

POGON ZA VIZUELIZACIJU ANIMIRANIH 3D MODELA**3D ANIMATED MODEL VIEWING ENGINE**Jovan Ivošević, *Fakultet tehničkih nauka, Novi Sad***Oblast – ELEKTROTEHNIKA I RAČUNARSTVO**

Kratak sadržaj – Rad se bavi analizom i implementacijom problema vezanih za prikazivanje animiranih 3D modela. Predstavljena je struktura modela datoteke i način učitavanja i čuvanja podataka prilagođen za efikasnu upotrebu. Analizirani su metodi kompresije, animacije i prikazivanja takvih modela. Za dalji rad razmatrane su mogućnosti implementiranja podrške virtualne realnosti i potencijalni problemi koje takva implementacija donosi.

Ključne reči: Animacija, 3D, model, pogon, virtuelna, realnost, inverzna, kinematika, kompresija, gltf

Abstract – The paper attempts to provide an analysis of and solutions to problems related to rendering 3D animated models. An analysis of the current leading model structure is provided, alongside with solutions to loading and storing the data in an efficient manner. Methods of compression, animation and rendering animated 3D models are considered. Virtual reality support is considered, alongside potential problems such support can lead to.

Keywords: Animation, 3D, model, engine, virtual, reality, inverse, kinematics, compression, gltf

1. UVOD

Sa visokom zastupljenošću tehnike u svakom aspektu života, sasvim je prirodna pojava potrebe za prikazom trodimenzionalne grafike.

U kompjuterskim video igrama i filmskoj industriji 3D grafika služi kao alat za vizuelno, ne nužno realno, predstavljanje sveta autora prikazivanog dela. U simulatorima i u medicini služi kao alat za što verniji prikaz posmatranog domena te industrije. Za takve alate najbitniji su brzina odziva i vernost prikaza.

Predstava objekata u digitalnom svetu najčešće se realizuje pomoću modela. Učitavanje 3D modela, parsiranje podataka u internu reprezentaciju zgodnu za upotrebu na platformi, proračuni vezani za prikazivanje tih modela i prikazivanje modela su glavni problemi koji će se razmatrati u ovom radu.

2. OPIS GLTF FORMATA

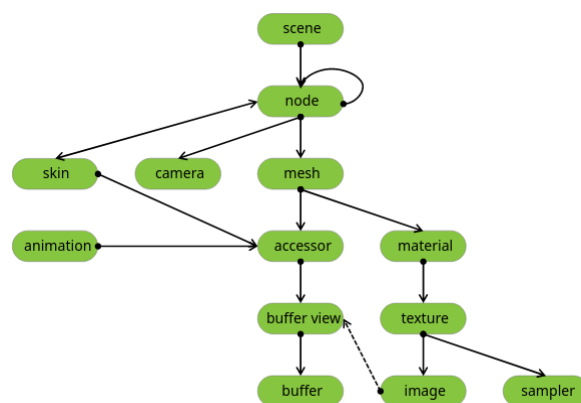
Olakšani prenos i opisivanje 3D modela su prednosti glTF formata [1]. Kako se izveze iz softvera za 3D modeli-

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Dragan Ivetić, red. prof.

ranje, u istom formatu se može i uvesti u aplikaciju. Datoteka može biti predstavljena u binarnom “.glb” ili tekstualnom “.gltf” obliku. Tekstualni oblik pogotovo je zgodan za prenos između više različitih API-eva jer koristi JSON predstavu podataka. Dodatna prednost JSON predstave jeste u njenoj razumljivosti za ljude.

Slika 1 prikazuje shemu glTF formata. Može se primetiti da datoteka ne čuva samo informacije o modelu, već postoji i podrška za čuvanjem informacija koje opisuju scenu u kojoj se model nalazi, kao i “kameru” iz čije perspektive je namenjeno posmatranje te scene. Za potrebe ovog rada će se posmatrati samo oni delovi formata koji direktno opisuju model, a ne i samu scenu ili kameru.



Slika 1. Shema glTF formata

Datoteka je logički organizovana kao niz stabala. Veze između čvorova svakog stabla fizički su realizovani preko indeksa čvorova (eng. nodes) u nizovima podataka. U JSON predstavi, na primer, svi čvorovi svih scena nalaze se u nizu “nodes” u korenu glavnog objekta modela. Struktura stabla dobija se tako što svaki čvor ima niz svojih potomaka, ukoliko taj čvor ima potomke. Niz potomaka ne sastoji se od samih objekata čvorova, već od celobrojnih nenegativnih vrednosti koje predstavljaju redni broj čvora u nizu čvorova.

3. KOMPRESIJA

Kao glavni vid kompresije i dekompresije glTF modela koristi se Draco biblioteka razvijena od strane Guglovog “Chrome Media” tima. Draco je softver otvorenog koda koji služi za komprimovanja 3D geometrijskih mreža i oblaka tačaka. U trenutku pisanja ovog rada, izovni kod biblioteke služi se sledećim metodama kompresije:

- Edgebreaker algroritam [2, 3],
- aritmetičko kodiranje i
- rANS.

3.1 ANS

Asymmetric Numeral Systems je metod kodiranja razvijen od strane Jaroslava Duda 2009. Ideja je relativno prosta: poruka se posmatra kao niz simbola i kodira se u jedinstven prirodan broj x . Ukoliko je x mali broj biće potrebno manje bitova za njegovo kodiranje i obrnuto ukoliko je x veći broj. Drugi način posmatranja jeste da se statističke osobine same poruke mogu upotrebiti tako da se učestalije poruke kodiraju u manji broj bitova, a manje učestale poruke u veći broj i tako dostigne efikasna kompresija. Ovaj metod je aproksimacija aritmetičkog kodiranja i nudi sličnu efikasnost kompresije, s tim da ANS ima prednost u odnosu na aritmetičko kodiranje u tome što se informacija smešta u jedan prirodan broj, umesto u dva broja koja opisuju opseg.

Motivšući primer za ANS može biti primer kodiranja sekvence binarnog broja $b_1b_2\dots b_i$ u kom je b_1 najznačajnija cifra. Inkrementalno kodiranje, to jest pretvaranje binarnog broja u prirodni broj dekadnog sistema moguće je izvršiti kodirajućom funkcijom C definisanom formulom 1:

$$C(x_i, b_{i+1}) := 2x_i + b_{i+1} = x_{i+1} \quad (1)$$

Funkcija C kao rezultat daje novu kodiranu vrednost x_{i+1} . Analogno tome inverzna funkcija za inkrementalno dekodiranje broja nazad u binarnu reprezentaciju definisana je formulom 2:

$$D(x_{i+1}) := (\lfloor \frac{x_{i+1}}{2} \rfloor, x_{i+1} \bmod 2) = (x_i, b_{i+1}) \quad (2)$$

Funkcija D prima binarnu predstavu kao ulaz i vraća prethodnu binarnu predstavu i poslednji dodati bit. Kako bi se izbegli problemi pri kodiranju niza nula, npr. "000" potrebno je x_0 dodeliti početnu vrednost "1". Ukoliko je učestalost jedinica i nula uniformna, to jest $P(0) = P(1) = 0.5$, primenom formule 1 dobija se da je entropija jednog bita jednaka jedinici, to jest da entropija kodirane poruke raste za jedan pri kodiranju svakog bita.

Pošto optimalna kodirajuća funkcija uzima prethodno kodiranu vrednost x_i i novi simbol b_{i+1} i kodira te vrednosti sa minimalnom entropijom, važi formula 3:

$$\begin{aligned} H(x_{i+1}) &= H(C_{opt}(x_i, b_{i+1})) \\ &= H(x_i) + H(b_{i+1}) \\ &= \log_2(x_i) - \log_2(p_{b_{i+1}}) \\ &= \log_2\left(\frac{x_i}{p_{b_{i+1}}}\right) \\ \Rightarrow C_{opt}(x_i, b_{i+1}) &\approx \frac{x_i}{p_{b_{i+1}}} \end{aligned} \quad (3)$$

Dakle ukoliko kodirajuća funkcija prati ovakvu aproksimaciju, onda se garantuje optimalno kodiranje. Uzimajući neuniformnu distribuciju parnih i neparnih brojeva (nula i jedinica) nekog alfabeta, tako da je verovatnoća pojave jedinice $p_1 = p < 1 - p = p_0$, gde je p_0 verovatnoća pojave nule, poželjno je da se u prvih N brojeva pojavi $N \cdot p$ jedinica i $N \cdot (1-p)$ nula. Kodirajuća funkcija koja ispunjava ovaj uslov prikazana je u formuli 4 uz svoju dekodirajuću funkciju.

$$\begin{aligned} C(x_i, b_{i+1}) &= \begin{cases} \left\lfloor \frac{x_{i+1}}{1-p} \right\rfloor - 1 & \text{za } b_{i+1} = 0 \\ \left\lfloor \frac{x_i}{p} \right\rfloor & \text{inace} \end{cases} \\ b_{i+1} &= \lceil (x_{i+1} + 1) \cdot p \rceil - \lceil x_{i+1} \cdot p \rceil \\ x_i &= \begin{cases} x_{i+1} - \lceil x_{i+1} \cdot p \rceil & \text{za } b_{i+1} = 0 \\ \lceil x_{i+1} \cdot p \rceil & \text{inace} \end{cases} \end{aligned} \quad (4)$$

Ovako definisane kodirajuće i dekodirajuće funkcije rađaju ANS varijantu definisanu za binarni alfabet poznata kao uABS - uniform Asymmetric Binary System. Funkcionisanje ovakvog sistema kodiranja zasnovano je na principu deljenja N prirodnih brojeva na parne i neparne (odgovarajući nulama i jedinicama) proporcionalno zastupljenošću jedinica i nula koji čine poruke za kodiranje.

Proširujući ovaj koncept na alfabet koji se sastoji od više od 2 različita simbola dobija se rANS - range variant Asymmetric Numeral Systems metoda kompresije. Za alfabet od n simbola s potrebno je izvršiti kvantizaciju raspodele simbola sa 2^n , to jest podeliti skup od N prirodnih brojeva na 2^n podskupova. Ovim se garantuje aproksimacija verovatnoće pojave simbola $s - p_s$ na $f_s/2^n$, gde je f_s prirodan broj koji označava broj elemenata odgovarajućeg podskupa za simbol s . Izračunavajući funkciju raspodele $\Phi(s) = f_0 + f_1 + \dots + f_{s-1}$ dobijamo kodirajuće i dekodirajuće funkcije u formuli 5:

$$\begin{aligned} C(x_i, s_{i+1}) &= \left\lfloor \frac{x_i}{f_s} \right\rfloor \cdot 2^n + \Phi(s) \\ &\quad + (x_i \bmod f_s) \\ s_{i+1} &= \text{symbol}(x_{i+1} \bmod 2^n) \\ |\Phi(s)| &\leq x_{i+1} \bmod 2^n < \Phi(s+1) \\ x_i &= D(x_{i+1}) = f_s \cdot \left\lfloor \frac{x_{i+1}}{2^n} \right\rfloor - \Phi(s) \\ &\quad + (x_{i+1} \bmod 2^n) \end{aligned} \quad (5)$$

Kodirajuća funkcija pronalazi odgovarajući opseg vrednosti prirodnih brojeva i "smešta" simbol u taj opseg, kodirajući ga jedinstvenim prirodnim brojem.

Dekodirajuća funkcija prima jedinstven prirodni broj i obrnutim procesom pronalazi njemu odgovarajući simbol. Zanimljiva posledica kvantizacije sa 2^n jeste ta što se celokupno množenje, deljenje i sve moduo operacije mogu svesti na binarno pomeranje što čini programsku implementaciju efikasnijom.

4. SKELETALNI SISTEM MODELA

Skeletalni sistem modela je mehanizam koji omogućuje pravilno prenošenje animiranog pokreta s jedne celine modela na drugu. Sačinjen je od hijerarhijski organizovanih zglobova. Nad zglobovima je moguće primeniti prostorne transformacije. Svaki zglob utiče na temena mrežne strukture modela u nekoj meri. Jedan zglob može uticati na više temena i na jedno teme može uticati više zglobova. Najčešće je dozvoljeno da najviše četiri zglobova utiču na jedno teme. Primena prostornih transformacija nad zglobom odražava se na temena koja su pod uticajem tog zgloba i na druge zglobove u sprezi. Uticaj prostorne transformacije zgloba nad temenom proporcionalan je koeficijentu uticaja tog zgloba nad tim temenom. Za svako teme čuvaju se identifikatori zglobova koji utiču na to teme i koeficijenti uticaja tih zglobova. Koeficijenti uticaja zglobova nad temenom takođe se nazivaju i “težine” (eng. weights). Suma svih težina jednog temena mora da bude jednaka jedinici.

Zglobovi su u hijerarhiji organizovani kao stablo. Jedan zglob može imati više potomaka ili ni jednog, a najviše jednog roditelja. Uticaj prostorne transformacije nad jednim zglobom prenosi se na njegovog roditelja, ukoliko postoji. Roditelj dalje prenosi uticaj transformacije na svog roditelja i uticaj se tako prenosi kroz stablo sve do korenskog zgloba, uključujući i njega. Ovaj način prenošenja pokreta od listova stabla ka korenu naziva se inverzna kinematika.

Svaki zglob poseduje lokalnu i globalnu transformaciju i inverznu transformaciju veznosti. Pozicija, orijentacija i veličina zgloba u njegovom lokalnom prostoru definisane su u lokalnoj transformaciji, a u globalnom prostoru definisane su u globalnoj transformaciji. Svaki 3D model ima svoju početnu poziciju, orijentaciju i veličinu - početnu pozu. Dok je model u početnoj pozi, animator vezuje zglobove skeletalnog sistema za model, te se ova poza takođe naziva i “poza vezivanja” (eng. bind pose). Matrica transformacije veznosti zglobova smešta zglobove u inicijalna stanja poze vezivanja. Premeštanje temena iz prostora mrežne strukture u lokalni prostor zgloba za koji je vezan vrši se množenjem inverznom matricom transformacije veznosti. Pošto je inverzna matrica veznosti poznata u trenutku pravljenja modela, često se izvozi zajedno sa modelom kako bi se izbegli skupi proračuni invertovanja matrica u toku njene učestale upotrebe pri prikazivanju modela. Lokalni prostor zgloba je relativan u odnosu na prostor njegovog roditelja, osim za korenski zglob koji nema roditelja. Iz ovog sledi da su transformacije nad zglobovima relativne u odnosu na roditelja, a ne na globalni prostor u kom se model nalazi.

Novo stanje temena v' u odnosu na staro stanje temena v dobijeno pod uticajem transformacije T_{world} zgloba j koeficijentom uticaja w dobija se formulom 6.

$$v' = w \cdot T_{world}(j) \times v \quad (6)$$

Novo izračunato stanje temena je deformacija temena pod uticajem transformacije zgloba. Deformacija dobijena primenom proračuna transformacije zglobova inverznom

kinematikom daje relativno realistične rezultate i izbegava pojavu priklještene mrežne strukture. Realizam pokreta i deformacije mrežne strukture moguć je zahvaljujući načinu dobijanja globalne transformacije T_{world} zgloba. Množenjem globalne transformacije roditelja p lokalnom transformacijom zgloba j dobija se globalna transformacija zgloba j kao što se može videti u formuli 7.

$$T_{world}(j) = T_{world}(p) \times T_{local}(j) \quad (7)$$

Pošto je globalna transformacija korenskog zgloba jednaka njegovoj lokalnoj transformaciji koja je najčešće inicijalizovana u samom modelu i pošto se sve uzastopne transformacije daljih potomaka mogu posmatrati kao relativni “pomeraji” u odnosu na roditeljsku transformaciju - sledi da se konačna transformacija nekog zgloba potomka dobija rekurzivnim prolaskom kroz stablo počevši od korenskog zgloba. Ilustracija pojednostavljenog primera zglobova nalazi se na slici 2.



Slika 2 Pojednostavljena hijerarhija zglobova

Na slici 2 zglobovi su označeni kao j_i . Zglob j_0 je korenski zglob, a j_1 njegov zglob potomak. Isto tako j_2 je potomak od j_1 i tako dalje sve do j_4 . Da bi se dobila globalna transformacija zgloba j_4 potrebna je globalna transformacija zgloba j_3 . Isto tako, za treći zglob je potrebna globalna transformacija drugog. Jasno je da je u pitanju rekurzivni proračun počevši od korenskog zgloba. Formula 8 prikazuje postupke dobijanja globalnih transformacija svih zglobova.

$$\begin{aligned} T_{world}(j_0) &= T_{local}(j_0) \\ T_{world}(j_1) &= T_{world}(j_0) \times T_{local}(j_1) \\ T_{world}(j_2) &= T_{world}(j_1) \times T_{local}(j_2) \\ T_{world}(j_3) &= T_{world}(j_2) \times T_{local}(j_3) \\ T_{world}(j_4) &= T_{world}(j_3) \times T_{local}(j_4) \end{aligned} \quad (8)$$

5. ZAKLJUČAK

U radu je opisana struktura 3D modela, metodi kompresije i skeletalnog sistem animacije modela.

Struktura 3D modela u glTF formatu pokazala se kao prililčno jednostavna za analizu i upotrebu. U tekstualnom formatu moguće je “golim okom” ili alatima za analizu JSON formata izvući korisne informacije o prirodi modela. Razvoj softverskog rešenja za upotrebu ovog formata olakšan je zbog jasno definisane

specifikacije za širok spektar upotreba. Pokrivene su čak i upotrebe svojstvenih ekstenzija.

Metodi kompresije opisani u radu značajno utiču na performanse 3D programa, pogotovo kada je u pitanju prenos 3D modela preko mreže. Razvoj asimetričnih numeričkih sistema kao metod kodiranja značajno je doprinelo mogućnostima kompresije ovih tipova podataka što se može primetiti primenom ove metode u razvoju Draco [5], Zstandard [6], LZFSE [7] i sličnih kompresora.

U radu je opisana skeletalna animacija kao glavni metod animiranja 3D modela. Animatori inverznom kinematikom mogu vrlo lako napraviti "organski" realne animacije koje se pridržavaju ograničenja modela uz smislenu transformaciju mrežne strukture. Skeletalna animacija oslikava ovakav način animiranja i pojednostavljuje logiku potrebnu za implementaciju ovakvog sistema. Dovoljno je uzeti u obzir hijerarhiju skeletalne strukture i redosled primene transformacija. Kretanjem kroz stablo skeetalnog sistema relativno jednostavno dobijaju se konačne transformacije temena mrežne strukture modela.

6. LITERATURA

[1] Khronos group, glTF™ 2.0 Specification, <https://registry.khronos.org/glTF/specs/2.0/glTF-2.0.html>.

[2] J. Rossignac, "Edgebreaker: connectivity compression for triangle meshes," in IEEE Transactions on Visualization and Computer Graphics [3] J. Rossignac, "Edgebreaker: connectivity compression for triangle meshes," in IEEE Transactions on Visualization and Computer Graphics

[3] M. Isenburg and P. Alliez, "Compressing polygon mesh geometry with parallelogram prediction," IEEE Visualization, 2002. VIS 2002., 2002

[4] Google, Draco, <https://github.com/google/draco>

[5] Facebook, Zstandard, <http://facebook.github.io/zstd/>

[6] Apple, LZFSE, <https://github.com/lzfse/lzfse>

Kratka biografija:



Jovan Ivošević je rođen 1998. godine u Novom Sadu. Završio je Tehničku školu "Pavle Savić" u Novom Sadu, 2017. godine. Fakultet Tehničkih Nauka u Novom Sadu upisao je iste godine. Ispunio je sve obaveze i položio je sve ispite predviđene studijskim programom.