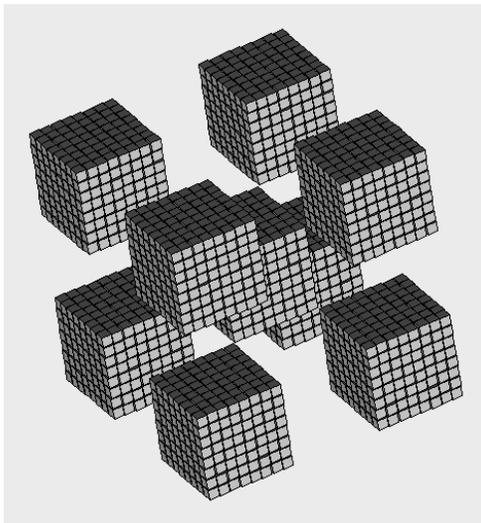


PK-8-C

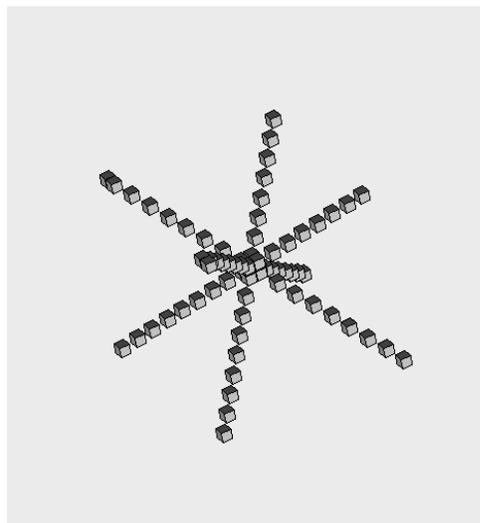


PK-12-C

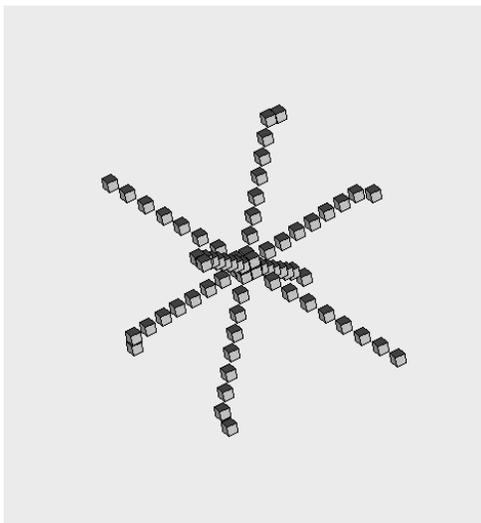
5.1. ábra A három irány-szekvenciális, középvonalakat kivonó vékonyító algoritmusunk eredménye egy $24 \times 24 \times 24$ -es átfűrt kockára. Megállapíthatjuk, hogy a 8 és a 12 törlési irányt alkalmazó eljárások (PK-8-C, PK-12-C) — szemben a hagyományos 6 al-iterációssal (PK-6-C) — nem őrzik meg a koordinátatengelyek irányába eső éleket. A 8 és a 12 al-iterációs algoritmusok eredménye azért szabálytalan, mivel a túl kicsi tesztobjektumra azok az eljárások egyetlen iterációs lépésben terminálnak.



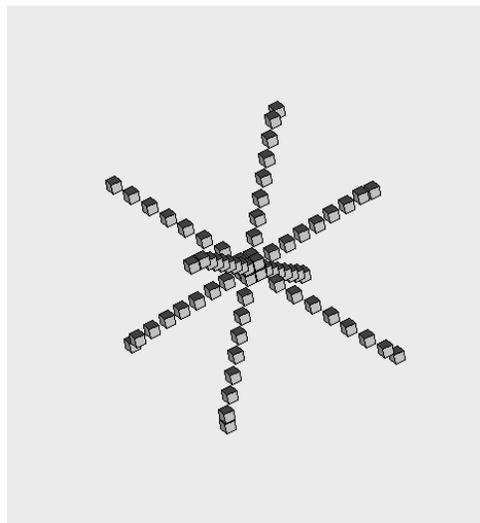
eredeti objektum



PK-6-C

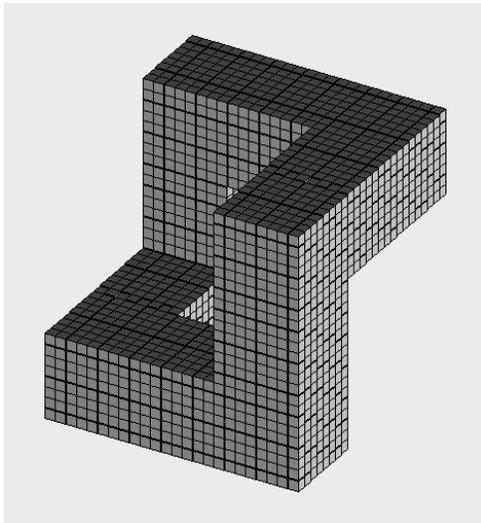


PK-8-C

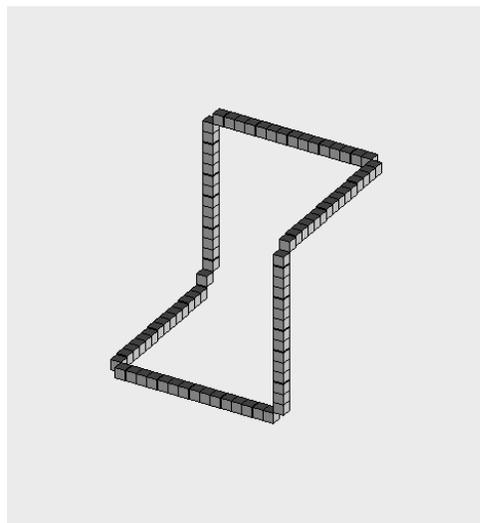


PK-12-C

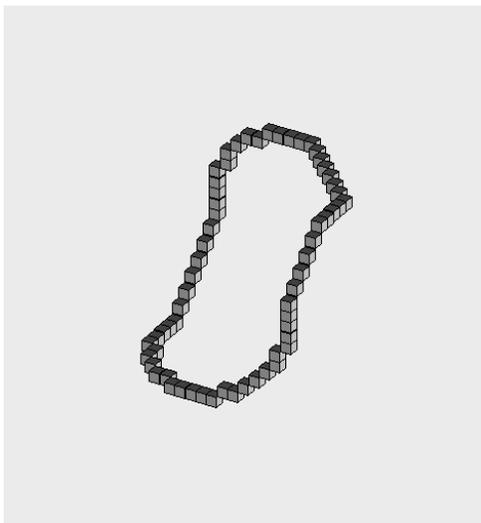
5.2. ábra A három irány-szekvenciális, középvonalakat kivonó vékonyító algoritmusunk eredménye egy $24 \times 24 \times 24$ -es szintetikus objektumra. Mindhárom eljárás szabályos és szimmetrikus “vázat” produkált. Megjegyzendő, hogy a 8 és a 12 al-iterációs algoritmusok (PK-8-C, PK-12-C) ez esetben egyetlen iterációs lépést igényeltek.



eredeti objektum



PK-6-C

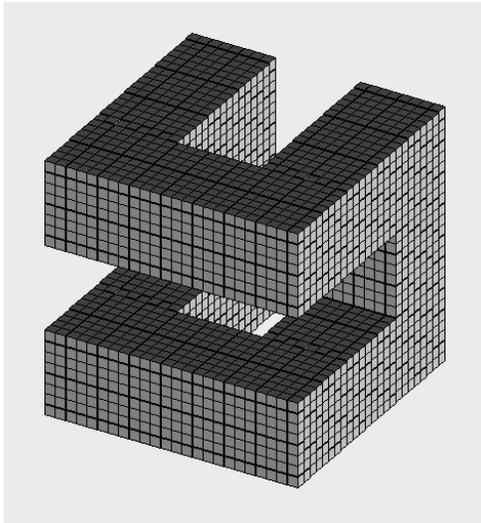


PK-8-C

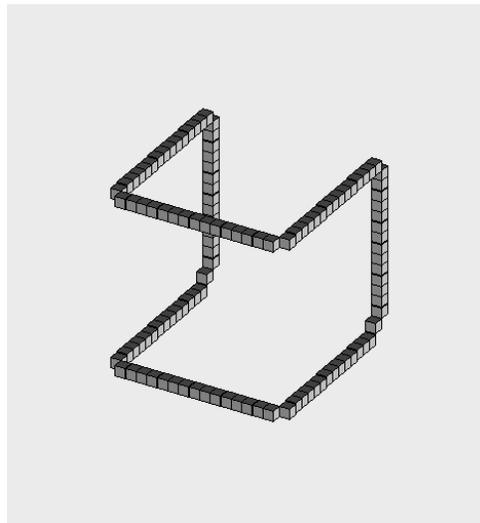


PK-12-C

5.3. ábra A három irány–szekvenciális, középvonalakat kivonó vékonyító algoritmusunk eredménye egy $24 \times 24 \times 24$ -es szintetikus objektumra. Megállapíthatjuk, hogy a 8 és a 12 törlési irányt alkalmazó eljárások (PK-8-C, PK-12-C) — szemben a hagyományos 6 al–iterációssal (PK-6-C) — nem őrzik meg a koordinátát engelyek irányába eső éleket. Ez a “lekerekítő” hatás előnyös a természetes objektumok vékonyításakor.



eredeti objektum



PK-6-C



PK-8-C



PK-12-C

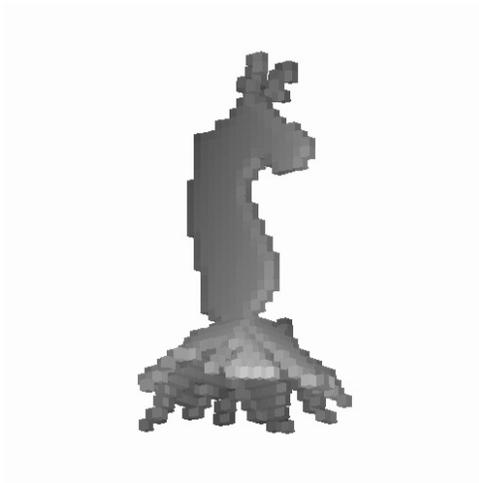
5.4. ábra A három irány–szekvenciális, közép vonalakat kivonó vékonyító algoritmusunk eredménye egy $24 \times 24 \times 24$ -es szintetikus objektumra. Megállapíthatjuk, hogy a 8 és a 12 törlési irányt alkalmazó eljárások (PK-8-C, PK-12-C) — szemben a hagyományos 6 al–iterációssal (PK-6-C) — nem őrzik meg a koordinátát engelyek irányába eső éleket. Ez a “lekerekítő” hatás előnyös a természetes objektumok vékonyításakor.



eredeti objektum



PK-6-S

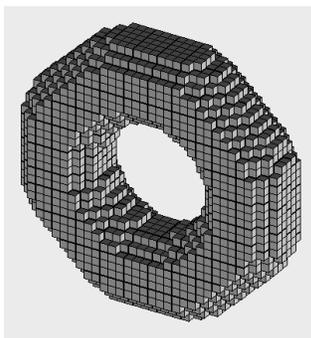


PK-8-S

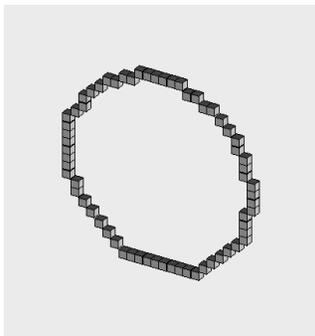


PK-12-S

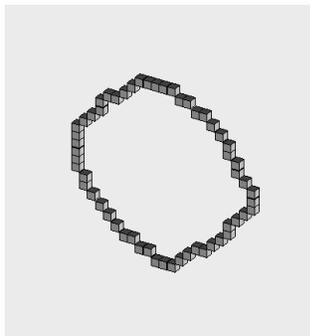
5.5. ábra A középfelszín kivonására javasolt vékonyító algoritmusaink eredményei egy sakkfigurára. Mindhárom eljárás “egy voxel vékony” felszínszegmenseket produkált. A kiindulási objektum mérete: $25 \times 40 \times 25$. A tesztképet Pierre Tellier (LSIIT, Université Louis Pasteur, Strasbourg) biztosította.



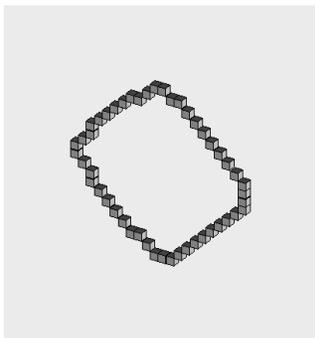
eredeti objektum



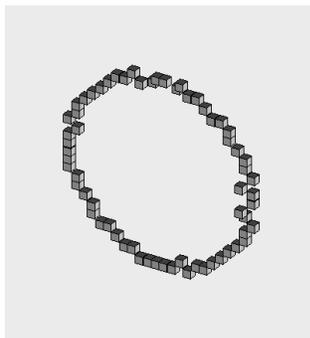
PK-6-C



PK-8-C

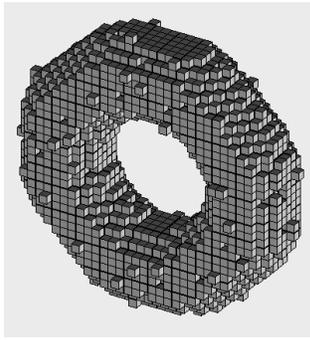


PK-12-C

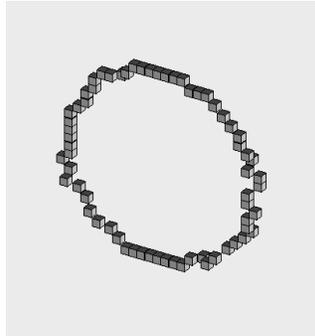


PK-H-C

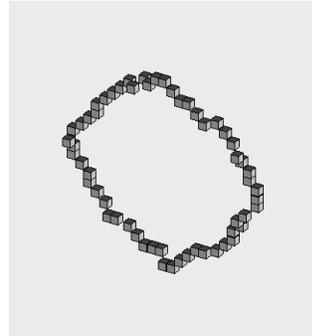
5.6. ábra A középvonalakat kivonó vékonyító algoritmusaink eredményei egy szintetikus tóruszra. Az eredeti tesztkép mérete: $40 \times 40 \times 20$. Valamennyi eljárás vékony zárt görbét produkált. (A “cakkozást” a sakktáblaszerű almezős felbontás okozza a hibrid (PK-H-C) algoritmus eredményén.)



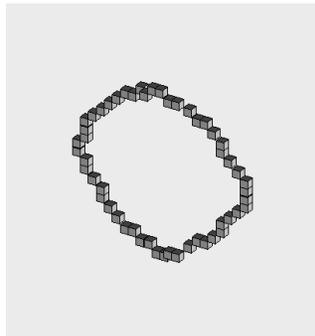
eredeti objektum



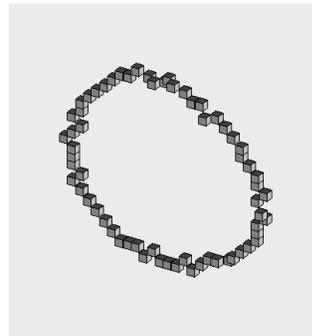
PK-6-C



PK-8-C

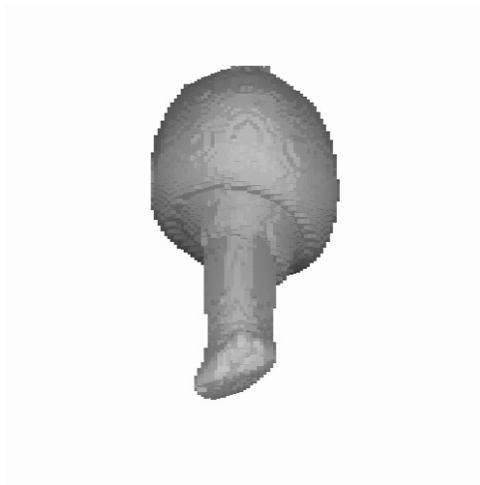


PK-12-C

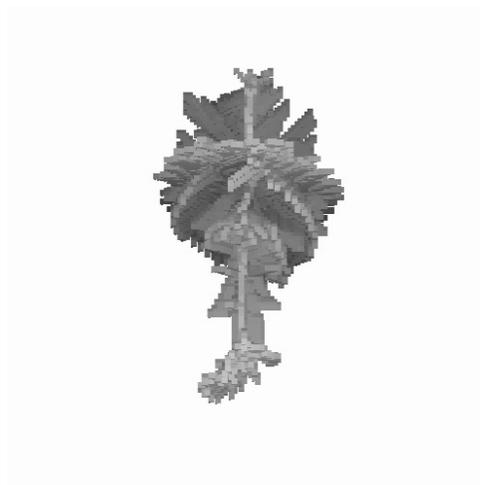


PK-H-C

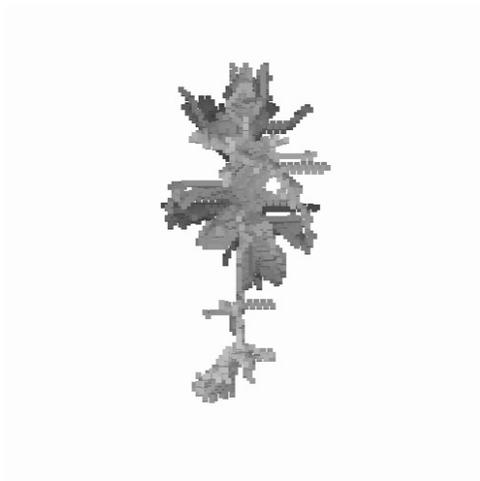
5.7. ábra A középvonalakat kivonó vékonyító algoritmusaink eredményei egy zajos tóruszra. Az objektumot “vegyes kontúzájjal” terheltük, vagyis bizonyos fekete határpontokat fehérre, bizonyos fekete határpontokkal 6-szomszédos fehér pontokat pedig feketére változtattunk. Az eredeti tesztkép mérete: $40 \times 40 \times 20$. Valamennyi eljárás vékony és parazita ágaktól mentes zárt görbét produkált.



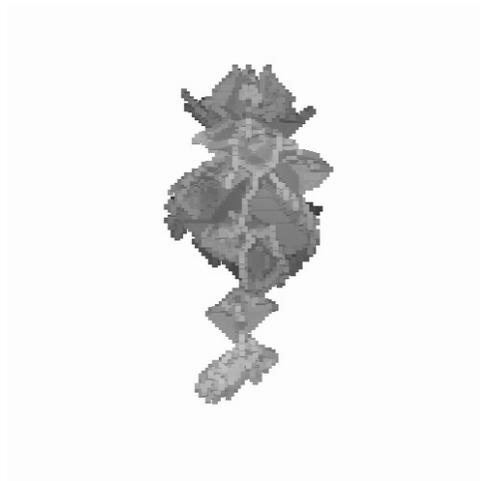
eredeti objektum



PK-6-S



PK-8-S

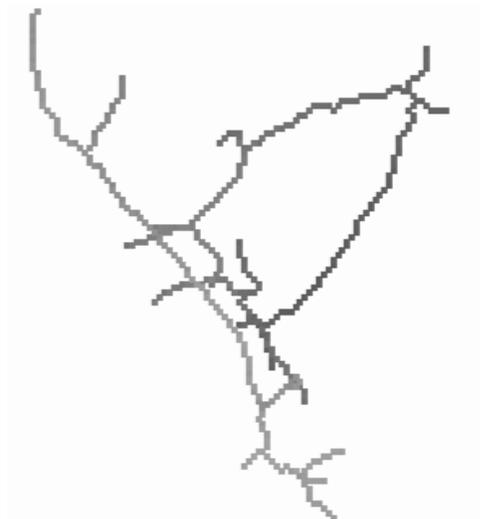


PK-12-S

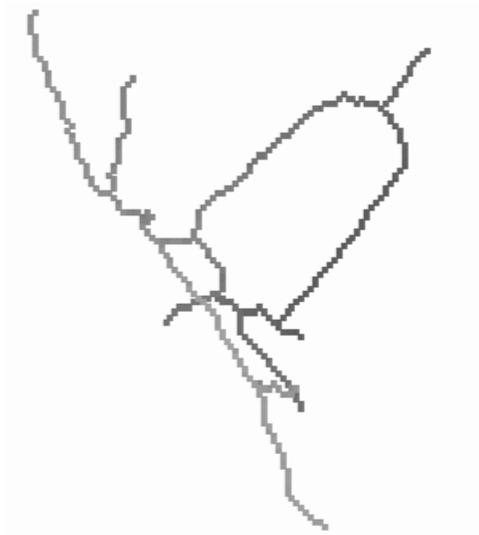
5.8. ábra A középfelzín kivonására javasolt vékonyító algoritmusaink eredményei egy gombára, ahol az eredeti objektum $88 \times 101 \times 57$ -es. A tesztképet Pierre Tellier (LSIIT, Université Louis Pasteur, Strasbourg) biztosította.



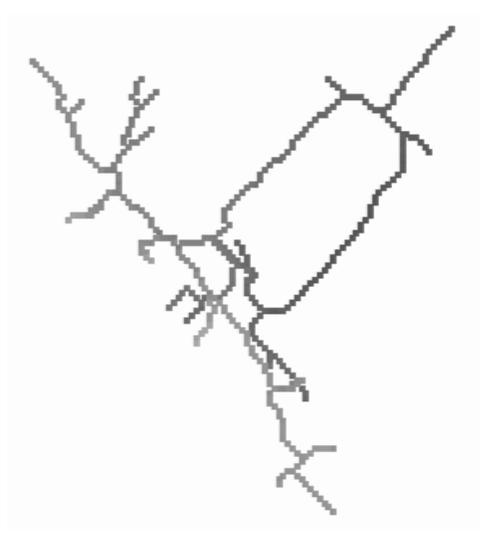
eredeti objektum



PK-6-C

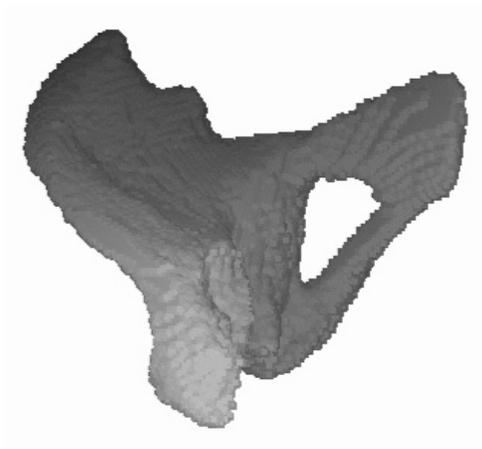


PK-8-C

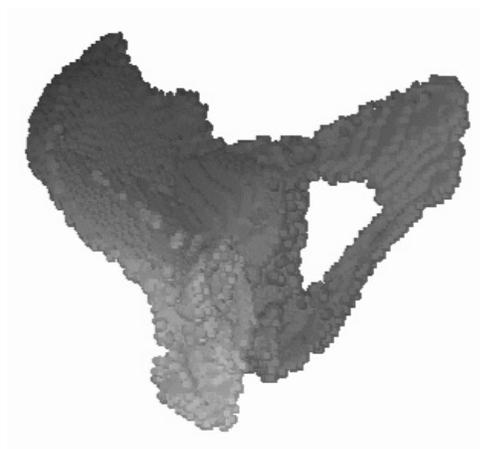


PK-12-C

5.9. ábra A három irány-szekvenciális, közép vonalakat kivonó vékonyító algoritmusunk eredménye egy CT vizsgálatból szegmentált csípőcsont $128 \times 128 \times 57$ -es részletére. A tesztképet Gábor Székely (ETH, Zürich) biztosította. Az eljárások “hasznoló” eredményt produkáltak. Ezen objektumra a legkevesebb ágat a 8 al-iterációs (PK-8-C) eljárás növesztette.



eredeti objektum



PK-6-S



PK-8-S



PK-12-S

5.10. ábra A középfelület kivonására javasolt vékonyító algoritmusaink eredményei egy CT vizsgálatból szegmentált csípőcsont $128 \times 128 \times 57$ -es részletére. A tesztképet Gábor Székely (ETH, Zürich) biztosította. Első ránézésre az ábrán négy egyforma objektum látható. A hasonlóság oka az, hogy az eredeti csípőcsont sem volt túlságosan vastag.



eredeti objektum



PK-6-C



PK-8-C



PK-12-C

5.11. ábra Három, a közép vonalakat kivonó vékonyító algoritmusunk eredménye egy MR agyvizsgálatból szegmentált agykamrára. A $80 \times 52 \times 72$ -es teszt képet Gábor Székely (ETH, Zürich) biztosította. Az eljárások “szép” eredményt produkáltak annak ellenére, hogy az agykamra egy meglehetősen bonyolult, többszörösen csavarodó objektum.



eredeti objektum



PK-6-S

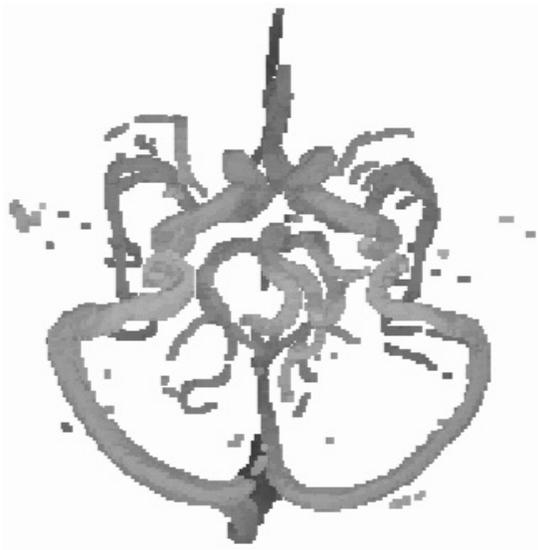


PK-8-S



PK-12-S

5.12. ábra A középfelszín kivonására javasolt vékonyító algoritmusaink eredményei egy MR agyvizsgálatból szegmentált agykamrára. A $80 \times 52 \times 72$ -es tesztképet Gábor Székely (ETH, Zürich) biztosította.



eredeti objektum
felülnézetből



PK-H-C
felülnézetből



eredeti objektum
oldalnézetből

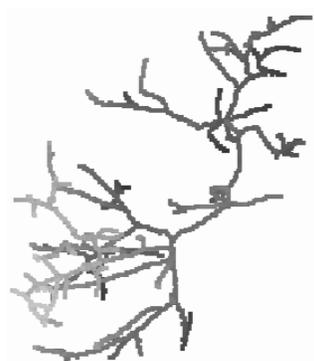


PK-H-C
oldalnézetből

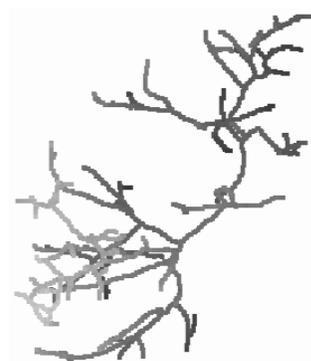
5.13. ábra A középvezonalakat kivonó hibrid (PK-H-C) algoritmusunk eredménye MRA agyvizsgálatból szegmentált vérerekre. A $256 \times 256 \times 124$ -es tesztképet Derek L.G. Hill (UMDS, Guy's Hospital, London) biztosította.



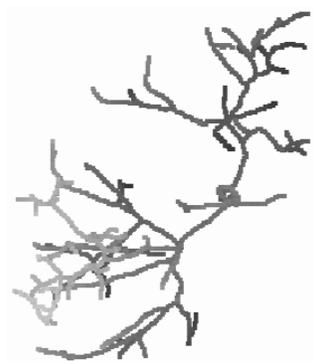
eredeti objektum



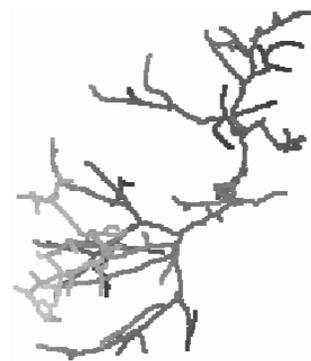
PK-6-C



PK-8-C



PK-12-C



PK-H-C

5.14. ábra A középvonalakat kivonó vékonyító algoritmusaink eredménye májból szegmentált vérerekre. A $272 \times 285 \times 241$ -es tesztképet Pierre Tellier (LSIIT, Université Louis Pasteur, Strasbourg) biztosította. Valamennyi eljárás “egy voxel vékony” középvonalakat produkált.

6. A 3D váz alkalmazásai

A alakreprezentáció két alapvető megközelítése az objektumok határvonalának vagy az objektumok által elfoglalt régióknak a leírásán alapul.

A határ-alapú megközelítés általában egyszerű, hatékony és a differenciálgeometria eszközeivel matematikailag is jól megalapozott. A határ-alapú reprezentáció gyakorlati alkalmazhatóságát — különösen magasabb dimenziókban — viszont erősen korlátozzák az alábbiak (Székely, 1996):

- a differenciálgeometriai vizsgálat nagyon zajérzékeny,
- a határ egyenletlensége erősen zavaró,
- a határ leírása nem mutatja ki a globális tulajdonságokat (pl. a szimmetriát),
- a határ leírásával az objektumok hierarchikus felépítése rejtve marad.

A régió-alapú jellemzők — köztük a váz is — képesek a globális tulajdonságok megragadására és lehetővé teszik a vizsgált objektumok hierarchikus leírását. A régió-alapú megközelítés bizonyára nem helyettesítheti, de kiegészíti és támogatja a határ-alapút.

A 2D vázat gyakran használják az alakfelismerő rendszerekben, pl. karakterek, ujjlenyomatok, vonalrajzok, biológiai sejtek felismerése, azonosítása és elemzése során (Lam, Lee, Suen, 1992; Suen, Wang, 1994; Marchand-Maillet, Sharaiha, 2000). 3D-ben jóval kevesebb hasznos alkalmazás merült fel, aminek két fő oka:

- a 3D képpalkotó berendezések később alakultak ki és szűkebb körben érhetők el, mint a 2D eszközök,
- a topológia és a topológia-megőrzése jóval bonyolultabb 3D-ben, mint 2D-ben.

A fenti gátló tényezők ma már nem állnak fenn, mivel egyrészt a 3D képpalkotó berendezések elterjedtek az orvosi diagnosztikában (pl. hazánkban is már számos kórházban és klinikán üzemelnek CT, MR és SPECT berendezések, Debrecenben pedig egy PET készülék is található), másrészt pedig rendelkezésre állnak a 3D digitális topológiára vonatkozó elméleti eredmények is.

Az alábbiakban áttekintjük a 3D vázkijelölés általunk ismert eddigi gyakorlati alkalmazásait és a 6.1. pontban ismertetjük saját algoritmusaink alkalmazását vérerek és légutak átmérőjének mérésére.

Mivel a 3D orvosi képpalkotó berendezések a leginkább elterjedtek, így a 3D váz alkalmazásai is leginkább az orvosi képfeldolgozás területén merültek fel (Sonka, Hlaváč, Boyle, 1999). Egyetlen kivételről tudunk: a Lee, Kashyap és Chu az általuk kidolgozott 3D vékonyító algoritmust (2.1.4.) és vázmodellt gyárilag előállított szerszámok öntési, kovácsolási vagy préselési hibáinak vizsgálatára javasolta (Lee, Kashyap, Chu, 1994).

Számosan javasoltak a kivont képjellemzők távolságtérképein alapuló regisztrációs módszert, ami precízebb eredményt ad, ha a kivont képjellemzők “vékonyak” (Borgefors, 1988; Jiang, Robb, Holton, 1992; van den Elsen, Maintz, Pol, Viergever, 1992).

Az agyi vérerek szimbólikus és szerkezeti leírásához is hasznosnak bizonyult a vékonyítás mint előfeldolgozó lépés (Gerig et al., 1993; Székely et al., 1995). MRA vagy spirális CT vizsgálatokból szegmentált vérerek vékonyítására (középvonalának meghatározására) számosan javasoltak 3D vékonyító/vázkijelölő algoritmust (Ma, Sonka, 1996; Palágyi, Kuba, 1997/1; Palágyi, Kuba, 1997/3; Nyström, 1998; Palágyi, Kuba, 1998/2; Palágyi, Kuba, 1998/3; Pudney, 1998; Gagvani, Silver, 1999; Yim, Choyke, Summers, 1999;

Palágyi, Kuba, 1999/2; Palágyi, Sorantin, Halmai, Kuba, 1999). Ma és Sonka vérerek mellett légutak vékonyítására is alkalmazta az algoritmusukat (Ma, Sonka, 1996).

A vérerek és légutak középvezetékének ismerete a sebészeti navigációt is segíti (Gagvani, Silver, 1999). Ge, Stelts és Vining virtuális végbéltükrözésre javasolt módszere is a vázkijelölésen alapul (Ge, Stelts, Vining, 1996). A vérerek középvezetékének meghatározása szükséges a virtuális endoszkópiához is (Rubin, Johnston, 1998). A virtuális technikák elsősorban nem is a költségtakarékosságuk, hanem a betegbarát voltuk miatt rendkívül fontosak.

A Voronoi-vázat alkalmasnak tartják rendkívül összetett szervek (pl. a homloklebény vagy a teljes agy) jellemzésére vagy a csontok vastagságának a mérésére (pl. csípőprotézis tervezésekor) is (Székely et al., 1992; Näf et al., 1997). Mivel a vázpontok lokális szimmetriát reprezentálnak, így a szegmentált agy vázából a szkizofréniára való hajlam is kimutatható, mivel az abban szenvedők agytekéi kevésbé asszimmetrikusak.

A vékonyítás — mint topológia-megőrző képművelet — alkalmas a csontok topológiai elemzésére nagyfelbontású mikro CT és MR vizsgálatok alapján (Gomberg et al., 1999).

6.1. Vérerek és légutak átmérőjének mérése

3D (bináris) objektumok vékonyítására többféle módszert is kidolgoztunk (2.1.6.–2.1.8., 2.4.1.), amelyek első alkalmazására a grazi egyetemen került sor a vese alatti artéria tágulat (*infra-renal aortic aneurysm*) és a légcső szűkület (*laryngotrachealstenosis*) diagnózisában (Palágyi, Sorantin, Halmai, Kuba, 1999; Palágyi és mtrrsai, 2000).

A 6.1a. ábra (spirál CT-ből) szegmentált vérere egy vese alatti artéria tágulatban szenvedő betegről készült. Ez a betegség a 60 évesnél idősebbek mintegy 2%-ánál jelentkezik. Ha az artéria átmérője meghaladja a kritikus 5 cm-t (!), akkor azonnali műtéti beavatkozás szükséges, mivel az elvékonyodott érfal megrepedése az esetek 70–90%-ában halált okoz. A 6.1b. ábrán látható (spirál CT-ből származó) szegmentált légcső egy nyelv alatti szűkületben (*subglottic stenosis*) szenvedő betegé. Mindkét bináris objektum középvonalát a 6 al-iterációs irányszekvenciális algoritmusunkkal (2.1.6.) határoztuk meg.

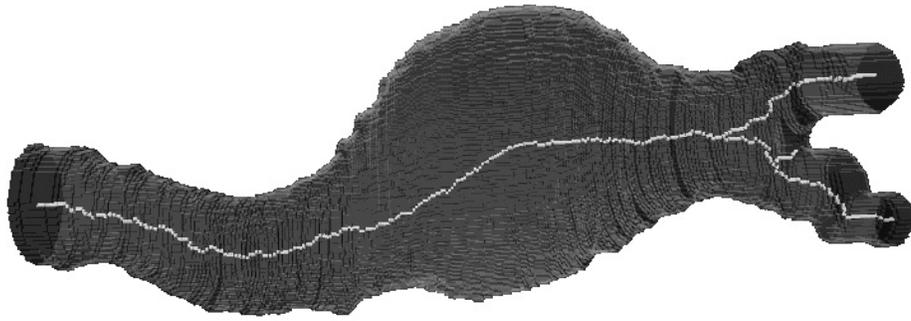
A vérerek és légutak keresztmetszetét/átmérőjét kiszámító módszerünk a következő lépésekből áll (Palágyi, Sorantin, Halmai, Kuba, 1999; Palágyi és mtrrsai, 2000):

- előfeldolgozás:
 - szegmentálás (interaktív módszerrel),
 - interpoláció (kocka-alakú voxeleket tartalmazó képtömb előállítás),
 - simítás morfológiai szűréssel;
- középvonal meghatározása:
 - vékonyítás középvonalra,
 - tisztítás (parazita ágak eltávolítása),
 - vektorizálás (raszter-vektor konverzió),
 - vektoros simítás;
- kiértékelés:
 - keresztmetszet (*cross-sectional profile*) meghatározása a középvonal minden (diszkrét) pontjára (a keresztmetszetet az adott pontra merőleges sík jelöli ki),
 - átmérő becslése a keresztmetszet alapján (a keresztmetszet területét a hozzá tartozó sík objektumpontjainak a száma adja, az átmérőt pedig az ugyanazon területű körlap átmérőjével becsüljük).

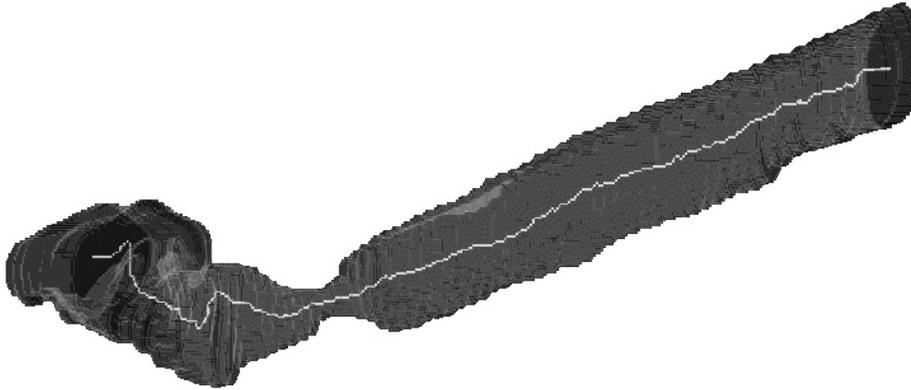
A vékonyítás előtt az objektumokat tartalmazó bináris tömbben megkeressük z -irányban az első és az utolsó nemnulla 2D szeletet. Ezen 2D képszeletek zsugorításával olyan izolált pontokat kapunk, amelyeket a vékonyítás horgony-pontnak tekint. A horgony-pontokat a vékonyítás nem törli, így azok végpontjai lesznek a középvonalaknak (mivel a vékonyítás topológia-megőrző). Ha valamely vizsgált 2D képszeletre a zsugorítás egynél több izolált pontot talál, akkor a vizsgált objektum elágazó²¹. Elágazó objektumok esetén az ágak találkozásának 2D rétegét is zsugorítjuk és az így kapott izolált ponto(ka)t is felvesszük a horgony-pontok közé. Ily módon egyfelől “szebb”, másrészt pedig a teljes szegmentált ér- vagy légút-szakaszt végigérő középvonalhoz jutunk (6.1. ábra). A 2D zsugorításra egy 4 al-iterációs irányszekvenciális eljárást (Palágyi, 1993) alkalmazunk.

Egy VRML (*Virtual Reality Modelling Language*) szerkesztővel megadhatunk a középvonalon érdekes anatómiai pontokat is, mely pontokra megjelöljük azon a görbén, ami az átmérőt adja meg a középvonalon (6.2. ábra).

²¹A szegmentált objektumokra bizonyos előfeltételekkel élünk, pl. feltesszük, hogy valamennyi ág az objektumot tartalmazó bináris tömb z -irányú szeletein “lép ki”.

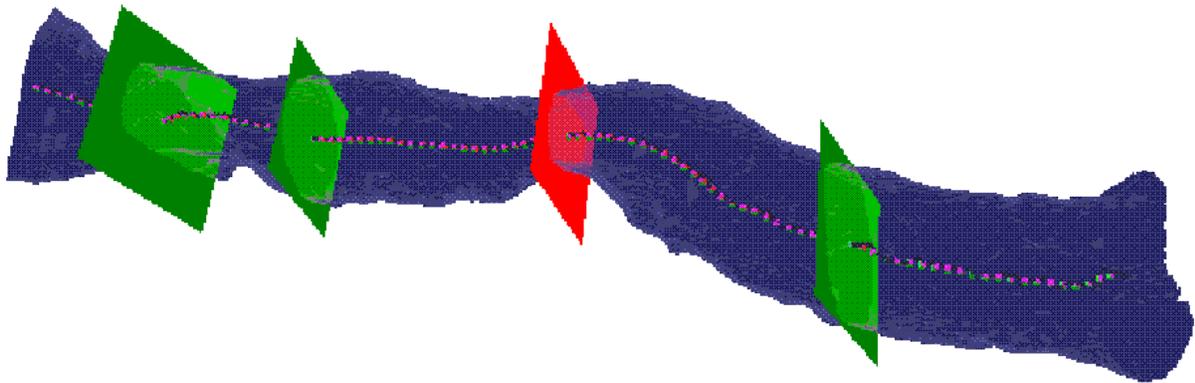


a

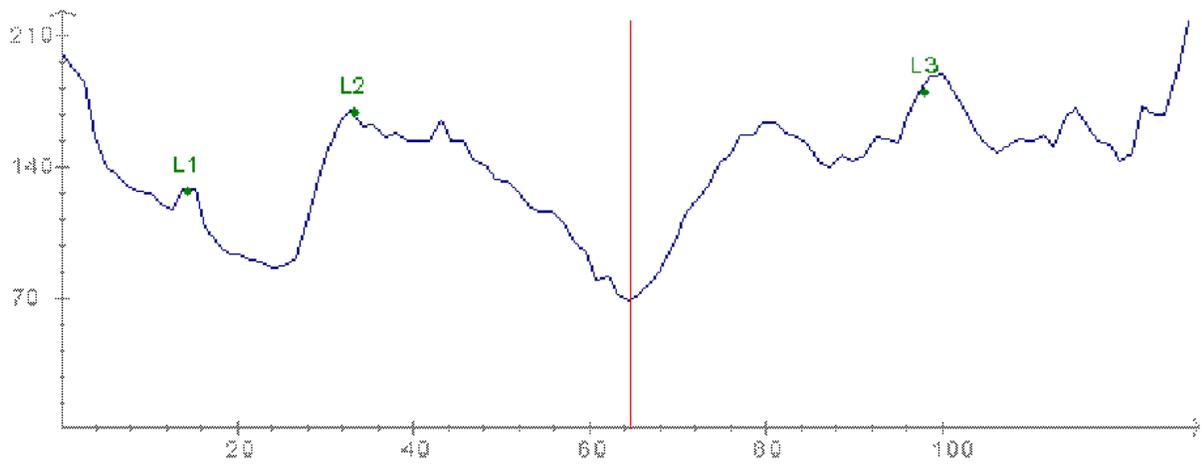


b

6.1. ábra Vese alatti szegmentált artériaszakasz és annak középvonala (a), szegmentált légcsőrészlet és annak középvonala (b). A horgony-pontok felvétele a teljes ér-, illetve légút-szegmenst végigérő középvonalakat eredményezett.



a



b

6.2. ábra Szegmentált légcső, a középvonallával és a középvonatra merőleges síkok három (interaktívan) megadott anatómiai pontban, továbbá a minimális átmérőnél (a). A grafikon a légcső becsült átmérőjét jeleníti meg a középvonalon pontjain végighaladva, ahol a megadott anatómiai pontok pozícióit is megjelöltük (b).

Irodalom

AKERS, 1978

S.B. Akers, Binary Decision Diagrams, *IEEE Trans. Computers* **27**, 1978, 509–516.

ALB, TANÁCS, 1997

Alb P., Tanács A., *Lokális és párhuzamos képfeldolgozó rendszer 2- és 3-dimenziós bináris képekre*, Tudományos diákköri dolgozat, József Attila Tudományegyetem, Szeged, 1997.

ALI, PAVLIDIS, 1982

F. Ali, T. Pavlidis, Noise filtering in binary pictures by combinatorial techniques, *Pattern Recognition* **15**, 1982, 131–135.

ARCELLI, 1981

C. Arcelli, Pattern thinning by contour tracking, *Computer Graphics and Image Processing* **17**, 1981, 130–144.

ARCELLI, CORDELLA, LEVIALDI, 1981

C. Arcelli, L.P. Cordella, S. Levialdi, From local maxima to connected skeletons, *IEEE Trans. Pattern Analysis and Machine Intelligence* **3**, 1981, 134–143.

ARCELLI, FRUCCI, 1992

C. Arcelli, M. Frucci, Reversible skeletonization by (5,7,11)-erosion, in *Visual form — Analysis and Recognition*, Plenum Press, 1992, 21–28.

ARCELLI, LEVIALDI, 1972

C. Arcelli, S. Levialdi, Parallel shrinking in three dimensions, *Computer Graphics and Image Processing* **1**, 1972, 21–30.

ARCELLI, SANNITI DI BAJA, 1988

C. Arcelli, G. Sanniti di Baja, Finding local maxima in a pseudo-Euclidean distance transform, *Computer Vision, Graphics, and Image Processing* **43**, 1988, 361–367.

ARCELLI, SANNITI DI BAJA, 1992

C. Arcelli, G. Sanniti di Baja, Ridge points in Euclidean distance maps, *Pattern Recognition Letters* **13**, 1992, 237–243.

ARCELLI, SANNITI DI BAJA, 1996

C. Arcelli, G. Sanniti di Baja, Skeletons of planar patterns, in *Topological Algorithms for Digital Image Processing*, Machine Intelligence and Pattern Recognition 19, North-Holland, 1996, 99–143.

ARCELLI, SANNITI DI BAJA, 1998

C. Arcelli, G. Sanniti di Baja, Picture editing by simultaneously smoothing figure protusions and dents, in *Proc. 9th Int. Conf. Pattern Recognition* 948–950 1998.

ATTALI, MONTANVERT, 1994

D. Attali, A. Montanvert, Semicontinuous skeletons in 2D and 3D shapes, in *Proc. 2nd Int. Workshop on Visual Form*, World Scientific, 1994, 32–41.

ATTALI, MONTANVERT, 1997

D. Attali, A. Montanvert, Computing and simplifying 2D and 3D continuous skeletons, *Computer Vision and Image Understanding* **67**, 1997, 261–273.

- AURENHAMMER, 1985
F. Aurenhammer, Voronoi diagrams: A survey of a fundamental geometric data structure, *ACM Comput. Surveys* **23**, 1985, 345–405.
- AWCOCK, THOMAS, 1996
G.W. Awcock, R. Thomas, *Applied image processing*, McGraw–Hill, Inc., 1996.
- BAXES, 1994
G.A. Baxes, *Digital Image Processing — Principles and Applications*, John Wiley & Sons, Inc., 1994.
- BERTRAND, 1995/1
G. Bertrand, Sufficient conditions for 3D parallel thinning algorithms, in *Proc. Conf. Vision Geometry IV*, Proc. SPIE **2573**, 1995, 52–60.
- BERTRAND, 1995/2
G. Bertrand, A parallel thinning algorithm for medial surfaces, *Pattern Recognition Letters* **16**, 1995, 979–986.
- BERTRAND, 1996
G. Bertrand, A Boolean characterization of three–dimensional simple points, *Pattern Recognition Letters* **17**, 1996, 115–124.
- BERTRAND, AKTOUF, 1994
G. Bertrand, Z. Aktouf, A 3D thinning algorithms using subfields, in *Proc. Conf. on Vision Geometry III*, Proc. SPIE **2356**, 1994, 113–124.
- BERTRAND, MALANDAIN, 1995
G. Bertrand, G. Malandain, A note on “Building skeleton models via 3–D medial surface/axis thinning algorithms”, *CVGIP: Graphical Models and Image Processing* **57**, 1995, 537–538.
- BLUM, 1964
H. Blum, A transformation for extracting new descriptors of shape, Symposium on Models for the perception of Speech and Visual Form, 1964.
- BLUM, 1967
H. Blum, Biological shape and visual science, *J. Theoretical Biology* **38**, 1967, 205–287.
- BOISSONAT, KOFAKIS, 1985
J.D. Boissonat, P. Kofakis, Use of the Delaunay triangulation for the identification and the localization of objects, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 1985, 398–401.
- BORGEFORS, 1984
G. Borgefors, Distance transformations in arbitrary dimensions, *Computer Vision, Graphics, and Image Processing* **27**, 1984, 321–345.
- BORGEFORS, 1988
G. Borgefors, Hierarchical chamfer matching: A parametric edge matching algorithm, *IEEE Trans. Pattern Analysis and Machine Intelligence* **10**, 1988, 849–865.
- BORGEFORS, 1996
G. Borgefors, On digital distance transforms in three dimensions, *Computer Vision and Image Understanding* **64**, 1996, 368–376.

- BORGEFORS, NYSTRÖM, SANNITI DI BAJA, 1999
 G. Borgefors, I. Nyström, G. Sanniti di Baja, Computing skeletons in three dimensions, *Pattern Recognition* **32**, 1999, 1225–1236.
- BOW, 1992
 S.–T. Bow, *Pattern recognition and image preprocessing*, Marcel Dekker, Inc., 1992.
- BRANDT, 1992
 J.W. Brandt, Describing a solid with the three–dimensional skeleton, in *Proc. Conf. Curves and Surfaces in Comp. Vision and Graphics III*, Proc. SPIE **1830**, 1992, 258–269.
- BRANDT, ALGAZI, 1992
 J.W. Brandt, V.R. Algazi, Continuous skeleton computation by Voronoi diagram, *CVGIP: Image Understanding* **55**, 1992, 329–338.
- BREU, GIL, KIRKPATRICK, WERMAN, 1995
 H. Breu, J. Gil, D. Kirkpatrick, M. Werman, Linear time Euclidean Transform algorithms, *IEEE Trans. Pattern Analysis and Machine Intelligence* **17**, 1995, 529–533.
- BRYANT, 1992
 R.E. Bryant, Symbolic boolean manipulation with ordered binary–decision diagrams, *ACM Computing Surveys* **24**, 1992, 293–317.
- BROWN, 1992
 L.G. Brown, A Survey of Image Registration Techniques, *ACM Computing Surveys* **24** 1992, 325–376.
- CALABI, 1965
 L. Calabi, *A study of the skeleton of plane figures*, Technical report 60429 Parke Mathematical Laboratories, 1965.
- CALABI, HARTNETT, 1968
 L. Calabi, W.E. Hartnett, Shape recognition, prairie fires, convex deficiencies and skeletons, *Am. Math. Monthly* **75**, 1968, 335–342.
- CUISENAIRE, MACQ, 1999
 O. Cuisenaire, B. Macq, Fast Euclidean distance transformation by propagation using multiple neighborhoods, *Computer Vision and Image Understanding* **76**, 1999, 163–172.
- DANIELSSON, 1980
 P.E. Danielsson, Euclidean distance mapping, *CVGIP: Graphical Models and Image Processing* **14**, 1980, 227–248.
- DEUTSCH, 1968
 E.S. Deutsch, Thinning algorithms on rectangular, hexagonal, and triangular arrays, *Commun. ACM* **15**, 1972, 827–837.
- DORST, 1986
 L. Dorst, Pseudo–euclidean skeletons, in *Proc. 8th Int. Conf. Pattern Recognition*, 1986, 286–288.
- EGGERS, 1996
 H. Eggers, Parallel Euclidean distance transformations in Z_g^n , *Pattern Recognition Letters* **17**, 1996, 751–757.

EGGERS, 1998

H. Eggers, Two fast Euclidean distance transformations in \mathbb{Z}^2 based on sufficient propagation, *Computer Vision and Image Understanding* **69**, 1998, 106–116.

EMBRECHTS, ROOSE, 1996

H. Embrechts, D. Roose, A parallel Euclidean distance transformation algorithm, *Computer Vision and Image Understanding* **63**, 1996, 15–26.

FAZEKAS, 1998/1

Fazekas A., Vázkiyelölő algoritmusok a digitális képfeldolgozásban, *Alkalmazott Matematikai Lapok* **19**, 1998, 155–180.

FAZEKAS, 1998/2

Fazekas A., *Vázkiyelölő algoritmusok*, Doktori értekezés, KLTE, Debrecen, 1998.

FERNÁNDEZ–VIDAL, MALANDAIN, 1995

S. Fernández–Vidal, G. Malandain, Digital euclidean skeleton in n -D, in *Proc. 9th Scandinavian Conf. Image Analysis SCIA '95*, 1995, 517–524.

FOLEY ET AL., 1991

J.D. Foley, A. Van Dam, S.K. Feiner, J.F. Hughes, *Computer Graphics — Principles and Practice*, Addison–Wesley Publishing Company, Reading, Massachusetts, 1991.

GAGVANI, SILVER, 1999

N. Gagvani, D. Silver, Parameter–controlled volume thinning, *Graphical Models and Image Processing* **61**, 1999, 149–164.

GE, STELTS, VINING, 1996

Y. Ge, D.R. Stelts, D.J. Vining, 3D skeleton for virtual colonoscopy, in *Proc. 4th Int. Conf. Visualization in Biomedical Computing, VBC'96*, Lecture Notes in Computer Science **1131**, Springer 1996, 449–454.

GERIG ET AL., 1993

G. Gerig, Th. Koller, G. Székely, Ch. Brechbühler, O. Kübler, Symbolic descriptions of 3–D structures applied to cerebral vessel tree obtained from MR angiography volume data, in *Proc. 13th Int. Conf. Information Processing in Medical Imaging, IPMI'93*, Lecture Notes in Computer Science **687**, Springer, 1993, 94–111.

GLASBEY, HORGAN, 1995

C.A. Glasbey, G.W. Horgan, *Image Analysis for the Biological Sciences*, John Wiley & Sons, 1995.

GOMBERG ET AL., 1999

B.R. Gomberg, P.K. Saha, H.K. Song, S.N. Hwang, F.W. Wehrli, Topological analysis of trabecular bone MR images, submitted to *IEEE Trans. Medical Imaging*, 1999.

GONG, BERTRAND, 1990

W.X. Gong and G. Bertrand, A simple parallel 3D thinning algorithm, in *Proc. 10th IEEE Int. Conf. Pattern Recognition*, 1990, 188–190.

GONZALES, WOODS, 1992

R.C. Gonzales, R.E. Woods, *Digital image processing*, Addison–Wesley, Reading, Massachusetts, 1992.

GÖKMEN, HALL, 1990

M. Gökmen, R.W. Hall, Parallel shrinking algorithms using 2-subfields approaches, *Computer Vision, Graphics, and Image Processing* **52**, 1990, 191–209.

GRAHAM, 1972

R.L. Graham, An efficient algorithm for determining the convex hull of a finite planar set, *Information Processing Letters* **1**, 1972, 132–133.

HAFFORD, PRESTON, 1984

K.J. Hafford, K. Preston, Three-dimensional skeletonization of elongated solids, *Computer Vision, Graphics, and Image Processing* **27**, 1984, 78–91.

HALL, 1996

R.W. Hall, Parallel connectivity-preserving thinning algorithm, in *Topological Algorithms for Digital Image Processing*, Machine Intelligence and Pattern Recognition 19, North-Holland, 1996, 145–179.

HALL, KONG, ROSENFELD, 1996

R.W. Hall, T.Y. Kong, A. Rosenfeld, Shrinking binary images, in *Topological Algorithms for Digital Image Processing*, Machine Intelligence and Pattern Recognition 19, North-Holland, 1996, 31–98.

HALL, KÜÇÜK, 1992

R.W. Hall, Ş. Küçük, Parallel 3D shrinking algorithms using subfields notions, in *Proc. 11th IEEE Int. Conf. Pattern Recognition*, 1992, 395–398.

HERMAN, 1992

G.T. Herman, Discrete multidimensional Jordan surfaces, *CVGIP: Graphical Models and Image Processing* **54**, 1992, 507–515.

HURT, ROSENFELD, 1984

S.L. Hurt, A. Rosenfeld, Noise reduction in three-dimensional digital images, *Pattern Recognition* **17**, 1984, 407–421.

JAISIMHA, 1994

M.Y. Jaisimha, R.M. Haralick, D. Dori, Quantitative performance evaluation of thinning algorithms in the presence of noise, in *Aspects of Visual Form Processing*, World Scientific, 1994, 261–286.

JARVIS, 1973

R.A. Jarvis, On the identification of the convex hull of a finite set of points in the plane, *Information Processing Letters* **2**, 1973, 18–21.

JÄHNE, 1993

B. Jähne, *Digital Image Processing: Concepts, Algorithms and Scientific Applications*, Springer-Verlag, 1993.

JIANG, ROBB, HOLTON, 1992

H. Jiang, A. Robb, K.S. Holton, A new approach to 3-D registration of multimodality medical images by surface matching, in *Proc. Conf. Medical Imaging V: Image Processing*, Proc. SPIE **1808**, 1992, 196–213.

KAMESWARA RAO, DANIELSSON, KRUSE, 1978

C.V. Kameswara Rao, P.E. Danielsson, B. Kruse, Checking connectivity preservation properties of some types of picture processing operations, *Computer Vision, Graphics, and Image Processing* **8**, 1978, 299–309.

KATONA, PALÁGYI, TÓTH, 1995

E. Katona, K. Palágyi, N. Tóth, Signature verification using neural nets, in *Proc. 9th Scandinavian Conf. Image Analysis SCIA '95*, 1995, 1115–1122.

- KIRKPATRICK, 1979
D.G. Kirkpatrick, Efficient computation of continuous skeletons, in *Proc. 20th Annual Symposium on the Foundations of Computer Science*, 1979, 18–27.
- KONG, 1986
T.Y. Kong, *Digital topology with application to image processing*, Doctoral dissertation, University of Oxford, 1986.
- KONG, 1989
T.Y. Kong, A digital fundamental group, *Computer and Graphics* **13**, 1989, 159–166.
- KONG, 1993
T.Y. Kong, On the problem of determining whether a parallel reduction operator for n -dimensional binary images always preserves topology, in *Proc. Conf. Vision Geometry II*, Proc. SPIE **2060**, 1993, 69–77.
- KONG, 1995
T.Y. Kong, On topology preservation in 2-D and 3-D thinning, *International Journal of Pattern Recognition and Artificial Intelligence* **9**, 1995, 813–844.
- KONG, ROSENFELD, 1989
T.Y. Kong, A. Rosenfeld, Digital topology: Introduction and survey, *Computer Vision, Graphics, and Image Processing* **48**, 1989, 357–393.
- KONG, ROSCOE, ROSENFELD, 1992
T.Y. Kong, A.W. Roscoe, A. Rosenfeld, Concepts of digital topology, *Topology and its Applications* **46**, 1992, 219–262.
- KUOSMANEN, KOSKINEN, ASTOLA, 1992
P. Kuosmanen, L. Koskinen, J. Astola, Detail preserving morphological filtering, in *Proc. of the 11th IAPR, Vol. III*, IEEE Computer Society Press, Los Vaqueros Circle, 1992, 236–239.
- KWOK, 1988
P.C.K. Kwok, A thinning algorithm by contour generation, *Comm. ACM* **31**, 1988, 1314–1324.
- LAM, LEE, SUEN, 1992
L. Lam, S.-W. Lee, C.Y. Suen, Thinning methodologies — A comprehensive survey, *IEEE Trans. Pattern Analysis and Machine Intelligence* **14**, 1992, 869–885.
- LATECKI, 1997
L. Latecki, Well-composed pictures, *Graphical Models and Image Processing* **59**, 1997, 164–172.
- LATECKI, ECKHARDT, ROSENFELD, 1995
L. Latecki, U. Eckhardt, A. Rosenfeld, Well-composed sets, *Computer Vision and Image Understanding* **61**, 1995, 70–83.
- LATECKI, MA, 1996
L. Latecki, C.M. Ma, An algorithm for simplicity test, *Computer Vision and Image Understanding* **63**, 1996, 388–393.
- LEE, POSTON, ROSENFELD, 1991
C. Lee, T. Poston, A. Rosenfeld, Winding Euler numbers for 2D and 3D digital images, *CVGIP: Graphical Models and Image Processing* **53**, 1991, 522–537.

- LEE, POSTON, ROSENFELD, 1993
 C. Lee, T. Poston, A. Rosenfeld, Holes and genus of 2D and 3D digital images, *CVGIP: Graphical Models and Image Processing* **55**, 1993, 20–47.
- LEE, KASHYAP, CHU, 1994
 T. Lee, R.L. Kashyap, C. Chu, Building skeleton models via 3–D medial surface/axis thinning algorithms, *CVGIP: Graphical Models and Image Processing* **56**, 1994, 462–478.
- LEVIALDI, 1972
 S. Levialdi, On shrinking binary patterns, *Comm. ACM* **15**, 1972, 7–10.
- LOBREGT, VERBEEK, GROEN, 1980
 S. Lobregt, P.W. Verbeek, F.C.A. Groen, Three–dimensional skeletonization: Principle and algorithm, *IEEE Trans. Pattern Analysis and Machine Intelligence* **2**, 1980, 75–77.
- MA, 1993
 C.M. Ma, Topology preservation on 3D images, in *Proc. Conf. Vision Geometry*, Proc. SPIE **2060** 1993, 201–207.
- MA, 1994
 C.M. Ma, On topology preservation in 3D thinning, *CVGIP: Image Understanding* **59**, 1994, 328–339.
- MA, 1995
 C.M. Ma, A 3D fully parallel thinning algorithm for generating medial face, *Pattern Recognition Letters* **16**, 1995, 83–87.
- MA, 1996
 C.M. Ma, Connectivity preservation of 3D 6–subiteration thinning algorithms, *Graphical Models and Image Processing* **58**, 1996, 382–386.
- MA, SONKA, 1996
 C.M. Ma, M. Sonka, A fully parallel 3D thinning algorithm and its applications, *Computer Vision and Image Understanding* **64**, 1996, 420–433.
- MALANDAIN, BERTRAND, 1992
 G. Malandain, G. Bertrand, Fast characterization of 3D simple points, in *IEEE Int. Conf. Pattern Recognition*, 1992, 232–235.
- MANZANERA ET AL., 1999/1
 A. Manzanera, T.M. Bernard, F. Pret  ux, B. Longuet, Ultra–fast skeleton based on an isotropic fully parallel algorithm, in *Proc. 8th Int. Conf. on Discrete Geometry for Computer Imagery, DGCI'99*, Lecture Notes in Computer Science **1568**, Springer, 1999, 313–324.
- MANZANERA ET AL., 1999/2
 A. Manzanera, T.M. Bernard, F. Pret  ux, B. Longuet, A unified mathematical framework for a compact and fully parallel n–D skeletonization procedure, in *Proc. Int. Conf. Vision Geometry IV*, 57–68, 1999.
- MANZANERA ET AL., 1999/3
 A. Manzanera, T.M. Bernard, F. Pret  ux, B. Longuet, Medial faces from a concise 3D thinning algorithm, in *Proc. 7th IEEE Int. Conf. Computer Vision, ICCV'99*, 337–343, 1999.

- MARAGOS, SCHAFFER, BUTT, 1996
 P. Maragos, R.W. Schaffer, M.A. Butt (Eds.), *Mathematical morphology and Its Applications to Image and Signal Processing*, Computational Imaging and Vision, Kluwer Academic Publishers, 1996.
- MARCHAND-MAILLET, SHARAIHA, 2000
 S. Marchand-Maillet, Y.M. Sharaiha, *Binary Digital Image Processing: A discrete Approach*, Academic Press, 2000.
- MATHERON, 1975
 G. Matheron, *Random sets and integral geometry*, John Wiley & Sons, 1975.
- MATHERON, 1988
 G. Matheron, Examples of topological properties of skeletons, in *Image analysis and mathematical morphology, Vol. 2: Theoretical Advances*, Academic Press, 217–238, 1988.
- MEYER, 1989
 F. Meyer, Skeletons and perceptual graphs, *Signal Processing* **16**, 1989, 335–363.
- MORGENTHALER, 1981
 D.G. Morgenthaler, *Three-Dimensional Simple Points: Serial Erosion, Parallel Thinning and Skeletonization*, TR-1005, Computer Vision Laboratory, University of Maryland, 1981.
- MUKHERJEE, DAS, CHATTERJEE, 1990
 J. Mukherjee, P.P. Das, B.N. Chatterjee, On connectivity issues of MESPTA, *Pattern Recognition Letters* **11**, 1990, 643–648.
- NACCACHE, SHINGHAL, 1984
 N.J. Naccache, R. Shinghal, A proposed algorithm for thinning binary patterns, *IEEE Trans. Syst. Man Cybern.* **14**, 1984, 409–418.
- NÄF, 1996
 M. Näf, *3D Voronoi skeletons: A semi-continuous implementation of the ‘Symmetric Axis Transform’ in 3D Euclidean space*, Ph.D. thesis, Swiss Federal Institute of Technology Zürich, 1996.
- NÄF ET AL., 1997
 M. Näf, G. Székely, R. Kikinis, M.E. Shenton, G. Kübler, 3D Voronoi skeletons and their usage for the characterization and recognition of 3D organ shape, *Computer Vision, Graphics, and Image Processing* **66**, 1997, 147–161.
- NYSTRÖM, 1998
 I. Nyström, Skeletonization applied to magnetic resonance angiography images, in *Proc. Conf. Medical Imaging: Image Processing*, Proc. SPIE **3338**, 1998, 693–701.
- OGNIEWICZ, ILG, 1992
 R.L. Ogniewicz, M. Ilg, Voronoi skeletons: Theory and applications, in *IEEE Conf. Computer Vision and Pattern Recognition*, 1992, 63–69.
- OLTYÁN, 1997
 Oltyán G., *3-dimenziós bináris képek megjelenítése*, Tudományos diákköri dolgozat, József Attila Tudományegyetem, Szeged, 1997.

OSHER, SETHIAN, 1988

S. Osher, J.A. Sethian, Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton–Jacobi formulations, *J. Computational Physics* **72**, 1988, 12–49.

PALÁGYI, 1993²²

Palágyi K., Lokális párhuzamos eljárás bináris képek zajszűrésére, *Alkalmazott Matematikai Lapok* **17**, 1993, 373–396.

PALÁGYI, KUBA, 1997/1

K. Palágyi, A. Kuba, A thinning algorithm to extract medial lines from 3D medical images, in *Proc. 15th Int. Conf. on Information Processing in Medical Imaging, IPMI'97*, Lecture Notes in Computer Science **1230**, Springer, 1997, 411–416.

PALÁGYI, KUBA, 1997/2

K. Palágyi, A. Kuba, A parallel 12-subiteration 3D thinning algorithm to extract medial lines, in *Proc. 7th Int. Conf. on Computer Analysis of Images and Patterns, CAIP'97*, Lecture Notes in Computer Science **1296**, Springer, 1997, 400–407.

PALÁGYI, KUBA, 1997/3

K. Palágyi, A. Kuba, An Algorithm for Thinning 3D Medical Images, in *Proc. 1st KEPAF Conf. Image Analysis and Pattern Recognition*, 1977, 64–71.

PALÁGYI, KUBA, 1998/1

K. Palágyi, A. Kuba, A 3D 6-subiteration thinning algorithm for extracting medial lines, *Pattern Recognition Letters* **19**, 1998, 613–627.

PALÁGYI, KUBA, 1998/2

K. Palágyi, A. Kuba, A hybrid thinning algorithm for 3D medical images, *J. Computing and Information Technology* **6**, 1998, 149–164.

PALÁGYI, KUBA, 1998/3

Palágyi K., Kuba A., Vékonyító algoritmusok 3D orvosi képekre, *A számítástechnika orvosi és biológiai alkalmazásai, A XXI. Neumann Kollokvium Kiadványa*, NJSZT Orvosbiológiai Szakosztálya, 1998, 63–66.

PALÁGYI, KUBA, 1999/1

K. Palágyi, A. Kuba, Directional 3D thinning using 8 subiterations, in *Proc. 8th Int. Conf. on Discrete Geometry for Computer Imagery, DGCI'99*, Lecture Notes in Computer Science **1568**, Springer, 1999, 325–336.

PALÁGYI, KUBA, 1999/2

K. Palágyi, A. Kuba, A parallel 3D 12-subiteration thinning algorithm, *Graphical Models and Image Processing* **61**, 1999, 199–221.

PALÁGYI, SORANTIN, HALMAI, KUBA, 1999

K. Palágyi, E. Sorantin, Cs. Halmai, A. Kuba, 3D thinning and its applications to medical image processing, *TASK Quarterly* **3**, 1999, 397–408.

PALÁGYI ÉS MTRSAI, 2000

Palágyi K., E. Sorantin, Halmai Cs., Martonossy L., Erdőhelyi B., Kuba A., 3D vékonyítás és alkalmazása vérerek és légutak átmérőjének meghatározására, *A Magyar Képfeldolgozók és Alakfelismerők 2. Országos Konferenciája 2000*, 95–100.

²²Megjelent 1997-ben.

- PRATT, KABIR, 1985
W.K. Pratt, I. Kabir, Morphological binary image processing with a local neighbourhood pipeline processor, in *Frontiers in computer graphics (Proc. of Computer Graphics Tokyo '84)*, Springer-Verlag, 1985. 321-343.
- PREPARATA, SHAMOS, 1990
F.P. Preparata, M.I. Shamos, *Computational Geometry*, Springer-Verlag, 1990.
- PUDNEY, 1998
C. Pudney, Distance-ordered homotopic thinning: A skeletonization algorithm for 3D digital images, *Computer Vision and Image Understanding* **72**, 1998, 404-413.
- RAGNEMALM, 1990
I. Ragnemalm, Contour processing distance transforms, in *Progress in Image Analysis and Processing*, World Scientific, Singapore, 1990, 202-212.
- RAGNEMALM, 1992/1
I. Ragnemalm, The Euclidean distance transform in arbitrary dimensions, in *Proc. 4th Int. Conf. in Image Analysis and Processing*, 1992, 290-293.
- RAGNEMALM, 1992/2
I. Ragnemalm, Fast edge smoothing in binary images using Euclidean metric, in *Progress in Image Analysis II.*, World Scientific, 1992.
- RAY, 1988
S. Ray, A heuristic noise reduction algorithm applied to handwritten numeric characters, *Pattern Recognition Letters* **7**, 1988, 9-12.
- REDDY, TURKIYYAH, 1995
J.M. Reddy, G.M. Turkiyyah, Computation of 3D skeletons using a generalized De-launey triangulation technique, *Computer-Aided Design* **27**, 1995, 677-694.
- ROBERT, MALANDAIN, 1998
L. Robert, G. Malandain, Fast binary image processing using binary decision diagrams, *Computer Vision and Image Understanding* **72**, 1998, 1-9.
- RONSE, 1986
C. Ronse, A topological characterization of thinning, *Theoretical Computer Sci.* **43**, 1986, 31-41.
- RONSE, 1988
C. Ronse, Minimal test patterns for connectivity preservation in parallel thinning algorithms for binary digital images, *Discrete Appl. Math.* **21**, 1988, 67-79.
- ROSENFELD, 1970
A. Rosenfeld, Connectivity in digital pictures, *J. Assoc. Comput. Mach.* **17**, 1970, 146-160.
- ROSENFELD, 1975
A. Rosenfeld, A characterization of parallel thinning algorithms, *Information and Control* **29**, 1975, 286-291.
- ROSENFELD, PFALTZ, 1968
A. Rosenfeld, J.L. Pfaltz, Distance functions on digital pictures, *Pattern Recognition* **1**, 1968, 33-61.

- RUBIN, JOHNSTON, 1998
P. Rubin, N. Johnston, Measurement of the aorta and its branches with helical CT, *Radiology* **206**, 1998, 823–829.
- SAHA, CHAUDHURI, 1994
P.K. Saha, B.B. Chaudhuri, Detection of 3-D simple points for topology preserving transformations with application to thinning, *IEEE Trans. Pattern Analysis and Machine Intelligence* **16**, 1994, 1028–1032.
- SAHA, CHAUDHURI, 1996
P.K. Saha, B.B. Chaudhuri, 3D digital topology under binary transformation with applications, *Computer Vision and Image Processing* **63**, 1996, 418–429.
- SAHA, CHAUDHURI, MAJUMDER, 1997
P.K. Saha, B.B. Chaudhuri, D.D. Majumder, A new shape-preserving parallel thinning algorithm for 3D digital images, *Pattern Recognition* **30**, 1997, 1939–1955.
- SAITO, TORIWAKI, 1994
T. Saito, J. Toriwaki, New algorithm for Euclidean distance transformation of an n -dimensional digitized picture with applications, *Pattern Recognition* **27**, 1994, 1551–1565.
- SAITO, TORIWAKI, 1995
T. Saito, J. Toriwaki, A sequential thinning algorithm for three dimensional digital pictures using the Euclidean distance transformation, in *Proc. 9th Scandinavian Conf. Image Analysis, SCIA '95*, 1995, 507–516.
- SCHALKOFF, 1989
R.J. Schalkoff, *Digital Image Processing and Computer Vision*, John Wiley & Sons, 1989.
- SCHMITT, 1989
M. Schmitt, Some examples of algorithms analysis in computational geometry by means of mathematical morphological techniques, in *Proc. Conf. Geometry and Robotics*, Lecture Notes in Computer Science **391**, Springer-Verlag, 1989, 225–246.
- SERRA, 1982
J. Serra, *Image analysis and mathematical morphology*, Academic Press, London, 1982.
- SERRA, SOILLE, 1994
J. Serra, P. Soille (Eds.), *Mathematical morphology and Its Applications to Image Processing*, Computational Imaging and Vision 2, Kluwer Academic Publishers, 1994.
- SETHIAN, 1990
J.A. Sethian, Numerical algorithms for propagating interfaces: Hamilton–Jacobi equations and conservation laws, *J. of Differential Geometry* **31**, 1990, 131–161.
- SHAKED, BRUCKSTEIN, 1998
D. Shaked, A.M. Bruckstein, Pruning medial axes, *Computer Vision and Image Understanding* **69**, 1998, 156–169.
- SHEEHY, ARMSTRONG, ROBINSON, 1996
D.J. Sheehy, C.G. Armstrong, D.J. Robinson, Shape-description by medial surface construction, *IEEE Trans. Visualization Comput. Graphics* **2**, 1996, 62–72.

- SHERBROOKE, PATRIKALAKIS, BRISSON, 1996
 E.C. Sherbrooke, N.M. Patrikalakis, E. Brisson, An algorithm for the medial axis transform of 3D polyhedral solids, *IEEE Trans. Visualization Comput. Graphics* **2**, 1996, 44–61.
- SHI, RITTER, 1995
 H. Shi, G.X. Ritter, A new parallel binary image shrinking algorithm, *IEEE Trans. Image Processing* **4**, 1995, 224–226.
- SMITH, 1987
 R.W. Smith, Computer processing of line images: A survey, *Pattern Recognition* **20**, 1987, 7–15.
- SONKA, HLAVÁČ, BOYLE, 1999
 M. Sonka, V. Hlaváč, R. Boyle, *Image Processing Analysis and Machine Vision*, Brooks/Cole Publishing Company, 1999.
- SRIHARI, UDUPA, YAU, 1979
 S.N. Srihari, J.K. Udupa, M.M. Yau, Understanding the bin of parts, in *Proc. IEEE Conf. Decision Control*, 1979, 44–49.
- SUEN, WANG, 1994
 C.Y. Suen, P.S.P. Wang (eds.), *Thinning methodologies for pattern recognition*, Series in Machine Perception and Artificial Intelligence **8**, World Scientific, 1994.
- SVENSSON, NYSTRÖM, BORGEFORS, 1999
 S. Svensson, I. Nyström, G. Borgefors, Fully reversible skeletonization for volume images based on anchor-points from the D^{26} distance transform, in *Proc. 11th Scandinavian Conf. Image Analysis, SCIA '99*, 1999, 601–608.
- SZÉKELY, 1996
 G. Székely, *Shape characterization by local symmetries*, Habilitationsschrift, Eidgenössische Technische Hochschule Zürich, 1996.
- SZÉKELY ET AL., 1992
 G. Székely, Ch. Brechbühler, O. Kübler, R. Ogniewicz, T. Budinger, Mapping the human cerebral cortex using 3D medial manifolds, in *Proc. Conf. Medical Imaging V: Image Processing*, Proc. SPIE **1808**, 130–144, 1992.
- SZÉKELY, KOLLER, KIKINIS, GERIG, 1995
 G. Székely, Th. Koller, R. Kikinis, G. Gerig, Structural description and combined 3–D display for superior analysis of cerebral vascularity from MRA, in *Medical Imaging: Analysis of multimodality 2D/3D images*, IOS Press, 1995, 183–194.
- TALBOT, VINCENT, 1992
 H. Talbot, L. Vincent, Euclidean skeletons and conditional bisectors, in *Proc. Conf. Medical Imaging V: Image Processing*, Proc. SPIE **1808**, 1992, 862–876.
- TANÁCS, PALÁGYI, 1998
 A. Tanács, K. Palágyi, Efficient implementation of morphological and local neighbourhood operations, in *Extended Abstracts of Conference of PhD Students in Computer Science*, József Attila University, Szeged, 1998, 100–101.
- TANÁCS, PALÁGYI, KUBA, 1998
 A. Tanács, K. Palágyi, A. Kuba, Medical image registration based on interactively defined anatomical landmark points, *Machine Graphics & Vision* **7**, 1998, 151–158.

- TANÁCS, PALÁGYI, KUBA, 1999
 A. Tanács, K. Palágyi, A. Kuba, Target registration error of point-based methods assuming rigid-body and linear motions, in *Proc. Int. Workshop on Biomedical Image Registration WBIR'99*, 1999, 223–233.
- TSAO, FU, 1981
 Y.F. Tsao, K.S. Fu, A parallel thinning algorithm for 3-D pictures, *Computer Graphics and Image Processing* **17**, 1981, 315–331.
- TSAO, FU, 1982
 Y.F. Tsao, K.S. Fu, A 3D parallel skeletonwise thinning algorithm, in *Proc. IEEE Conf. Pattern Recognition Image Processing*, 673–683, 1982.
- UDUPA, 1993
 J.K. Udupa, *Three dimensional display*, Technical Report No. MIPG196, Medical Image Processing Group, University of Pennsylvania, Philadelphia, 1993.
- UDUPA ET AL., 1994
 J.K. Udupa, D. Odhner, S. Samarasekera, R. Goncalves, K. Iyer, K. Venugopal, S. Furuie, 3DVIEWNIX: An open, transportable, multidimensional, multimodality, multiparametric imaging software system, in *Proc. Conf. Medical Imaging: Image Processing*, Proc. SPIE **2164**, 1994, 58–73.
- VAN DEN ELSSEN, MAINTZ, POL, VIERGEVER, 1992
 P.A. van den Elsen, J.B.A. Maintz, E.J.D. Pol, M.A. Viergever, Image fusion using geometrical features, in *Proc. Conf. Medical Imaging V: Image Processing*, Proc. SPIE **1808**, 1992, 172–186.
- VAN DEN ELSSEN, POL, VIERGEVER, 1993
 P.A. van den Elsen, E.J.D. Pol, M.A. Viergever, Medical Image Matching — a Review with Classification, *IEEE Trans. Engineering in Medicine and Biology* **12**, 1993, 26–39.
- VOSS, 1993
 K. Voss, *Discrete Images, Objects, and Functions in Z^n* , Springer-Verlag, 1993.
- WATT, POLICARPO, 1998
 A. Watt, F. Policarpo, *The Computer Image*, Addison-Wesley, 1988.
- YAMADA, 1984
 H. Yamada, Complete Euclidean distance transformation by parallel operation, in *Proc. 7th Int. Conf. Pattern Recognition*, 69–71, 1984.
- YIM, CHOYKE, SUMMERS, 1999
 P.J. Yim, P.L. Choyke, R.M. Summers, Grey-scale skeletonization of small vessels in magnetic resonance angiograms, submitted to *IEEE Trans. Medical Imaging*, 1999.
- YU, YAN, 1997
 D. Yu, H. Yan, An efficient algorithm for smoothing, linearization and detection of structural feature points of binary image contours, *Pattern Recognition*, **30**, 1997, 57–69.

Összefoglaló

A váz gyakran alkalmazott régió-alapú alakjellemző, ahol egy folytonos, d -dimenziós bináris objektum vázát az objektumba beírt d -dimenziós hipergömbök középpontjai alkotják. A váz egy szemléletes hasonlattal, a prérítűz terjedésével is illusztrálható: Ha a vizsgált objektum határának minden pontját egyidejűleg meggyújtjuk és feltételezzük, hogy a tűzfrontok minden irányban egyenletes sebességgel terjednek, akkor a váz azokból a pontokból áll, ahol az objektum belsejében a tüzek találkoznak, kioltják egymást.

Diszkrét objektumok esetén az “igazi” váznak csak egy közelítése határozható meg, amivel szemben az alábbi — geometriai és topológiai — követelményeket fogalmazták meg:

- A váz a geometriailag korrekt pozíción, az objektum “közepén” helyezkedjen el és legyen invariáns a lineáris geometriai transzformációkra.
- A váz őrizze meg az eredeti objektum topológiáját (vagyis a kijelölt váz legyen topológikusan ekvivalens a kiindulási objektummal).

Diszkrét képterekre a vázkijelölés három fő technikáját különböztetjük meg:

- a távolság-transzformáció,
- a vékonyítás és
- a Voronoi-diagram számítása.

A távolság-transzformációval történő vázkijelölés (amennyiben az a pontos euklidészi távolságon alapul) képes geometriailag korrekt “vázat” produkálni, de a topológiai kritérium teljesülését nem garantálja.

A Voronoi-vázra mindkét feltétel teljesül, de nagyon drága (idő- és számításigényes) technika, mivel egy diszkrét ponthalmazból kiindulva a Voronoi-diagram folytonos elemek halmazából (3D-ben: pontok, élek, konvex poligonok és konvex poliéderek) áll.

A vékonyítás a tűzfrontok terjedését modellezi diszkrét képtereken. A front-terjedés a diszkrét térben diszkrét folyamat: egy időegység alatt a tűz frontja a tőle “egységnyi” távolságra lévő rácpontokat éri el. A vékonyítás egy iterációs lépése tehát a front-terjedés egy időegységének felel meg. Ez a vázkijelölő technika a topológia-megőrzésre helyezi a hangsúlyt, ugyanakkor nem invariáns az elforgatásra.

A vékonyítás természeténél fogva párhuzamos, mivel — a prérítűz analógia szerint — a határ valamennyi pontja egyidejűleg kap lángra, továbbá a tűzfrontok minden irányban és egyenletes sebességgel terjednek. Megjegyezzük, hogy javasoltak szekvenciális vékonyító algoritmusokat is, melyek a kontúrkijelölés és a kontúrkövetés technikáját alkalmazva imitálják a külső réteg pontjainak egyidejű eltávolítását. A front-terjedés nem csak párhuzamos, hanem lokális is. Ezen második fontos tulajdonság szerint az, hogy egy objektumpont a “tűz áldozata” lesz-e az eljárás következő fázisában vagy sem, eldönthető a kérdéses pont egy — az objektum nagyságától független méretű — szűk környezete alapján.

A vékonyító algoritmusok egy iterációs lépése tehát egy olyan lokális és párhuzamos redukciós képművelettel adható meg, ami csak határpontokat töröl. A teljes eljárás ilyen képműveletek sorozatából áll. A vékonyítás menetét az alábbi, teljesen általános sémával írhatjuk le:

repeat
a “törölhető” határpontok eltávolítása
until nem történt változás

A **repeat**-ciklus magja a vékonyítás egy iterációs lépésének felel meg. Az egyes vékonyító eljárások az iterációs lépések megszervezésében és a “törölhető” pontokra adott feltételekben különböznek egymástól. A párhuzamos vékonyító algoritmusok — a topológia megőrzése érdekében — az iterációs lépések megszervezésére a következő három módszert alkalmazzák:

- Az irány-szekvenciális módszerek egy iterációs lépést kettő vagy több egymást követő al-iterációra bontanak, amelyek során csak bizonyos irányokba eső határpontok távolíthatók el.
- Az almező-szekvenciális technika a képteret kettő vagy több diszjunkt részhalmazzra, almezőre bontja, ahol az almezők felváltva aktivizálódnak és mindig csak az aktív almezőbe eső pontok törölhetők.
- A teljesen párhuzamos módszer nem igényel al-iterációkat, viszont ennek fejében az előző technikáknál nagyobb lokális környezetet vizsgál.

A 2D vékonyítás könnyű feladatnak tűnhet, mégis nagy kihívásnak bizonyult, mivel közel 500 publikáció foglalkozik vele. Míg 2D-ben számos vékonyító algoritmus létezik, addig 3D-ben csak néhány eljárást közöltek. Ennek oka az, hogy egyrészt a 2D képpalkotó módszerek és eszközök korábban terjedtek el és így 2D-ben a vékonyítást igénylő alkalmazások is korábban jelentkeztek, másrészt pedig a topológia megőrzése jóval nehezebb probléma 3D-ben, mint 2D esetén.

Az értekezés kulcsproblémája a vékonyításon alapuló vázkijelölés 3D-ben.

A dolgozat 1. fejezete ismerteti a folytonos váz fogalmát és tulajdonságait, a digitális topológiai alapfogalmait és a topológia megőrzésére vonatkozó eredményeket, a fő vázkijelölő technikákat, a váz és a vékonyítás leírását a matematikai morfológia eszközeivel, valamint a morfológiai műveletek és a maszkillesztés kapcsolatát.

A dolgozat fő eredményei a 2. fejezetben találhatóak, ahol nem csak az általunk javasolt új algoritmusokat ismertetjük, hanem rendszerezzük a 3D vékonyító technikákat és bemutatjuk az ezidáig publikált valamennyi párhuzamos vékonyító algoritmust is. Megjegyzendő, hogy eddig csak a 2D vékonyításra közöltek összefoglaló, áttekinthető tanulmányokat. Valamennyi javasolt algoritmusunkat általánosított maszkillesztéssel írtuk le, ami több szempontból is előnyös: áttekinthető, könnyen érthető, kapcsolódik a morfológiai megközelítéshez és hatékony implementációt tesz lehetővé.

A 3D vékonyítás módszerei és algoritmusai után a 3. fejezetben a zsugorítással foglalkozunk. A zsugorítás egyrészt azért fontos, mivel az a vékonyítással rokon művelet, másrészt pedig jelentős a szerepe a váz utófeldolgozásakor.

A 4. fejezetben tárgyaljuk a váz elő- és az utófeldolgozását. Az 5. fejezetben az új vékonyító algoritmusok hatását mutatjuk be néhány “mesterséges” és “természetes” bináris objektumra. Végül, a 6. fejezet a 3D vázkijelölés és a vékonyítás alkalmazásait tárgyalja.

Az értekezésben található saját eredményeink:

- Új elégséges feltételt adtunk a 3D (26,6) képek topológia-megőrző párhuzamos redukcióira (1.2.2.).
- Megmutattuk, hogy a maszkillesztéssel és a pontonkénti Boole-műveletek segítségével megadhatók a morfológiai műveletek. (1.4.4.).

- Bevezettük az általánosított maszkillesztést és igazoltuk azt, hogy az általánosított maszkillesztéssel megadható tetszőleges lokális és párhuzamos bináris képművelet (1.4.4.).
- Bemutattunk három új irány–szekvenciális vékonyító algoritmust. Az első (2.1.6.) az általánosan alkalmazott 6 törlési iránnyal dolgozik, míg a második (2.1.7.) és a harmadik (2.1.8.) algoritmusunk elsőként javasol 8, illetve 12 törlési irányt. Mindhárom — középvonalak és középfelszínek meghatározására egyaránt alkalmas — vékonyító algoritmus topológia–megőrző a $(26,6)$ képekre.
- Ma és Sonka teljesen párhuzamos algoritmusának (2.3.2.) tesztelése során kiderült, hogy eljárásuk nem őrzi meg a topológiát. Az algoritmust módosítottuk úgy, hogy a javasolt új eljárás már topológiailag korrekt legyen (2.3.3.).
- Bevezettünk egy új vékonyító stratégiát, ami kombinálja az irány–szekvenciális és az almező–szekvenciális technikákat (2.4.). Ismertettünk egy hibrid algoritmust is, ami középvonalakra vékonyít és topológia–megőrző a $(26,6)$ képekre (2.4.1.).
- Bemutattunk egy új 3D zsugorító algoritmust, ami topológia–megőrző a $(26,6)$ képekre (3.1.).

Summary

The skeleton is a frequently used region-based shape feature. The skeleton of a continuous d -dimensional binary object is the locus of the centers of all maximal inscribed d -dimensional hyperspheres. In discrete spaces, only an approximation to the “true skeleton” can be produced. During the last two decades skeletonization has become a challenging research field. There are two major requirements to be complied with:

- the geometrical one (i.e., requiring that the “skeleton” is in the “middle” of the original object and invariant under global linear transformations) and
- the topological aspect (i.e., forcing the “skeleton” to retain the topology of the original object).

There are three major skeletonization techniques in discrete spaces:

- distance transformation,
- thinning, and
- generating Voronoi diagram.

The result of the distance transformation depends on the selected distance and the ridge extraction is a rather difficult task. The distance map based method fulfills the geometrical requirement if a good approximation to the Euclidean distance is applied, but the topological correctness is not guaranteed.

The Voronoi diagram of a discrete set of points (called generating points) is the partition of the given space into cells so that each cell contains exactly one generating point and the locus of all points which are nearer to this generating point than to other generating points. It has been shown that the skeleton of an object which is described by a set of boundary points can be approximated by a subgraph of the Voronoi diagram of that generating points. Both requirements can be fulfilled by the skeletonization based on Voronoi diagrams but it is regarded as an expensive process, especially for large and complex objects.

The thinning process is a frequently used method for producing an approximation to the skeleton in a topology-preserving way. It is based on digital simulation of the fire front propagation: border points (i.e., object points that are “adjacent” to the background) of a binary object that satisfy certain topological and geometric constraints are deleted in iteration steps. The entire process is repeated until only the “skeleton” is left. The topologically oriented thinning pays less attention to the metric properties of the object, therefore, the invariance under rotation (object orientation) is not guaranteed. Our attention is focussed on thinning, since it

- is the fastest,
- can be implemented easily,
- can be executed in parallel,
- produces topologically correct “skeletons”, and
- provides satisfactory results for complex and strongly elongated objects.

Each thinning algorithm can be sketched by the following program:

```
repeat  
    changing “deletable” border points to white  
until no points are deleted
```

Existing 3D thinning algorithms differ from each others in two regards:

- How to organize an iteration step (i.e., the kernel of the **repeat** cycle)?
- Which types of border points are regarded as “deletable”?

A simple point is an object point whose deletion does not alter the topology of the picture. Note that the simplicity of a 3D point can be decided by investigating its $3 \times 3 \times 3$ neighborhood. Thinning algorithms delete simple points which are not end-points, since preserving end-points provides important information relative to the shape of the objects. Most of the existing thinning algorithms are parallel, since the fire front propagation is by nature parallel. (It means that all border points satisfying the deletion condition of the actual phase of the process are simultaneously deleted.) Those algorithms delete a set of simple points that can alter the topology. Three methods have been proposed to overcome this problem:

- Algorithms belonging to the first type do not divide an iteration step into subiterations. In order to preserve topology, these fully parallel algorithms investigate some additional points that are in the $5 \times 5 \times 5$ neighborhood but not in the $3 \times 3 \times 3$ neighborhood.
- The second type of algorithms examines the $3 \times 3 \times 3$ neighborhood of each border point. Iteration steps are divided into a number of successive subiterations, where only border points of certain kind can be deleted in each subiteration. Consequently, each subiteration uses a different deletion rule. These algorithms are called directional or border sequential ones. Since there are six kinds of major directions in 3D images, six-subiteration directional thinning algorithms were generally proposed. Note that we have been proposed an eight-subiteration algorithm and a twelve-subiteration one, too.
- The third approach is the subfield sequential method. The set of points \mathbb{Z}^3 is divided into more disjoint subsets which are alternatively activated. At a given iteration step, only border points of the active subfield are designated to be deleted. The existing two subfield sequential 3D algorithms use eight subfields. Note that we developed a hybrid thinning algorithm using both subfield sequential and directional approaches.

The above three thinning technique are proposed to answer the first question: How to organize an iteration step of parallel thinning for providing topology preservation? The second important question is: Which types of black points are designated to be deleted? Some algorithms (in each phase) delete all simple points of a given type which are not end-points, others give the prescribed neighborhood of deletable points. Curve thinning (or medial line thinning) preserves line end-points while surface thinning (or medial surface thinning) do not delete surface end-points. Note that different surface end-point characterizations have been proposed by various authors.

This thesis deals with 3D thinning algorithms. Section 1 is devoted to the continuous skeleton and its properties, it reviews the fundamental concepts of digital topology and surveys the major theoretical results in topology preservation, describes the three major skeletonization techniques, and presents the morphological tools for skeletonization. In 1.2.2., new sufficient conditions are established for parallel reductions to preserve topology on $(26,6)$ 3D images. It is shown in 1.4.4. that any local parallel operation (including morphological ones) can be described by generalized template matching.

Section 2 reviews the thinning methodologies and the existing 3D parallel thinning algorithms. We have proposed the following four topology-preserving ones:

- 6-subiteration border sequential algorithm (using the usual six deletion directions),
- 8-subiteration border sequential algorithm (demonstrating a possible way for constructing non-conventional directional thinning algorithms),
- 12-subiteration border sequential algorithm (suggesting the first topology preserving 3D thinning algorithm that applies the twelve deletion directions), and
- hybrid algorithm (presenting a brand-new method that uses both directional and subfield sequential approaches).

Section 3 deals with shrinking. Shrinking is fairly important for us from two viewpoints. It is commonly used in pre-processing the noisy objects and in post-processing the “skeleton” to remove unwanted branches. On the other hand, thinning and shrinking are strongly connected operations (i.e., shrinking can be regarded as a thinning process that produces topological kernels). A new 6-subiteration parallel shrinking algorithm is presented in 3.1. (Its topological correctness is also proved.)

Techniques concerning pre- and post-processing are reviewed in Section 4. The morphological filtering and a complex noise reduction algorithm (Palágyi, 1993) are illustrated in 4.1. and the morphological pruning is described in 4.2.

Our thinning algorithms have been tested for several synthetic and medical images. Their results are presented in Section 5. Finally, the emerged applications to 3D skeletonization are described in Section 6.

This thesis contains the following new results:

- New sufficient condition is given for topology preservation (1.2.2.).
- The generalized template matching is introduced. We showed that either binary morphological operations or Boolean local parallel operations can be described by generalized template matching.
- Four topology-preserving 3D thinning algorithms are presented. Three of them are directional (or border-sequential) (2.1.6.–2.1.8.) using the general 6, the proposed 8, and 12 deletion directions, respectively. The last one (2.4.1.) proposes a new technique called hybrid method in which border-sequential and subfield-sequential approaches are combined.
- The fully parallel algorithm developed by Ma and Sonka (2.3.2.) is modified to preserve topology (2.3.3.).
- A non-published topology-preserving 3D shrinking algorithm is presented.

Köszönetnyilvánítás

Szüleimen és a családomon kívül még sokaknak köszönöm azt, hogy eljutottam odáig, hogy a dolgozat megírása felmerülhetett és az el is készült.

Köszönetet mondok Katona Endrének (SZTE, Szeged), aki a szakdolgozatom és a diplomamunkám témavezetőjeként megszerettette velem a párhuzamos algoritmusokat, majd munkatársként segített a pályámon.

Munkámat jelentősen előrelendítették azok a hallgatók is, akik témavezetőjüknek választottak és létrehozták azokat az eszközöket, amelyek megkönnyítették a 3D vékonyító algoritmusok tervezését és tesztelését: Alb Péter és Tanács Attila egy programrendszert fejlesztett ki 2D és 3D lokális és párhuzamos képműveletek megadására és végrehajtására, Oltyán Gábor 3D bináris képmegjelenítője pedig hasznos eszköznek bizonyult az algoritmusok eredményeinek vizsgálatakor.

A vázkijelölés módszereivel, a matematikai morfológiával és a digitális topológiával foglalkozó szakirodalom megismerésében, feldolgozásában is többen támogattak: Fazekas Attilával (KLTE, Debrecen) már sok éve közösen gyűjtjük és vitatjuk meg a fenti területek közleményeit, Gilles Bertrand (ESIEE, Paris), Toyofumi Saito (Nagoya University), Ingela Nyström (Centre for Image Analysis, Uppsala University) és Antoine Manzanera (Aérospatiale E/SCS/V, Chatillon) cikkeik különnyomatait küldték el nekem, Gábor Székelytől (ETH, Zürich) pedig megkaptam a habilitációs dolgozatát.

Személyes konzultációk során és levelezés útján számos hasznos észrevétellel és javaslattal segített: Gilles Bertrand (ESIEE, Paris), Christian Brechbühler (ETH, Zürich), Guido Gerig (ETH, Zürich; University of North Carolina at Chapel Hill), Gábor T. Herman (MIPG, University of Pennsylvania, Philadelphia), Václav Hlaváč (Czech Technical University, Prague), Yukido Kenmochi (Chiba University, Japan), Tat Y. Kong (Queens College, CUNY, Flushing), Dewey Odhner (MIPG, University of Pennsylvania, Philadelphia), Milan Sonka (University of Iowa), Gábor Székely (ETH, Zürich), Jean-Philippe Thirion (INRIA, Sophia-Antipolis) és Jayaram K. Udupa (MIPG, University of Pennsylvania, Philadelphia). Velük egy lapon kell említenem a (konferenciákra és folyóiratokhoz) benyújtott kéziratok névtelen bírálóit, akiknek a megjegyzései és javaslatai nemcsak a közleményeim színvonalát emelték, hanem szélesítették a látóteremet is. Ugyancsak nagy hasznomra voltak azok az észrevételek, amelyeket a tanszékcsoport szemináriumain, a nemzetközi képfeldolgozó nyári iskolákon (SSIP'94, SSIP'96–SSIP'99) és a meglátogatott hazai és külföldi intézetekben tartott előadások kapcsán kaptam.

Hálás vagyok Gábor Székelynek, akinek a meghívására kilenc hónapot tölthettem el Zürichben az ETH-n. Az idő alatt nem csak a Voronoi-diagramokon alapuló vázkijelöléssel ismerkedhettem meg, hanem arra is lehetőségem nyílt, hogy ott írjam meg a dolgozatom jelentős részét.

Orvosi és szintetikus tesztképeket biztosított számomra: Derek L.G. Hill (UMDS, Guy's Hospital, London), Gábor Székely (ETH, Zürich) és Pierre Tellier (LSIIT, Université Louis Pasteur, Strasbourg). Különösen hálás vagyok Erich Sorantin-nak (Department of Radiology, Karl Franzens University, Graz), akinek a vékonyító eljárásaink első gyakorlati alkalmazásait köszönhetjük.

Köszönöm több hazai és külföldi szervezetnek és alapítványnak, hogy lehetővé tették a 3D vékonyításban elért eredményeink megmérettetését rangos nemzetközi konferenciákon (IPMI'97 – Poultney, USA; CAIP'97 – Kiel; DGCI'99 – Paris) és támogatták a tanulmányutaimat (INRIA, Sophia-Antipolis; MIPG, University of Pennsylvania, Philadel-

phia; ETH, Zürich; Karl Franzens University, Graz). Jelentős támogatást kaptam az Országos Tudományos Kutatási Alapprogramoktól az OTKA T/9 023604 pályázat révén.

Észrevételeikkel sokat segítettek azok, akik a dolgozat korábbi változatait elolvasták: Fazekas Attila (KLTE, Debrecen), Szirányi Tamás (MTA, SZTAKI) és Tanács Attila (SZTE, Szeged).

A végére hagytam Kuba Attilát (SZTE, Szeged), akinek a legtöbbet köszönhetem: nemcsak a tőle kapott sokrétű és nélkülözhetetlen szakmai segítséget, hanem az Alkalmazott Informatikai Tanszéken megteremtett nyugodt, alkotó légkört, valamint az emberi gesztusokat, a biztatást és a bizalmat is.