

VIZUALIZÁCIA KOSTÍ VO VIRTUÁLNO-REALITNOM SYSTÉME

(BONES VISUALIZATION IN VIRTUAL-REALITY SYSTEM)

Branislav SOBOTA

Katedra počítačov a informatiky, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach,
Letná 9, 042 00 Košice, Slovenská republika, tel. 055/602 2563, E-mail: branislav.sobota@tuke.sk

SUMMARY

This paper described the morphological simulation of bones for visual frame in the virtual reality system. The paper is divided to three parts. In the first part are described introduction to virtual reality, modelling of bones and fractal geometry. This part also described a method choice for realisation of these problems. As initial tool for solving of this problem, was used L-system. Second, main part contained algorithms for generating of bone body and for bone surface. Algorithm for generating of bone body use fractal geometry functions based on L-systems. Algorithm for generating of bone surface use shrinkwrap method. This process is described as iterative algorithm. The final points set define triangular net of bone surface. By both algorithms is added a possibility of using random parameters values. This feature increased abilities of this process. In the last part are described advantages and disadvantages of this solving. Also are here suggestions for using these bones in virtual models and worlds and examples of outputs from program implementation.

Keywords: virtual reality, modelling, bones, skeleton model, L-systems, fractals

1. ÚVOD

Systémy virtuálnej reality sa počas svojho vývoja zaoberali aj vizualizáciou stále zložitejších scén [9], ktoré sa čoraz viac približovali realite. Preto sa stalo nevyhnutným zobrazovanie aj prírodných, humanoidných a animálnych objektov. Keďže tieto objekty často nie sú jednoduché, ani ich modelovanie a simulácia nie je jednoduchá. V minulosti sa pre tieto účely využívala aproximácia pomocou jednoduchých geometrických telies. Väčšina modelov používaných k simulácii ľudského či zvieracieho tela využíva ako základ skeletálny model [2][7]. V tomto prípade kosti objektu sú najčastejšie lineárne aproximované a nie je kladený dôraz na vizuálnu, materiálovú či bunkovú štruktúru kosti. Je to aj z dôvodu následnej jednoduchšej aplikácii transformačných a deformačných funkcií z pohľadu konečného spracovania. Už na tejto úrovni je možné zapracovanie niektorých fyzikálnych princípov. Ak je potrebná vizualizácia kostí poväčšine je používaný povrchový model [4][7][13]. V prípade simulácie pre potreby medicíny resp. genetiky je nutné okrem predchádzajúcich možností zahrnúť do modelu aj materiálovú a bunkovú štruktúru [3][5][6]. Simulácie častokrát používajú na tejto úrovni metódu konečných prvkov. V tomto prípade je problematické tieto algoritmy využívať v reálnom čase.

Na virtuálno-realitné systémy sa kladú stále väčšie požiadavky na dosiahnutie realistickej podoby a to aj z pohľadu implementácie humanoidných a animálnych objektov a to najmä v reálnom čase. Preto je potrebné hľadať novšie a dokonalejšie postupy ich tvorby. Jedným z týchto postupov je využitie fraktálov a fraktálnej geometrie[9][14] a v rámci nej využitie L-systémov [1][15].

V dnešnej dobe predstavujú L-systémy veľmi vhodnú metódu na tvorbu fraktálnych objektov. Nakoľko sa v prírodných objektoch často uplatňujú fraktály a prvky fraktálnej geometrie, stávajú sa L-systémy vhodným prostriedkom pre modelovanie a simuláciu skôr spomínaných objektov. L-systémy pre svoju grafickú reprezentáciu využívajú tzv. „korytnačiu grafiku“, kde kresliace pero predstavuje vlastne „virtuálna korytnačka“, ktorá sa pohybuje v príslušnom smere. Najväčšie možnosti použitia ponúkajú otvorené L-systémy. Pomocou vhodnej množiny pravidiel je možné modelovať širokú škálu objektov. Otvorenosť systému umožňuje systému interaktívne reagovať na okolie, a naopak, okolie môže byť informované o stave a reakciách objektu. Parametrizácia zabezpečuje, že rovnaký L-systém môže vytvárať podobné, ale rôzne objekty. V prírode hrá veľkú úlohu náhodnosť, ktorá zabezpečuje rôznorodosť inak rovnakých objektov a javov. Pri modelovaní objektov pomocou L-systémov je možné náhodnosť do modelu zahrnúť nedeterministickým výberom pravidiel z množiny použiteľných (aj genetických pravidiel). V tomto prípade sa uplatňuje nedeterminizmus L-systému.

V rámci vývoja virtuálno-realitného systému PROLAND [8][9], ktorý je vyvíjaný na KPI FEI TU Košice sa veľká miera pozornosti venuje aj modelovaniu rôznych prírodných útvarov a javov [11][12]. V procese tvorby prírodných objektov sa väčšinou uplatňujú princípy fraktálnej geometrie a časticových systémov, len pri určitých procesoch a častiach objektov, zvyšné procesy a časti sú vytvárané inými metódami a prvkami. Preto je vhodné L-systémy uplatňovať len pri časti simulovania objektu. Pri zvyšnom procese tvorby je nutné využiť iné postupy modelovania. Z hľadiska modelovania kostí, popisovaných v tomto článku sa sledujú najmä dve línie: modelovanie kostry objektu kosti a modelovanie povrchu kosti.

L-systémy sa výhodne uplatňujú pri generovaní kostry objektu. Táto kostra určuje základné tvary a základné rozmery objektu. Pre iné aspekty objektu, ako napríklad povrch, fyzikálne vlastnosti a podobne, je nutné využiť iné postupy.

Dôležitým prvkom objektu je jeho povrch. Jeho podoba výrazne ovplyvňuje výslednú podobu objektu. Existuje viac postupov, ako vytvoriť povrch okolo objektu, ktorý je definovaný svojou kosterou. Jedným z nich je algoritmus shrinkwrap [13].

2. TVORBA KOSTRY OBJEKTU

Kostra objektu vytvára základ, od ktorého závisia parametre objektu, akými sú jeho morfológická štruktúra, rozmery a proporcie. Jej vytváranie je základnou časťou v procese tvorby objektu. V tejto fáze sa využívajú prostriedky fraktálnej geometrie a fraktály.

3. VÝBER METÓDY GENEROVANIA KOSTRY

Na základe uvedenom skôr sa ako metóda pre tvorbu kostí zvolila L-systémy. L-systémy sú osvedčenou metódou pri generovaní rastlín [12][15]. Keďže postupy pri tvorbe rastlín je možné využiť aj pri tvorbe kostí, bolo možné očakávať, že L-systémy budú schopné generovať ich kostru. Výhodnou vlastnosťou L-systémov je aj ich schopnosť reagovať na rôzne hodnoty parametrov, ktoré má mať vytváraný objekt, alebo ktoré tento objekt majú ovplyvňovať. Z tohto dôvodu boli zvolené nedeterministické kontextové parametrické L-systémy. Nedeterminizmus týchto systémov umožňuje, aby sa pri tvorbe kostry uplatňovala náhoda, čo umožňuje zvýšiť realistickosť objektov ich rôznorodosťou. Využívanie parametrov umožňuje prispôsobovanie sa vlastností objektu rôznym požiadavkám a vplyvom. Kontext pravidiel umožňuje vytváranie zložitejšieho súboru pravidiel, schopného tvoriť zložitejšie štruktúry. Možná otvorenosť L-systémov použitím komunikačných modulov nebola využitá, pretože systém slúži na vytváranie objektov mimo prostredia. Teda komunikácia L-systém s prostredím, sprostredkovaná komunikačnými modulmi, nie je potrebná.

4. POPIS KOSTRY OBJEKTU

Kostra objektu je tvorená vrcholmi, prepojenými hranami. Pre výslednú podobu objektu sú dôležité vrcholy, hrany slúžia len pre vizualizáciu kostry. Počas reprezentácie predpisu, ktorý je vytvorený L-systémom, sa vytvára množina vrcholov kostry. Ak sa bezprostredne po sebe vytvoria dva vrcholy, vytvorí sa aj hrana, ktorá ich spája. Bezprostrednosťou sa myslí to, že medzi vytvorením dvoch bodov nedochádza k zmene pozície v radiacom bode, zmena smerovania (uhla) je prípustná. 2D L-systémy sú upravené, prispôbené do trojrozmerného priestoru. Stav je potom reprezentovaný usporiadanou päticou (x, y, z, α, β) ,

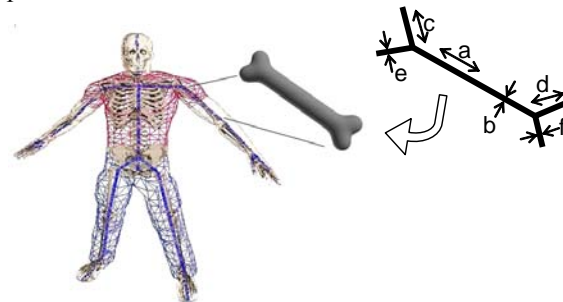
kde x, y, z reprezentujú polohu v smere danej osi a uhly α a β jednoznačne určujú smer rozvoja v trojrozmernom priestore.

4.1. Geometrická reprezentácia

Predchádzajúce časti systému vytvorili reťazec, ktorý je potrebné geometricky reprezentovať, a tým vytvoriť kostru objektu. Aktuálny stav reprezentuje polohu, smerovanie a smerovanie lokálnej osi z . Pre oba vektory smerovania existujú normálové vektory. Navyše je vytvorený zásobník, do ktorého je možné ukladať stav iterácií L-systému.

Výpočet 3D L-systému sa vykonáva v trojrozmernom priestore a jeho stav je vzťahovaný k dvom súradnicovým systémom. Prvým z nich je globálny súradnicový systém priestoru, v ktorom dochádza k základného rozvoju L-systému. Vzhľadom k tomuto súradnicovému systému sa určujú polohy a vektory smeru a lokálnej osi spolu s normálami. V každej iterácii sa využíva aj lokálna súradnicová sústava, ktorá je určená lokálnou osou z a jej normálou, ktorá predstavuje lokálnu os x . Kolmo k nim v pravotočivom zmysle existuje, ale nie je definovaná, os y . Vzhľadom k tejto súradnicovej sústave sa vykonávajú všetky rotácie. Rotácie vektorov sú uskutočňované pomocou kvaterniónov.

Na Obr. 1 je uvedená ukážka popisu kostry jednoduchej kosti s prípadnými definovateľnými parametrami.



Obr. 1 Príklad popisu kostry jednoduchej kosti
Fig. 1 Example of description of simple bone skeleton

5. TVORBA POVRCHU OBJEKTU

Po tom, čo je kostra objektu vytvorená, je možné pokračovať vo vytváraní objektu definovaním jeho povrchu. Povrch objektu je reprezentovaný množinou trojuholníkov. Každý trojuholník je definovaný troma vrcholmi, ktoré sú spojené troma hranami. Tvar povrchu je vypočítavaný pomocou upraveného algoritmu shrinkwrap, ktorý je popísaný ďalej. Takto je umožnené používateľovi definovať niekoľko parametrov, ktoré priamo ovplyvňujú výslednú podobu povrchu. Najdôležitejším z nich je hodnota funkcie $V(r)$ (pozri vzťah (6)), ktorá definuje konečnú polohu vrcholov povrchu. Jej zvyšovanie približuje povrch ku kostre, čím sa dosahuje väčšie vykreslenie detailov. Od určitej

hodnoty ale môže povrch kostru vykresľovať tak výrazne, že to vedie k strate realistikosti zobrazenia. Preto je potrebné zvoliť konečnú hodnotu $V(r)$ dôsledne.

Druhým parametrom, zásadne ovplyvňujúcim tvorbu povrchu, je ΔV (pozri vzťah (7)). Táto hodnota určuje krok, o ktorý sa hodnota $V(r)$ zväčší pri každej iterácii. Od tejto hodnoty závisí, kam sa presunie každý vrchol v nasledujúcej iterácii. Čím je táto hodnota vyššia, tým je beh celého algoritmu rýchlejší. Nakoľko ale presnosť diskrétného výpočtu, využívajúceho spojité derivácie, závisí od miery diskretizácie, zvýšenie hodnoty ΔV zvyšuje chybu výpočtu novej polohy bodu (pozri vzťah (12)), pretože gradient používaný pri výpočte je určený pomocou prvej derivácie povrchu (pozri vzťah (4)). Preto príliš veľká hodnota ΔV môže viesť k lokálnym chybám na povrchu alebo až ku skolabovaniu celého povrchu.

Poslednými hodnotami, ktoré určujú tvorbu povrchu, sú počet poludníkov a rovnobežiek inicializačnej gule. Pri generovaní inicializačnej gule sa vrcholy vytvárajú na priesečníkoch rovnobežiek s poludníkmi a na vzájomných priesečníkoch poludníkov. Preto je počet vrcholov inicializačnej gule rovný súčinu počtu poludníkov a rovnobežiek zväčšený o dva. Počet poludníkov musí byť deliteľný štyrmi, počet rovnobežiek musí byť nepárny.

Pre určenie povrchu je potrebné určiť hodnotu $V(r)$ v ľubovoľnom bode podľa vzťahu (6). V prípade, že je počet elementov kostry väčší ako jeden, je potrebné určiť váhu každého z nich pre výpočet hodnoty funkcie V v bode r . Pre výpočet váhy elementu platí vzťah:

$$\rho_i = \frac{1}{(x - R_x)^2 + (y - R_y)^2 + (z - R_z)^2} \quad (1)$$

kde x, y, z sú súradnice polohy bodu r , R_x, R_y, R_z sú súradnice polohy elementu, a pre ρ platí vzťah:

$$\rho = \sum_i \frac{1}{(x - R_{xi})^2 + (y - R_{yi})^2 + (z - R_{zi})^2} \quad (2)$$

kde i predstavuje i -ty element kostry. Pre výpočet funkcie $V(r)$ platí vzťah:

$$V(r) = \sum_i \frac{\rho_i}{\sqrt{(x - R_{xi})^2 + (y - R_{yi})^2 + (z - R_{zi})^2}} \quad (3)$$

Ako zo vzťahu (12) vyplýva, pre výpočet novej polohy bodu r je potrebné poznať gradient funkcie V v bode r . Pre gradient platí vzťah:

$$\nabla V = \left(\frac{\partial V}{\partial x}, \frac{\partial V}{\partial y}, \frac{\partial V}{\partial z} \right) \quad (4)$$

ktorý po dosadení zo vzťahu (3) nadobudne podobu:

$$\begin{aligned} \nabla V = & \sum_i \frac{-\rho_i(x - R_{xi})^2}{\sqrt{((x - R_{xi})^2 + (y - R_{yi})^2 + (z - R_{zi})^2)^3}} \bar{x} \\ & + \sum_i \frac{-\rho_i(y - R_{yi})^2}{\sqrt{((x - R_{xi})^2 + (y - R_{yi})^2 + (z - R_{zi})^2)^3}} \bar{y} \\ & + \sum_i \frac{-\rho_i(z - R_{zi})^2}{\sqrt{((x - R_{xi})^2 + (y - R_{yi})^2 + (z - R_{zi})^2)^3}} \bar{z} \end{aligned} \quad (5)$$

Ako je u algoritmu shrinkwrap uvedené, inicializačná guľa sa vytvára takmer v nekonečnu, kde je hodnota $V(r)$ takmer rovná nule. V praxi pri realizácii nie je možné umiestniť povrch gule do takejto vzdialenosti. V systéme PROLAND ako polomer inicializačnej gule bola zvolená hodnota o 30% väčšia ako najväčšia vzdialenosť vrcholu kostry od stredu kostry. Stred gule je totožný so stredom kostry, ktorý je určený aritmetickým priemerom jednotlivých súradníc elementov. Pretože v prípade, že inicializačný povrch gule nie je vytvorený takmer v nekonečne, hodnoty funkcie $V(r)$ sa pre vrcholy povrchu môžu podstatne líšiť. Preto sa určí stredná hodnota $V(r)$ a všetky vrcholy sa presunú do polôh, ktoré zodpovedajú danej strednej hodnote $V(r)$. Preto už inicializačný povrch nadobúda približný tvar objektu namiesto tvaru gule.

Ako už bolo spomenuté, priebeh vytvárania konečného povrchu prebieha v iteráciách, v ktorých sa vrcholy povrchu presúvajú na miesta odpovedajúce novej hodnote $V(r)$, určenej podľa vzťahu (7). Po presunutí všetkých vrcholov je potrebné určiť, či sa hrany spájajúce vrcholy neodchýľujú od povrchu. Preto sa určí hodnota V aj pre stred každej hrany. V prípade, že sa hodnota V stredu niektorej hrany líši od aktuálnej hodnoty V_0 o viac ako 5%, tento stred sa stane novým vrcholom povrchu. Hrana, ktorú rozpoľoval, sa rozdelí na dve. Po tom, čo sa všetky hrany, ktoré sa odchyľovali od povrchu, rozdelili, trojuholníky, ktorým niektorá z rozdelených hrán tvorila jednu zo strán, sa rozdelia, aby sa prispôbili novej topológii vrcholov a hrán.

6. ALGORITMUS SHRINKWRAP

Pre vytvorenie povrchu objektu okolo jeho kostry je možné použiť algoritmus shrinkwrap [13]. Pre jeho implementáciu je nutné zaviesť termín izo-povrch. Izo-povrch je množina bodov $r \in R^3$, pre ktoré funkcia $V(r) = V_0$ a kde

$$V(r) = \sum_i \frac{\rho_i}{|r - R_i|} \quad (6)$$

Sumácia cez i ukazuje, že funkcia V je súčtom niekoľkých komponentov s relatívnou váhou ρ_i . Tieto komponenty (elementy) tvoria kostru (skelet) objektu, okolo ktorého sa izo-povrch určuje. Môžu to byť body, čiary alebo konvexné polygóny.

Ak je komponent tvorený bodom, potom R_i je jeho pozícia a $(\rho_i/|r - R_i|) - V_0$ určuje povrch gule okolo R_i s polomerom ρ_i/V_0 .

Ak je komponentom úsečka AB , potom R_i závisí od bodu r . R_i je projekciou bodu r na AB a $(\rho_i/|r - R_i|) - V$ určuje valec s pologuľovými ukončeniami s polomerom ρ_i/V_0 , kde úsečka AB leží v osi valca.

Podobne, ak element tvorí konvexný polygón, R_i je projekciou bodu r do tohto polygónu. Izo-povrch potom tvoria body, ktoré sú od tohto povrchu vzdialené o ρ_i/V_0 s valcovitými okrajmi s týmto polomerom.

Izo-povrch okolo celého objektu je potom jemne zaoblené zjednotenie izo-povrchov jednotlivých elementov.

Množina všetkých bodov, ktoré tvoria izo-povrch, je nahradená sieťou bodov, ktoré definujú sieť trojuholníkov. Dôležité je, aby hustota týchto bodov bola dostatočná pre zachytenie detailov vytváraného objektu. Pri vytváraní tejto siete sa uplatňuje takzvaná adaptívna triangulácia [13]. V prípade, že sa spojnice dvoch bodov siete príliš vzdialia od izo-povrchu, táto spojnica sa rozdelí novovzniknutým vrcholom, ktorý sa presunie do správnej polohy na izo-povrchu. Vznikom nového vrcholu sa zvýši počet hrán a trojuholníkov. Táto adaptívna triangulácia zabezpečuje, aby sa tvar trojuholníkov prispôboval zakriveniu povrchu. V smeroch, v ktorých je zakrivenie povrchu malé, sa trojuholníky predlžujú, a opačne, v smeroch, v ktorých sa povrch zakrivuje viac, je aj hustota bodov vyššia, a tým veľkosť strán trojuholníkov menšia. To umožňuje, aby sa na zachytení detailov povrchu využíval primeraný počet vrcholov.

Výpočet novej polohy bodu resp. vrcholu je pomerne jednoduchý. Pre každý iteračný krok výpočtu sa hodnota V_0 zväčší o ΔV . Čím je táto hodnota menšia, tým presnosť aproximácie povrchu sa zvyšuje, ale zvyšuje sa aj počet potrebných matematických operácií. Hodnota ΔV ale nesmie byť veľká natoľko, aby sa zmenila topologická štruktúra trojuholníkovej siete. Pre určenie novej polohy bodu, keď sa V_0 zväčší o hodnotu ΔV , je možné použiť Taylorov rozvoj prvého rádu.

Predpokladajme, že bod r je na povrchu, kde $V(r) = V_0$. Pre novú polohu bodu platí:

$$V(r + \delta) = V_0 + \Delta V \quad (7)$$

Taylorov rozvoj vyzerá takto:

$$V(r + \delta) = V(r) + (\delta \cdot \nabla V(r)) + O(|\delta|^2) = V_0 + \Delta V \quad (8)$$

alebo

$$\Delta V \approx (\delta \cdot \nabla V(r)) \quad (9)$$

kde $\nabla V(r)$ je gradient povrchu v bode r .

To ale nehovorí o tom, v ktorom smere zmena o δ nastane. Preto je dobré zvoliť:

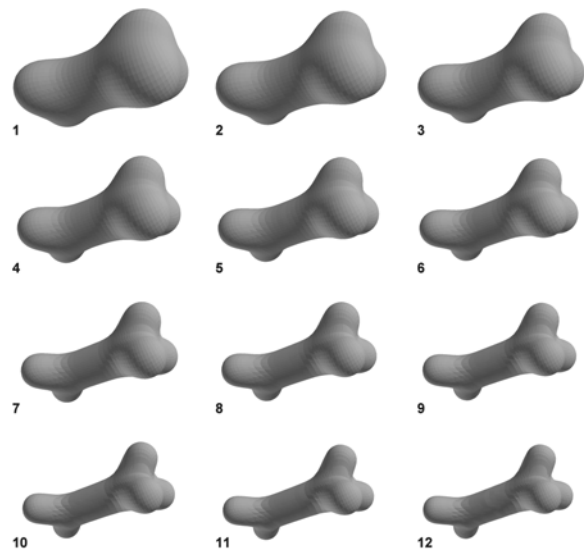
$$\delta = \lambda \cdot \nabla V(r) \quad (10)$$

a potom

$$\lambda = \frac{\Delta V}{(\nabla V(r) \cdot \nabla V(r))} \quad (11)$$

Pre novú polohu bodu r teda platí:

$$r + \frac{\Delta V \nabla V(r)}{(\nabla V(r) \cdot \nabla V(r))} \quad (12)$$



Obr. 2 Ukážka postupného modelovania kosti (KPI FEI TU Košice)

Fig. 2 Example of bone modelling (DCI FEI TU Košice)



Obr. 3 Ukážka kosti s prvkom náhodnosti (KPI FEI TU Košice)

Fig. 3 Example of bone modelling with random element (DCI FEI TU Košice)

7. ZÁVER

Systém vďaka svojej parametrizácii umožňuje vytvárať objekty, ktorých vlastnosti môžu byť ovplyvnené rozličnými vonkajšími skutočnosťami. Z hľadiska gramatiky L-systému je optimalizovaný pre kosti. To umožňuje vytvárať rôzne typy kostí s popisom kostry modelovanej kosti a aj jej morfológického vzhľadu. Pomocou vytvoreného programového vybavenia je možné experimentovať s rôznymi typmi, prakticky bez obmedzenia a získané výsledky ukladať a prípadne použiť v nadväzných systémoch. Vychádzajúc z [16] môže byť tento systém vhodný na experimentálne účely v procese výučby biológie a biofyziky. Potom pomocou systému je možné modelovať rôzne druhy kostí aj s prípadnými externými vplyvmi a používateľ získa vizualizáciu výsledku. Na Obr. 2 je uvedený príklad postupného generovania kosti v dvanástich iteračných krokoch. Na Obr. 3 je ukážka kosti, pri ktorej boli použité rovnaké generačné pravidlá ako pri ukážke z Obr. 2, ale s už určitým prvkom náhodnosti.

Napriek svojim zjednodušeniam, je tento model vhodný na vizualizáciu údajov z priamej syntézy skeletálnych modelov. Problém popisovaný v tomto príspevku bol riešený ako časť projektu riešenia paralelných architektúr špecializovaných vysokovýkonných počítačových systémov za účelom simulácie a aplikácie skeletálneho modelu človeka v prostrediach virtuálnej reality.

LITERATÚRA

- [1] Lindenmayer, A.: Mathematical models for cellular interaction in development. *Journal of Theoretical Biology*, vol. 18, pp.280-315, 1968
- [2] Hohne, K.H. – Bomans, M. – Riemer, M. – Schubert, R. – Tiede, U. – Lierse, W.: A 3D anatomical atlas based on a volume model. *IEEE Visualization 1992*, 12 (1992) 72-78
- [3] Morris, D. – Sewell, Ch. – Blevins, N. – Barbagli, F. – Salisbury, K.: A Collaborative Virtual Environment for the Simulation of Temporal Bone Surgery Stanford University, Department of Computer Science and Department of Otolaryngology, Stanford USA, 2002
- [4] Bryan, J. – Stredney, D. – Wiet, G. – Sessanna, D.: Virtual Temporal Bone Dissection: A Case Study. *Proc. of IEEE Visualization 2001*, Ertl et. Al., (Eds): 497-500, October 2001
- [5] Taylor, M. – Cotton, J. – Zioupos, P.: Finite Element Simulation of the Fatigue Behaviour of Cancellous Bone; Kluwer Academic Publishers 2002, *Meccanica* Vol. 37, pp. 419–429, 2002
- [6] Mota, A. – Klug, W. S. – Ortiz, M. – Pandolfi, A.: Finite-element simulation of firearm injury to the human cranium *Computational; Springer-Verlag, Mechanics* 31 (2003), pp. 115–121, 2003
- [7] Serge Van Sint, J.: The VAKHUM Project: Presentation and Preliminary Results; ISB2001 Congress, Zurich, July, 2001
- [8] Sobota, B. – Janošo, R. – Jánsky, M. – Valigurský, M.: PROLAND – the basic conception of creating of virtual reality system; *Proceedings of FEI International Conference - Electronic Computers & Informatics ECI'99, Košice-Herľany, 14.-15. október. 1999*, ss. 224-229
- [9] Sobota, B.: Príspevok k riešeniu vizualizačného rámca systémov virtuálnej reality; *Dizertačná práca, KPI FEI TU Košice, Košice dec. 1998 máj 1999*, ps. 117
- [10] Sobota, B.: Počítačová grafika – Fraktály, ELFA Košice, Košice október 2000, ps.64
- [11] Sobota, B. – Janošo, R. – Krahulík, P.: Simulácia niektorých prírodných javov v systéme virtuálnej reality; *Proceedings of the Scientific Conference with International Participation "Modelling and Simulation in Management and Control"*, Súľov, 6.-8. okt.1997, ss.84-88
- [12] Sobota, B. – Janošo, R. – Rozický, R.: Simulácia prírodných útvarov v systéme virtuálnej reality; *Proceedings of the Scientific Conference with International Participation "Modelling and Simulation in Management and Control"*, Súľov, 6.-8. okt.1997, ss.98-102
- [13] Brian Wyvill, Kees van Overveld: Tiling Techniques for Implicit Skeletal Models. *Implicit Surfaces for Geometric Modeling and Computer Graphics, SIGGRAPH 1996*
- [14] Harold Brochmann: *Fractal Geometry*, <http://www.saltspring.com/brochmann/math/Fractals/fractal-0.00.html>, 2. 9. 1999
- [15] Prusinkiewicz P. – Lindenmayer A.: *The Algorithmic Beauty of Plants*. Springer-Verlag, New York, 1990
- [16] Bernátová, R. – Bernát, M.: Technika vizualizácie systému logickej štruktúry biologického a biofyzikálneho učiva na báze jeho kybernetickej podstaty. *Technológia vzdelávania, V-XII, 03/2004*, s. 12-14, ISSN 1335-003X

BIOGRAPHY

Branislav Sobota was born on 22.05.1967. In 1990 he graduated (MSc.) with honours at the Department of Computers and Informatics of the Faculty of Electrical Engineering and Informatics at Technical University in Košice. He defended his PhD. in the field of virtual reality and computer graphics in 1999; his thesis title was "A contribution to solution of visualisation frame of virtual reality system". Since 1991 he is working as a lecturer at the Department of Computers and Informatics. His scientific research is focusing on computer graphics and virtual reality. In addition to this, he also investigates the questions related to the parallel computing and modelling and simulation.