

CSG operácie s voxelizovanými modelmi

Príspevok na Študentskú vedeckú konferenciu 2003

Autor: Pavol Novotný, študent 5MPG
Fakulta matematiky, fyziky a informatiky
Univerzita Komenského v Bratislave

Konzultant: Ing. Miloš Šrámek
Komisia pre vedeckú vizualizáciu
Rakúska akadémia vied, Viedeň

Abstrakt

V tejto práci predstavujeme novú techniku realizácie CSG operácií s voxelizovanými geometrickými objektami, ktoré sú reprezentované pomocou diskretných limitovaných vzdialenostných polí doplnených o informáciu o povrchovej normále. Táto metóda odstraňuje artefakty, ktoré vznikajú pri použití iných objemových CSG operácií, berúc do úvahy podmienky reprezentovateľnosti objektov, podľa ktorých ostré detaily nie sú v diskretných poliach reprezentovateľné. Navrhovaná technika rieši tento problém zaoblením hrán a iných ostrých detailov. Pracuje na úrovni voxelov bez potreby rekonštruovať spojité modely z ich diskretnej reprezentácie.

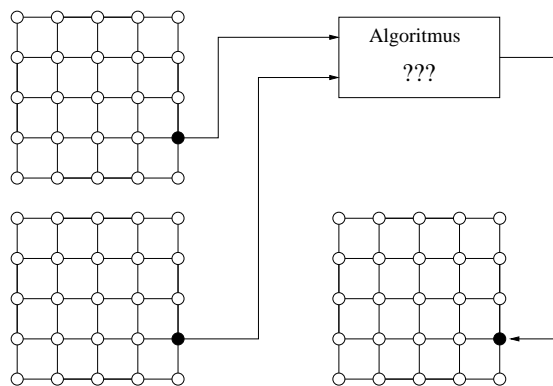
1 Úvod do problematiky

Objemová grafika sa ako podoblasť počítačovej grafiky zaoberá štúdiom priestorových objektov v ich skutočnej trojrozmernej podstate. Základom objemovej reprezentácie je voxel, elementárna jednotka objemu (skratka z anglického *volume element*). Každý voxel obsahuje určitú informáciu, ktorá lokálne charakterizuje daný objem. Typ tejto informácie závisí od konkrétnej implementácie. Voxely môžu byť vo vnútri objemu rozložené rôznym spôsobom, obvykle sa však používa karteziánska mriežka.

V našej práci vychádzame z techniky *limitovaných vzdialenostných polí*. Každý voxel má definovanú hodnotu hustoty, ktorá je odvodená od jeho vzdialenosti od povrchu. Hlavnou myšlienkou limitovaných vzdialenostných polí je poznamka, že na rekonštrukciu objektu potrebujeme len informáciu z voxelov, ktoré ležia v blízkosti povrchu. Hustota sa preto mení lineárne len v úzkej prechodovej oblasti okolo povrchu a vo vzdialenejšom okolí je konštantná. Formálne je daná funkciou

$$D(x, y, z) = \begin{cases} 2T & \text{pre } d(x, y, z) < -\delta \\ 0 & \text{pre } d(x, y, z) > \delta \\ T \left(1 - \frac{d(x, y, z)}{\delta}\right) & \text{inak} \end{cases}$$

kde $d(x, y, z)$ je vzdialenosť bodu od povrchu (vo vnútri objektu záporná), 2δ je šírka prechodovej oblasti a T je prahová hodnota definujúca povrch objektu. Na rozdiel od pôvodnej metódy



Obrázok 1: Schéma ideálnej realizácie CSG operácie na úrovni voxelov

limitovaných vzdialenostných polí si vo voxeloch ležiacich v prechodovej oblasti pamätáme navyše normalizovaný gradient hustoty, ktorý nám slúži na presnejšiu rekonštrukciu povrchovej normály.

Cieľom nášho skúmania boli CSG operácie (zjednotenie, prienik, rozdiel) s objemovými dátami. Jednoduché metódy prezentované v minulosti vedú často k nežiaducim artefaktom. Problémom je fakt, že výsledok CSG operácie vo všeobecnosti nemusí byť pomocou vzdialenostných polí reprezentovateľný. Podľa záverov článku [1] je geometrický objekt X vhodný na voxelizáciu s daným rozlíšením, ak X je S_r -otvorená aj S_r -uzavretá množina. Toto kritérium sa dá intuitívne popísať ako vlastnosť povrchu, po ktorom sa dá z oboch strán „kotúľ at“ guľa s polomerom r . Pritom pri realizácii CSG operácií bežne vznikajú hrany, kvôli ktorým výsledný objekt toto kritérium nespĺňa. Preto pri snahe vyhnúť sa artefaktom potrebujeme hrany zaobliť.

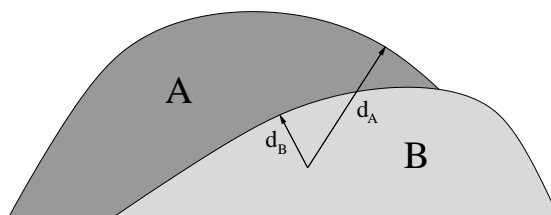
Naším zámerom bolo vyvinúť CSG operácie, ktoré budú hrany vyhladzovať bez potreby rekonštruovať spojité objekty z ich diskretnej reprezentácie. Ideálne by bolo nájsť algoritmus, ktorý na vstupe dostane dva voxelizované objekty a vytvorí žiadaný výsledok jedným prechodom voxelmi, pričom na určenie hodnoty voxelu v novom objeme sa použije vždy len informácia z dvojice korešpondujúcich voxelov zo vstupných objemov (schéma je na obrázku 1).

Nami vytvorené CSG operácie vychádzajú z tejto schémy, aj keď sa jej nedržia úplne striktné. Pri sofistikovanejších metódach totiž potrebujeme malej množine voxelov zo vstupných objemov dodať určitú chýbajúcu informáciu na základe hodnôt voxelov z ich blízkeho okolia. Hlavný princíp pracovať len na úrovni voxelov sa nám však podarilo zachovať.

Každý voxel má definovanú hustotu $d \in \langle 0, 1 \rangle$. Mimo objektu je hustota nulová, vo vnútri je jednotková a na povrchu má hodnotu 0,5. Voxel v prechodovej oblasti majú navyše zapamätaný normalizovaný gradient \vec{n} . Algoritmus vo všeobecnosti pracuje s hodnotami $d_a, \vec{n}_a, d_b, \vec{n}_b$ vstupných voxelov a ich analýzou získa výsledné hodnoty d, \vec{n} .

Skôr, ako sa začneme venovať konštrukcii CSG operácií, je dobré si uvedomiť, že medzi zjednotením, prienikom a rozdielom platia nasledovné vzťahy

$$\begin{aligned} A \cap B &= (A^c \cup B^c)^c & A \cup B &= (A^c \cap B^c)^c \\ A \setminus B &= (A^c \cup B)^c & A \setminus B &= A \cap B^c, \end{aligned}$$



Obrázok 2: Ilustrácia k minmaxovému kritériu

A, B sú vstupné objekty. d_A je vzdialenosť od povrchu A a zároveň od $A \cup B$, d_B je vzdialenosť od povrchu B a zároveň od $A \cap B$.

kde A^c (doplnok k objektu A) získame na voxelovej úrovni jednoducho:

$$\begin{aligned} d^c &= 1 - d \\ \vec{n}^c &= -\vec{n} \end{aligned}$$

To znamená, že všetky operácie môžeme vykonať jednotným spôsobom. Stačí vedieť vytvoriť zjednotenie (respektíve prienik) a ostatné operácie naň prevedieme. Ďalej budeme na ilustráciu používať ľubovoľnú CSG operáciu podľa toho, ktorá bude v danej situácii názornejšia.

2 Jednoduchá CSG operácia

Prvú sme implementovali jednoduchú CSG operáciu, ktorá výsledný voxel vytvorí priamo prebratím celej informácie (respektíve jej negáciou) z jedného zo vstupných voxelov. O tom, ktorý z nich to bude, rozhodne „minmaxové“ kritérium:

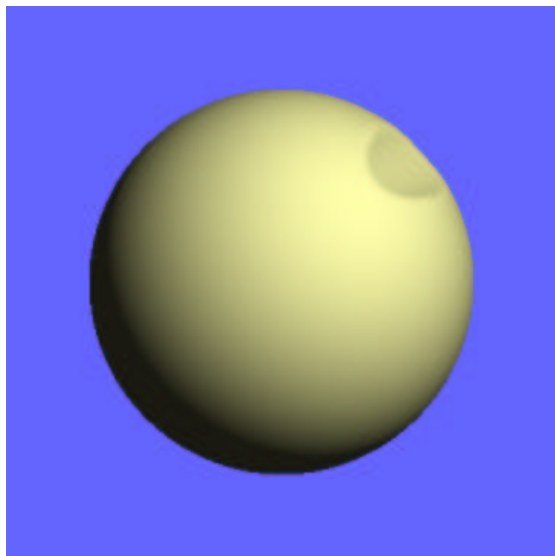
$$\begin{aligned} d_{A \cup B} &= \max(d_a, d_b) \\ d_{A \cap B} &= \min(d_a, d_b) \\ d_{A \setminus B} &= \min(d_a, 1 - d_b) \end{aligned}$$

Gradient preberieme z toho istého voxelu ako hustotu, v prípade rozdielu ho niekedy potrebujeme otočiť (ak je výsledná hustota $1 - d_b$).

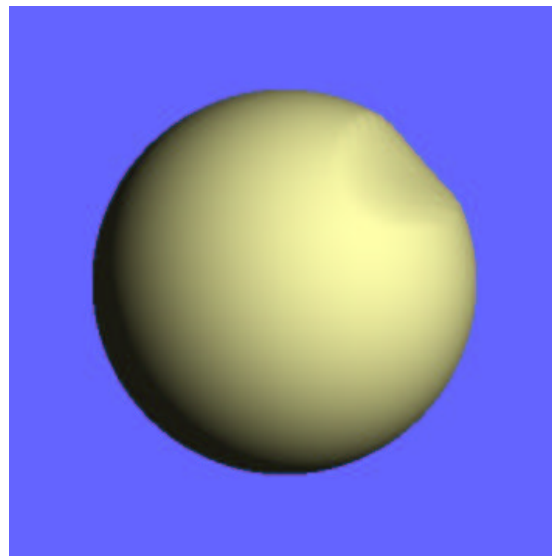
Minmaxové kritérium už bolo v minulosti viackrát aplikované (napríklad v [2][3][4]), motiváciu k nemu poskytuje obrázok 2. V článku [2] bol navrhnutý alternatívny spôsob na výpočet hustoty rozdielu objektov:

$$d_{A \setminus B} = \max(0, d_a - d_b)$$

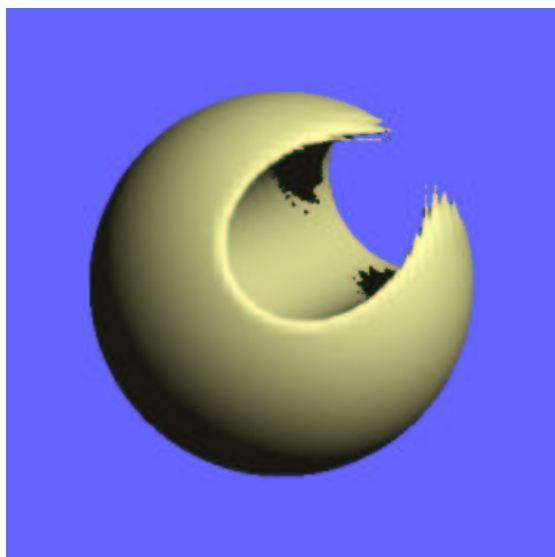
Táto realizácia rozdielu vedie k rozšíreniu hrany. Ostré hrany potom vyzerajú lepšie (sú menej zubaté), ale pri tupých hranách dochádza k vizuálnemu zhoršeniu. Na obrázkoch 3 a 4 môžeme obe metódy porovnať — ťažko stanoviť, ktorá je lepšia. My sme sa rozhodli kvôli uniformite používať prvú metódu, pretože zachováva platnosť vzťahov medzi zjednotením, prienikom a rozdielom.



(1a)



(2a)



(1b)



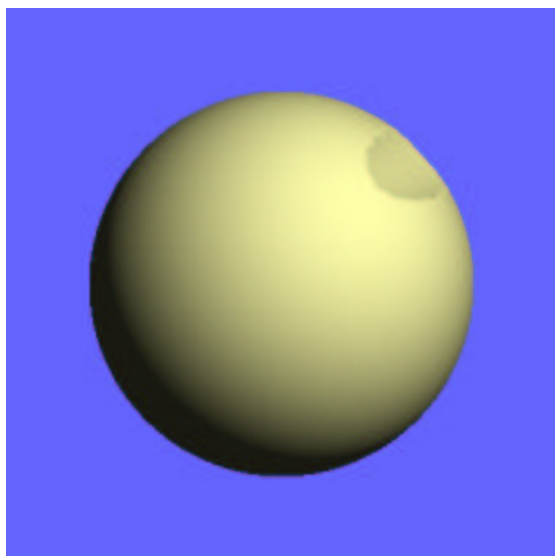
(2b)

Obrázok 3: Porovnanie dvoch spôsobov realizácie rozdielu objektov pomocou jednoduchého minmaxového kritéria (objem bez gradientu)

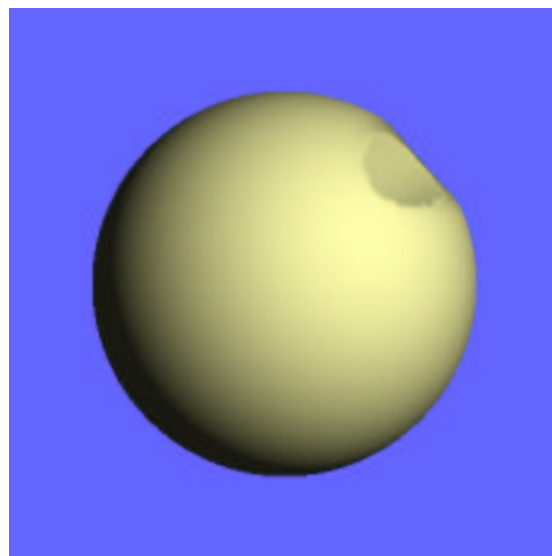
Model: guľa minus valec. Rôzne výpočty výslednej hustoty:

(1) $d_{A \setminus B} = \min(d_a, 1 - d_b)$

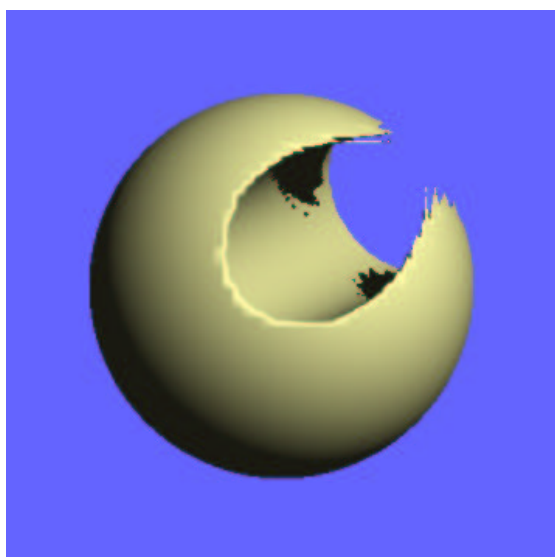
(2) $d_{A \setminus B} = \max(0, d_a - d_b)$



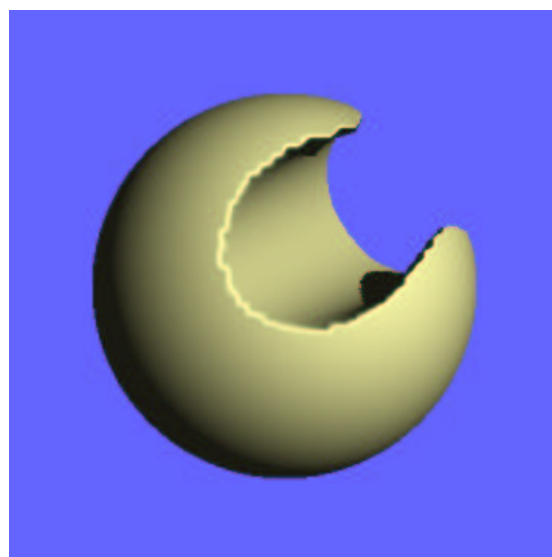
(1a)



(2a)



(1b)



(2b)

Obrázok 4: Porovnanie dvoch spôsobov realizácie rozdielu objektov pomocou jednoduchého minmaxového kritéria (objem s gradientom)

Model: guľa minus valec. Rôzne výpočty výslednej hustoty:

(1) $d_{A \setminus B} = \min(d_a, 1 - d_b)$

(2) $d_{A \setminus B} = \max(0, d_a - d_b)$

3 Vylepšená CSG operácia

Minmaxové kritérium dáva dobré výsledky skoro všade, zlyháva len v blízkosti prieniku daných povrchov. To je práve kritická oblasť, kde vznikajú hrany, ktoré potrebujeme zaobliť tak, aby výsledok bol korektné reprezentovateľný. Obrázok 5 ilustruje situáciu v okolí hrany, ktorá vznikne prienikom objektov A, B . Kvôli prehľadnosti sú zobrazené len hranice prechodových oblastí (nazývame ich *vnútorný* a *vonkajší povrch*) — skutočný povrch prechádza stredom medzi nimi. Metódu si popíšeme na rovinnom prípade, ktorý sa ľahko zovšeobecní pre trojrozmerný priestor. Vychádzame z prieniku dvoch polrovín, pričom treba rozlíšiť dve možnosti podľa toho, aký uhol zvierajú:

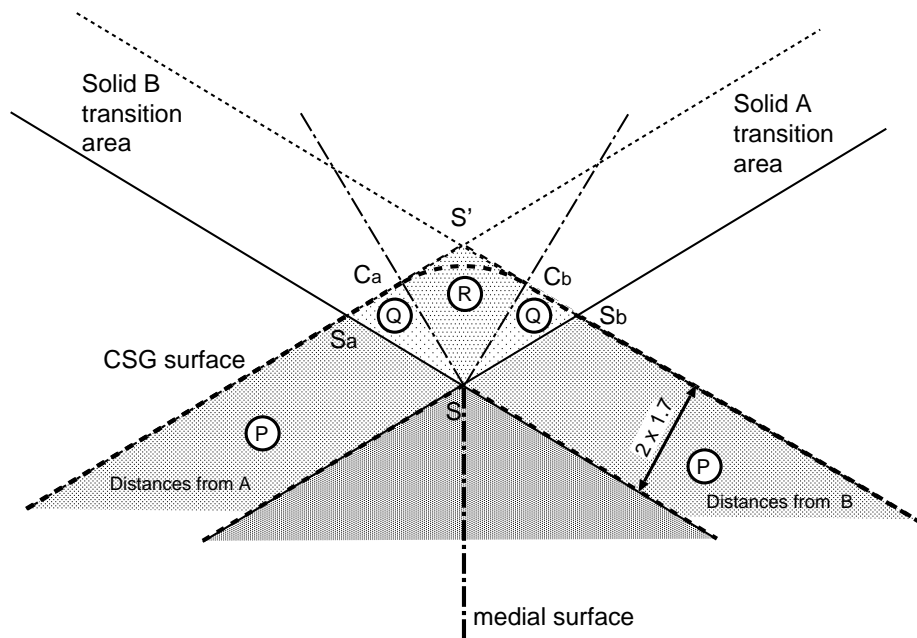
tupý uhol: V oblastiach P, Q získame správny výsledok pomocou minmaxového kritéria. V oblasti R upravíme hustotu a jej gradient tak, aby korešpondovala so vzdialenosťou od bodu S , ktorý je priesečníkom vnútorných povrchov. Vonkajší povrch prieniku vytvorí v oblasti R kružnicový oblúk so stredom S , polomerom $2r$ a koncovými bodmi C_a, C_b , kde r je polomer prechodovej oblasti. Podobne i skutočný povrch bude v R tvorený kružnicovým oblúkom, ale s polomerom r . Týmto presne splníme kritérium otvorenosti a uzavretosti, ktoré v rovinnom prípade znamená, že sme schopní „kotúľat“ kruh s polomerom r po oboch stranách povrchu.

ostrý uhol: Tu je situácia zložitejšia. Potrebovali by sme dať voxelom v oblastiach Q, R informáciu, ktorá by zodpovedala ich vzdialenosti od bodu S (podobne, ako v prípade tupého uhlu). Problém je v tom, že body v Q ležia len v prechodovej oblasti jedného objektu, preto v nich poznáme hustotu a gradient len pre jeden povrch. Na základe hodnôt voxelov nedokážeme odlíšiť oblasti P a Q . Jedine v R môžeme použiť iný postup ako minmaxové kritérium. Aby sme sa čo najviac priblížili ideálnemu riešeniu, voxelom v R priradíme informáciu tak, aby vonkajší povrch vytvoril kružnicový oblúk spájajúci body S_a, S_b . Polomer volíme tak, že sa oblúk v S_a, S_b dotýka vonkajších povrchov objektov A, B . Prirodzene, týmto sa nám nepodarí splniť kritérium otvorenosti a uzavretosti — čím menší uhol povrchy zvierajú, tým väčšiu chybu dostaneme.

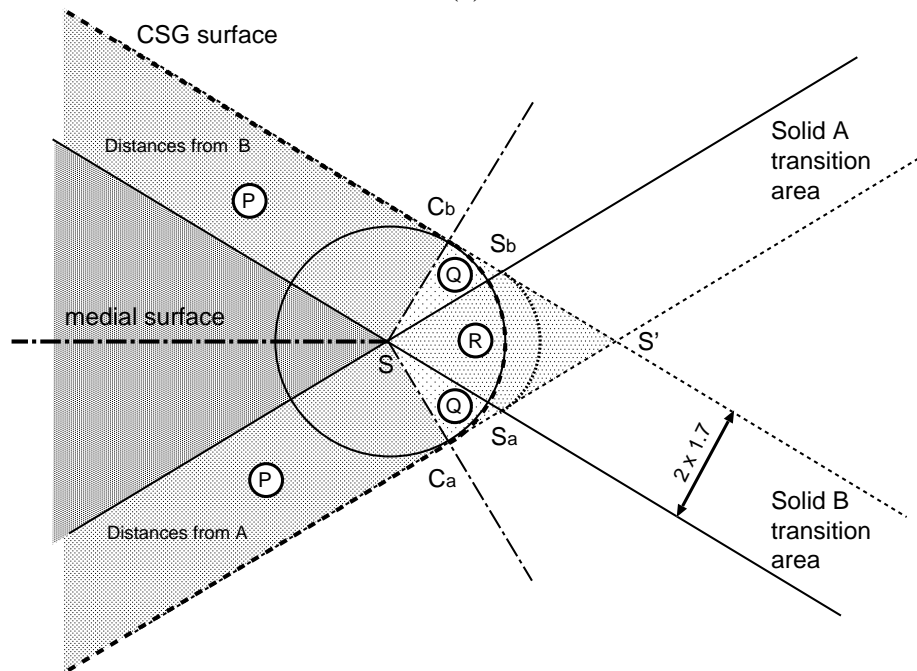
Ktorý z uvedených prípadov nastal, rozpoznáme na základe skalárneho súčinu normalizovaných gradientov.

Týmto sme si vysvetlili filozofiu metódy, teraz si popíšeme konkrétny výpočet hustoty a gradientu v kritickej oblasti. Kľúčovým je riešenie úlohy načrtnutej na obrázku 6. Máme daný bod P a lineárne nezávislé vektory \vec{u}, \vec{v} , ktoré určujú roviny α, β — roviny α, β sú porade kolmé na vektory \vec{u}, \vec{v} a od bodu P sú vzdialené $\|\vec{u}\|, \|\vec{v}\|$. Ďalej X je ten bod priesečnice rovín, ktorý je najbližšie k bodu P . Naším cieľom je nájsť vektor $\vec{x} = \overrightarrow{PX}$. Využijeme tri fakty:

1. $(\vec{x} - \vec{u}) \perp \vec{u}$
2. $(\vec{x} - \vec{v}) \perp \vec{v}$
3. \vec{x} je lineárnou kombináciou vektorov \vec{u}, \vec{v}

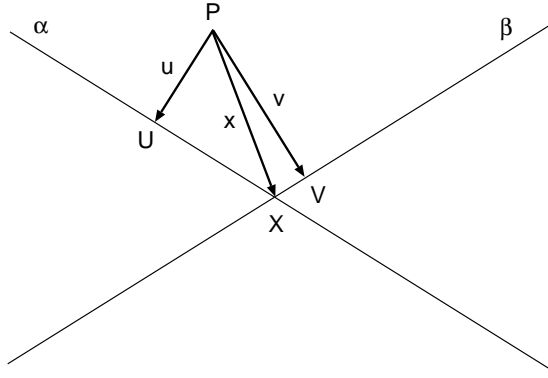


(a)



(b)

Obrázok 5: Schéma prieniku dvoch objektov A, B
 (a) hrana s tupým uhlom, (b) hrana s ostrým uhlom (b)



Obrázok 6: Prienik dvoch rovín

Roviny α , β sú dané bodom P a vektormi \vec{u} , \vec{v} . X je najbližší bod priesečnice rovín k bodu P . Vektor \vec{x} nezávisí od voľby bodu P , ale len od vektorov \vec{u} , \vec{v} .

Na základe týchto poznatkov zostavíme sústavu rovníc a jej riešením získame vzťah

$$\vec{x} = \frac{(\|\vec{u}\|^2 - \vec{u}\vec{v}) \cdot \|\vec{v}\|^2}{\|\vec{u}\|^2\|\vec{v}\|^2 - (\vec{u}\vec{v})^2} \cdot \vec{u} + \frac{(\|\vec{v}\|^2 - \vec{u}\vec{v}) \cdot \|\vec{u}\|^2}{\|\vec{u}\|^2\|\vec{v}\|^2 - (\vec{u}\vec{v})^2} \cdot \vec{v} \quad (*)$$

Na obrázku 7 je naznačená schéma, z ktorej vyplýva výpočet gradientu a hustoty pre voxely ležiace v kritickej oblasti v okolí tupej hrany zjednotenia dvoch objektov. Vo V máme dané hustoty d_a , d_b a normalizované gradienty \vec{n}_a , \vec{n}_b vstupných objemov A , B a chceme získať hodnoty d a \vec{n} výsledného zjednotenia. Využijúc, že $\vec{a} = -d_a\vec{n}_a$ a $\vec{b} = -d_b\vec{n}_b$, na základe (*) dostávame:

$$\begin{aligned} \vec{g} &= -K\vec{n}_a - L\vec{n}_b \\ K &= \frac{d_a - d_b\vec{n}_a\vec{n}_b}{1 - (\vec{n}_a\vec{n}_b)^2} \\ L &= \frac{d_b - d_a\vec{n}_a\vec{n}_b}{1 - (\vec{n}_a\vec{n}_b)^2} \end{aligned}$$

Pre $K > 0$, $L > 0$ leží V v oblasti R , preto mu priradíme hustotu a gradient na základe vektora \vec{g} :

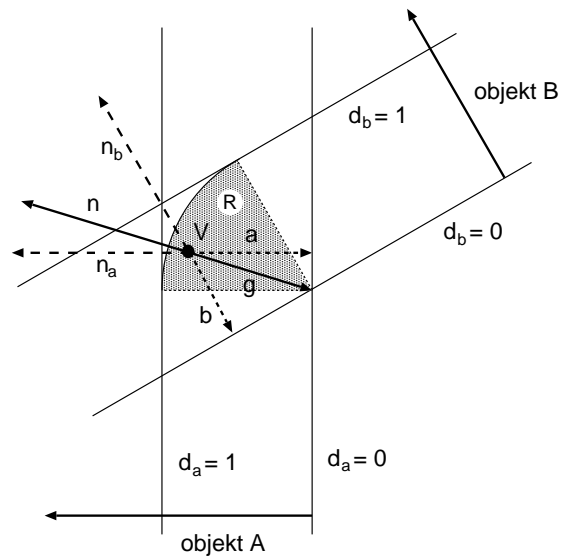
$$\begin{aligned} d &= \|\vec{g}\| \\ \vec{n} &= -\frac{\vec{g}}{\|\vec{g}\|} \end{aligned}$$

Inak využijeme minmaxové kritérium:

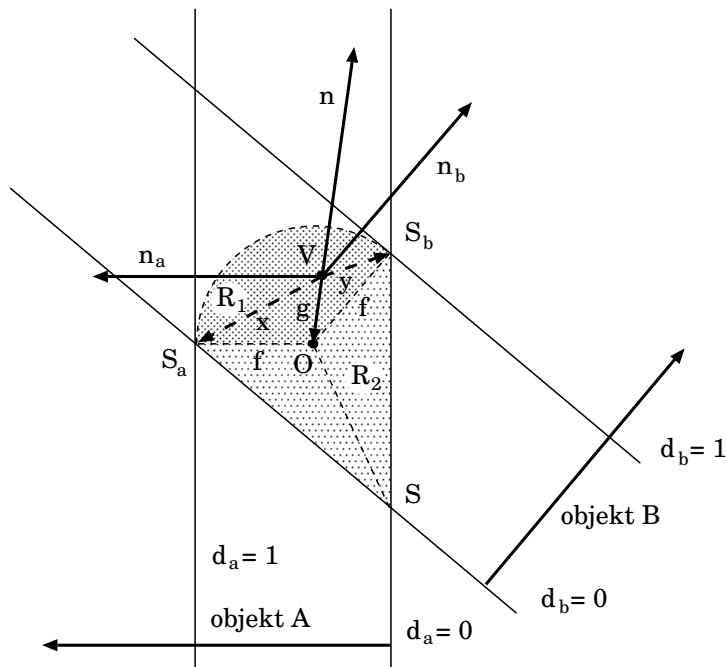
$$\begin{array}{ll} \text{pre } K \leq 0 : & \text{pre } L \leq 0 : \\ d = d_a & d = d_b \\ \vec{n} = \vec{n}_a & \vec{n} = \vec{n}_b \end{array}$$

V okolí ostrej hrany je situácia komplikovanejšia (pozri obrázok 8). Vektory \vec{x} , \vec{y} môžeme vyjadriť na základe (*) nasledovne:

$$\begin{aligned} \vec{x} &= K_x\vec{n}_a - L_x\vec{n}_b & \vec{y} &= -K_y\vec{n}_a + L_y\vec{n}_b \\ K_x &= \frac{(1-d_a)+d_b\vec{n}_a\vec{n}_b}{1-(\vec{n}_a\vec{n}_b)^2} & L_x &= \frac{d_b+(1-d_a)\vec{n}_a\vec{n}_b}{1-(\vec{n}_a\vec{n}_b)^2} \\ K_y &= \frac{d_a+(1-d_b)\vec{n}_a\vec{n}_b}{1-(\vec{n}_a\vec{n}_b)^2} & L_y &= \frac{(1-d_b)+d_a\vec{n}_a\vec{n}_b}{1-(\vec{n}_a\vec{n}_b)^2} \end{aligned}$$



Obrázok 7: Kritická oblasť zjednotenia objektov — tupý uhol
V leží v kritickej oblasti zjednotenia $A \cup B$. Jeho hustotu a gradient odhadneme na základe vektora \vec{g} .



Obrázok 8: Kritická oblasť zjednotenia objektov — ostrý uhol
V leží v kritickej oblasti zjednotenia $A \cup B$. Jeho hustotu odhadneme na základe vektora \vec{g} .

Z týchto vzťahov vieme ďalej odvodiť:

$$\begin{aligned}\vec{g} &= -K_y\vec{n}_a - L_x\vec{n}_b \\ f &= K_x + K_y = L_x + L_y = \frac{1}{1-\vec{n}_a\vec{n}_b}\end{aligned}$$

Keďže bod O je od vnútorného povrchu vzdialený o dĺžku f , musí mať hustotu $1 - f$. Voxel V je k povrchu o $\|\vec{g}\|$ bližšie, preto mu priradíme hodnotu

$$d = (1 - f) + \|\vec{g}\|.$$

Oblasť R_1 je charakterizovaná podmienkou $K_y > 0, L_x > 0$. Voxely mimo nej dostanú hustotu podľa minmaxového kritéria.

Rozhodnúť, aký smer gradientu by mali mať voxely v oblastiach R_1, R_2 , nie je triviálne. Ak by sme sa rozhodli pre smer k bodu S , na úsečkách S_aS, S_bS by sa gradient menil nespojite, pretože mimo R_1, R_2 je smer gradientu kolmý na jeden z pôvodných povrchov. Ak by sme v oblasti R_2 ponechali gradient podľa minmaxového kritéria a v R_1 by sme zvolili smer k bodu O , vznikla by nespojitosť na úsečke OS . Preto počítame gradient v oblastiach R_1, R_2 týmto spôsobom:

$$\vec{n} = \frac{d_a\vec{n}_a + d_b\vec{n}_b}{\|d_a\vec{n}_a + d_b\vec{n}_b\|}$$

Tento vzťah je odvodený z prieniku povrchov, ktoré sú na seba kolmé. Takto stanovený gradient nemá v kritickej oblasti presne smer normály na povrch, ale aspoň sa všade mení spojite (okrem bodu S , kde to nevádi). Chyba, ktorá týmto vznikne, je menej závažná, ako nespojitý gradient.

Špeciálne treba ošetriť situácie, keď sú blízko seba rovnobežné povrchy (presnejšie povedané, takmer rovnobežné s určitou toleranciou). Pre rovnako orientované povrchy priamo použijeme minmaxové kritérium. Pre opačne orientované povrchy vychádzame s limitného prípadu ostrého uhla, preto zjednotenie realizujeme na základe hustôt nasledovne:

Ak $d_a + d_b \leq 1$, potom máme:

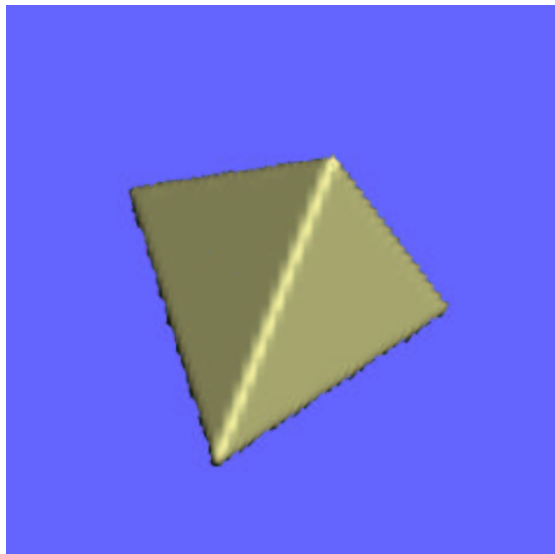
$$\begin{array}{ll} \text{pre } d_a \geq d_b : & \vec{n} = \vec{n}_a \\ & d = d_a \end{array} \qquad \begin{array}{ll} \text{pre } d_a < d_b : & \vec{n} = \vec{n}_b \\ & d = d_b \end{array}$$

Ak $d_a + d_b > 1$, potom stanovíme: $d = 1$

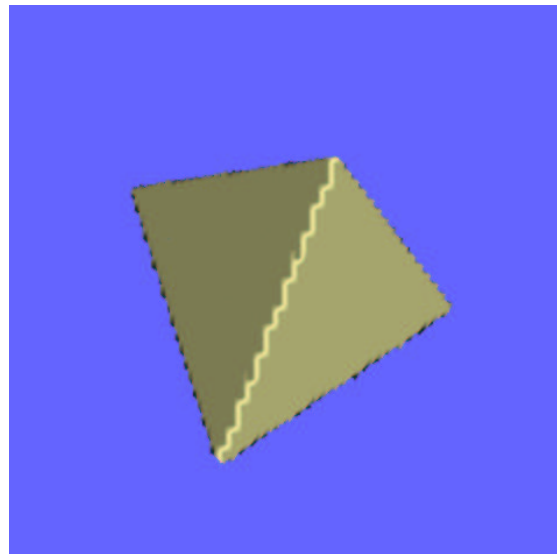
Vylepšená CSG operácia korektne vyhladzuje tupé hrany a na ostrých hranách do určitej miery koriguje artefakty. Na obrázku 9 vidno pravidelný štvorsten, na ktorom touto metódou dosahujeme viditeľne lepšie výsledky ako jednoduchou CSG operáciou, hoci steny zvierajú ostrý uhol (asi $75,5^\circ$). Na veľmi ostrých hranách (obrázok 10(c)) sa však nedostatky prejavajú viac, dochádza k podobnému „zúbkateniu“ ako pri jednoduchej metóde. Preto sme pristúpili k návrhu dokonalejšej techniky, ktorú prezentujeme na ďalších stranách.

4 Pokročilá CSG operácia

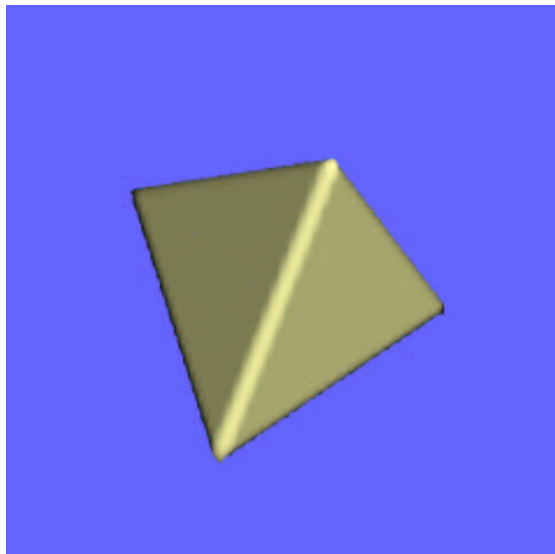
Pokročilá metóda sa snaží odstrániť artefakty aj na ostrých hranách. Hlavným problémom je spomínaná oblasť (Q na obrázku 5(b)), v ktorej máme len informáciu o jednom povrchu, ale



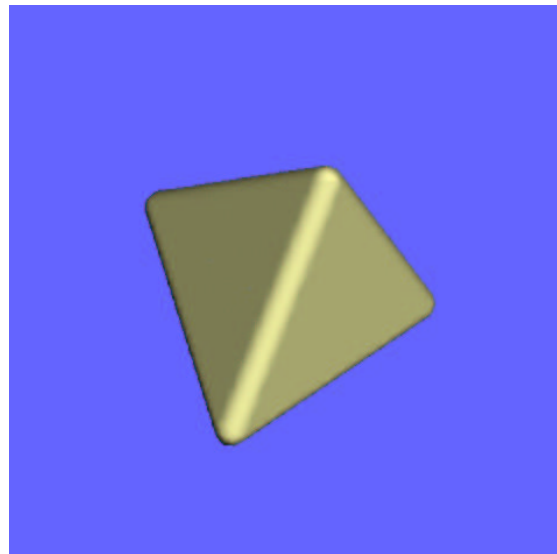
(a)



(b)

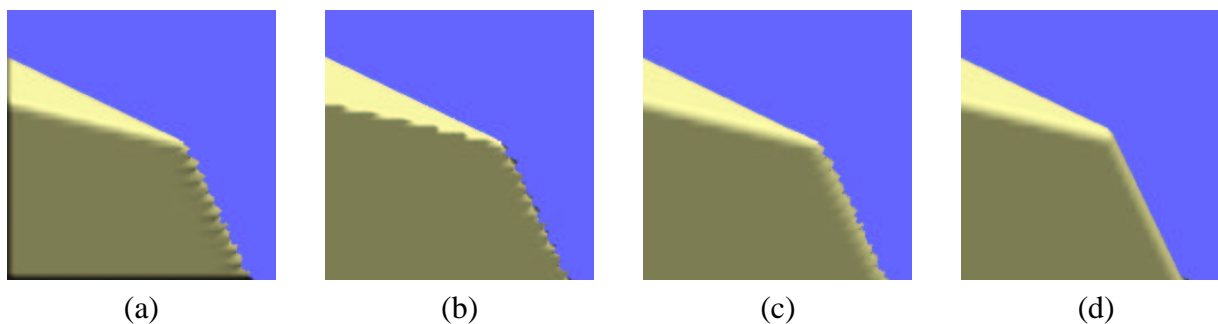


(c)



(d)

Obrázok 9: Artefakty na hranách pravidelného štvorstenu pri rôznych výpočtoch CSG operácií
Objem bez gradientu: (a) jednoduchá operácia. Objem s gradientom: (b) jednoduchá operácia,
(c) vylepšená operácia, (d) pokročilá operácia.



Obrázok 10: Artefakty na ostrých hranách pri rôznych výpočtoch CSG operácií
Objem bez gradientom: (a) jednoduchá operácia. *Objem s gradientom:* (b) jednoduchá operácia, (c) vylepšená operácia, (d) pokročilá operácia.

na správne zaoblenie hrany potrebujeme aj údaje o druhom povrchu. Keď ich budeme mať, môžeme výpočet určený doteraz len pre tupé hrany rozšíriť na ľubovoľný typ hrany. Za týmto účelom musíme pridať do algoritmu nasledovné kroky:

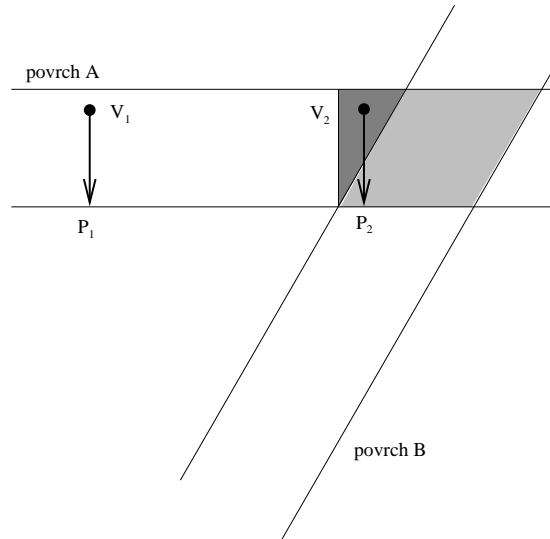
1. Pre voxely, ktoré ležia len v okolí jedného povrchu, treba rozlíšiť, či patria do oblasti P alebo Q .
2. Voxelom v oblasti Q musíme pred vykonaním CSG operácie doplniť chýbajúcu informáciu o vzťahu k druhému povrchu.

Hoci hustota pamätaná vo voxeloch má len hodnoty z intervalu $(0, 1)$, pri realizácii CSG operácií pracujeme aj s hodnotami mimo tohto intervalu. Pritom tieto hodnoty vypočítame vždy tesne pred realizáciou CSG operácie s danou dvojicou voxelov, preto túto informáciu nepotrebujeme do voxelov ukladať.

Test, ktorým rozpoznávame voxely v oblasti Q , je ilustrovaný obrázkom 11. Vo voxelu V máme informáciu o povrchu A , na základe ktorej vieme určiť súradnice bodu P (päta kolmice na hranicu prechodovej oblasti). Ak bod P leží v prechodovej oblasti objektu B , potom voxel V sa nachádza v kritickej oblasti Q , inak je mimo nej.

Pravda, testovať polohu bodu P vzhľadom na povrch B nie je úplne triviálne. P je totiž bod spojitého priestoru, ktorý leží v niektorej mriežkovej bunke. Vo vrcholoch bunky sú voxely, z ktorých niektoré môžu byť vo vnútri a niektoré mimo prechodovej oblasti B . Pomocou tejto neúplnej informácie potrebujeme odhadnúť hustotu v bode P . S týmto faktom sa vysporiadavame nasledovne:

- Ak sú všetky voxely danej bunky mimo prechodovej oblasti B , potom aj bod P je mimo.
- Ak sú všetky voxely danej bunky vo vnútri prechodovej oblasti B , potom aj bod P je vo vnútri.
- Inak skontrolujeme všetkých 26 susedných buniek (susediacich stenou, hranou alebo vrcholom) a z nich si vyberieme tie, ktoré majú všetky voxely v prechodovej oblasti B .



Obrázok 11: Test na rozlíšenie voxelov, ktoré ležia v kritickej oblasti

Z každej takejto bunky získame trilineárnou interpoláciou odhad hustoty v bode P . Bunkám priradíme váhy na základe ich vzdialeností od bodu P a vytvoríme váhovaný priemer z odhadnutých hustôt. Výslednú hustotu priradíme bodu P a podľa nej určíme, či voxel V je v kritickej oblasti.

V druhom kroku sa snažíme voxelu V dodať hodnoty hustoty a gradientu týkajúce sa povrchu B . Rozlišujeme dva prípady:

- (i) Ak aspoň v jednom smere ležia najbližšie dva voxelov v prechodovej oblasti, využijeme informáciu z týchto voxelov (obrázok 12(a)). Ak napríklad chceme vypočítať hustotu a gradient vo voxelu $V_{i,j,k}$ pomocou známych voxelov $V_{i+1,j,k}$, $V_{i+2,j,k}$, stanovíme:

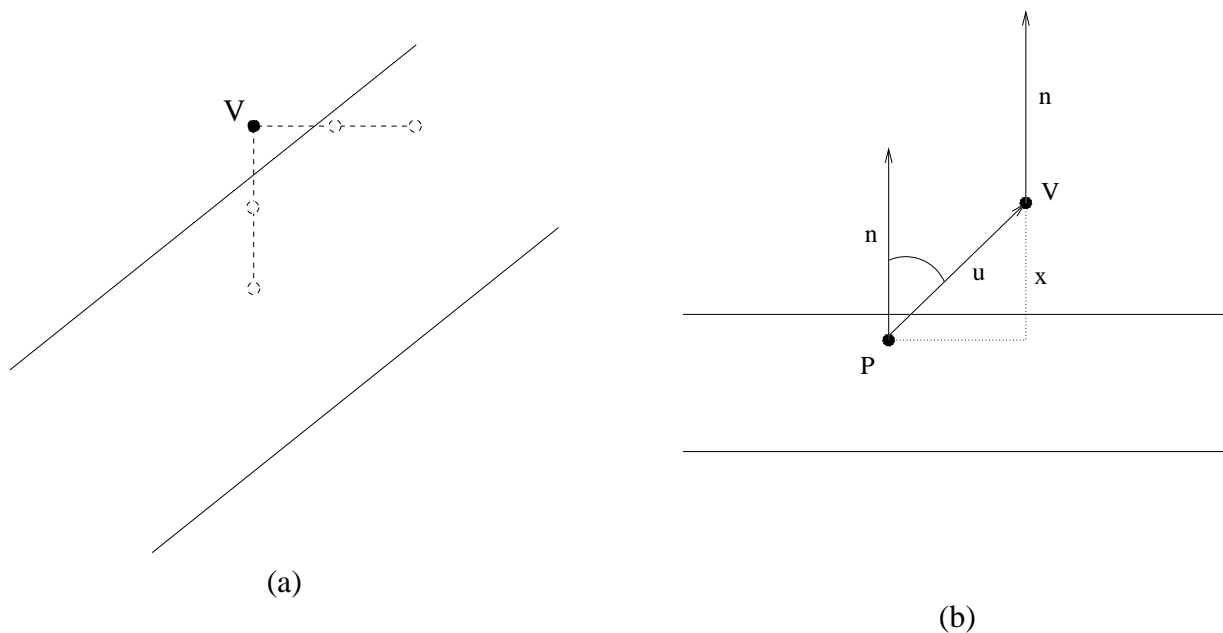
$$\begin{aligned} d_{i,j,k} &= 2d_{i+1,j,k} - d_{i+2,j,k} \\ \vec{n}_{i,j,k} &= 2\vec{n}_{i+1,j,k} - \vec{n}_{i+2,j,k} \end{aligned}$$

V prípade, že môžeme takýto odhad spraviť pre dvojicu voxelov vo viacerých smeroch, výsledok vytvoríme spriemerovaním týchto odhadov.

- (ii) Keď sa v okolí V nenachádza žiadna potrebná dvojica voxelov, použijeme informácie zistené v bode P (obrázok 12(b)). Gradient \vec{n} bodu P určíme rovnakým postupom ako jeho hustotu. Na základe vektorov \vec{n} , $\vec{u} = \vec{PV}$ vypočítame rozdiel hustôt x v bodoch V , P . Voxel V potom získa hodnoty

$$\begin{aligned} d_V &= d_P + x \\ \vec{n}_V &= \vec{n}_P . \end{aligned}$$

Opäť treba osobitne riešiť výskyt rovnobežných povrchov. Pri zjednotení použijeme v prípade rovnako orientovaných povrchov minmaxové kritérium. Ak sú blízko seba opačne orientované povrchy, medzeru medzi nimi nie sme schopní reprezentovať, preto ju vyplníme vnútornou hustotou. Analogicky postupujeme pri realizácii prieniku a rozdielu.



Obrázok 12: *Doplnenie informácie do voxelu v kritickej oblasti*
 (a) pomocou susedných voxelov, (b) pomocou predtým nájdenej bodu P

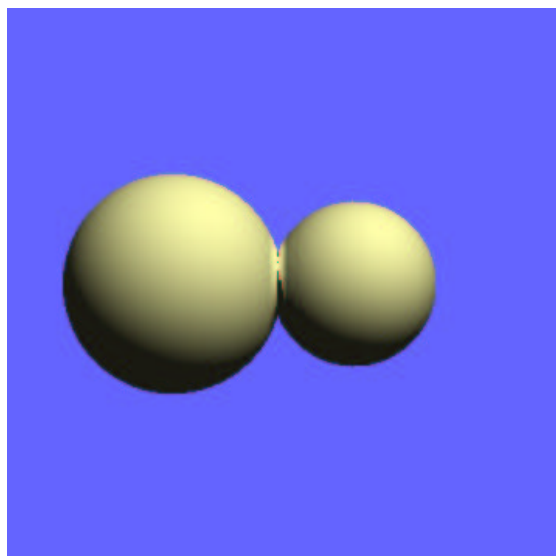
Ako vidno na viacerých obrázkoch (9, 10, 14, 16), touto metódou sa nám podarilo vyriešiť problém prakticky všetkých hrán. V prípade ostrých hrán to síce vedie k ich viditeľnému zaobleniu (obrázky 14, 16), čo je však v súlade s kritériom otvorenosti a uzavretosti. Pri zjednotení dvoch takmer sa dotýkajúcich objektov vzniká trochu iný typ artefaktov ako pri použití predošlých metód (obrázok 13). Problém je v tom, že úzka medzera medzi objektami nie je reprezentovateľná a pokročilá CSG operácia ju zaplní tak, aby bol medzi povrchmi hladký prechod.

Hoci pri popise metódy sme predpokladali voxely so zapamätaným gradientom, túto techniku je možné použiť aj pre bezgradientové voxely a v prípade potreby gradient dopočítať (pomocou stredových diferencií). Ako však vidno na obrázku 15, týmto spôsobom získame na ostrých hranách horšie výsledky, ako pri použití gradientového voxelu. To potvrdzuje náš predpoklad, že ukladať si gradient je výhodné napriek zvýšeným požiadavkám na pamäť.

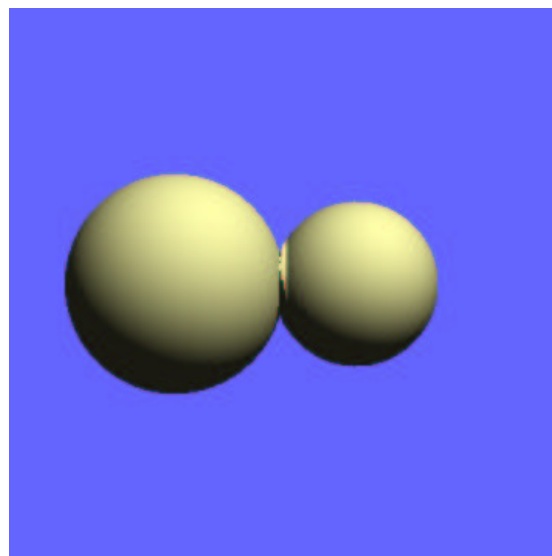
Z pozorovania vyplýva, že pokročilá CSG operácia dosahuje veľmi uspokojivé výsledky. Okrem zvláštnych prípadov, kde sa stále objavujú artefakty (obrázok 17). S týmto problémom sme sa pokúsili vysporiadať pomocou špeciálnej CSG operácie.

5 Špeciálna CSG operácia

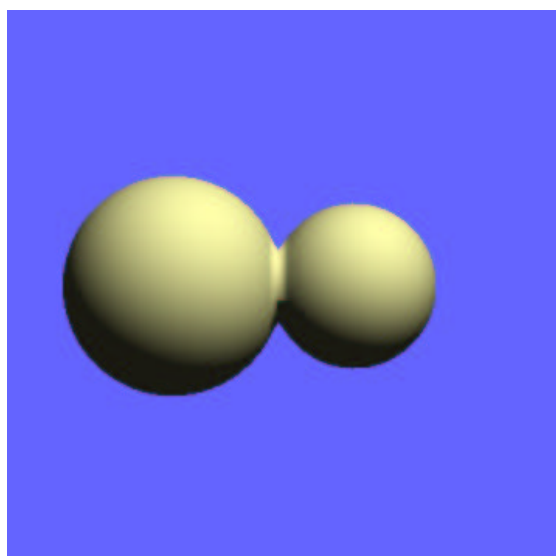
Špeciálna CSG operácia vychádza s pokročilejšou CSG operáciou, pričom sa snaží odhaľovať regióny, kde by mohli nastať problémy. Pokročilá CSG operácia pracuje presne len v situáciách, keď sa pretínajú lokálne rovinné povrchy. Čím viac sú povrchy zakrivené, tým k väčšej chy-



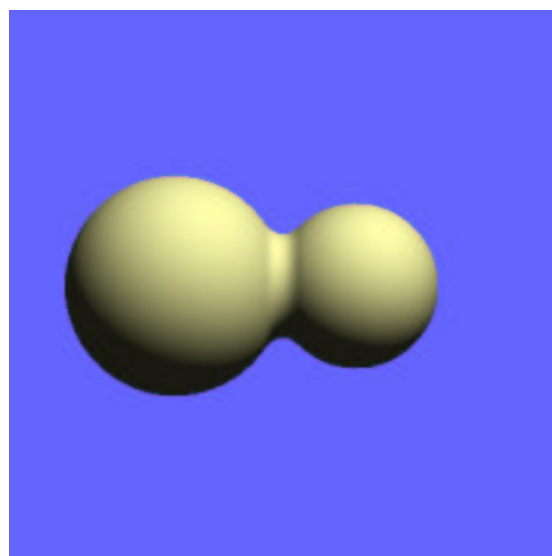
(a)



(b)



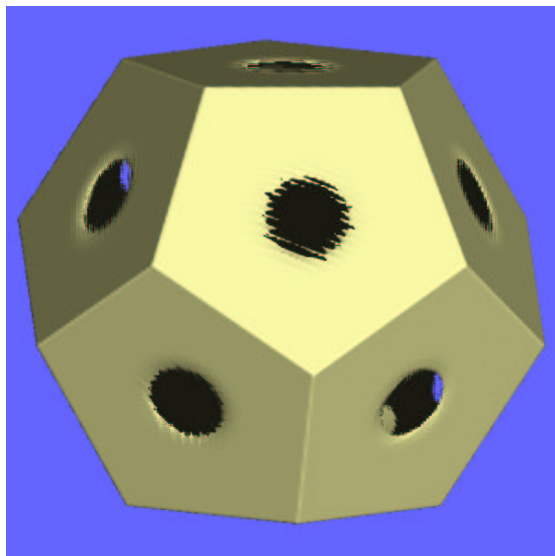
(c)



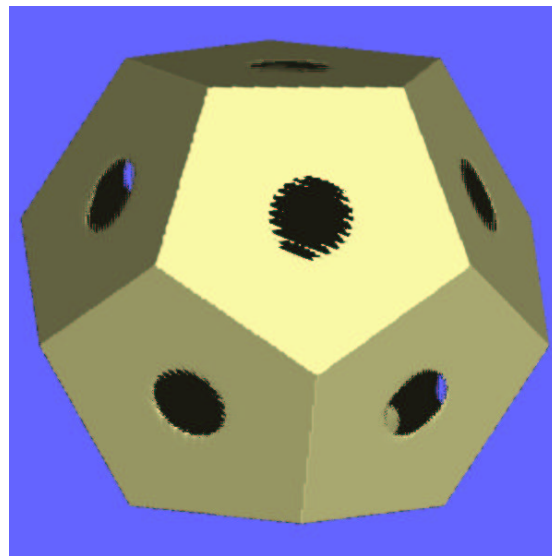
(d)

Obrázok 13: Artefakty v okolí dvoch dotýkajúcich sa povrchov pri rôznych výpočtoch CSG operácií

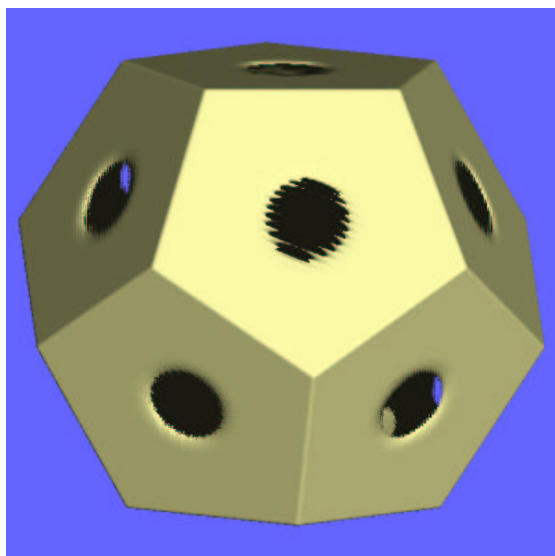
Objem bez gradientu: (a) jednoduchá operácia. Objem s gradientom: (b) jednoduchá operácia, (c) vylepšená operácia, (d) pokročilá operácia.



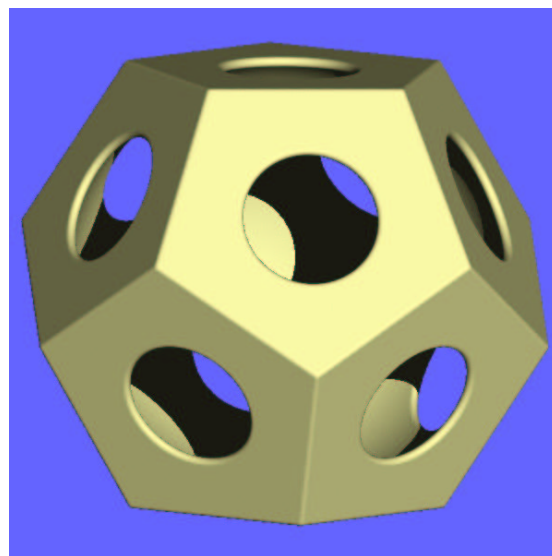
(a)



(b)



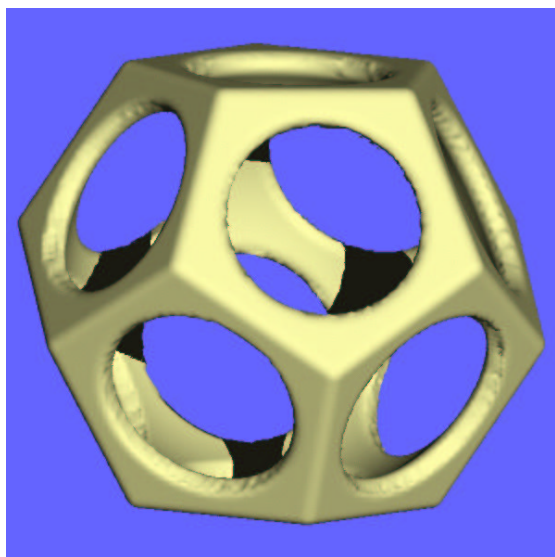
(c)



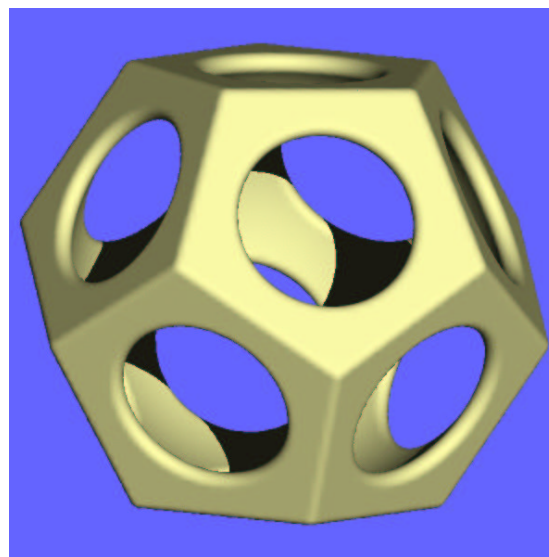
(d)

Obrázok 14: Artefakty na hranách pri rôznych výpočtoch CSG operácií (model: pravidelný 12-sten mínus guľa)

Objem bez gradientu: (a) jednoduchá operácia. Objem s gradientom: (b) jednoduchá operácia, (c) vylepšená operácia, (d) pokročilá operácia.



(a)



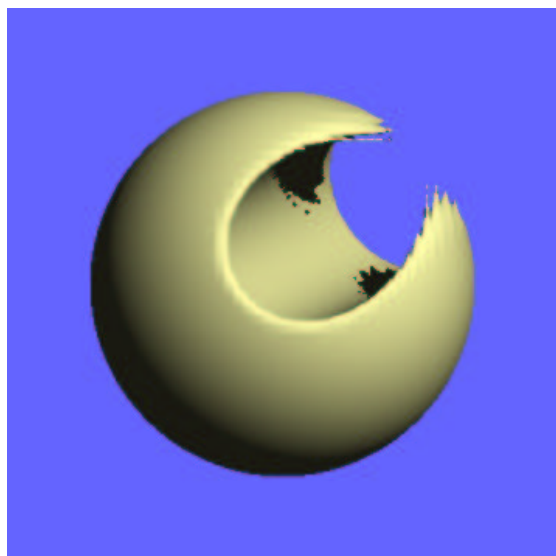
(b)

Obrázok 15: Porovnanie pokročilej CSG operácie pre rôzne typy voxelov
 (a) voxel bez gradientu, (b) voxel s gradientom. Model: pravidelný 12-sten mínus guľa.

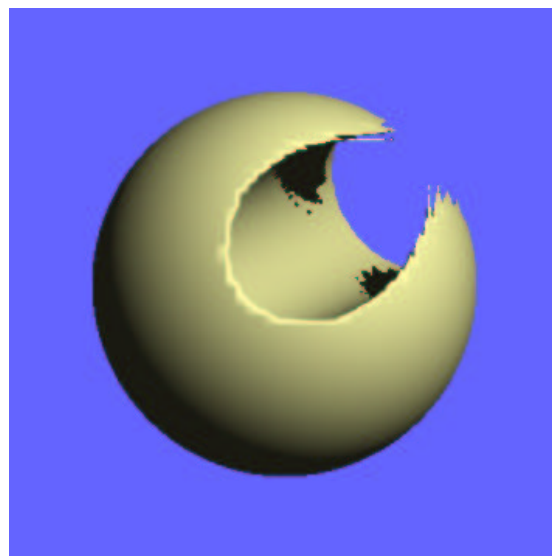
be dochádza. Vizuálne sa táto chyba začne prejavovať, ak sa zakrivené povrchy stretajú pod veľmi malým uhlom. Vtedy je totiž výpočet priesečníku kriviek (v priestore povrchov) značne nepresný. Špeciálna CSG operácia sa tento problém snaží riešiť nasledovne:

1. Počas vykonávania pokročilej CSG operácie si označujeme voxely, ktoré vznikli kombináciou vstupných voxelov nesúcich „takmer“ opačné gradienty. Experimentálne sme ich stanovili ako tie, ktorých uhol sa od priameho líši menej ako o 7° . Tieto voxely sú nazvané *nebezpečné*.
2. Po uskutočnení CSG operácie kontrolujeme nebezpečné voxely. Ich hodnoty porovnávame so susednými (iba s bezpečnými) voxelmi a ak sa od nich výrazne líšia, upravíme ich tak, aby boli v súlade s „bezpečným okolím“. Podrobnosti sú dosť zložité vzhľadom na rôzne možné situácie, preto ich nebudeme presne popisovať.

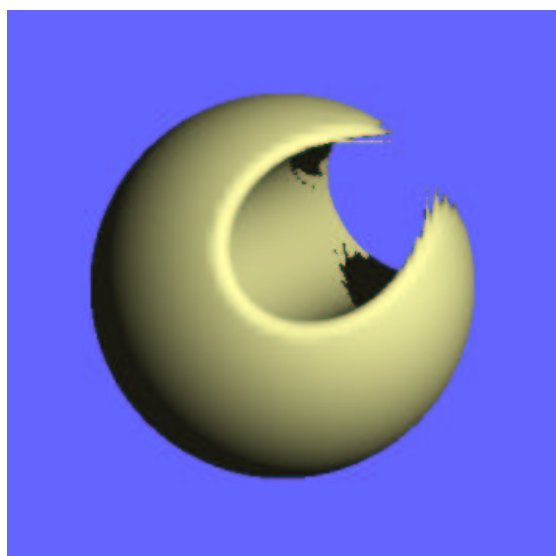
Týmto sa nám podarilo dosiahnuť isté vizuálne zlepšenie, aj keď je zrejmé, že nedostaneme úplne korektný výsledok (pozri obrázok 17). Narážame tu totiž na principiálny problém, ktorý je ilustrovaný obrázkom 18. Máme dva objekty *A*, *B*. Pokým sú od seba dosť ďaleko, ich povrchy sú oddelené. Keď ich navzájom priblížime tak, že sa budú výrazne pretínať, ich povrchy sa spoja. Je zrejmé, že medzi týmito dvoma polohami nemôže existovať spojitý prechod. Preto bude pri ľubovoľnom rozlíšení existovať kritická vzdialenosť, kedy bude zjednotenie týchto dvoch objektov nereprezentovateľné. Pritom tento problém nastane pri približovaní ľubovoľných dvoch povrchov. Našťastie kritická je len vzdialenosť objektov vo veľmi úzkom pásme hodnôt, mimo tohto pásma sa CSG operácia realizuje korektne.



(a)



(b)



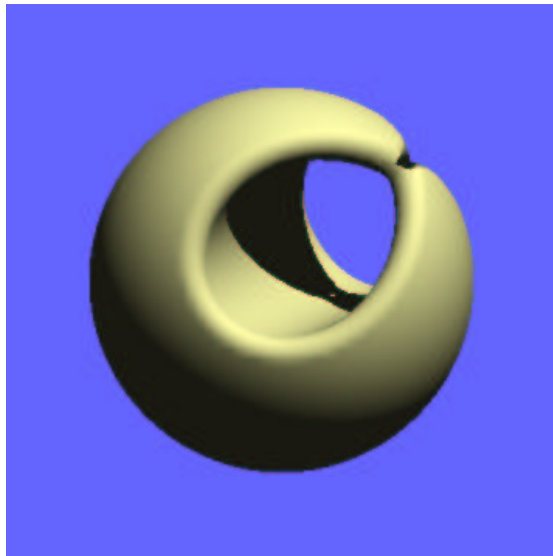
(c)



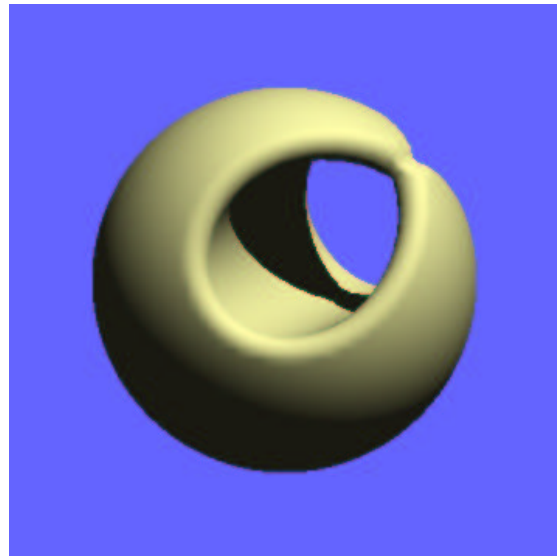
(d)

Obrázok 16: Artefakty na hranách pri rôznych výpočtoch CSG operácií (model: guľa mínus valec)

Objem bez gradientu: (a) jednoduchá operácia. Objem s gradientom: (b) jednoduchá operácia, (c) vylepšená operácia, (d) pokročilá operácia.

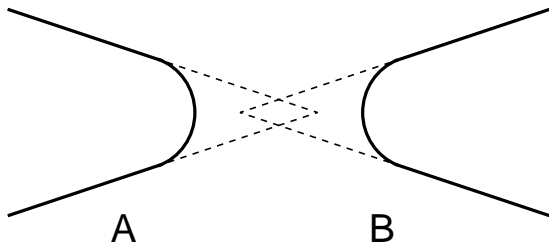


(a)

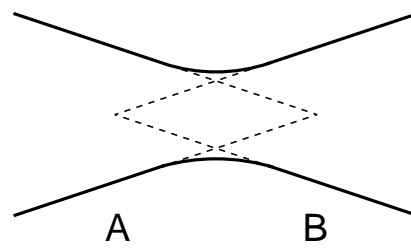


(b)

Obrázok 17: *Problémy pokročilej CSG operácie (model: guľa mínus valec)*
 Chyby vznikajú pri povrchoch, ktoré sa takmer dotýkajú. Typy CSG operácií: (a) pokročilá,
 (b) špeciálna.



(a)



(b)

Obrázok 18: *Problém približujúcich sa objektov*
 Snažíme sa reprezentovať zjednotenie dvoch objektov A, B. Medzi polohami (a), (b) neexistuje spojitý prechod, preto v kritickej vzdialenosti vznikajú artefakty.

6 Záver

V predloženej práci sme navrhli a implementovali novú metódu realizácie CSG operácií medzi voxelizovanými modelmi na úrovni voxelov bez potreby rekonštruovať ich analytický popis. Táto metóda je v súlade s kritériom reprezentateľnosti voxelizovaných objektov, vďaka čomu odstraňuje nežiaduce artefakty na rozdiel od jednoduchších techník, ktoré boli používané doposiaľ.

Referencie

- [1] J. Andreas Bærentzen, M. Šrámek, and Niels Jorgen Christensen. A morphological approach to voxelization of solids. In *The 8-th International Conference in Central Europe on Computer Graphics, Visualization and Digital Interactive Media 2000*, pages 44–51, Pilsen, Czech republic, 2000.
- [2] M. Šrámek and A. Kaufman. `vxt`: a c++ class library for object voxelization. In Min Chen, Arie E. Kaufman, and Roni Yagel, editors, *Volume Graphics*, pages 119–134. Springer Verlag, London, 2000.
- [3] S. F. Frisken, Ronald N. Perry, Alyn P. Rockwood, and Thouis R. Jones. Adaptively sampled distance fields: A general representation of shape for computer graphics. In Kurt Akeley, editor, *Siggraph 2000, Computer Graphics Proceedings*, Annual Conference Series, pages 249–254. ACM Press / ACM SIGGRAPH / Addison Wesley Longman, 2000.
- [4] J. Andreas Baerentzen and N. J. Chistensen. A technique for volumetric CSG based on morphology. In Klaus Mueller, editor, *Volume Graphics'01*, pages 71–79, Stony Brook, NY, June 2002.
- [5] A. Kaufman, D. Cohen, and R. Yagel. Volume graphics. *IEEE Computer*, 26(7):51–64, July 1993.
- [6] S.F.F. Gibson. Using distance maps for accurate surface reconstruction in sampled volumes. In *IEEE Symposium on Volume Visualization*, pages 23–30, 1998.
- [7] Miloš Šrámek, Leonid I. Dimitrov, and J. Andreas Baerentzen. Correction of voxelization artifacts by revoxelization. In Klaus Mueller and Arie Kaufman, editors, *Volume Graphics'01, Proceedings of the Joint IEEE TVCG and Eurographics Workshop*, pages 265–275, Stony Brook, NY, June 21–22, 2001.