

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 6

**INTERAKTIVNO UPRAVLJANJE
SADRŽAJEM VIRTUALNE SCENE: PRIKAZ
UPRAVLJANE SCENE**

Vedran Pavlić

Zagreb, lipanj 2010.

Sadržaj

Uvod	3
1. Interaktivno upravljanje sadržajem virtualne scene	4
1.1. Primjena.....	5
1.2. Pregled dosadašnjih radova u području	8
2. Aplikacija za prikaz upravljane scene	12
2.1. Opis strukture aplikacije.....	12
2.2. Opis modela grafa scene i objekata	13
2.2.1. Razredi modela.....	13
2.2.2. Osvježavanje stanja scene	14
2.3. Opis kamera i pogleda na scenu	16
2.4. Opis komunikacije.....	18
2.5. Opis upravljanja aplikacije	22
2.5.1. Opis jezgre aplikacije	22
2.5.2. Postupak umetanja objekta	23
3. Način korištenja aplikacije s primjerima	25
3.1. Postavljanje početnih postavki i korištenje aplikacije.....	25
4. Rezultati i usporedba s postojećim sustavima	27
4.1. Usporedba s postojećim sustavima.....	27
4.2. Modeli korišteni za razvoj i testiranje aplikacije.....	32
4.3. Performanse i ograničenja sustava	33
5. Moguća proširenja	34
Zaključak	35
Literatura	36
Sažetak.....	37
Summary.....	37

Uvod

Virtualna okruženja u računalnoj grafici nalaze sve više primjena. Virtualne scene se koriste u mnogo područja počevši od računalnih igara preko simulatora za obuku pilota do sustava za liječenje psiholoških poremećaja. Interaktivno upravljanje sadržajem scene svoje primjene nalazi baš u liječenju psiholoških poremećaja kao što su fobije, PTSP i mnogi drugi.

Virtualna okruženja i scene su se u svojim počecima sastojala od statičkih objekata i okolina kroz koje se korisnik može kretati. Danas virtualna okruženja uključuju i pomične dinamičke objekte te objekte s kojima korisnik može interaktivno reagirati ili razgovarati. Najnoviji ovakav sadržaj je interaktivno upravljanje sadržajem scene kod kojeg scenom upravlja instruktor ili terapeut, a pacijent ili učenik sudjeluje u toj sceni i reagira na objekte i pobude koje instruktor stavlja u scenu preko svog sučelja.

Cilj ovog rada je prikazati način ostvarivanja interaktivnog upravljanja sadržajem scene te pokazati moguće koristi koje takvo upravljanje može donijeti. Za potrebe rada izrađena je aplikacija koja prikazuje virtualnu scenu kojom se može interaktivno upravljati.

1. Interaktivno upravljanje sadržajem virtualne scene

Interaktivno upravljanje sadržajem virtualne scene odnosi se na mogućnost promjene i upravljanja objektima u sceni za vrijeme njenog prikaza. Scene koje se prikazuju u računalnim sustavima su obično statične ili proceduralno upravljane. Interaktivno upravljanje sadržajem scene ima nekoliko prednosti pred tradicionalnim scenama, ali ima i nekih dodatnih zahtjeva te nije uvijek primjenjivo.

Jedna od najvažnijih prednosti interaktivnog upravljanja je mogućnost prilagodbe scene trenutnom stanju korisnika i mogućnost pobuđivanja određenih reakcija ili stanja kod korisnika scene. Ovo se postiže interaktivnim umetanjem objekata za koje se očekuje reakcija korisnika. Ta reakcija se obično bilježi te se daljnji prikaz scene nastavlja ovisno o toj reakciji. Druga prednost interaktivnog upravljanja je u broju različitih kombinacija na koje se može upravljati objektima. Svaka kombinacije može izazvati drukčiju reakciju kod korisnika.

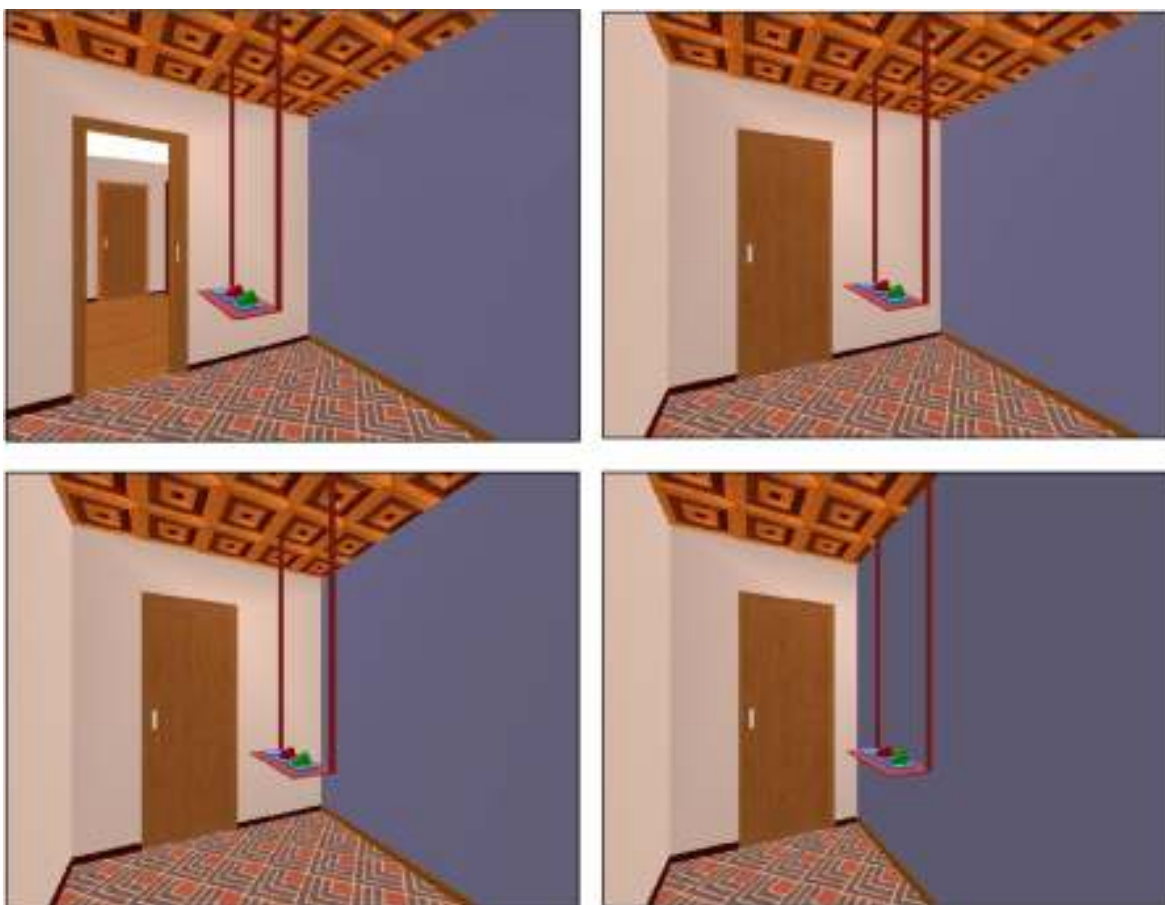
Dodatni zahtjevi interaktivnog upravljanja je potreba izvođenja programa na računalu s više monitora, odnosno više izlaza za prikaz. Pri tom bi se jedan monitor koristio za prikaz dijela sustava koji upravlja sadržajem scene dok se na drugom monitoru prikazuje sadržaj scene. Ovakav sustav također zahtijeva da sadržajem scene upravlja kvalificirana osoba.

Nabrojene prednosti i zahtjevi usmjeravaju ovakve aplikacije na primjene u specifičnim područjima kao što je terapija i liječenje psihičkih poremećaja izlaganjem te uvježbavanje i školovanje pilota i drugih zanimanja koja zahtijevaju visok stupanj kvalifikacije.

1.1. Primjena

Kao što je već spomenuto interaktivno upravljanje sadržajem virtualne scene najviše primjena nalazi u liječenju različitih psiholoških poremećaja i fobija kao što su strah od visina, pauka, javnih nastupa, klaustrofobija, PTSP i mnogi drugi. Tehnika koja se pritom primjenjuje naziva se terapija izlaganjem.

Terapija izlaganjem (engl. Exposure therapy, ET) je oblik kognitivno-bihevioralne psihoterapije (engl. Cognitive behavioral therapy, CBT) i koristi se za smanjenje reakcija straha i razine anksioznosti, posebice u liječenju fobija, a temelji na postupcima habituacije (navikavanja) i kognitivne disonancije. ET u kombinaciji sa tehnikama prividne stvarnosti (engl. Virtual reality, VR) naziva se VR terapija izlaganjem (engl. Virtual reality exposure therapy, VRET). Trenutno najviše korišteni oblik ovakve vrste liječenja za prevenciju mentalnih bolesti i poteškoća izazvanih stresom je dozirano izlaganje stresu (engl. Stress inoculation training).



Slika 1. Virtualno okruženje za liječenje klaustrofobije doziranim izlaganjem.

Cilj i svrha doziranog izlaganja stresu je povećanje otpornosti pojedinca na stres. Ona se postiže korištenjem dviju tehnika:

- Učenjem novih strategija i vještina suočavanja sa stresom
- Osnaživanjem primjenom već naučenih vještina

Da bi se omogućio napredak i uspješno usvajanje naučenih strategija i vještina dozirano izlaganje stresu provodi se u tri koraka:

- Konceptualizacija
- Usvajanje i uvježbavanje vještina
- Primjena i praćenje

Cilj faze konceptualizacije (engl. Conceptualization) je identifikacija odlučujućih faktora koje subjekt ima u suočavanju sa stresom. Faza konceptualizacije provodi se u obliku suradnje kroz dijalog eksperta (terapeuta) i ispitanika (subjekta). Potrebno je u suradnji s ispitanikom razdvojiti opće uzročnike stresa na pojedinačne stresne situacije, te ispitanik mora uvidjeti razlike između promjenjivih i nepromjenjivih aspekata stresnih situacija. Također, ispitanik mora postati svjestan kratkoročnih, srednjoročnih i dugoročnih ciljeva u svom željenom ponašanju prema uzročnicima stresa. Nakon toga terapeut mora odrediti razinu nedostatka strategija suočavanja sa stresom kod subjekta i neadekvatne primjene već naučenih strategija.

Druga faza SIT-a je faza usvajanja i uvježbavanja vještina (engl. Skills acquisition and rehearsal). U ovoj fazi terapeut mora odrediti koja strategija suočavanja sa stresom je najprikladnija te zajedno s ispitanikom istražiti kako se ona može primijeniti na njegovu situaciju. Također je potrebno istražiti uzroke koji mogu sprečavati primjenu te strategije. Složene stresne situacije se dijele na manje dijelove kojima je lakše upravljati te se uvježbavaju vještine suočavanja. Pri tom se za isprobavanje vještina ispitanika nastoji dovesti u stresnu situaciju. U ovom koraku može pomoći primjena virtualne stvarnosti i interaktivno upravljanje sadržajem scene.

Posljednja faza terapije izlaganjem je primjena i praćenje. U ovoj fazi se pacijenta potiče da koristi naučene vještine. Preporuča se izlaganje subjekta stupnjevanim uzročnicima stresa pri čemu se može koristiti interaktivno upravljanje scenom.

SIT je uspješno korišten u više kliničkih studija kao što su priprema bolesnika za invazivne i kirurške postupke, suočavanje sa akutnim i kroničnim bolom, liječenje pacijenata s PTSP-

om, napadima panike, agresivnim ponašanjem, strahom od letenja i drugim fobijama [19][20]. Također, SIT je korišten i van kliničkog okruženja u svrhu pripreme subjekata za suočavanje sa stresnim životnim i društvenim situacijama poput smrti u užoj obitelji, napetostima na poslu, gubitku posla, posljedicama velikih životnih promjena i dr. SIT je korišten i nad subjektima s profesionalnim zanimanjima koje uključuju visoke razinama stresa u svrhu smanjenja percepcije stresa i povećanja radnog učinka [19] [20]. Svi postignuti rezultati jasno ukazuju da je SIT efikasan način reduciranja anksioznosti i povećanja ukupnih performansi pod utjecajem stresa.

Primjena virtualnog okruženja i interaktivnog upravljanja sadržajem scene u terapiji izlaganjem može povećati njenu uspješnost jer se pacijenta mnogo lakše dovodi u stresnu situaciju. To je ujedno i najveća prednost korištenja virtualnih okruženja u terapiji izlaganjem. Još jedna prednost VRET-a jest lakša primjena te povećana sigurnost i mogućnost nadzora od izlaganja in vivo.

VRET računalno sučelje istodobno koriste terapeut i pacijent, ali na različite načine. Pacijent je uronjen u okolinu prividne stvarnosti pomoću kacige montirane na glavu (engl. Head Mounted Display, HMD) i senzora pokreta koji mu dopuštaju kretanje i interakciju s okolinom. Prirodni oblik interakcije omogućuje osjećaj prisutnosti te omogućuje kvalitetniji emocionalni angažman pacijenta, a to uvelike doprinosi ukupnom uspjehu terapije. Terapeut najčešće koristi grafičko korisničko sučelje za odabir i dizajniranje sadržaja virtualnih pobuda.



Slika 2. Oprema za terapiju izlaganjem uz korištenje virtualnog okruženja.

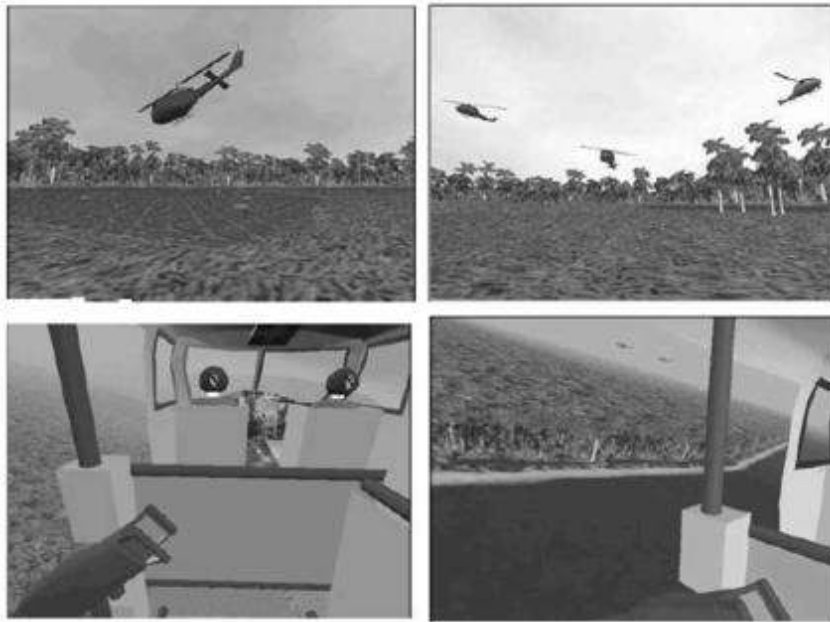
Osim primjena u terapiji moguća je i primjena u treningu i obučavanju za suočavanje sa stresnim situacijama ili obuci za zahtjevnija i stresna zanimanja kao što su piloti. Pri tom se virtualna stvarnost i interaktivno upravljanje scenom koristi za uvježbavanje prethodno naučenih i usvajanje novih vještina.

Simulator je posebno značajan za ekstremne situacije, koje je neuputno učiti na stvarnom sustavu zbog prevelike opasnosti, velikih troškova i sl. Interaktivno upravljanje provodi instruktor, sukladno mogućnostima koje dopušta simulator – npr. ubacivanje neprijateljskih zrakoplova ako se vježba borba u zraku.

1.2. Pregled dosadašnjih radova u području

U ovom području postoji nekoliko radova koji su služili kao smjernice pri izradi aplikacije koja dolazi uz rad. U ovom poglavlju bit će spomenuti neki od njih.

Prvi rad dolazi nam od autora Albert Rizzo, Barbara Rothbaum i Ken Graap. Ime rada je Primjena virtualne stvarnosti u liječenju PTSP-a (engl. Virtual Reality Applications for the Treatment of Combat-Related PTSD). Oni su pokušali primijeniti virtualno okruženje u sklopu liječenja ratnih veterana koji pate od PTSP-a terapijom izlaganjem. U tu svrhu izradili su sustav koji simulira situacije slične onima koje su pacijenti doživjeli te koje su uzrok njihovog sadašnjeg stanja. Sustav uključuje scene slične onima iz ratova u Vietnamu, Afganistanu i Iraku.



Slika 3. Scene iz virtualnog Vijetnama.

Pacijent za vrijeme korištenja sustava na glavi nosi naočale za virtualnu stvarnost dok za kretanje korisni mali joystick. Terapeut na svojem sučelju vidi pogled pacijenta i može aktivirati neke prethodno definirane događaje te tako utjecati na scenu i promatrati reakciju pacijenta.



Slika 4. Izgled sustava za liječenje PTSP-a.

Sustav je testiran na 16 ratnih veterana u 13 tretmana koji su se protezali kroz 5 do 7 tjedana. Rezultati su pokazali da je došlo do smanjivanja simptoma PTSP-a. Osam pacijenata nastavilo je tretman kroz 6 mjeseci. Simptomi PTSP-a kod tih pacijenata smanjeni su za 15-67%.

Drugi rad koji treba spomenuti je sustav NeuroVR koji je razvio tim znanstvenika s Guiseppe Rivom na čelu. Sustav se koristi na način da terapeut unaprijed priprema scene koje prezentira pacijentu. Implementacija sustava je dostupna na internetu, a komponente sustava su otvorenog koda. NeuroVR se sastoji od dvije komponente: komponenta za prikaz interaktivnih scena i komponenta za uređivanje scene. U izgradnji komponente za uređivanje scene kao polazna točka je korišten program Blender koji je nadograđen u te svrhe.



Slika 5. Komponenta za uređivanje scene.

Nakon izgradnje sustava provedeno je probna studija u kojoj je sudjelovalo 211 pacijenata koji pate od ekstremne prekomjerne težine. U sklopu studije usporedili su rezultate liječenja virtualnom terapijom uz kontroliranje prehrane s tradicionalnim načinima liječenja kontroliranjem prehrane i kognitivno bihevioralne terapije. Rezultati nakon 6 mjeseci pokazali su povećanje samopouzdanja, povećanje zadovoljstva sa slikom o sebi te

smanjenje prekomjernosti težine. Sustav je također korišten za poboljšanje liječenja anoreksije, bulimije i agorafobije.



Slika 6. Pacijent i terapeut za vrijeme liječenja.

U Sjedinjenim Američkim državama u gradu San Diegu nalazi se klinika koja se bavi liječenjem psihičkih poremećaja virtualnom terapijom izlaganjem. Pacijenti klinike su većinom ratni veterani koji pate od PTSP-a. Sustav koji se koristi u klinici sličan je po opremi Rizzovom sustavu. Pacijenti koji se nalaze u programu klinike postižu pozitivne rezultate te je otprilike 66% pacijenata nakon tretmana spremno za povratak na radna mjesta.

Ovi radovi i postojeći sustavi daju dobru podlogu i nameću neke zahtjeve koje ovakve aplikacije moraju zadovoljiti. Ti zahtjevi korišteni su kao polazna točka pri izradi sustava koji će biti opisan u sljedećim poglavljima ovog rada.

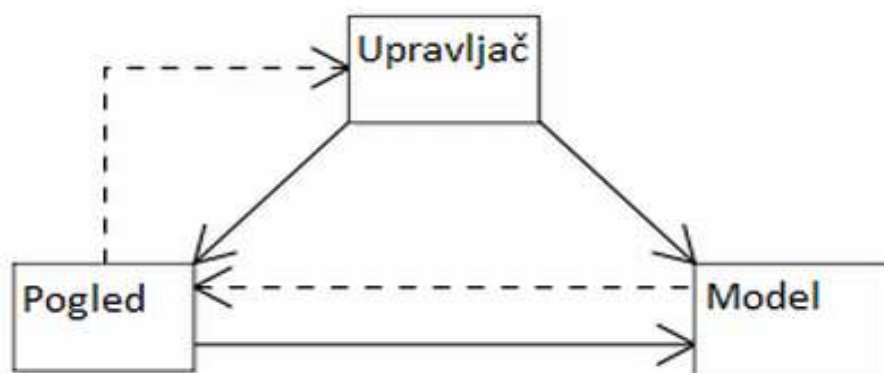
2. Aplikacija za prikaz upravljane scene

Sustav za interaktivno upravljanje i prikaz interaktivno upravljanih objekata sastoji se od dvije aplikacije: aplikacije za upravljanje sadržajem scene kojom najčešće upravlja terapeut i aplikacije za prikaz scene koju koristi pacijent. Terapeut pomoću svoje aplikacije može nadzirati i upravljati stanjem scene koju vidi pacijent te u nju umetati objekte i uklanjati objekte iz nje. Pacijent na svom zaslonu vidi trenutno stanje scene te se može kretati po sceni i doći u interakciju s objektima.

Aplikacija za prikaz scene izrađena je u programskom jeziku C++ uz korištenje OSG biblioteke. U sljedećim poglavljima bit će detaljno objašnjeni dijelovi te aplikacije.

2.1. Opis strukture aplikacije

Arhitektura aplikacije slijedi oblikovni obrazac model-pogled-upravljač(engl. Model-view-controller). Model pri tome predstavlja objekte scene i odgovoran je za njihovo osvježavanje. Upravljač je odgovoran za komunikaciju s aplikacijom za upravljanje sadržajem scene te umetanje i uklanjanje objekata iz scene. Pogled je odgovoran za prikazivanje scene na zaslonu pacijenta te za osvježavanje prikaza tlocrta scene. Aplikacija sadrži i pomoćne klase koje se koriste u svim dijelovima arhitekture.



Slika 7. Obrazac Model-pogled-upravljač.

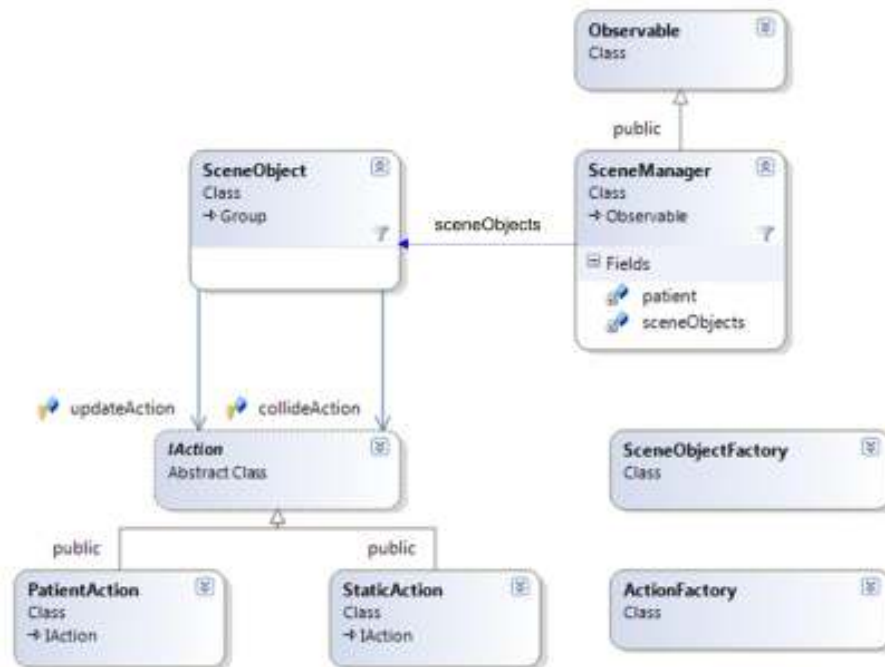
2.2. Opis modela grafa scene i objekata

2.2.1. Razredi modela

Model je dio arhitekture koji je odgovoran za objekte u sceni i za osvježavanje stanja tih objekata.

Scena se sastoji od objekata i terena. Svakom objektu pridruženo je ponašanje u slučaju sudara te ponašanje u slučaju kad sudara nema. Teren je predstavljen kao jedan od objekata i ima postavljeno statično ponašanje. Osim objekata i terena u sceni se nalazi i pacijent. Pacijent se ne iscrtava kao objekt već se na njegovu poziciju postavlja kamera iz koje se iscrtava scena.

Model se sastoji od više razreda, to su razredi `SceneObject` i `SceneObjectFactory` koji su odgovorni za objekte, razredi `ActionFactory`, `IAction`, `PatientAction` i `StaticAction` koji su odgovorni za kretanje objekata i osvježavanje stanja objekata i razreda `SceneManager` koji predstavlja graf scene te je odgovoran za detekciju sudara i osvježavanje stanja cjelokupne scene pri čemu on poziva odgovarajuće metode za osvježavanje svakog objekta.



Slika 8. Model scene.

Razred `SceneObject` predstavlja jedan objekt u sceni. Objekt scene predstavljen je svojim modelom te su uz njega vezane matrice za rotaciju, translaciju i skaliranje. Svaki od objekata ima uz sebe vezana i dva razreda koji implementiraju sučelje `IAction`, a služe za osvježavanje stanja objekta. Prvi takav razred osvježava stanje objekta kad nema sudara, a drugi osvježava stanje u slučaju sudara. Objekt ima definirani životni vijek koji može biti ograničen ili neograničen.

Razredi `SceneObjectFactory` i `ActionFactory` odgovorni su za stvaranje objekata i akcija te za postavljanje početnog stanja tih objekata i akcija. Ovi razredi implementiraju oblikovni obrazac tvornice (engl. `Factory`).

Razred `IAction` je sučelje koje moraju implementirati svi razredi koji žele mijenjati stanje objekta. Sučelje se sastoji od metoda `update` i `collide` koje osvježavaju stanje objekta u odgovarajućoj situaciji. U finalnoj verziji implementiran je razred `PatientAction` koji osvježava poziciju pacijenta te razred `StaticAction` koji se koristi kad se objekt želi stvoriti kao nepokretan. Razred `PatientAction` informaciju o pokretima pacijenta dobiva od razreda `PatientAdapter` koji služi kao posrednik između ulaznih uređaja i modela. Na ovaj način moguće je proširiti aplikaciju dodavanjem drugih ulaznih uređaja pri čemu je potrebno naslijediti `PatientAdapter`. Kao jedna od implementacija ponuđena je `KeyboardAdapter` koja koristi tipkovnicu kao ulazni uređaj.

Moguća je daljnja nadogradnja te dodavanje novih akcija pri čemu je potrebno naslijediti sučelje `IAction` te izmijeniti `ActionFactory` tako da može kreirati novu vrstu akcija.

Koordinaciju i usklađivanje svih ovih razreda nadzire `SceneManager`. `SceneManager` predstavlja graf scene i u sebi sadrži popis objekata scene pri čemu je prvi objekt teren. Početne pozicije objekata postavljaju se relativno na teren.

2.2.2. Osvježavanje stanja scene

Osvježavanje scene započinje s detekcijom sudara. Detekcija sudara provodi se unutar razreda `SceneManager`, a računa se upotrebom kvadrata obujmica. Inicijalni korak je pri tome osvježavanje kvadrata obujmica za svaki objekt u sceni. Nakon toga se za svaki par objekata ispituje da li kvadrati obujmice imaju presjek te se oni koji imaju dodaju u listu objekata koji se sudaraju. Detekcija sudara između pacijenta i objekata je jedinstvena jer pacijent nema modela objekta pa kvadrat obujmica ima unaprijed definiranu veličinu.



Slika 9. Objekti s kvadrima obujmicama za detekciju sudara.

Za detekciju sudara između objekta i terena zadužen je objekt jer reakcija na taj sudar može biti različita počevši od eksplozije, praćenja terena, zaustavljanja i drugih. Za neke od tih reakcija potrebno je računati plohu sudara dok je za neke ona suvišna. Računanje plohe sudara zahtjevan je postupak te ga je poželjno izbjeći ako je to moguće. Detekcijom sudara unutar razreda objekta postiže se upravo taj učinak.

Nakon detekcije sudara počinje osvježavanje stanja objekata. Najprije se izračunava koliko je vremena prošlo od prethodnog osvježavanja te se objekti osvježavaju za to vrijeme pozivom odgovarajuće funkcije za osvježavanje.

Objekt pacijenta osvježava se tako da se provjeri stanje tipki strelica te se ovisno o pritisnutim tipkama dobiva pozicija na koju bi se pacijent trebao pomaknuti. Najprije se izračunava visina terena u toj točki. Pacijent se pomiče samo ako visinska razlika trenutne pozicije i sljedeće pozicije nije prevelika. Nakon toga se provjerava da li na tom mjestu već stoji neki drugi objekt te se u slučaju da je mjesto slobodno pomiče pacijent.

Na kraju `SceneManager` obavještava pogled da je scena osvježena te započinje iscrtavanje scene.

2.3. Opis kamera i pogleda na scenu

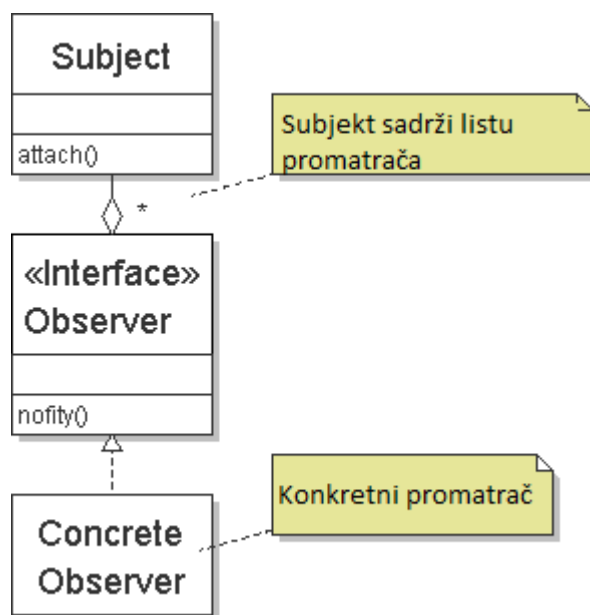
Pogled na scenu i prikaz scene na zaslonu implementiran je u klasi `Renderer`. Funkcije za prikaz pozivaju se nakon što je osvježena pozicija svih objekata. Razred `Renderer` predstavlja apstrakciju sustava za iscrtavanje. Sama implementacija parsiranja grafa scene, primjene odgovarajućih transformacija te konačnog prijenosa podataka za iscrtavanje je ostvarena ugrađenom OSG-ovom funkcionalnošću.

Scena se snima s 3 kamere. Prva kamera prati poziciju pacijenta i prikazuje njegov pogled na zaslonu aplikacije. Druga kamera također prati poziciju pacijenta te sliku sprema i šalje aplikaciji za upravljanje sadržajem scene. Treća kamera prikazuje tlocrt scene. Postavljena je visoko iznad scene i snima cijelu površinu terena. Ta slika se također šalje aplikaciji za upravljanje sadržajem scene.



Slika 10. Slike iz perspektive kamera koje snimaju scenu.

Isertavanje scene poziva se nakon osvježavanja modela pri čemu je za povezivanje korišten oblikovni obrazac promatrač (eng. observer).



Slika 11. Oblikovni obrazac promatrač.

2.4. Opis komunikacije

Usporedno s prikazivanjem sadržaja scene odvija se komunikacija s aplikacijom za upravljanje sadržajem scene. Komunikacija se odvija preko IP protokola UDP paketima. Aplikacija za prikaz scene prima poruke na adresi 192.168.1.16 na portu 10010, a šalje ih na adresu 192.168.1.15 na port 10011.

Svi podaci koji se razmjenjuju pakiraju se u tip `Message` koji ima implementirane metode `ToBytes` i `FromBytes` koje su odgovorne za pretvaranje podataka u polje bitove i rekonstrukciju podataka iz polja bitova. Iz klase `Message` nasljeđivanjem se izvode klase za konkretne vrste poruka. U finalnoj verziji aplikacije postoji 5 vrsta poruka:

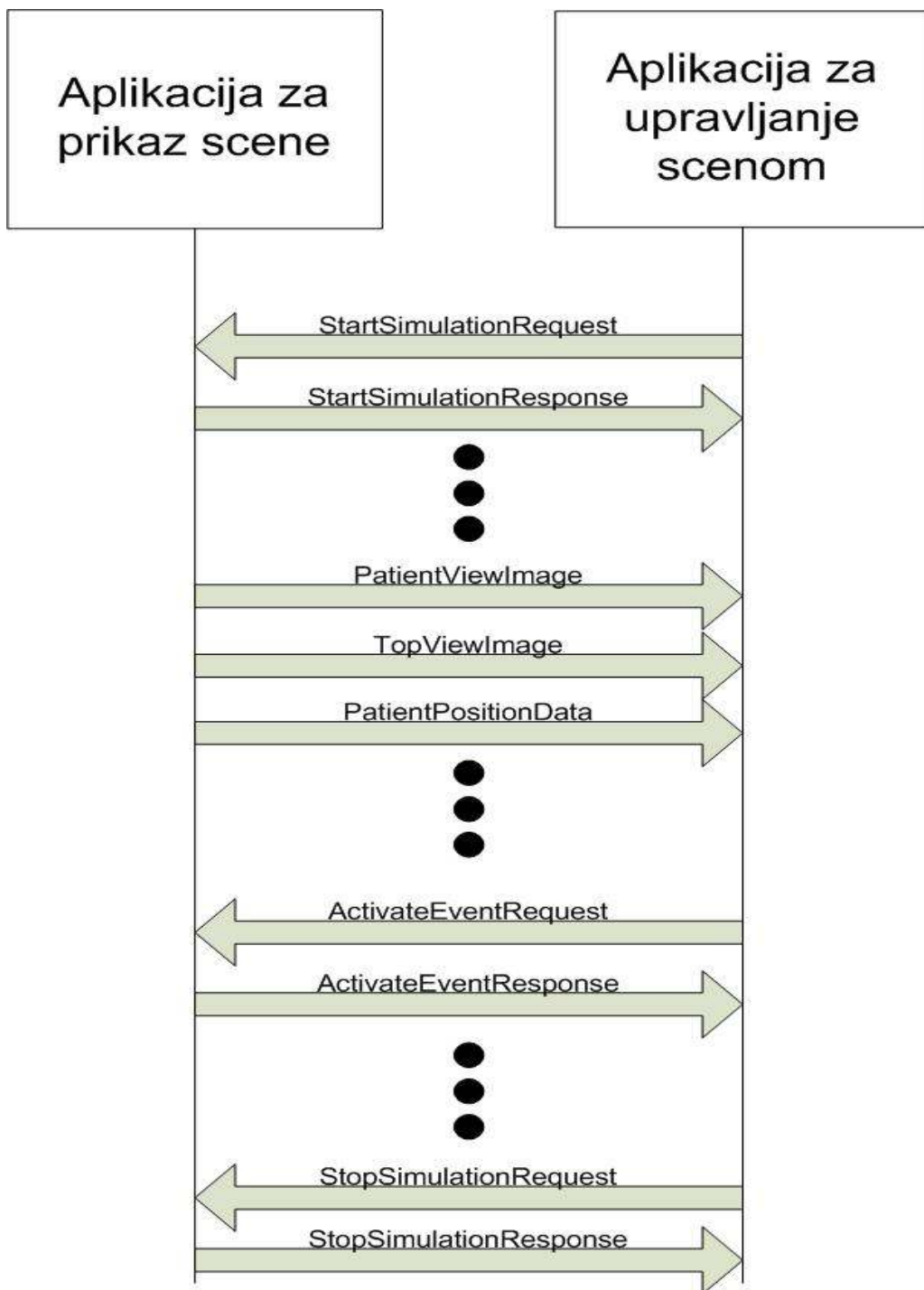
Tablica 1 Vrste poruka

Vrsta poruke	Značenje
<code>StartSimulation</code>	Zahtjev za početak simulacije
<code>StopSimulation</code>	Zahtjev za završetak simulacije
<code>ActivateEvent</code>	Zahtjev za umetanje objekta
<code>DeactivateEvent</code>	Zahtjev za uklanjanje objekta iz scene
<code>GenericResponse</code>	Generički odgovor

Aplikacija za upravljanje scenom pri slanju pridjeljuje poruci ID te očekuje odgovor koji govori o uspješnosti provođenja određenog zahtjeva.

Kad aplikacija za prikaz scene primi poruku, ona pokušava izvršiti zahtjev te šalje natrag odgovarajuću povratnu poruku s informacijom o uspješnosti zahtjeva.

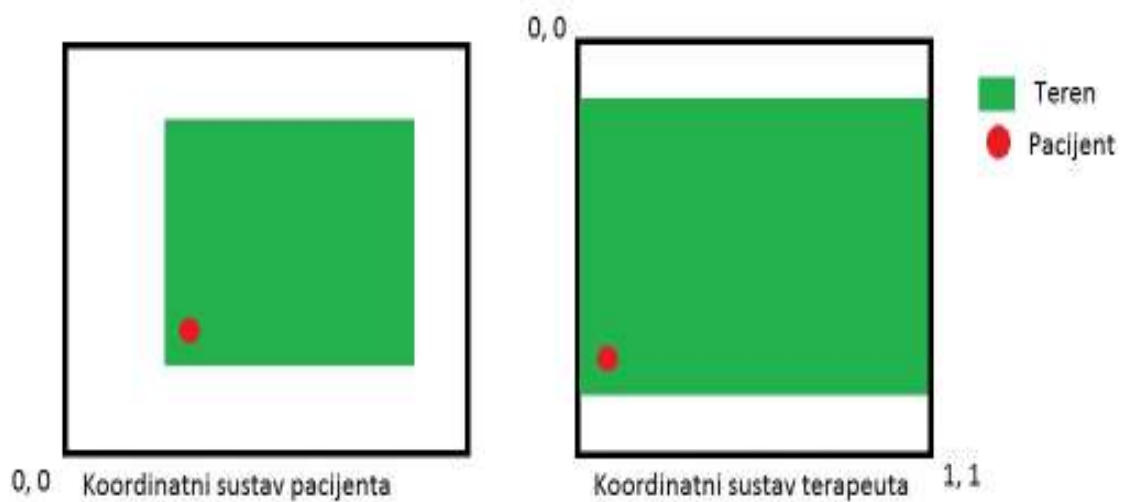
Aplikacija za prikaz scene redovito nakon osvježavanja šalje aplikaciji za upravljanje scenom trenutnu poziciju pacijenta te sliku scene iz pogleda pacijenta i tlocrt scene s objektima. Slike se šalju u rezoluciji 800 x 600 u jpg formatu. Količina poslanih slika u sekundi ograničena je na maksimalno 20 slika u sekundi, a maksimalna veličina slike iznosi 65kB.



Slika 12. Tijek komunikacije.

Pri slanju pozicije pacijenta koordinate se moraju pretvoriti iz koordinatnog sustava pacijenta u koordinatni sustav terapeuta. Koordinatni sustav pacijenta ima apsolutne koordinate dok su koordinate u koordinatnom sustavu terapeuta skalirane na interval [0, 1]

pri čemu je teren pozicioniran u centar koordinatnog sustava. Ova transformacija koordinata obavlja se unutar funkcije `ConvertToNormalizedCoordinates` razreda `util`.



Slika 13. Koordinatni sustavi pacijenta i terapeuta.

Za izračun ove transformacije najprije se računa omjer dimenzija terena te omjer dimenzija slike u aplikaciji terapeuta. Iz razlike ovih omjera može se vidjeti s koje će strane terena u koordinatnom sustavu terena biti praznine. Ako je omjer dimenzija slike veći od omjera dimenzija terena praznine će biti okomite, u suprotnom slučaju praznine su vodoravne. Nakon toga se računa debljina praznina te se teren pomiče na potrebnu poziciju. Zatim se koordinate skaliraju dijeljenjem s veličinom terena uvećanom za veličinu praznine. Na kraju se invertira y koordinata.

```

double iRatio = imageSize.x() / imageSize.y();
double terrainRatio = terrainSize.x() / terrainSize.y();

if (iRatio > terrainRatio)
{
    double crnilo = (imageSize.x() - imageSize.y() * terrainRatio) / 2;
    double offset = crnilo * terrainSize.y() / imageSize.y();

    object.x() = object.x() + offset;

    object.y() = object.y() / terrainSize.y();
    object.x() = object.x() / (terrainSize.x() + offset * 2);
}
else
{
    double crnilo = (imageSize.y() - imageSize.x() * terrainRatio) / 2;
    double offset = crnilo * terrainSize.x() / imageSize.x();

    object.y() = object.y() + offset;

    object.x() = object.x() / terrainSize.x();
    object.y() = object.y() / (terrainSize.y() + offset * 2);
}
object.y() = 1 - object.y();

```

Kod umetanja objekta se najprije provodi inverzna transformacija za dobivanje koordinate pacijenta iz koordinata terapeuta.

Razred koji je odgovoran za komunikaciju u aplikaciji za prikaz scene je Communicator. Taj razred sadrži implementaciju slanja i primanja poruka te osnovne obrade poruka, odnosno rekonstrukciju tijela poruke iz polja bitova. Ovaj razred pokrenut je na zasebnoj drevi te neprestano osluškuje i čeka poruke. Primljene poruke sprema u listu poruka te ih na poziv funkcije GetMessage prosljeđuje klasi ApplicationEngine.

2.5. Opis upravljanja aplikacije

2.5.1. Opis jezgre aplikacije

Aplikacijom upravlja razred `ApplicationEngine`. Ovaj razred je odgovorna za pozivanje osvježavanja scene, obrađivanje zahtjeva koji dolaze preko poruka te je odgovorna za pripremu podataka za slanje aplikaciji terapeuta. Osnovna logika koja izvršava ove zadatke sadržana je u metodi `loop`.

```
Čekaj zahtjev za početak simulacije;
Ponavljaj
{
    Dohvati primljene poruke;
    Obradi zahtjeve iz poruka;
    Osvježi stanje objekata u sceni;
    Pošalji poziciju pacijenta i sliku iz pacijentovog pogleda te
        tlocrt;
    Pošalji odgovore na primljene zahtjeve;
    Ako je primljen zahtjev za prekid izađi;
}
```

Ova metoda poziva se na početku rada aplikacije i izlaskom iz te funkcije završava rad aplikacije. Simulacija počinje nakon što od aplikacije terapeuta pristigne zahtjev za početak simulacije. Nakon toga se ponavlja osnovni ciklus naredbi do pristignuća zahtjeva za prekid simulacije. Osnovni ciklus počinje dohvaćanjem i obradom primljenih poruka u kojoj se u scenu dodaju ili iz nje uklanjaju odgovarajući objekti. Nakon toga se osvježava stanje objekata u sceni te se iscertava slika na zaslonu korisnika. Na kraju se aplikaciji za upravljanje sadržajem scene šalje pozicija pacijenta i potrebne slike.

Razred `ApplicationEngine` sadrži i pokazivač na razred za prikaz scene, pokazivač na razred za komunikaciju i pokazivač na razred modela scene. Preko ovih pokazivača pozivaju se odgovarajuće funkcije tih razred. Događaji u sustavu kao što su pritisak tipke, klik miša i sl. prvo dolaze do ove klase i ona ih obrađuje unutar funkcije `handle`. Ukoliko nije definirana metoda za obradu događaja on se prosljeđuje dodatnim razredima koje su se

pretplatile na događaje. Razredi se pretplaćuju na događaje pozivanjem funkcije `setAdditionalHandler` pri čemu se kao parametar predaje instanca željene klase.

Pri inicijalizaciji se poziva funkcija `preloadData` koja učitava modele iz direktorija `Models` zbog bržeg umetanja tokom rada aplikacije. Razred `ApplicationEngine` implementira Singleton oblikovni obrazac te ne može postojati više instanci ove klase.

2.5.2. Postupak umetanja objekta

U poglavlju o komunikaciji navedene s vrste zahtjeva koji dolaze preko poruka. Jedna od tih vrsta je zahtjev za umetanje objekta. U ovom poglavlju bit će objašnjen postupak obrade tog zahtjeva. Na slici je prikazana scena pacijenta bez objekata.



Slika 14. Scena bez objekata.

1. Komunikator prima poruku te ju dodaje u listu primljenih poruka
2. ApplicationEngine traži od komunikatora da mu proslijedi primljene poruke
3. ApplicationEngine obrađuje zahtjev za umetanje objekta
 - a. Kreira se novi objekt
 - b. Postavljaju se parametri objekta
 - c. Objekt se umeće u scenu
 - d. Sprema se informacija o uspješnosti umetanja objekta
4. Osvježava se stanje objekata u sceni
5. Šalje se pozicija pacijenta i odgovarajuće slike
6. Šalje se odgovor na zahtjev za umetanje s informacijom o uspješnosti umetanja

Visina na koju se objekt umeće računa se relativno s terenom. Najprije se izračuna visina terena na mjestu umetanja objekta te se nakon toga visina objekta postavlja na tu visinu uvećanu za parametar visine iz zahtjeva za umetanje objekta. Visina terena u željenoj točki dobiva se tako da se povuče pravac paralelan s osi z koji prolazi kroz tu točku te se gleda visina sjecišta tog pravca i terena.

Scena s umetnutim objektom prikazana je na slici.



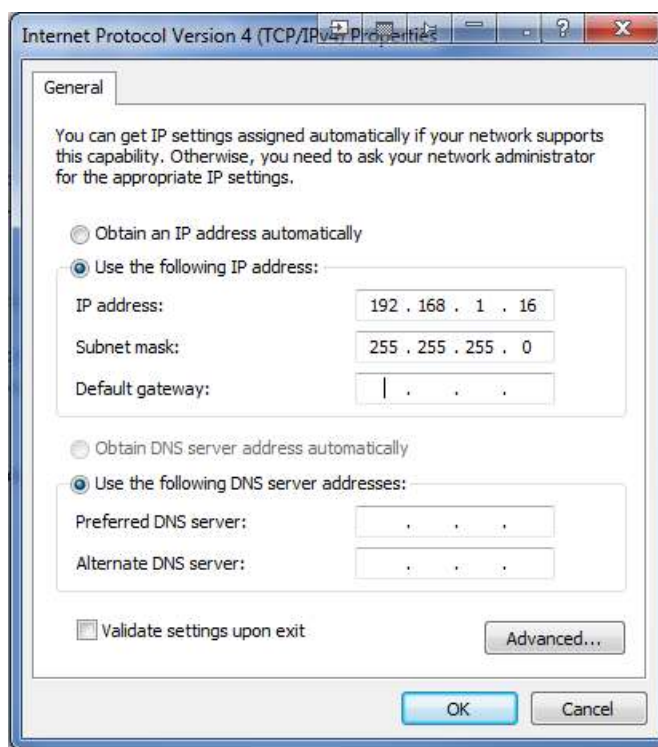
Slika 15. Scena s umetnutim objektom krave.

3. Način korištenja aplikacije s primjerima

3.1. Postavljanje početnih postavki i korištenje aplikacije

Za pokretanje aplikacije potrebno je instalirati biblioteke boost i OSG. Biblioteka boost dostupna je na adresi <http://www.boost.org/>, dok se biblioteku OSG može dohvatiti na adresi <http://www.openscenegraph.org/projects/osg/wiki/Downloads>.

Prije pokretanja aplikacije potrebno je postaviti postavke lokalne mreže. Način podešavanja prikazan je na slici.



Slika 16. Mrežne postavke aplikacije za prikaz scene.

Aplikacija kojom upravlja terapeut mora biti pokrenuta na računalu koje se nalazi na IP adresi 192.168.1.15.

Nakon postavljanja mrežnih postavki možemo pokrenuti aplikaciju za prikaz scene i aplikaciju za upravljanje sadržajem scene. U aplikaciji za upravljanje sadržajem scene

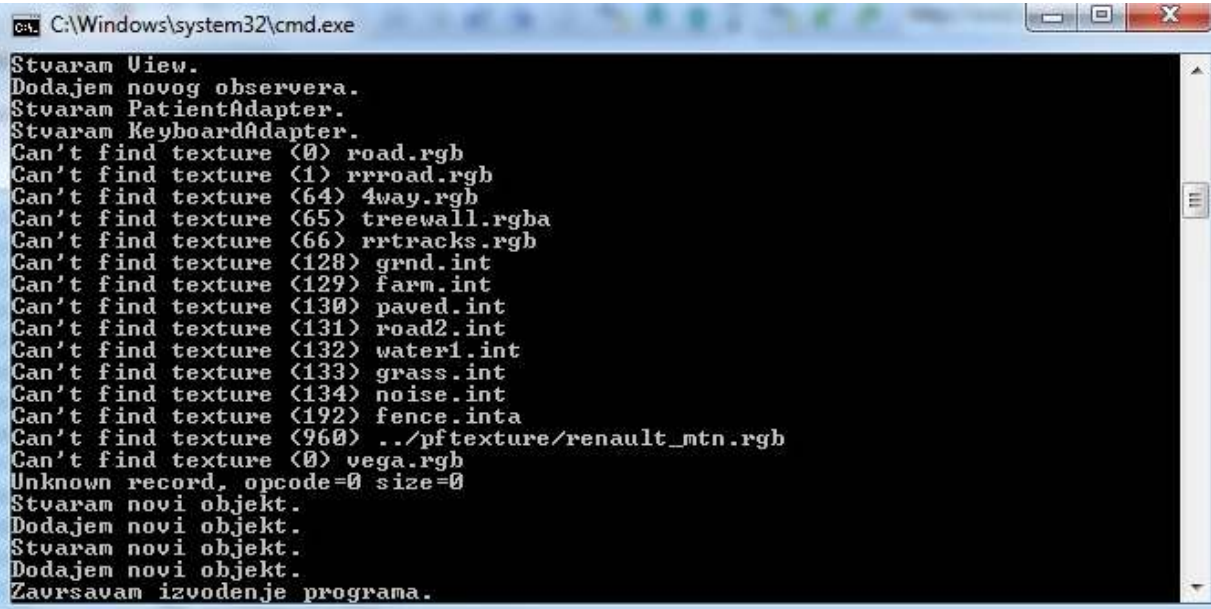
izabiremo teren koji ćemo koristiti te početnu poziciju pacijenta te pritisnemo gumb start. Pritiskom gumba start pokreće se prikaz scene.

Modeli koje želimo koristiti i umetati u scenu moraju se nalaziti u poddirektoriju Models, a teren se mora nalaziti u poddirektoriju Terrains.

Korisnik se scenom kreće strelicama na tipkovnici. Terapeut može preko aplikacije za upravljanje sadržajem scene pokretati događaje i umetati objekte u scenu. Događaji se mogu učitavati iz odgovarajućih datoteka ili ih terapeut može sam kreirati te podesiti po vlastitoj želji. Terapeut također može izgraditi scenarij koji određuje vremenski redoslijed pojavljivanja događaja te pokretanjem tog scenarija izvršiti lanac događaja bez da pokreće svaki zasebno. Na taj način je moguće isplanirati cijele seanse za određenog pacijenta.

U aplikaciji za upravljanje sadržajem scene prikazana je scena iz pogleda pacijenta te tlocrt scene s iscrtanom pozicijom pacijenta i pozicijama objekata.

Simulacija se prekida pritiskom na gumb stop u aplikaciji za upravljanje scenom. Aplikacija za prikaz u tom trenutku zaustavlja prikaz scene i čeka na ponovni pritisak tipke start u aplikaciji za upravljanje scenom. Iz aplikacije za prikaz scene izlazi se pritiskom tipke END na tipkovnici. Važniji događaji u aplikaciji ispisuju se u konzoli.



```
C:\Windows\system32\cmd.exe
Stvaram Uiew.
Dodajem novog observera.
Stvaram PatientAdapter.
Stvaram KeyboardAdapter.
Can't find texture (0) road.rgb
Can't find texture (1) rrroad.rgb
Can't find texture (64) 4way.rgb
Can't find texture (65) treewall.rgba
Can't find texture (66) rrtracks.rgb
Can't find texture (128) grnd.int
Can't find texture (129) farm.int
Can't find texture (130) paved.int
Can't find texture (131) road2.int
Can't find texture (132) water1.int
Can't find texture (133) grass.int
Can't find texture (134) noise.int
Can't find texture (192) fence.inta
Can't find texture (960) ../pftexture/renault_mtn.rgb
Can't find texture (0) vega.rgb
Unknown record, opcode=0 size=0
Stvaram novi objekt.
Dodajem novi objekt.
Stvaram novi objekt.
Dodajem novi objekt.
Zavorsavam izvodenje programa.
```

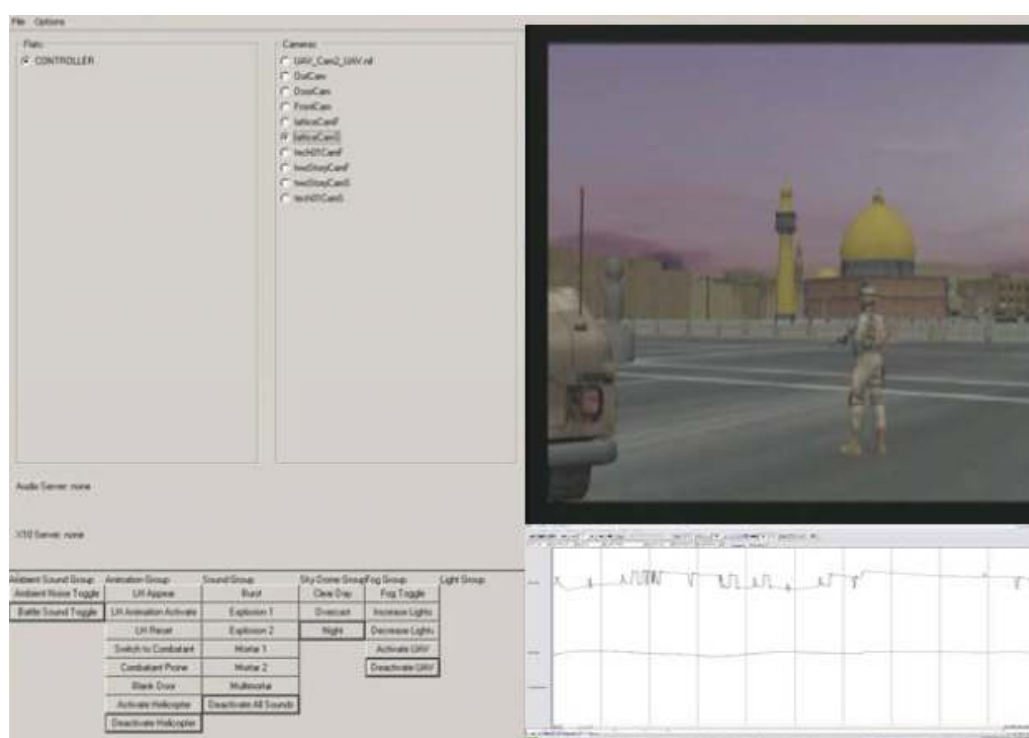
Slka 17. Ispis aplikacije u konzoli.

4. Rezultati i usporedba s postojećim sustavima

4.1. Usporedba s postojećim sustavima

Krajnji izrađeni sustav sastoji se od aplikacije terapeuta i aplikacije pacijenta. Svaka aplikacije pokreće se na zasebnom računalu pri čemu se prije pokretanja moraju postaviti postavke navedene u prethodnom poglavlju.

U odnosu na sustave koji su navedeni u uvodu ovog rada ova aplikacija nudi neka proširenja i mogućnosti koji ti sustavi nemaju. Neke funkcionalnosti tih sustava izostavljene su pri izradi ove aplikacije jer se mogu nadomjestiti korištenjem dodatnih pomoćnih programa. Na slici je prikazan sustav Virtualni Irak s Wizard of Oz sučeljem.



Slika 18. Sučelje Wizard of Oz.

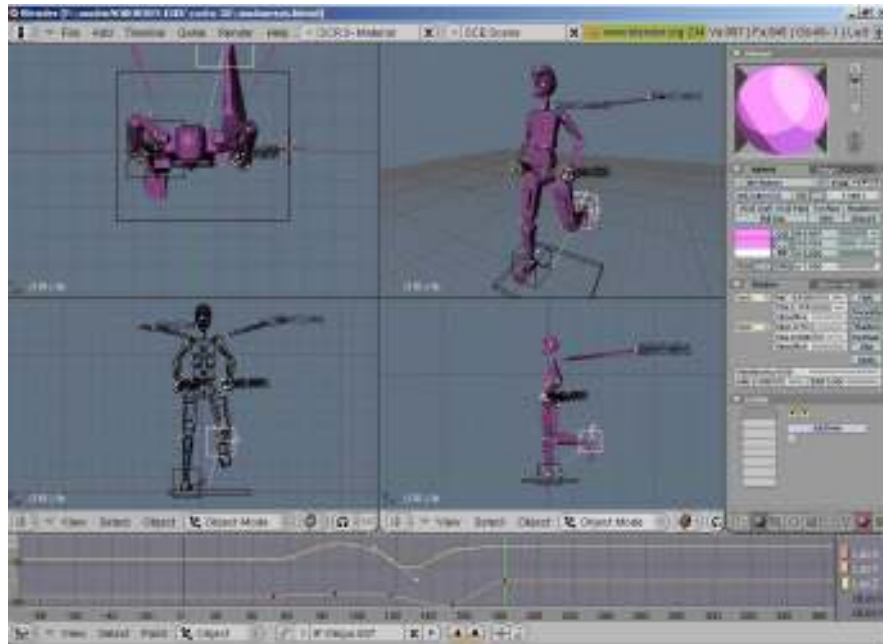
Zajedničko svojstvo postojećih sustava je da imaju predefimirani skup događaja koji se mogu pojaviti u sceni, odnosno ograničen skup objekata i načina upravljanja scenom. Kod izrađenog sustava skup objekata koji se mogu umetati je beskonačan i ovisi samo o

dostupnosti modela za te objekte. Umetnuti objekti su statični te je za omogućavanje kretanja objekta potrebno proširiti sustav. Izrađeni sustav ne omogućuje reprodukciju zvuka koja je omogućena u nekim postojećim sustavima te je i za to potrebno proširenje sustava. Velika prednost izrađenog sustava je što se objekti mogu umetnuti na bilo koje mjesto u sceni te tako nije potrebno pacijenta navoditi kamo da ide i indirektno upravljati njegovim kretanjem već se može slobodno kretati i istraživati prostor oko sebe kako bi to činio u stvarnom svijetu. Događaje u sustavu, odnosno objekte i pozicije na koje se umeću moguće je spremirati u datoteku i učitati u kasnijim uporabama sustava.

Pregled trenutnog stanja scene u aplikaciji terapeuta je dovoljan za obavljanje svih potrebnih operacija jer nudi pogled iz pozicije pacijenta čime terapeut dobiva uvid u ono što vidi pacijent i može procijeniti stanje pacijenta. Ponuđen je također i tlocrt scene čime terapeut dobiva pregled stanja cjelokupne scene. Na tlocrtu je istaknuta pozicija pacijenta te smjer pogleda pacijenta. Pogled pacijenta je ograničen te je vektor smjera uvijek paralelan s vodoravnom ravninom xy.

Unos naredbi korištenjem tipkovnice uobičajen je za ovakve sustave jer pacijentu nudi mogućnost kretanja scenom pri čemu može sjediti ili ležati što povećava razinu opuštenosti pacijenta i može pospješiti terapiju.

Neki od postojećih sustava nude sučelje za kreiranje objekata i terena. Izrađeni sustav ne nudi navedenu funkcionalnost jer je procijenjeno da se ona može postići korištenjem programa za modeliranje kao što su Blender, 3Dstudio i dr.



Slika 19. Blender.

Novitet u interaktivnom upravljanju objektima u sceni je mogućnost upravljanja scenom pomoću predefiniranog scenarija. Scenarij je slijed događaja koji se izvode određenim redoslijedom. Terapeut pri kreiranju scenarija izabire događaje i slijed kojim se izvode. Događaji mogu biti umetanje objekata, uklanjanje objekata ili čekanje određenog vremenskog raspona. Nakon kreiranja scenarija slijed događaja se pokreće klikom miša te nakon toga terapeut više ne mora upravljati scenom već to obavlja aplikacija umjesto njega. Ovim se terapeut oslobađa od dijela terapije koji mu može odvlačiti pažnju te se može bolje usredotočiti na pacijenta i njegove trenutne potrebe.



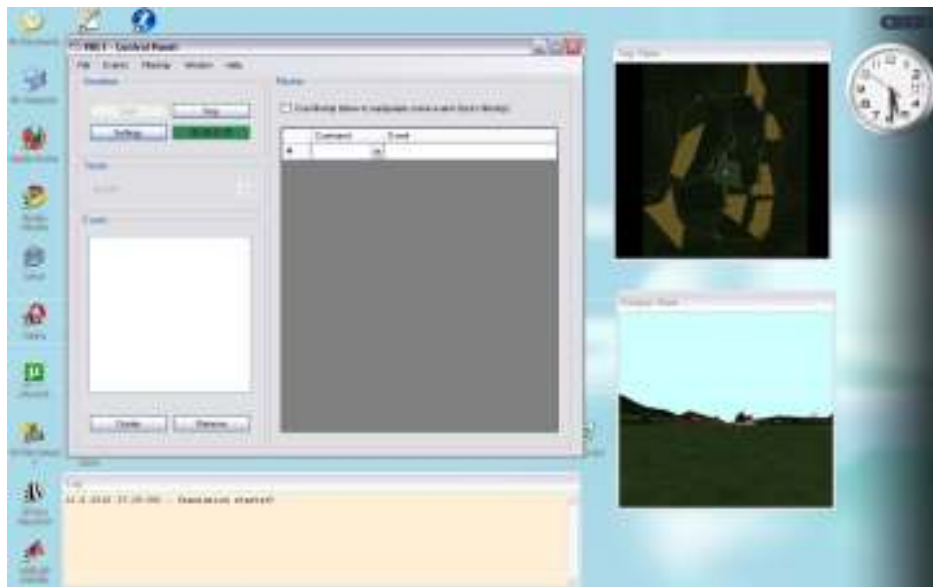
Slika 20. Upravljanje scenom uz korištenje scenarija.

Scenarij je moguće spremirati u datoteku te učitati pri sljedećem pokretanju sustava. Prije učitavanja scenarija iz datoteke i pokretanja istog potrebno je iz datoteke učitati ili definirati događaje koji se koriste u tom scenariju. U protivnom izvođenje scenarija nije moguće. Uz pomoć scenarija terapeut može prije dolaska pacijenta isplanirati tijekom cijelog sata terapije te se može pripremiti za situacije koje se mogu dogoditi.

Scenariji također mogu služiti kao standardizirani oblici terapije te je moguće za svaki poremećaj izraditi nekoliko scenarija ovisno o stupnju poremećaja i napretku i stanju pacijenta. Postupnim mijenjanjem scenarija i manjim promjenama između dvije terapije može se polako priviknuti pacijenta na neke podražaje i smanjiti njegovu reakciju na iste.

Sučelje aplikacije za upravljanje sadržajem scene sastoji se od 4 prozora. Prvi prozor služi za kreiranje i aktiviranje događaja, scenarija te upravljanje aplikacijom. Drugi prozor prikazuje poruke koje dolaze iz aplikacije pacijenta pri čemu se za neuspjelo aktiviranje događaja prikazuje razlog neuspješnog umetanja. Na preostala dva prozora prikazana je slika scene iz pozicije pacijenta i tlocrt scene. Terapeut može slobodno mijenjati poziciju i

veličinu navedenih prozora te ih može postaviti na različite monitore čime dobiva veću preglednost sustava.

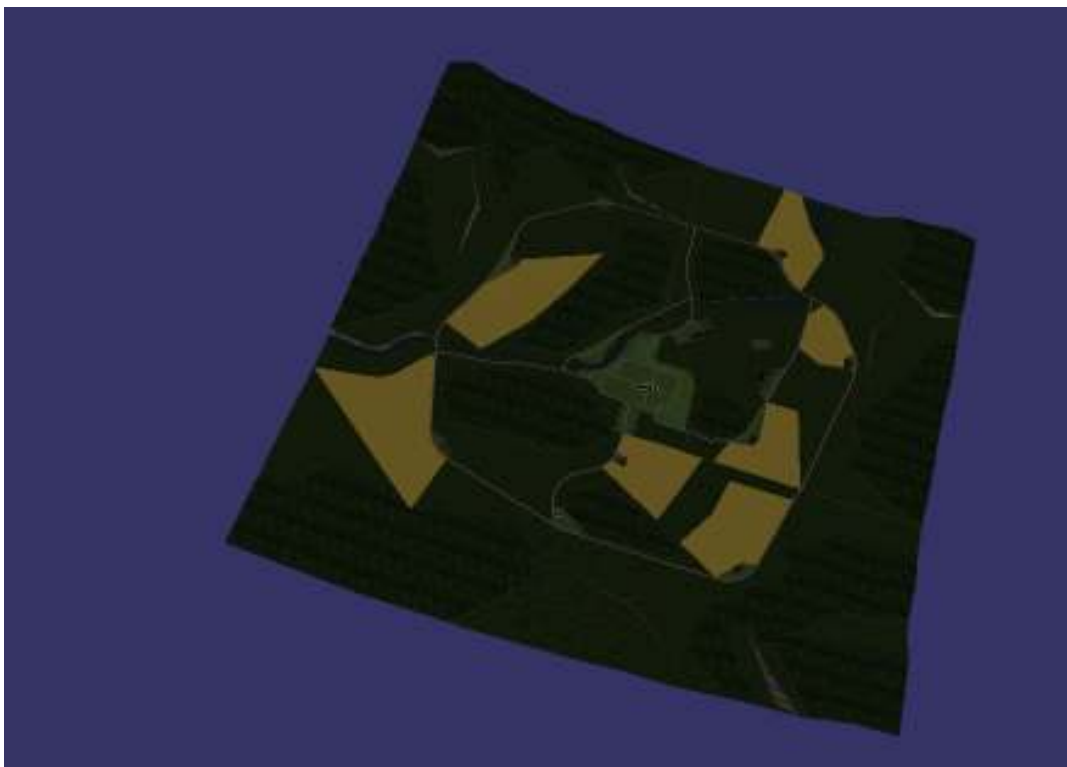


Slika 21. Prozori aplikacije terapeuta.

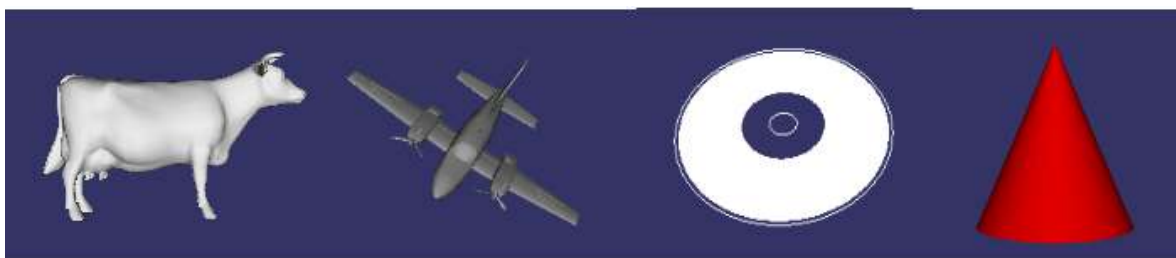
Nakon pregleda rezultata i usporedbe s funkcionalnosti s postojećim sustavima uočena su neka proširenja kojima je moguće nadograditi izrađeni sustav.

4.2. Modeli korišteni za razvoj i testiranje aplikacije

U razvoju aplikaciju korišten je model terena planinskog naselja. Teren je dugačak i širok 5000m i sastoji se od 13400 poligona i 160 datoteka s teksturama. Za neke poligone i u terenu nedostaju teksture prekrivanja pa se iscrtavaju crni. Objekti koji se umeću u scenu učitavaju se iz osg i 3ds modela. Kao ulazni uređaj je pri razvoju korištena tipkovnica. Pritiskom tipke s može se na ekranu dobiti ispis statistike s podacima o sceni i brzine iscrtavanja. Teren i objekti korišteni u razvoju prikazani su na slikama.



Slika 22. Teren korišten u razvoju i testiranju aplikacije.



Slika 23. Objekti korišteni u razvoju i testiranju aplikacije.

4.3. Performanse i ograničenja sustava

Završna verzija aplikacije ima određena ograničenja. Pogled pacijenta uvijek je paralelan s xy ravninom. U scenu se mogu umetati samo objekti učitani iz .osg i .3ds datoteka. Detekcija sudara se ne provodi za objekte iz scene i billboard objekte te pacijent može prolaziti kroz takve objekte. Brzina kretanja, visina i širina kvadra obujmice pacijenta je fiksna i ne može se mijenjati.

Aplikacije je razvijena i testirana na prijenosnom računalu ASUS K61IC s Nvidia 220m grafičkom karticom i 4 GB radne memorije. Pri početnom pokretanju scene bez objekata brzina iscrtavanja iznosi 15 slika u sekundi. Ovako mali broj slika u sekundi uzrokuje da kretanje izgleda isprekidano. Najviše se vremena troši u metodi `ConvertImage` na pretvaranje slika iz tipa podataka `osg::Image` u tip podataka `System::Drawing::Image` koji se šalje aplikaciji terapeuta. Kad bi se eliminirala potreba za tom pretvorbom slike brzina iscrtavanja bi porasla na 40 slika u sekundi. S porastom broja objekata u sceni brzina iscrtavanja polako opada zbog potrebe za osvježavanjem i detekcije sudara više objekata. Značajnije smanjenje performansi se pri tom primjećuje tek kod umetanja oko 1500 objekata pri čemu brzina iscrtavanja pada s 15 na 10 slika u sekundi. Za optimizaciju detekcije sudara kod većeg broja objekata moguće je implementirati neke od efikasnijih struktura kao što su četvorno stablo, oktalno stablo ili uniformna mreža

5. Moguća proširenja

Najveći broj proširenja moguće je izvesti na samom modelu. Završna verzija aplikacije podržava samo statičke objekte. Model ovdje možemo proširiti dodavanjem novog ponašanja objekata. Pri tome je potrebno napraviti razred koji implementira željeno ponašanje i naslijediti sučelje `IAction`. Potrebno je također proširiti metodu `createAction` razreda tvornice akcija `ActionFactory` tako da može stvarati objekte novog razreda.

Drugo proširenje je podržavanje reprodukcije zvuka i zvučnih efekata. Ovo proširenje moguće je izvesti proširenjem razreda `Renderer` koji prikazuje scenu ili implementacijom novog razreda koji će implementirati sučelje `Observer` i pratiti model.

Sklopovsko sučelje kojim pacijent upravlja aplikacijom moguće je proširiti ili zamijeniti pri čemu je potrebno napraviti razred koji nasljeđuje `PatientAdapter` i zadužen je za komunikaciju s novim sklopovskim sučeljem.

Za poboljšanje rada cijelog sustava moguće se napraviti aplikaciju koja procjenjuje stanje i emocije pacijenta. Ova aplikacija bi bilježila fiziološke znakove pacijenta kao što su puls, brzina disanja, vodljivost kože i drugi te na temelju tih značajki terapeutu prikazala pod kolikom razinom stresa se nalazi pacijent.

Nakon izrade ovih proširenja možemo dobiti sustav koji se može koristiti za terapiju izlaganjem u kliničkom okruženju.

Zaključak

Interaktivno upravljanje sadržajem scene je tehnika koja se može iskoristiti za poboljšanje mnogih vrsta simulatora. Najveća prepreka u popularizaciji liječenja korištenjem simulatora s interaktivnim upravljanjem sadržajem sceni i simulatora općenito je visoka cijena. Drugi problem može biti nedovoljna obuka i informiranost osoblja koje bi te simulatore trebalo koristiti.

Ipak cijena simulatora kao i druge računalne opreme polako pada, a računala polako ulaze u sve sfere ljudskog života te se njima zna koristiti sve veći broj ljudi.

Zbog toga se u dolazećim godinama može očekivati polagani porast popularnosti korištenja simulatora za liječenje i obuku.

Aplikacija izrađena u sklopu ovog rada može se iskoristiti kao polazna točka ili inicijalna verzija nekog budućeg sustava.

Literatura

- [1] A. Rizzo, et al., “A virtual reality exposure therapy application for Iraq War military personnel with post traumatic stress disorder: From training to toy to treatment”. *Proceedings of NATO Advanced Research Workshop on Novel Approaches to the Diagnosis and Treatment of Posttraumatic Stress Disorder*, IOS Press, pp.
- [2] B.O. Rothbaum, L.F. Hodges, D. Ready, K. Graap, R.D. Alarcon, “Virtual reality exposure therapy for Vietnam veterans with posttraumatic stress disorder”. *Journal of Clinical Psychiatry*, vol. 62, no. 8, pp. 617–622, Aug. 2001.
- [3] C. Botella, et al., “An Adaptive Display for the Treatment of Diverse Trauma PTSD Victims”. *Cyberpsychology, behavior, and social networking*, Volume 13, Number 1, 2010
- [4] Giuseppe Riva, Alessandra Gorini, Andrea Gaggioli, “Virtual Reality as Experiential Tool for Supporting Empowerment and Clinical Change”. *CHI 2008 Workshop on Technology in Mental Health*, 2008

Sažetak

(Interaktivno upravljanje sadržajem virtualne scene – Prikaz upravljane scene)

Rad opisuje izradu, korištenje i primjenu sustava za upravljanje sadržajem virtualne scene. Ovakvi sustavi najčešće se koriste u simulatorima za terapiju i obuku ljudi. Uvodni dio rada daje pregled drugih sustava koji se koriste u te svrhe. Središnja poglavlja opisuju aplikacija sustava koja se koristi za prikaz upravljane scene na sučelju pacijenta, a izrađena je u sklopu rada. Na kraju su prikazani rezultati i navedena moguća proširenja.

Ključne riječi: interaktivno upravljanje scenom, terapija izlaganjem, simulator, prikazivanje grafa scene

Summary

(Interactive virtual scene content management – Displaying the managed scene)

The thesis describes the development, use and possible applications of interactive virtual scene content management systems. These systems are most common in simulators that are used for therapy or training. The introduction provides an overview of current systems. The central chapters describe the application which displays the managed scene. This application was developed within the thesis. The final parts show the results and list possible upgrades to the system.

Keywords: Interactive virtual scene content management, saturation therapy, simulator, scene graph display