

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 4396

**IZRADA LJUDSKIH LIKOVA POMOĆU
UREĐAJA KINECT**

Marko Nadž

Zagreb, lipanj 2016.

**SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA
ODBOR ZA ZAVRŠNI RAD MODULA**

Zagreb, 15. ožujka 2016.

ZAVRŠNI ZADATAK br. 4396

Pristupnik: **Marko Nađ (0036479641)**
Studij: Računarstvo
Modul: Računarska znanost

Zadatak: **Izrada ljudskih likova pomoću uređaja Kinect**

Opis zadatka:

Proučiti osnove mogućnosti uređaja Microsoft Kinect u svrhu uzorkovanja površine trodimenzijskih objekta. Posebice obratiti pažnju na uzorkovanje ljudskih likova. Razraditi modele ljudskih likova temeljem uzorkovanih podataka te povezati s animiranim pokretima likova. Dobivene objekte uklopiti u virtualno prostorno okruženje. Ostvariti implementaciju okruženja uz mogućnost upravljanja likovima te primjenu proučenih tehnologija. Načiniti očjenu i usporedbu ostvarenih rezultata.
Izraditi odgovarajući programski proizvod. Rezultate rada načiniti dostupne putem Interneta. Radu priložiti algoritme, izvorne kodove i rezultate uz potrebna objašnjena i dokumentaciju. Citirati korištenu literaturu i navesti dobivenu pomoć.

Zadatak uručen pristupniku: 18. ožujka 2016.
Rok za predaju rada: 17. lipnja 2016.

Mentor:



Prof. dr. sc. Željko Mihajlović

Predsjednik odbora za
završni rad modula:



Prof. dr. sc. Siniša Srbljić

Djelovođa:



Doc. dr. sc. Tomislav Hrkać

Sadržaj

Sadržaj	4
1. Uvod	1
2. Uređaj Microsoft Kinect	2
3. Proces izrade likova	5
3.1 Uzorkovanje likova i ostalih objekata.....	5
3.2 Uređivanje modela.....	12
3.3 Definiranje kostura objekta.....	16
3.4 Vezanje površine modela i kostura	17
4. Integracija s virtualnim okruženjem.....	19
4.1 Izrada 3D aplikacije.....	19
4.2 Performanse	22
5. Zaključak.....	23
6. Literatura	24
7. Sažetak	25
7. Abstract	25

1. Uvod

Izrada ljudskih likova danas je česta potreba mnogih primjena. Modeli ljudskih likova prvenstveno se pojavljuju u računalnim igrama, raznim animiranim okruženjima, a imaju svoju primjenu i u medicini, vojsci i mnogim drugim područjima. S ubrzanim razvojem industrije igara javlja se potreba za što bržom i učinkovitijom izradom često korištenih vrsta modela, koji bi pri tome zadržali što veću razinu vjernosti. U takve modele se, s obzirom na raširenost uporabe, ubrajaju i modeli ljudskih likova.

U počecima razvoja 3D modeliranja, izrada modela bila je prilično ograničena sklopovskim mogućnostima tadašnjih računala, jer ona nisu mogla raditi sa zahtjevnim modelima na zadovoljavajući način. Zbog tog razloga modeli su morali biti jednostavni, s izrazito malim brojem poligona, a to je za posljedicu imalo relativno jednostavno i vremenski ne zahtjevno modeliranje. S razvojem sklopovlja tokom vremena, računala su bila u mogućnosti raditi sa sve kompleksnijim modelima. Kompleksniji modeli zahtijevaju veću vještina modeliranja, i vremenski su mnogo zahtjevniji, što može predstavljati problem ukoliko je brzina izrade važna.

Pojavom 3D kamera kao što je Microsoftov Kinect, skeniranje trodimenzijskih objekata postalo je dostupno širokom spektru korisnika prvenstveno zbog njihove relativno niske cijene. Kinect je uređaj za čije korištenje nije potrebno stručno znanje, i pored ostalog omogućava brzu izradu modela korištenjem uzorkovanih objekata. Izrada ljudskih likova pomoću Kinecta predstavlja određen kompromis između vremena izrade koje bi bilo potrebno profesionalnom modelaru i cijene kvalitetnih uređaja koji su potrebni, primjerice, za fotogrametriju. Modeli dobiveni skeniranjem pomoću Kinecta još uvijek nisu na razini modela koje bi bio u stanju izraditi profesionalac, ali su dovoljno dobri za određene primjene, i svakako predstavljaju dobru polaznu točku za izradu vjerodostojnijih modela uređivanjem rezultata uzorkovanja pomoću neke 3D aplikacije, što također olakšava posao izrade modela.

2. Uredaj Microsoft Kinect

Kinect je naziv za niz uređaja koje je razvio Microsoft prvenstveno za konzole Xbox 360 i Xbox One, i za računala s operacijskim sustavom Windows. Kinect je zamišljen kao periferni ulazni uređaj koji će korisnicima omogućiti interakciju s konzolom ili računalom kroz prirodno sučelje, koristeći geste i glasovne naredbe. U sklopu ovog Završnog rada korišten je uređaj Microsoft Kinect v2, koji će biti detaljnije opisan u nastavku.

Uređaj sadržava napredno senzorsko sklopovlje, u koje se ubrajaju dubinski senzor, RGB kamera i niz od četiri mikrofona. Uz pomoć ovih senzora moguće je koristiti uređaj za 3D skeniranje i detekciju pokreta, raspoznavanje izraza lica, prepoznavanje glasovnih naredbi i niz ostalih primjena koje mogu iskoristiti ovakvo sklopovlje.



Slika 1. Microsoft Kinect v2

Dubinski senzor se sastoji od infracrvenog projektor i infracrvene kamere. Senzor određuje dubinsku sliku korištenjem metode projiciranja strukturirane svjetlosti (engl. structured light), što se odvija kroz sljedeća tri osnovna koraka:

1. Infracrveni projektor projicira poznati uzorak točaka u prostor u kojem je smješten.

2. Infracrvena kamera snima uzorak kojeg je projektor projicirao. Uzorak će biti deformiran zbog objekata na koje je zraka naišla prilikom projiciranja – točke na dijelovima uzorka koji su projicirani na udaljenije objekte će biti jače razmaknute nego točke projicirane na bliže objekte. Ta činjenica će poslužiti kao temelj za daljnju analizu skeniranog prostora.
3. Usporedbom izvornog, projiciranog uzorka i snimljenog uzorka određuju se pojedine udaljenosti snimljenih predmeta, na temelju čega se konstruira dubinska slika.

Dubinska slika je predstavljena nijansama sive boje, pri čemu tamniji slikovni elementi (engl. pixel) predstavljaju manju dubinu, a svjetlijii veću. Crni slikovni elementi označavaju da za područja kojima pripadaju nisu dostupni dubinski podaci, što se može dogoditi zbog nekoliko razloga: ako su točke predaleko, pa se dubinske vrijednosti ne mogu precizno odrediti, objekti su u sjeni projektila, pa na taj dio ne dosegaju infracrvene točke, ili nedovoljno dobro reflektiraju infracrveni dio spektra, što se događa u slučaju snimanja reflektivnih površina, kao što su leće naočala, zrcalo i sl. Primjer dubinske slike dobiven pomoću Kinecta prikazan je na slici Slika 2.



Slika 2. Dubinska slika

Jedna od također važnih komponenata uređaja koja pridodaje značenje onome što je detektirano, i time omogućava prepoznavanje pokreta, je softver koji on koristi. Prilikom pokretanja uređaja, najprije se skenira prostor u kojem se korisnik nalazi. Nakon toga pronalaze se karakteristične točke na tijelu koje se zatim kontinuirano prate, što omogućava preslikavanje stvarnih kretnji u njihovu digitalnu reprezentaciju. Detekcija pokreta nije izvedena na način da je u uređaj najprije pohranjena svaka moguća akcija, svaki pokret osobe, koja bi se preslikavala u odgovarajuću reakciju u softveru, jer bi to bilo neizvedivo. Umjesto toga, Kinect „uči“ kako reagirati na ulaznu informaciju na sličan način kao što i ljudi uče – klasifikacijom gesti. Ovo je postignuto prikupljanjem velike količine podataka dobivene snimanjem ljudskih pokreta iz stvarnog svijeta, i obradom uz pomoć algoritama strojnog učenja. Rezultat ovog procesa bio je uspješno povezivanje tih podataka s modelima različito odjevenih ljudi, različite dobi, spola i fizičkog izgleda. To je imalo za posljedicu mogućnost zaključivanja o položajima osobe čak i u nekim slučajevima gdje se dijelovi tijela međusobno preklapaju, ili je dio tijela osobe zaklonjen primjerice komadom namještaja.

3. Proces izrade likova

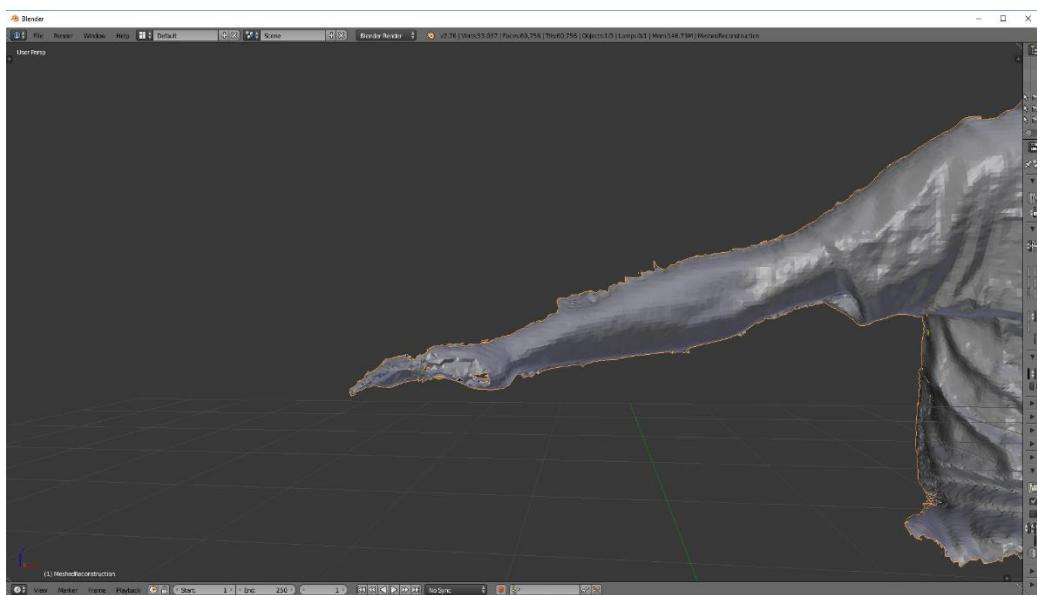
U sklopu ovog Završnog rada izradio sam dva modela ljudskih likova temeljena na uzorkovanju pomoću Kinecta, čiji postupak izrade će opisati u nastavku.

3.1 Uzorkovanje likova i ostalih objekata

Kako bih mogao koristiti Kinect, najprije sam morao preuzeti razvojni paket *Kinect for Windows SDK 2.0*, i instalirati odgovarajuće upravljačke programe (engl. driver). Nakon postavljanja potrebnog okruženja, preuzeo sam aplikaciju *3D Scan* koja omogućava uzorkovanje trodimenzijskih objekata. Aplikacija nudi mogućnost definiranja volumena u kojem će se odvijati uzorkovanje. To se postiže definiranjem vrijednosti visine, širine i dubine unošenjem željenih parametara. Za početak sam podesio volumen za uzorkovanje tako da odgovara visini osobe koju sam odlučio snimiti. Nakon što su ti parametri uneseni, potrebno je još odrediti razlučivost u kojoj će se uzorkovanje odvijati. Prije početka snimanja još je potrebno samo da osoba stane u odgovarajući položaj, što je za potrebe ovog rada bila simetrična T-poza. Ova poza je odabrana jer je u kasnijim fazama pripreme likova lakše raditi s modelima u T-pozi. Lakše je raditi na ovaj način prvenstveno iz razloga što se određeni postupci, kao što je postavljanje kostura (engl. rigging) i pridjeljivanje težinskih vrijednosti kostiju poligonalnoj mreži (engl. skinning), mogu provesti brže i s manje duplikacije posla ako radimo sa simetričnim likovima. Kad su svi parametri podešeni i osoba je spremna, snimanje može započeti. Pokretanjem procesa uzorkovanja aplikacija najprije definira prostor u kojem će uzorkovati objekte, a on je određen smjerom u kojem je kamera okrenuta i prethodno definiranim parametrima za volumen. 3D poligonalna reprezentacija prednje strane osobe se vrlo brzo pojavljuje na zaslonu, a da bi se dobio cjeloviti model, bilo je potrebno obići osobu i snimiti je sa svih strana.

Već prilikom prvog snimanja naišao sam na nekoliko problema. Da bi uzorkovanje rezultiralo vjerodostojnim modelom, nužno je da osoba stoji izrazito

mirno, što se kod dvominutnog snimanja pokazalo teško izvedivim. Svaki pokret, primjerice pomicanje ruke, uzrokovao bi ponovno izračunavanje položaja poligona modela, pri čemu bi poligoni postepeno počeli nestajati s mesta na kojem je ruka do tada bila i nastajati na novom mjestu ruke. Problem je u tome što prilikom snimanja sprjeda ruke koja se pomakne prema naprijed, aplikacija nema informaciju o tome da se ruka više ne nalazi straga, pa se na taj način dobiju modeli koji su na dijelovima rastegnuti, ili sadrže poligone na mjestima na kojima to ne bi smjeli. Primjer ovog problema prikazan je na slici Slika 3.



Slika 3. Deformacije uzrokovane pokretom

Svi dijelovi tijela koje bi osoba pomaknula za vrijeme snimanja uzrokovali bi neprirodne deformacije kod dobivenog modela, stoga je potrebno stajati što je moguće mirnije. Deformacije su bile najizraženije u području ruku, osobito na rubnim dijelovima kod zapešća i prstiju. To se događa iz razloga što je prilično naporno zadržati raširene ruke mirno, pa bi nakon nekog vremena i blago podrhtavanje ruku rezultiralo velikim otklonima od izvornog položaja – otklon je to veći što je udaljenost od ramena veća, pa su tako i deformacije bile najjače u području šake. Nepomični dijelovi kao što su noge i trup su snimljeni uz mnogo manje deformacije.

Kako bih pokušao ublažiti ovaj problem što je moguće više, uputio sam osobu koju sam snimao da stane u tzv. A-pozu, u kojoj je lakše zadržati ruke mirnima jer je napor u tom slučaju manji. A-poza također ima određena dobra svojstva koja bi

u kasnijim fazama uređivanja modela trebala olakšati posao. Prije svega je simetrična kao i T-poza, pa se postupci koje je potrebno provesti na jednoj strani modela mogu „zrcaliti“ i na drugu stranu. Ovdje postoji mogućnost da će biti potrebno malo više rada prilikom vezanja površine objekta i kostura jer su sada ruke bliže trupu, pa je moguće da kosti trupa počnu utjecati na dijelove poligonalne mreže ruku, ali to je svakako lakše rješiv problem nego što bi bilo ručno ispravljanje mreže da bi se dobio realističan izgled. Sljedećih nekoliko pokušaja snimanja je dalo vidljivo bolje rezultate nego ranije, premda su deformacije na nekim mjestima još uvijek postojale, samo blaže nego u prethodnom slučaju.

Sljedeći problem na koji sam naišao bio je nejednoliko osvjetljenje modela. Izvor svjetlosti bio je postavljen s prednje strane, pa bi stražnja strana bila u sjeni. Također, pokazalo se da jedan izvor svjetlosti nije dovoljan ni za snimanje samo prednje strane osobe, jer bi u slučaju samo jednog izvora nastajala sjena u području očiju, ispod nosa i ispod vrata. Sjena je bila vidljiva već i na bočnim dijelovima glave i trupa, što prilikom probnog postavljanja lika u virtualnu scenu nije izgledalo prihvatljivo. Potrebno je snimiti lik koji je osvijetljen što je moguće ravnomjernijim svjetлом, kako bi se kasnije izbjegle nepostojeće i neprirodne sjene. Rezultat takvog snimanja trebala bi biti vjerodostojnija interakcija modela sa simuliranom svjetlošću virtualne scene, gdje bi jedine sjene bile one koje su nastale kao rezultat nedostatka svjetlosti na određenom dijelu lika, umjesto neprirodnih zatamnjenja koja su rezultat slabog osvjetljenja prilikom snimanja. Kako bih postigao što uniformnije osvjetljenje osobe, odlučio sam u sljedećim pokušajima uvesti dvije promjene. Da bih osvijetlio osobu sa svih strana jednoliko, morao bih koristiti malo složeniji sustav osvjetljenja s puno više izvora svjetlosti. Zbog tog razloga odlučio sam se za način snimanja gdje osoba više neće mirno stajati dok ju kamera obilazi sa svih strana, već da ovaj put kamera stoji na mjestu, a osoba se okreće. Na ovaj način svjetlost će uvijek dolaziti s iste strane (izvor će se nalaziti iza Kinecta, i osvjetljavati scenu u smjeru snimanja), što znači da će uvijek ista količina svjetlosti osvjetljavati snimanu osobu. Za osvjetljavanje sam ovaj put koristio četiri zasebna izvora svjetlosti. Prva dva su bila postavljena iznad razine glave, jedan s desne, a drugi s lijeve strane, a bili su usmjereni prema dolje. Preostala dva izvora su bila smještena na podu, također jedan s lijeve a drugi s desne strane, i bili su usmjereni prema gore. Koristeći ovakvu konfiguraciju izvora svjetlosti uspio sam riješiti problem sjena u području očiju i

ostalih, jer je u ovom slučaju svaki dio tijela (okrenut prema kameri) bio osvijetljen. Bilo je još samo potrebno odrediti smjer osvjetljenja i udaljenosti izvora od snimane osobe kako bi se izbjeglo prejako osvjetljenje i prevelika preklapanja svjetlosti koju projiciraju različiti izvori.

Prilikom snimanja kod kojeg osoba više ne stoji nepomično, već se okreće, a kamera je statična, bilo je još potrebno pripaziti na već ranije opisan problem deformacija mreže kod pokreta. Kako i relativno mali otkloni od izvornog položaja uzrokuju deformacije, jasno je da se uzorkovanje neće moći provesti na način da se snimana osoba sama okreće pokretima nogu, jer bi to prouzročilo deformacije koje bi bile čak i izraženije nego u prethodnim pokušajima. Kako bi se izbjegla potreba za pomicanjem u svrhu rotiranja, odlučio sam za potrebe snimanja prilagoditi rotirajuću platformu na kojoj bi snimana osoba mogla mirno stajati, za što je bilo dovoljno iskoristiti uredsku stolicu koja ima mogućnost rotiranja, i kojoj je uklonjen naslon kako ne bi zaklanjao osobu prilikom snimanja. Kako bi ovakav način snimanja, kod kojeg je kamera statična, a snimani objekt ili osoba dinamična, bio uspješan, nužno je da se u kadru ne nalaze nikakvi sporedni predmeti. Ako se u kadru nalazi samo objekt na rotirajućoj platformi, moguće ga je rotirati pri čemu se ostvaruje isti učinak kao da je objekt statičan, a kamera ga obilazi. Aplikacija koja sastavlja poligonalnu mrežu naprsto se može „zavarati“ na ovaj način, jer se ponaša kao da je volumen koji se uzorkuje statičan, a kamera ga snima iz različitih kutova, premda je situacija obrnuta. Ako se u kadru nalaze i drugi objekti, kao što je primjerice zid prostorije, ovakvo snimanje neće uspjeti jer je sada moguće zaključiti da je kamera ipak statična, a u sceni se događaju promjene. Rezultat toga je neprestano osvježavanje scene, koje ovisi o trenutnom položaju osobe, pri čemu je uvijek samo jedna strana snimljena.

Uz ovakav način snimanja gdje je kamera statična, osoba se okreće na rotirajućoj platformi, i pri tome je osvijetljena s četiri izvora svjetlosti, sjene su uklonjene gotovo u potpunosti. Rotiranje također nije uzrokovalo ništa veće deformacije mreže nego što je to bio slučaj sa statičnom scenom i pomičnom kamerom. Problemi deformacija i sjena su na ovaj način uklonjeni gotovo u potpunosti, međutim, pojavio se novi problem kod stvaranja tekstura snimane osobe. Samo ona strana osobe koja bi bila zadnja uhvaćena u kadar bi uvijek imala

ispravno obojene poligone, a prethodno snimljene strane bi imale poligone kojima je boja izmijenjena na neodgovarajući način. Ova pojava je vidljiva na slici Slika 4.



Slika 4. Neodgovarajuće vrijednosti boje

Na slici se vidi kako je stražnjem dijelu glave i vrata pridružena svjetla nijansa sive boje, a rukama je pridružena tamno smeđa boja. Boje na prednjoj strani su bile u redu, odnosno nije došlo do ovakvih izmjena. Razlog zbog kojeg je došlo do ovakvih promjena u boji jest taj što aplikacija za uzorkovanje koju sam koristio, u trenutku kad već snimljena strana tijela izade iz kamera, počne prepostavljati koja boja bi se mogla nalaziti na dijelu mreže koji je netom izašao iz kamera. Te prepostavke će se donositi uvijek za one rubne dijelove tijela koji zbog rotacije izlaze iz kamera. Kako zbog raspršenja svjetlosti i refleksije od ostalih površina u prostoru na snimanu osobu ne pada samo svjetlost prethodno postavljenih četiriju izvora svjetlosti, već i svjetlost nastala refleksijom od zidova, namještaja i slično, dolazi do preuzimanja boje koja je nastala refleksijom. Primjerice, na slici se vidi kako na dijelovima mreže koji predstavljaju ruke, osim normalne komponente boje, postoji i tamno smeđa boja nastala zbog refleksije. U ovom slučaju boja je smeđa jer se u trenutku snimanja iza osobe u visini ruku nalazio komad namještaja smeđe boje koji je reflektirao svjetlost prema Kinectu. U području glave dobivena je svjetla boja koja je nastala kao rezultat svijetle boje zidova iza glave osobe. Ovo se dogodilo jer Kinect nije mogao u potpunosti razlučiti koje boje pripadaju samoj osobi, a koje pozadini, stoga bi se prilikom rotacije povremeno pogrešno zaključilo o boji u

rubnim točkama. Pogrešna interpretacija boje najizraženija je na konveksnijim dijelovima, kao što su ruke, glava, ili nabori odjeće. Udubljeniji i ravniji dijelovi nisu u tolikoj mjeri opterećeni ovim problemom jer slikovni elementi koji pripadaju udubljenjima ne mogu ni jednom trenutku doći u dodir sa slikovnim elementima pozadine u ulaznoj slici.

Ovaj problem sam nastojao riješiti postavljanjem što je moguće tamnije pozadine, kako bi smetnje bile što slabije. Najbolje rezultate sam dobio postavljanjem crnog zastora u pozadinu, i usmjeravanjem svjetlosti tako da pada što je moguće više na snimanu osobu, a što manje na zastor, kako bi kontrast u osvjetljenju bio maksimalan. Boje su u ovom slučaju bile uočljivo prirodne, ali još uvijek su na pojedinim mjestima postojala svjetlijia mjesta koja su nastala jer je i crni zastor koji sam koristio ipak reflektirao određenu količinu svjetlosti, koja je bila dovoljna da izazove ove smetnje. Prije ručnog pokušaja ispravljanja neispravno obojenih područja, odlučio sam provesti uzorkovanje s nekom drugom aplikacijom koja bi mogla riješiti ovaj problem na neki način. Preuzeo sam besplatnu inačicu aplikacije za 3D snimanje *KScan3D*, koja funkcioniра na malo drugačiji način. Umjesto kontinuiranog snimanja i nadograđivanja poligonalne mreže, *KScan3D* u diskretnim trenucima pamti ulazne vrijednosti za dubinu i boje, i na temelju tih vrijednosti u zadanim trenucima konstruira onaj dio modela koji je u tom trenutku vidljiv. Kako se snimani objekt ili osoba okreće, u zadanim razmacima, i zadani broj puta, uzimaju se snimke na temelju kojih nastaje model dio po dio – svaki novi dio se poravnava u odnosu na sve ostale prethodno snimljene dijelove i zatim se spaja s njima. S obzirom da *KScan3D* ne provodi uzorkovanje kontinuirano, ne pojavljuje se već opisan problem pogrešnog izračuna boje za neke dijelove mreže.

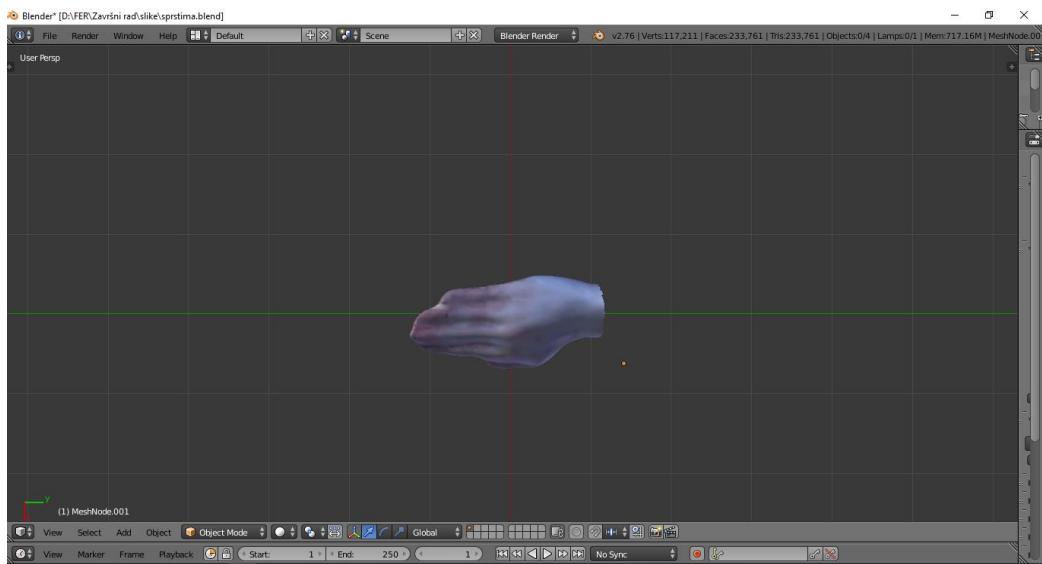
U ovom trenutku svi problemi koji su se pojavili za vrijeme uzorkovanja su uklonjeni ili ublaženi gotovo u potpunosti – deformacije uzrokovane neizbjježnim pomacima su još uvijek prisutne, ali u značajno manjoj mjeri nego u početku; sjene u području očiju i ispod vrata su gotovo neprimjetne; boje se više nimalo ne „razljevaju“ i doprinose vjernom izgledu modela. Ovako dobiven model dalje se može koristiti u raznim virtualnim okruženjima, moguće ga je animirati i koristiti za mnoge druge primjene.

Premda su svi značajni problemi otklonjeni, i dobiveni lik je dovoljno vjeran za određene primjene, odlučio sam pokušati dodatno povisiti kvalitetu snimljenog modela promjenivši metodu snimanja. Svi raniji modeli su dobiveni snimanjem čitavog lika odjednom. Da bi to bilo moguće, cijeli lik osobe mora moći stati u kadar, što podrazumijeva i razmjerno veliku udaljenost između osobe i Kinecta. Prednost ovog pristupa je brzina izrade – cijeli model je gotov nakon samo jednog snimanja. Najveći nedostatak je smanjena kvaliteta modela zbog povećanog razmaka između kamere i osobe, jer kvaliteta snimke pada s udaljenošću od kamere. Upravo zbog tog razloga odlučio sam izraditi model u dijelovima. Kod snimanja dio po dio, ne postoji potreba da osoba stoji na nekoliko metara udaljenosti od Kinecta, već može doći na minimalnu dozvoljenu udaljenost, koja za Kinect v2 iznosi 50 cm. Snimanjem s maksimalnom dostupnom razlučivošću i minimalnom udaljenošću postignuti su značajno bolji rezultati – prilikom snimanja na većoj udaljenosti finiji detalji bi bili izgubljeni, primjerice uši lika bi bile gotovo u potpunosti poravnate s ostatkom glave, što se ovako izbjegava. Koristeći ovaj pristup sam redom snimio najprije glavu i vrat, zatim trup od vrata do nogu i noge. Nakon toga sam snimio desnu ruku od ramena do zapešća. Kod uzorkovanja prstiju ruke javili su se novi problemi. Prsti ruke predstavljaju sitne dijelove s kojima Kinect ne barata baš najbolje – vrhovi prstiju su naprsto zanemareni kao da ni ne postoje. Ova pojava je prikazana na slici Slika 5.



Slika 5. Nedostatak detalja

Neovisno o korištenoj aplikaciji, razlučivosti uzorkovanja ili udaljenosti prstiju od kamere, problem je ostao prisutan u podjednakoj mjeri. S obzirom da se uzorkovanje tankih objekata pokazalo neostvarivim, odlučio sam snimiti prste koji ovaj put neće biti razmaknuti, već spojeni. Snimanje je ovaj put bilo uspješnije, a rezultat se može vidjeti na sljedećoj slici.



Slika 6. Rješenje problema s nedostatkom detalja

3.2 Uređivanje modela

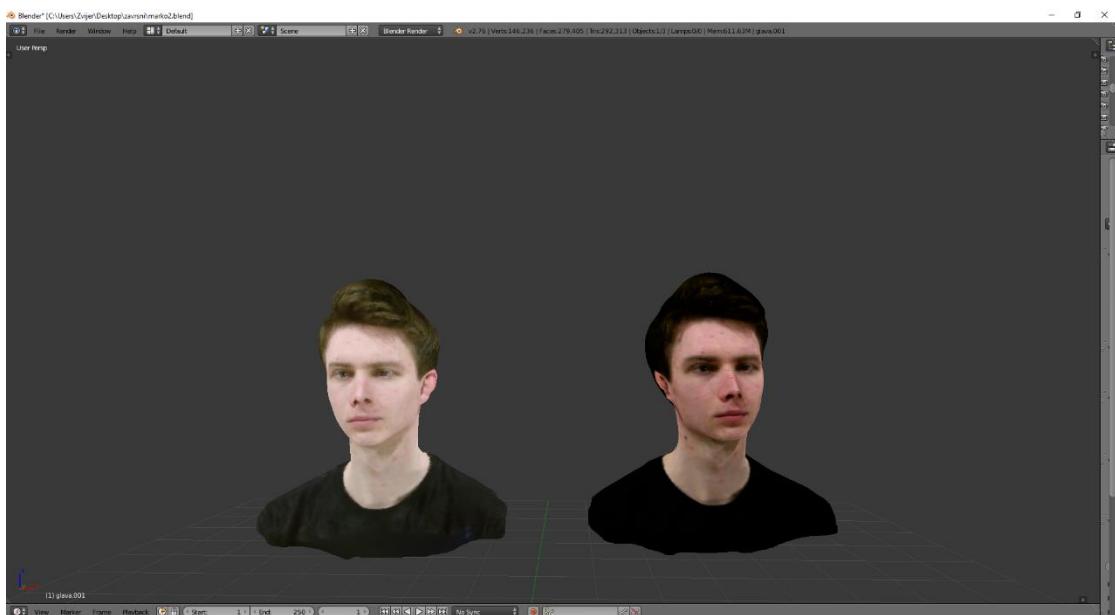
Nakon što su svi potrebni dijelovi modela uzorkovani, potrebno ih je povezati uz pomoć aplikacije za 3D modeliranje. Za uređivanje modela koristio sam Blender, 3D aplikaciju otvorenog koda za modeliranje, animiranje, izradu vizualnih efekata i druge primjene.

Najprije je potrebno uvesti dobivene mreže poligona u Blender. Za dobavljanje uzorkovanih podataka Bleneru sam koristio format .fbx, koji je pogodan jer pohranjuje i podatke o boji. S obzirom da je uzorkovanje provedeno u najvećoj dostupnoj razlučivosti, modeli koji su nastali na taj način su imali jako velik broj vrhova. Primjerice, model glave se sastojao od oko 250 000 vrhova i zauzimao je više od 20MB memorije. Svi dobiveni dijelovi zajedno imaju gotovo milijun i pol vrhova, što je svakako prevelika količina vrhova. Broj vrhova je potrebno smanjiti kako bi se smanjilo opterećenje na grafički procesor prilikom korištenja lika u

virtualnom okruženju. Osim što je ovako velik broj vrhova nepovoljan za performanse, također je i mnogo zahtjevниje uređivati takve likove. Zbog tih razloga prvi korak je bio izrada modela s manjim brojem vrhova. U Blenderu se ova operacija može izvesti jednostavno. Najprije je potrebno duplicirati model za koji se želi izraditi inačica s manje vrhova. Dobiveni model se smješta u novi sloj (engl. layer), na istim koordinatama kao i polazni model. Na ovaj način oba modela se nalaze na identičnom mjestu, ali se u svakom trenutku vidi samo jedan od njih, što olakšava rad s modelima. Dupliciranom modelu se potom dodaje modifikator *remesh modifier* koji generira topologiju s drugačijim brojem vrhova koji se može podešavati. Za sve modele sam odabrao minimalni broj vrhova za koji modeli ne mijenjaju izgled u velikoj mjeri. Najveća promjena postignuta je za trup i noge, jer se radi o površinama koje su većinom ravne – ako nabori odjeće polaznog modela nisu jako izraženi, promjena u broju vrhova nije jako uočljiva. U području glave ostavio sam najveći broj vrhova jer je smanjenje broja poligona najuočljivije na ljudskom licu. Polovica ukupnog broja svih vrhova se zbog toga nalazi u području glave lika. Smanjivanjem broja vrhova svih dijelova dobiveni dijelovi su se ukupno sastojali od oko 12 000 vrhova, što je značajna ušteda.

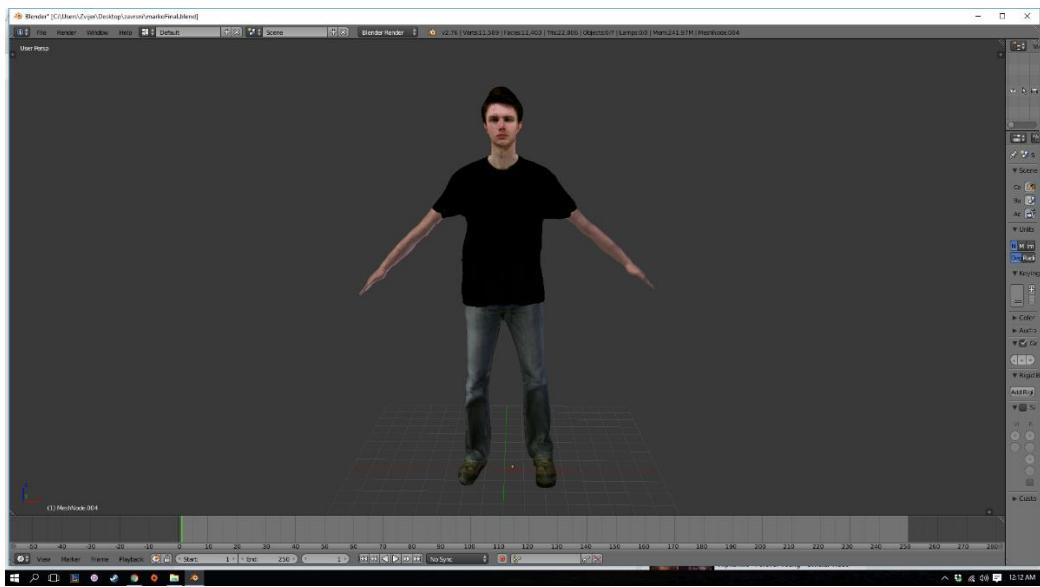
Dodatkom modifikatora *remesh* dobiveni lik odmah gubi informaciju o boji. Za generiranje teksture za novonastali model poslužit će izvorni model privremeno spremlijen u drugom sloju. Izvorni model sadrži potrebnu informaciju o boji za svaki vrh (engl. vertex paint) koja će biti projicirana na model s manje vrhova. Prije preuzimanja boje, potrebno je označiti *shadeless* svojstvo materijala kako se u generiranoj tekstuuri ne bi pojavile sjene i ostale modifikacije boje koje su vidljive u sceni zbog osvjetljenja u samoj sceni. Da bi projekcija bila moguća, najprije je potrebno provesti postupak UV preslikavanja (engl. UV mapping). To je postupak koji „razmata“ trodimenzionalni objekt uzduž njegovih bridova čime se dobiva dvodimenzionalna reprezentacija objekta. Razmotranom modelu se pridružuju nove koordinate u dvodimenzionalnom koordinatnom sustavu, što će omogućiti pridruživanje slikovnih elemenata tekture koordinatama modela u 3D prostoru. Naziv „UV preslikavanje“ potječe od oznaka „u“ i „v“ za koordinatne osi dvodimensijskog koordinatnog sustava, jer su oznake „x“, „y“ i „z“ rezervirane za trodimenijski sustav modela. U Blenderu je moguće ovaj postupak provesti na više načina, a za svoje modele koristio sam opciju *smart uv project* koja ovaj postupak

provodi automatski. Jednom kad je UV odmotavanje gotovo, može se pokrenuti postupak generiranja teksture projiciranjem s polaznog modela na onaj sa smanjenim brojem vrhova. Za svaki dio modela je na ovaj način generirana odgovarajuća tekstura. Na sljedećoj slici uspoređeni su modeli glave od kojih je lijevi polazni model, a desni model dobiven smanjenjem broja vrhova i generiranjem teksture.



Slika 7. Usporedba modela s različitim brojem vrhova

Kad je svim dijelovima smanjen broj vrhova i kreirana tekstura, dijelovi se mogu povezati u jedinstvenu cjelinu. Dijelovi su povezani jednostavno pomicanjem na odgovarajuća mjesta i podudaranjem susjednih dijelova. Prilikom podudaranja dijelova, u većini slučajeva postojali su dijelovi mreže koji su bili obujmljeni drugim dijelovima. Na primjer, gornji dio ruke je bio obuhvaćen poligonima rukava majice i skriven od pogleda iz svih kutova. Osim zbog uštede na broju vrhova, takve poligone je bilo potrebno ukloniti jer bi njihovo zaostajanje kasnije moglo prouzročiti probleme kod definiranja kostura i dodavanja animacija. Konačan izgled izrađenog modela prikazan je na slici Slika 8.



Slika 8. Konačan izgled modela

Osim prikazanog modela, u sklopu ovog završnog rada izrađen je još jedan model ljudskog lika dobiven snimanjem druge osobe. Modele sam izrađivao paralelno koristeći iste postupke uređivanja te sam nailazio na slične probleme koji su već ranije opisani, stoga izrada drugog modela neće biti detaljnije opisivana.

Pored dva modela ljudskih likova, uzorkovanjem pomoću Kinecta sam izradio još tri modela igračaka, koji će u nastavku također biti smješteni u isto virtualno okruženje kao i ljudski modeli. Jedan od modela je prikazan na sljedećoj slici.



Slika 9. Model igračke

Postupak snimanja predmeta je sličan snimanju ljudi. Koristio sam istu aplikaciju za 3D uzorkovanje i iste uvjete osvjetljenja. S obzirom da su igračke koje sam snimao razmjerno malih dimenzija i mase, odlučio sam dodatno prilagoditi metodu snimanja. Umjesto rotiranja objekta na pokretnoj platformi, zavezao sam igračku tankom vezicom i prije snimanja pustio da se okreće na vezici ispred Kinecta. Na ovaj način uspio sam snimiti i donju stranu objekta koja bi kod snimanja na rotirajućoj platformi bila skrivena kameri. Sama vezica nije predstavljala problem jer je dovoljno tanka tako da ju Kinect uopće nije ni registrirao. Dodatna prednost kod snimanja predmeta u odnosu na snimanje ljudi jest ta što su predmeti potpuno nepomični. Zbog toga modeli ni na jednom mjestu nisu bili deformirani. Također, snimljeni objekti su dovoljno malih dimenzija, pa je jedna snimka bila dovoljna da obuhvati cijeli objekt, što je u nastavku omogućilo uređivanje dobivenih modela uz minimalan napor.

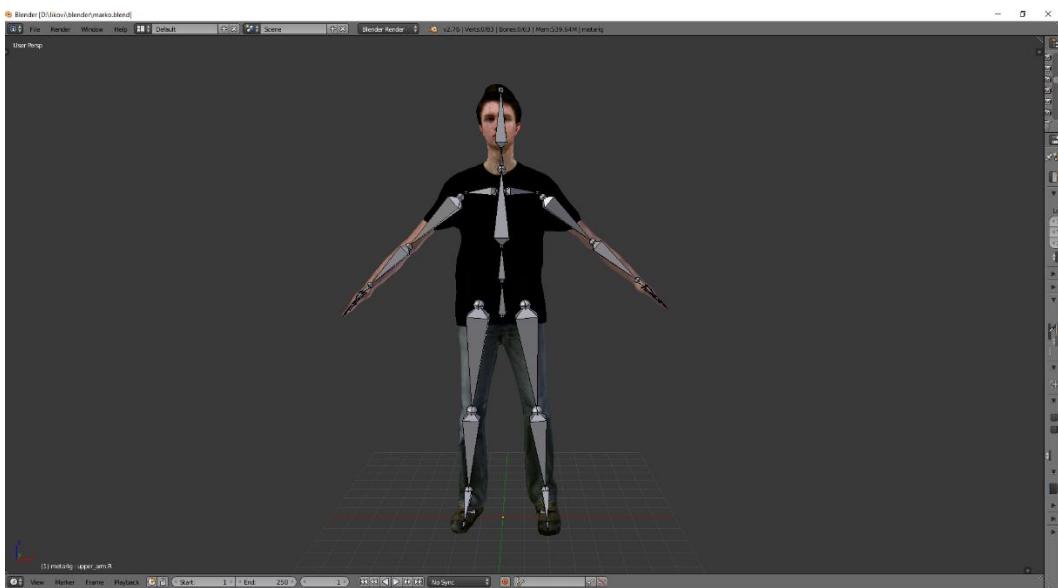
3.3 Definiranje kostura objekta

Kao rezultat uređivanja uzorkovanih podataka dobiveni su modeli s uređenim površinama i teksturama. Ti modeli su statičke mreže. Da bi se likovi mogli animirati, potrebno je provesti postupke definiranja kostura modela (engl. rigging) i težinskog vezanja površine modela i kostura (engl. skinning). Za oba postupka sam koristio Blender.

Definiranje kostura objekta je postupak određivanja skupa kostiju koji je pridružen mreži poligona. Svaka kost je povezana s točno određenim dijelom mreže i utječe samo na njegove transformacije. Kosti imaju svoju trodimenzionalnu transformaciju koja uključuje položaj, rotaciju i omjer, te u nekim slučajevima i roditeljsku kost. Uspostavljanje roditeljskih odnosa između kostiju omogućava stvaranje hijerarhije, pri čemu transformacije roditeljskih kostiju uzrokuju transformacije njima pridruženih poligona, i poligona pridruženih kostima niže u hijerarhiji. Tako na primjer kost nadlaktice utječe na ponašanje kosti podlaktice koja dalje utječe na ponašanje kostiju koje joj slijede sve do vrhova prstiju.

Blender omogućava izradu kostura na više načina. Prvi način je definiranje jedne po jedne kosti prema potrebi, što omogućava najveću fleksibilnost. Ova

metoda je korisna kod konstrukcije kostura za modele koji imaju specifičnu mrežu. Kako je animacija ljudskih likova česta potreba mnogih primjena, 3D aplikacije kao što je Blender omogućavaju izradu ljudskog kostura na mnogo brži način. Za svoje modele koristio sam opciju automatskog generiranja kostura, koja bi stvorila skup kostiju hijerarhijski povezanih na odgovarajući način koje odgovaraju ljudskom modelu. Stvoreni kostur u početku nije odgovarao veličinom i položajem mojem modelu, ali je poslužio kao dobra polazna točka. Najprije sam skalirao kostur tako da poprimi veličinu koja odgovara veličini modela. Nakon toga je bilo potrebno podesiti translaciju i rotaciju svake od kostiju tako da se podudari s odgovarajućim dijelom mreže. Lik sa svojim kosturom je prikazan na slici Slika 10.

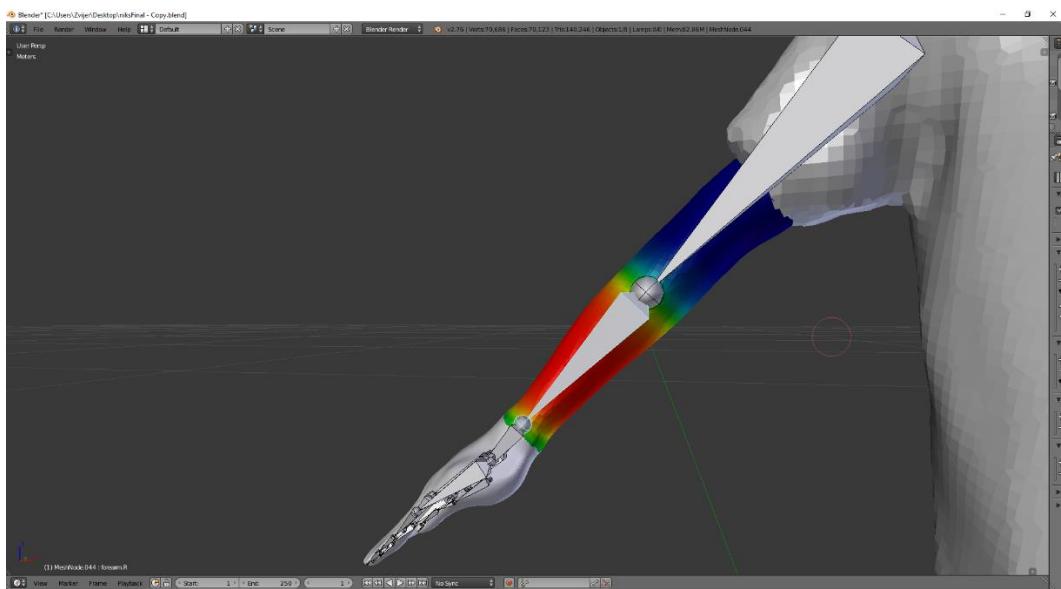


Slika 10. Lik s konstruiranim kosturom

3.4 Vezanje površine modela i kostura

Nakon postavljanja kostura može se prijeći na postupak težinskog vezanja površine modela i kostura. To je proces kojim se svaka kost povezuje s mrežom poligona koristeći težine koje predstavljaju mogućnost kosti da utječe na transformaciju mreže. Za dodjeljivanje težina koristio sam kombinaciju automatskog i ručnog dodjeljivanja, slično kao i za definiranje kostura. Automatsko dodjeljivanje težina određuje utjecaj kostiju na mrežu na temelju njihovih udaljenosti od pojedinih dijelova mreže. Zbog toga neke kosti, osim na svoje poligone, utječu i na poligone

koji pripadaju susjednim kostima, što je potrebno ručno ispraviti. Na primjer, kosti podlaktice u određenoj mjeri je pridijeljen utjecaj i na područje nadlaktice, koji ne smije postojati. Modificiranje ovih težina moguće je korigirati „bojanjem“ mreže koristeći Blenderovo grafičko sučelje. Boje koje su bliže crvenom dijelu spektra predstavljaju najjači utjecaj, a one bliže plavom dijelu predstavljaju slabiji ili nepostojeći utjecaj. Za spomenutu kost podlaktice potrebno je označiti poligone u području podlaktice crvenom bojom, jer je ta kost zadužena samo za te poligone. Poligoni koji odgovaraju području nadlaktice i šake su označeni plavom bojom, čime kost gubi utjecaj na ta područja. Prijelazna područja kao što su lakat i zapešće su označeni nijansama žute i zelene boje, a isto je učinjeno i za kosti nadlaktice i šake. Na taj način je postignuto da susjedne kosti imaju komplementaran utjecaj na prijelazna područja, što rezultira prirodnijim savijanjima ruke. Bez definiranja ovog prijelaznog područja, savijanje ruke u laktu rezultira prevelikim rastezanjem mreže u području lakta, zbog čega se formiraju oštiri rubovi koji ne odgovaraju stvarnoj situaciji. Utjecaj na mrežu nakon *skinninga* je prikazan za kost podlaktice na sljedećoj slici.



Slika 11. Prikaz utjecaja kosti podlaktice na mrežu

4. Integracija s virtualnim okruženjem

4.1 Izrada 3D aplikacije

Po završetku konstrukcije kostura, likovi su u potpunosti spremni za korištenje u virtualnom okruženju. Za kreiranje virtualnog okruženja koristio sam Unity, grafički pogon (engl. engine) za razvoj računalnih igara.

Pomoću Unityja sam izradio 3D računalnu igru u kojoj su korišteni izrađeni modeli. Tema igre je sakupljanje igračaka koje su skrivene u labirintu upravljanjem likom pomoću Kinecta, i pronašak izlaza iz labirinta. Na početku igre prikazana su dva lika koje sam prethodno izradio, i od igrača se očekuje da odabere lik s kojim želi igrati. Odabir se provodi podizanjem odgovarajuće ruke, nakon čega Kinect prepozna ruku koja je podignuta i igraču dodjeljuje odgovarajući lik. Po završetku odabira, lik se smješta u središte labirinta nakon čega, radi dinamičnosti igre, dobiva ograničenu količinu vremena da pronađe tri slučajno raspoređene igračke u labirintu. Brzina lika se kontrolira desnom rukom – visina ruke se očitava i na temelju nje se mijenja brzina igrača. Smjer igrača se kontrolira pomicanjem lijeve ruke. Zatvaranjem desne šake lik prestaje hodati, a zatvaranjem lijeve šake rotacija lika se zamrzava. Nakon što su otkrivene sve tri igračke, ili je vrijeme za potragu isteklo, igra prelazi u novu fazu u kojoj se na nasumičnom mjestu otvara izlaz iz labirinta, i u kojoj igrač mora pronaći taj izlaz također u ograničenom vremenu. Što je više igračaka pronađeno u fazi potrage, igrač će imati više vremena na raspolaganju za pronašak izlaza. Ako igrač pronađe izlaz u zadanim vremenima, igra završava uspjehom, a u suprotnosti završava neuspjehom. U bilo kojem trenutku se igra može ponovno pokrenuti pritiskom na tipku „R“, ili prekinuti rad aplikacije pritiskom na tipku „Q“.



Slika 12. Prikaz igre u tijeku

Prvi korak u izradi igre bio je definiranje topologije labirinta. U scenu sam dodao ravninu koja predstavlja tlo i određen broj kvadara koji predstavljaju zidove od živice. Odgovarajuću teksturu sam preuzeo s Interneta, i primijenio je na sve objekte. Nakon toga je bilo potrebno podesiti svojstva materijala, kao što je odsjaj, kako bi se dobio što stvarniji izgled. Nakon uređivanja labirinta, u projekt je bilo potrebno uvesti modele ljudi i igračaka. Za uvoz je korišten Blenderov vlastiti format .blend, koji je pogodan za korištenje u Unityju jer se sve promjene na uvezenoj .blend datoteci pojavljuju unutar projekta prilikom sljedećeg pokretanja, bez potrebe za ponovnim uvozom. Kod uvoza ljudskih modela bilo je potrebno naznačiti da se radi o humanoidnim modelima. Unityjev sustav za animaciju *Mecanim* detektirao je da struktura kostiju odgovara ljudskim likovima, što će kasnije omogućiti korištenje odgovarajućih animacija. Uvezeni modeli nemaju pridružene teksture, pa ih je potrebno zasebno uvesti. Tekture je zatim moguće pridružiti liku jednostavno metodom „drag and drop“ na odgovarajuće dijelove tijela, jer će Unity automatski provesti pridruživanje tekture koristeći UV mape izrađene u Blenderu.

Likove želimo animirati, a za to je najprije potrebno dodati im komponentu *Animator Controller*. Ta komponenta će određivati koja animacija će se odvijati u kojim uvjetima. Komponenta se ponaša kao konačan automat, kojemu sam definirao tri stanja i uvjete prijelaza među stanjima. Stanja su nazvana „Idle“, „Walk“

i „Run“ – svako stanje je aktivno ovisno o brzini lika koja je definirana skriptom, i sadrži odgovarajuću animaciju. S Unityjeve trgovine dodacima (engl. Asset Store) sam preuzeo niz animacija prilagođenih ljudskim likovima, i odabrao one koje su najbolje odgovarale mojim likovima. Animacije su uspješno pridružene likovima jer je Mecanim prethodno prepoznao njihov humanoidni kostur, i na taj način omogućio korištenje ovih animacija. Nakon ekstrakcije potrebnih dijelova odabralih animacija stajanja, hodanja i trčanja, svaka animacija je pridružena pripadnom stanju komponente *Animator Controller*. Time je posao izmjenjivanja animacija prepusten Unityju, i o njemu ne treba posebno voditi računa korištenjem skripti, jedino je potrebno sustavu periodički dojavljivati trenutačnu brzinu lika.

Za potrebe igre napisao sam nekoliko skripti u jeziku C#, koje će ukratko biti opisane u nastavku.

GameManager – glavna skripta koja upravlja tokom igre. Ovom skriptom definirana su stanja u kojoj se igra nalazi (kao što je odabir lika, potraga za igračkama itd.), vremenska ograničenja pojedinih stanja, i ispis o postignutim rezultatima i preostalom vremenu na zaslon. Skripta je zadužena za instanciranje odabranog lika na početku igre i vođenje računa o broju pronađenih predmeta.

PlayerController – skripta pridružena igraču koja upravlja njegovim kretanjem. Skripta koristi skriptu *BodySourceManager*, koja je preuzeta kao dio dodatka (engl. plugin) koji omogućava integraciju Kinecta i Unityja. Ta skripta dobavlja podatke o koordinatama i informaciju o zatvorenosti šaka skripti *PlayerController*, koja na temelju tih podataka regulira brzinu i rotaciju lika.

SpawnPointManager – slučajno odabire mjesta u labirintu na kojima će se instancirati igračke. Skripta sadrži reference na pet objekata postavljenih na različita mjesta u labirintu. Ti objekti su „prazni“, odnosno ne sadržavaju nijednu komponentu osim komponente *Transform*, u kojoj je pohranjena informacija o položaju objekta u labirintu. Na početku igre, skripta slučajno odabire tri od mogućih pet mjesta, i na njih postavlja modele igračaka.

Osim ovih skripti napisano je još nekoliko sporednih skripti za rotaciju igračaka kako ne bi bile potpuno statične, detekciju kolizije sa zidovima labirinta i dojavljivanje te informacije skripti koja upravlja likom i još neke manje važne akcije.

4.2 Performanse

Da bi se igra mogla pokrenuti, nužno je da računalo na kojemu se ona pokreće ispunjava zahtjeve koje nameće sam Kinect. Za Kinect v2 minimalno je potrebno računalo s dvojezgrenom procesorom frekvencije takta 3.1 GHz sa 64-bitnim instrukcijskim setom, USB 3.0 priključkom, 4 GB RAM memorije, grafičkom karticom kompatibilnom s DirectX 11 te operacijskim sustavom Windows 8 ili novijim. Igru sam testirao na prijenosnom računalu s procesorom Intel i7-4710HQ, grafičkom karticom NVIDIA GeForce GTX 850M, 4 GB RAM-a i operacijskim sustavom Windows 10.

Mjeranjem frekvencije osvježavanja dobio sam rezultat od stabilnih 60 okvira po sekundi. Ni u trenucima kad bi se u kadru našla oba animirana lika, ili animirani lik i neki od modela igračaka, iznos frekvencije ne bi pao ispod 60 okvira po sekundi. Razlog što je frekvencija osvježavanja ograničena na ovu vrijednost jest taj što Unity automatski ograničava osvježavanje jer većina monitora na tržištu ionako ne može postići veće frekvencije. Također, veća frekvencija u ovom slučaju nije ni potrebna jer je dobiveni prikaz prilično gladak, pa povećanje frekvencije osvježavanja ne bi doprinijelo vjernijem prikazu. Za primjene kod kojih je potrebna veća frekvencija, ovo Unityjevo ograničenje se može jednostavno isključiti.

Za svrhe testiranja isključio sam ograničenje od 60 okvira po sekundi i ponovio mjerenje. Najprije sam pokrenuo igru uz odabir prvog lika kojeg sam izradio, i mjeranjem utvrdio da se igra izvodi uz prosječnih 80 okvira u sekundi. U ovom slučaju su bila prisutna i povremena odstupanja koja su iznosila maksimalno do 10 okvira u sekundi. Pokretanje igre s drugim likom je dalo prilično različite rezultate – igra se u ovom slučaju izvodila s čak 125 okvira po sekundi u prosjeku. Razlika između ova dva slučaja je izuzetno velika, a razlog tome je razlika u broju vrhova među likovima. Prvi izrađeni lik se sastoji od oko 70 000 vrhova, dok je smanjenje broja vrhova kod drugog lika provedeno na bolji način – on se sastoji od oko 12 000 vrhova. Ovaj primjer pokazuje koliko je važno što je moguće više rasteretiti grafički procesor, te da je dodatak novih objekata u postojeći projekt moguć uz odgovarajuću „štедnju“ na broju vrhova.

5. Zaključak

U ovom radu opisane su osnovne mogućnosti uporabe uređaja Microsoft Kinect u svrhu uzorkovanja trodimenzijskih površina, a detaljnije su opisane mogućnosti uzorkovanja ljudskih likova s ciljem izrade ljudskih modela. Modeliranje korištenjem podataka dobivenih uzorkovanjem objekata se pokazalo kao metoda koja omogućava brzu izradu modela koji se potom mogu koristiti u različitim virtualnim okruženjima.

3D kamere cijenom pristupačne širokom spektru korisnika, kao što je Kinect, danas još uvijek nisu u mogućnosti producirati rezultate koje bi mogao proizvesti profesionalni modelar, ali svakako imaju svoje prednosti. Kinect se tako može koristiti za primjene kod kojih apsolutna preciznost nije od presudne važnosti, ili primjerice za brzu izradu prototipova igara. Premda ne može u potpunosti nadomjestiti vještina profesionalca, Kinect svakako može poslužiti kao dobra početna točka za izradu modela.

6. Literatura

[1] Kinect Fusion, <https://msdn.microsoft.com/en-us/library/dn188670.aspx>

[2] Kinect for Windows Sensor,

<https://msdn.microsoft.com/en-us/library/hh855355.aspx>

[3] Unity Manual, <http://docs.unity3d.com/Manual/>

[4] Blender Reference Manual, <https://www.blender.org/manual/>

7. Sažetak

U ovom radu opisane su osnovne mogućnosti izrade ljudskih likova temeljenih na podacima dobivenim uzorkovanjem pomoću uređaja Microsoft Kinect. U prvom poglavlju opisane su prednosti uzorkovanja ovom metodom. U drugom poglavlju su opisane tehničke značajke uređaja. Treće poglavlje se bavi postupcima uzorkovanja i obrade uzorkovanih podataka trodimenzijskih objekata. U posljednjem poglavlju opisana je izrada virtualnog okruženja u koje su uklopljeni izrađeni ljudski modeli.

Ključne riječi: 3D uzorkovanje, modeliranje, skinning, rigging, Kinect, Blender, Unity

7. Abstract

This paper describes the basic features of making human figures based on data obtained by sampling using a Microsoft Kinect device. The first chapter outlines the advantages of this method of sampling. The second chapter describes the device technical specifications. The third chapter deals with the sampling procedures and processing of sampled data of three-dimensional objects. The last chapter describes the development of the virtual environment that incorporates developed characters.

Keywords: 3D sampling, modeling, skinning, rigging, Kinect, Blender, Unity