Automatizirano generiranje 3D modela ljudi

Orić, Mihaela

Master's thesis / Diplomski rad

2023

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:200:153635

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-05-11**

Repository / Repozitorij:

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek

SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA OSIJEK

Sveučilišni studij

AUTOMATIZIRANO GENERIRANJE 3D MODELA LJUDI

Diplomski rad

Mihaela Orić

Osijek, 2023.

SADRŽAJ

1. UVOD 1

1.1. Zadatak diplomskog rada 1

2. PREGLED PODRUČJA 2

2.1. Sintetički podaci za treniranje 3

3. KORIŠTENE TEHNOLOGIJE 5

3.1. Blender 5

3.2. MakeHuman 6

3.3. Bpycv 7

3.4. Mixamo 8

4. DIZAJN I IMPLEMENTACIJA SUSTAVA ZA AUTOMATSKO GENERIRANJE 3D MODELA LJUDI 9

4.1. Grafičko sučelje 10

4.2. MakeHuman izrada modela čovjeka 12

4.3. Uređivanje izrađenog modela 15

4.4. Pomicanje postojećeg modela i raznolika pozadina 19

4.4.1. Mijenjanje poze modela 19

4.4.2. Mijenjanje pozadine 21

4.5. Dobivanje anotacija 21

4.5.1. Značajke na tijelu 22

4.5.2. Semantička segmentacija 23

4.5.3. Dubinska slika 24

5. TESTIRANJE 26

6. ZAKLJUČAK 31

LITERATURA 32

SAŽETAK 33

ABSTRACT 34

ŽIVOTOPIS 35

1. UVOD

Jedan od glavnih izazova strojnog učenja je pronalazak i priprema dovoljno velikog seta anotiranih podataka na kojemu će se vršiti treniranje. Sve češći pristup rješavanju toga problema je korištenje sintetičkih podataka. Sintetički podaci mogu ubrzati cijeli proces izrade modela jer se anotacija može automatizirati pri generiranju te tako izbjeći korak ručnog anotiranja podataka. Osim toga, korisniku se daje fleksibilnost kod odabira veličine seta podataka.

Rekreiranje vizualnih podataka može se postići korištenjem 3D modeliranja. U ovom radu za 3D modeliranje koristi se alat Blender [1] koji u sebi ima ugrađeni Python prevoditelj. Ugrađeni prevoditelj daje mogućnost automatiziranje cijelog rada u Blenderu. Sustav predstavljen u nastavku ovog rada generira 3D modele ljudi te sprema slike tih ljudi u različitim pozama i iz različitih perspektiva s anotacijama. Uz program Blender koriste se i javno dostupni resursi preuzeti sa stranica PolyHaven [2].

Generiranje podataka pokreće se putem grafičkog sučelja izrađenog pomoću Qt alata [3]. Korisnik u tome sučelju može odabrati veličinu seta podataka koji želi generirati, vrste anotacija te raznolikost generiranih modela ljudi.

U nastavku rada pregledat će se slična programska rješenja, objasniti primjena sintetičkih podataka za treniranje i motivacija izrade ovog rada, koje probleme rješava te objasniti korištene tehnologije. Opisat će se dizajn predstavljenog sustava, implementacija i testiranje sustava.

1.1. Zadatak diplomskog rada

Zadatak diplomskog rada je izraditi sustav za automatizirano generiranje 3D modela ljudi koji će se koristiti za generiranje sintetičkih podataka za treniranje modela za strojno učenje. Taj sustav pružat će korisniku više opcija pri generiranju podataka.

2. PREGLED PODRUČJA

Korištenje sintetičkih podataka za treniranje često je zastupljeno u današnje vrijeme, a to potvrđuje i činjenica da je Microsoft uložio resurse u izradu sintetičkog seta podataka koji sadrži raznolike slike lica s anotiranim značajkama na licu te semantičku segmentaciju dijelova lica [4]. Microsoft je u ovom radu koristio parametrizirano generiranje 3D modela lica. Osnovni modeli ljudske glave dobiveni su skeniranjem različitih stvarnih osoba i fotogrametrijom. Mijenjanjem parametara lica, od tih osnovnih modela moguće je dobiti beskonačan broj različitih lica. Ostali dijelovi lica poput kose, očiju i usta, također su dobiveni fotogrametrijom. Umjetnici za modeliranje napravili su dodatnu obradu modela dobivenih fotogrametrijom kako bi se mogli primijeniti na nove 3D modele. Ta obrada uključivala je izdvajanje teksture lica, uklanjanje bora, izraza lica, različitih modnih dodataka i slično. Kreirani set podataka je javno dostupan za nekomercijalno korištenje. Na slici 2.1. vidljiva je semantička segmentacija slika iz seta podataka Face Synthetics.

Slika 2.1. *Semantička segmentacija Microsoftovog Face Syntethics seta podataka* [4].

Synthesis AI [5] bavi se komercijalnom proizvodnjom sintetičkih podataka. Nude foto realistične fotografije i videozapise koji su nastali uz pomoć generativne umjetne inteligencije, proceduralnog 3D modeliranja i filmskih sustava za iscrtavanje (engl. *Rendering*), Korisnik može odabrati želi li anotirati segmentacijske mape, 2D/3D značajke, dubinske mape, normale površina ili nešto drugo, a primjeri anotacija vidljivi su na slici 2.2.

Slika 2.2. *Različite anotacije koje pruža Synthesis AI* [5].

Oni predlažu korištenje njihovih podataka za razvijanje modela u raznim primjenama. Neke od njih uključuju izradu avatara, nadziranje vozača u automobilskoj industrija, detekcija pješaka, sigurnost i virtualno isprobavanje proizvoda.

2.1. Sintetički podaci za treniranje

Uspješnost modela dobivenih strojnim učenjem ovisi o kvaliteti podataka koje taj model uzima pri treniranju i testiranju. Malen ili loše anotiran skup podataka ne može rezultirati odličnim modelom. Da bi podaci bili dovoljno dobri, potrebno je uložiti puno vremena u prikupljanje, anotiranje i provjeru svakog uzorka kako bi se potvrdilo da skup ne sadrži anomalije koje će utjecati na kvalitetu. To je iscrpljujući posao, ali je nekada bez obzira na vrijeme uloženo, nemoguće dobiti skup podataka koji se zahtjeva. Ako je primjena modela detektiranje opasnih radnji na radnome mjestu, podaci bi trebali sadržavati primjere gdje je radnik ozlijeđen na radnome mjestu zbog svojih radnji, a to bi bilo nemoguće reproducirati u svrhu prikupljanja podataka.

U nekim slučajevima prikupljanje podataka ne samo da je iscrpljujuće i zahtjevno, nego podaci mogu sadržavati informacije koje bi se trebale zaštititi. Te informacije su ljudska lica, identitet, registracijske oznake na vozilima i slično. A ako takvi setovi podataka već postoje, često su zaštićeni licencama koje posjeduju tvrtke.

Navedene probleme rješavaju umjetno generirani podaci koji nastaju računalnim putem. Modeliraju se objekti, okruženja i ljudi te reproduciraju različite radnje. Kvaliteta modela treniranog na sintetičkim podacima u nekim slučajevima može biti bolja od onih treniranih na stvarnim podacima. Razlog tomu je fleksibilnost koja se daje pri kreiranju podataka [6].

U težim slučajevima, kada je stvarne podatke teško reproducirati sintetičkim putem, proces generiranja podataka može biti mukotrpan. Nekada se koriste simulatori fizike kako bi se postiglo što realističnije pozicioniranje sintetičkih objekata te napredni strojevi za iscrtavanje da se postigne foto realističnost. Ako je nemoguće generirati dovoljno realistične sintetičke podatke, moguće je kombinirati stvarne i sintetičke podatke za treniranje.

S. Hinterstoisser u svome radu [7] iznosi način na koji je moguće trenirati model na sintetičkim podacima bez da se pokvare performanse modela na stvarnim podacima. Potrebno je „zamrznuti“ slojeve modela koji su zaduženi za izvlačenje značajki i koji su trenirani na stvarnim podacima. Potom trenirati ostale slojeve na sintetičkim podacima. Modeli trenirani na ovaj način imaju veću preciznost od modela treniranih isključivo na stvarnim ili isključivo na sintetičkim podacima.

Kao što je ranije navedeno, kvalitetni sintetički setovi podataka su često licencirani i nisu dostupni na besplatno korištenje. To je potaklo na izradu ovoga rada koji će omogućiti kreiranje slika ljudi za različite primjene.

3. KORIŠTENE TEHNOLOGIJE

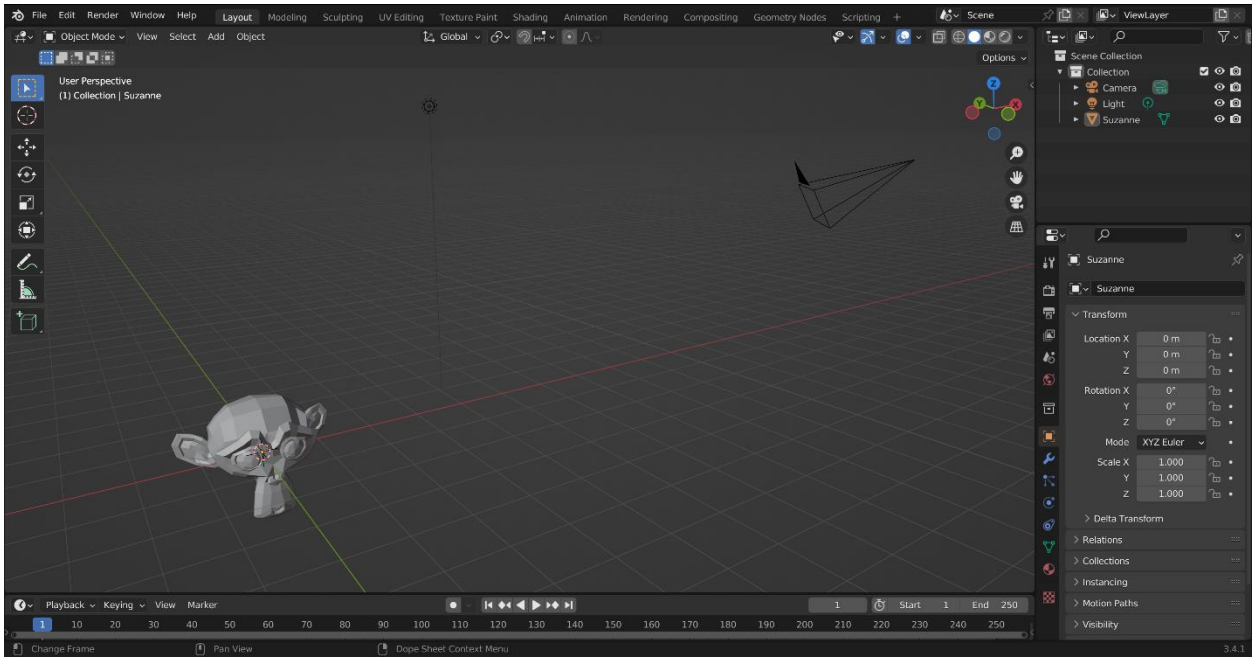
U izradi ovoga rada korišteno je 3D modeliranje i alati u kojima je moguće 3D modelirati. 3D modeliranje je proces kreiranja trodimenzionalne reprezentacije objekta ili površine. Tehnologije koje su korištene opisane su u nastavku.

3.1. Blender

Blender je programski alat u kojemu je moguće raditi poligonsko 3D modeliranje i animacije. Nudi velik raspon korisnih mogućnosti poput izrade realističnih 3D modela na brz i jednostavan način, digitalno kiparstvo pomoću više od 20 različitih vrsta kistova, simulacija fizike (u koju pripada simuliranje tekućine, kose, dima, odjeće, i slično), uređivanje animacija, uređivanje fotografija i videa. Zajednica korisnika Blendera je mnogobrojna i aktivna i moguće je koristiti proširenja koja su kreirali članovi zajednice ili podijeliti vlastita proširenja.

Besplatan je i izvorni kod mu je otvoren te ga to čini jako popularnim alatom za korištenje u hobijima ili kod honorarnog rada, ali također se često rabi i u profesionalne svrhe.

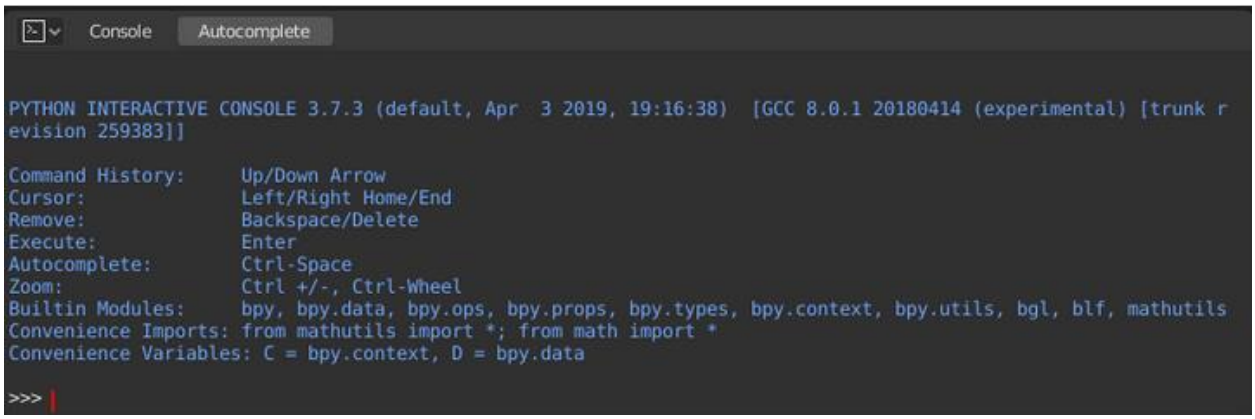
Moguće ga je besplatno preuzeti na službenim stranicama [1]. Početna scena Blender projekta prikazana je na slici 4.1.



Slika 4.1. *Početni zaslon koji se otvara pri kreiranju novog projekta u Blenderu.*

Blender u sebi ima ugrađeni Python prevoditelj koji se učitava pri pokretanju i ostaje aktivan sve dok se koristi program. Python konzola u Blenderu prikazana je na slici 4.2. U prozoru za pisanje

skripti, nakon uvođenja biblioteke „bpy“, sve značajke koje Blender nudi u grafičkom sučelju postaju dostupne i putem Python skripti. Na taj način, moguće je pristupiti svim klasama i funkcijama Blendera te svako izvršavanje Python programa mijenja scenu vidljivu u prozoru za 3D prikaz [8]. Ovo omogućuje pokretanje programa u pozadinskom načinu putem naredbenog retka bez da se troše resursi na vizualizaciju.



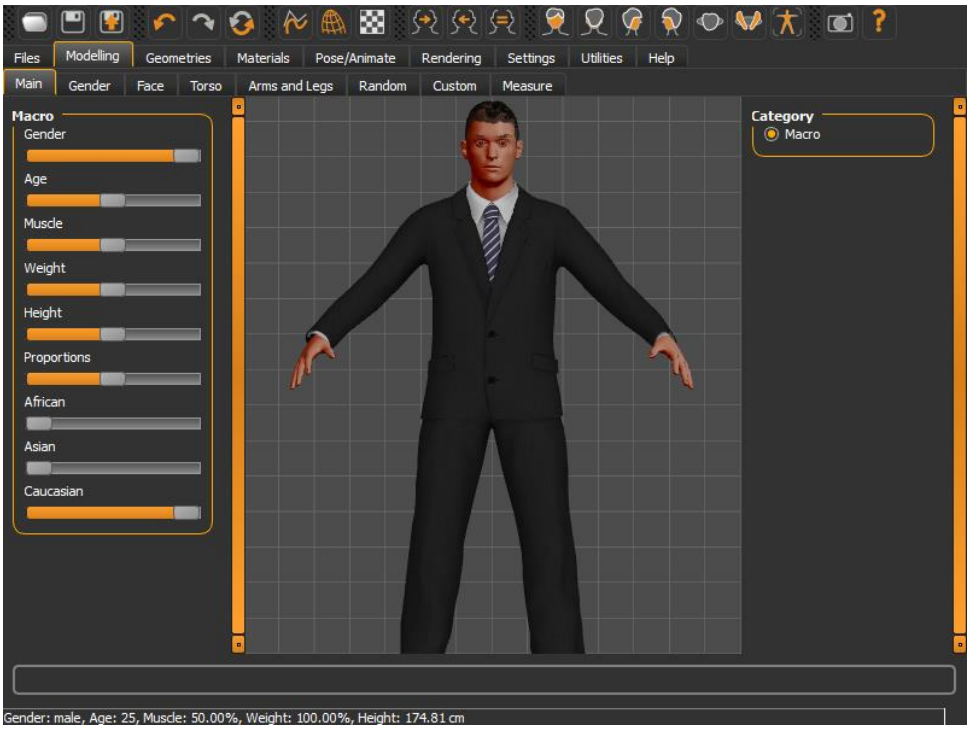
Slika 4.2. *Python konzola u Blenderu.*

Mogućnost pokretanja u pozadini te pisanje Python skripti glavni su razlozi odabira Blendera u izradi ovog rada. Cijeli proces pokretanja sustava, modeliranja i iscrtavanja je automatiziran.

3.2. MakeHuman

MakeHuman [9] je alat otvorenog koda koji je dizajniran da bi pojednostavio kreiranje virtualnih ljudi putem grafičkog sučelja. Glavni cilj je mogućnost kreiranja velikog broja realističnih virtualnih ljudi te iscrtavanje i izvoz u samo nekoliko koraka. Virtualnim ljudima manipulira se putem dvije vrste atributa: makro i detalji. Makro atributi kontroliraju dob, visinu, spol, masu i etničku pripadnost. Detalji upravljaju svim manji pojedinostima koji utječu na izgled, sve od oblika očiju, nosa, ušiju, do duljine prstiju. Izgled programa MakeHuman prikazan je na slici 4.3.

Osim što postoji kao zaseban program, MakeHuman postoji u verziji Blender proširenja. Putem API-ja moguće je upravljati kreiranjem ljudi Pythonom.



Slika 4.3. *Sučelje MakeHuman programa* [9].

3.3. Bpycv

Bpycv [10] je alat koji pomaže u računalnom vidu i dubokom učenju kroz Blender. Značajke koje bpycv nudi su iscrtavanje anotacija putem segmentacije, uključujući semantičku, po instancama i panoptičku segmentaciju, iscrtavanje dubinskih slika, generiranje poze te generiranje nasumične scene. Za lakše instaliranje, bpycv dostupan je kao docker datoteka. Primjer iscrtanih slika koje daje bpycv prikazan je na slici 4.4.

Slika 4.4. *Slika koju iscrtava bpycv segmentacija po instancama ,RBG slika i dubinska slika* [10].

3.4. Mixamo

Mixamo [11] je servis za animiranje 3D likova. Internetska baza nudi besplatan pristup likovima i animacijama uz alat za automatsko dodavanje i postavljanje kostura za animiranje na likove. Modeli likova koji se nude optimizirani su za korištenje u računalnim igrama te imaju dodane kosture, što znači da su spremni za animiranje. Ti modeli su kvalitetno izrađeni i korisni su za početnike koji razvijaju svoje igre. Također su korisni i naprednijim razvojnim inženjerima koji žele iskoristiti Mixamo modele privremeno, kao zamjenu za modele koje će razviti sami u budućnosti. Iako je baza likova velika, javno je dostupna besplatno svim korisnicima te stoga nije praktična za komercijalnu uporabu. Poput baze likova, javno je dostupna i velika baza animacija raznih ljudskih kretnji, poput ljudskog hoda, trčanja, borbe, plesa, i slično. Osim pružanja gotovih modela likova, na Mixamo servis moguće je prenijeti vlastite likove i korištenjem alata automatski dodati kostur koji se povezuje s modelom i omogućuje animiranje pomicanjem dijelova toga kostura. Modeli su nakon izvoza spremni za korištenje u programima za 3D modeliranje [12]. Postoje i programska proširenja za Blender koja ubrzavaju proces uvoza, izvoza i animiranja. Ta proširenja također omogućuju korištenje Mixamo alata putem programskog koda u Blender Python prevoditelju.

Adobe, američka multinacionalna kompanija za razvoj računalnih sustava, vlasnik je Mixamo servisa. Činjenica da je servis u vlasništvu tako velike kompanije poznate diljem svijeta pridonosi redovitom održavanju stranica te konstantnom povećanju baze animacija i likova.

4. DIZAJN I IMPLEMENTACIJA SUSTAVA ZA AUTOMATSKO GENERIRANJE 3D MODELA LJUDI

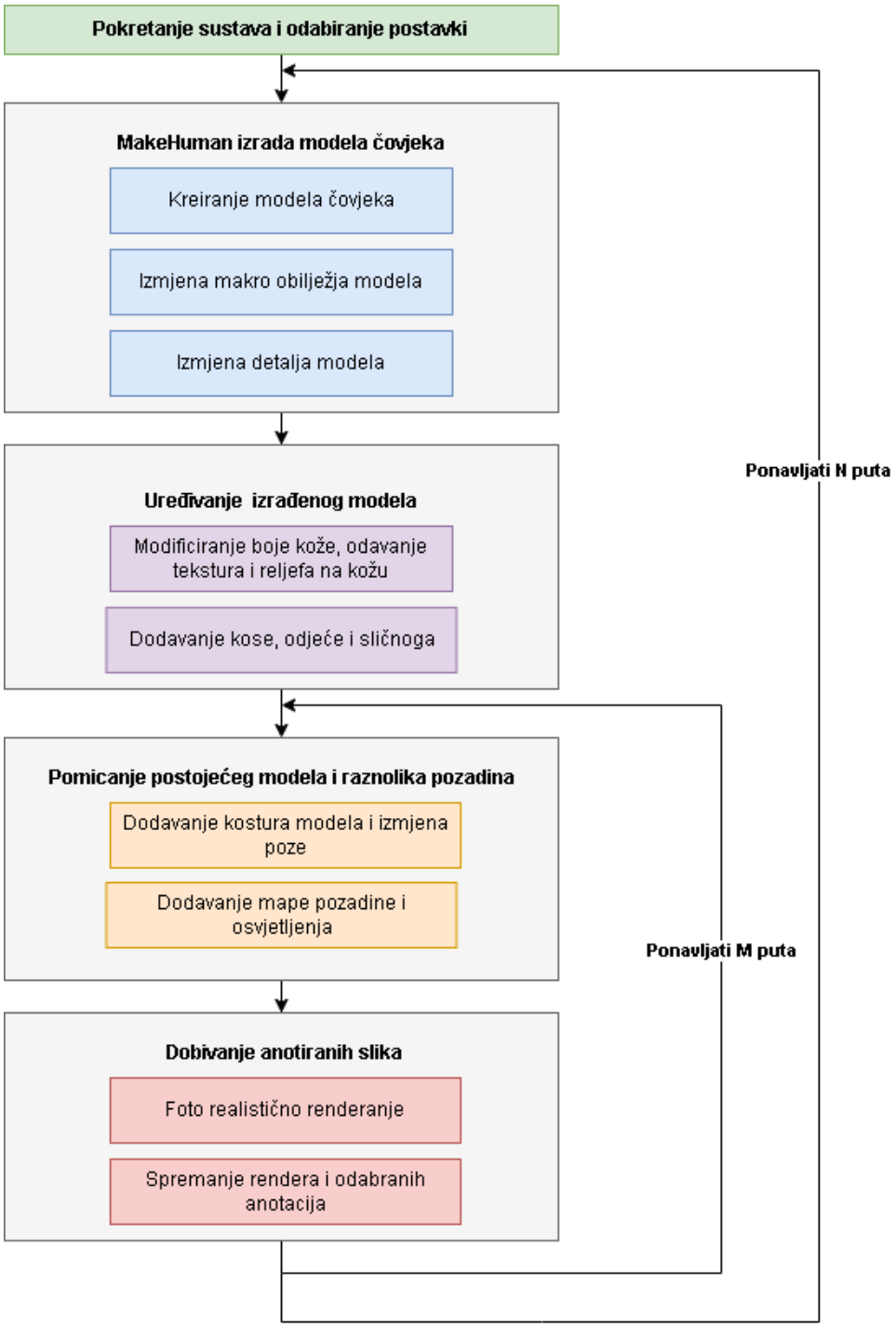
Ideja sustava je korisniku pružiti grafičko sučelje putem kojega će se pokretati generiranje modela ljudi. U sučelju korisnik može odabrati željene anotacija, raznolikost i veličinu skupa podataka. Ostatak pokretanja generiranja je automatiziran i sakriven od korisnika kako bi pokretanje bilo što jednostavnije.
Sljedeći korak je izrađivanje modela čovjeka putem MakeHuman Blender proširenja. Modelu čovjeka mijenjaju se makro atributi poput visine, spola i dobi. Nakon makro atributa, mijenjaju se detalji na modelu.

Poslije toga slijedi dodatna modifikacija koja nije isključivo vezana za MakeHuman proširenje. Iako MakeHuman zajednica pruža velik broj gotovih tekstura kože, većina njih nije dovoljno realistično izvedena. Također, nije moguće mijenjati boju tih tekstura kože te se time gubi jedna vrsta anotacije. Osim toga, tako bi raznolikost kreiranih ljudi bila manja jer bi svi imali isti ton kože. Ovim bi korakom završila sama izrada modela jednog čovjeka.

Nakon izrade modela, dodaje mu se kostur te se modificira poza. Dodavanjem HDRI (High Dynamic Range Image) mapa na scenu, lako se postiže realistična pozadina i osvjetljenje. Mijenjanjem tih mapa povećava se raznolikost kreiranih slika. Zadnji korak je iscrtavanje slike i izvlačenje anotacija odabranih u grafičkom sučelju. Ovisno o odabranom broju različitih slika koje se žele kreirati za jedan model čovjeka, u petlji se vraća na korak izmjene kreiranog modela.

Isto tako, ovisno o broj različitih modela ljudi, u petlji se vraća na korak izrade novog modela čovjeka. Shema dizajna predloženog sustava prikazana je na slici 5.1.

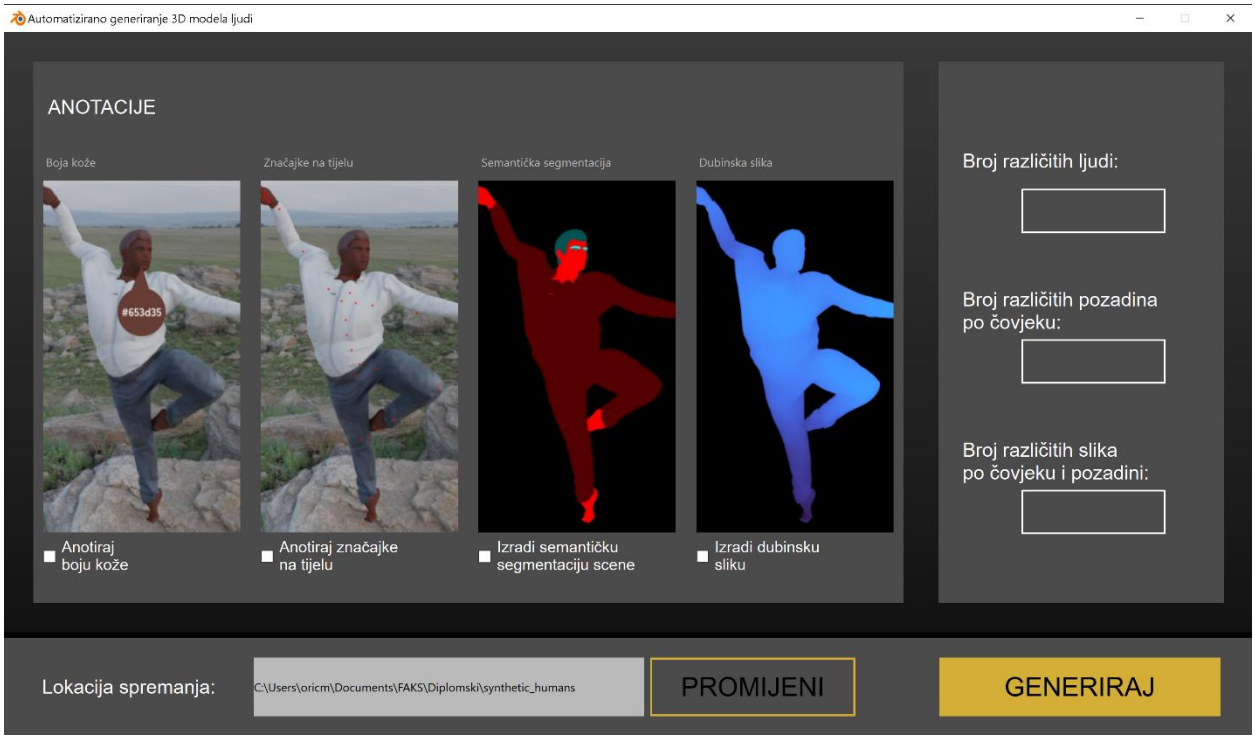
Za svaki model čovjeka sprema se Blender datoteka u slučaju da korisnik želi dodatno iskoristiti model.



Slika 5.1. *Dizajn sustava.*

4.1. Grafičko sučelje

Pozadina sustava i tijek izrade modela skriveni su od korisnika. Njemu se pruža grafičko sučelje putem kojega se pokreće generiranje modela ljudi. U sučelju je moguće odrediti veličinu i raznolikost skupa podataka koji se kreira. Izgled grafičkog sučelja prikazan je na slici 5.2.



Slika 5.2. *Izgled grafičkog sučelja za sustav automatiziranog generiranja 3D modela ljudi.*

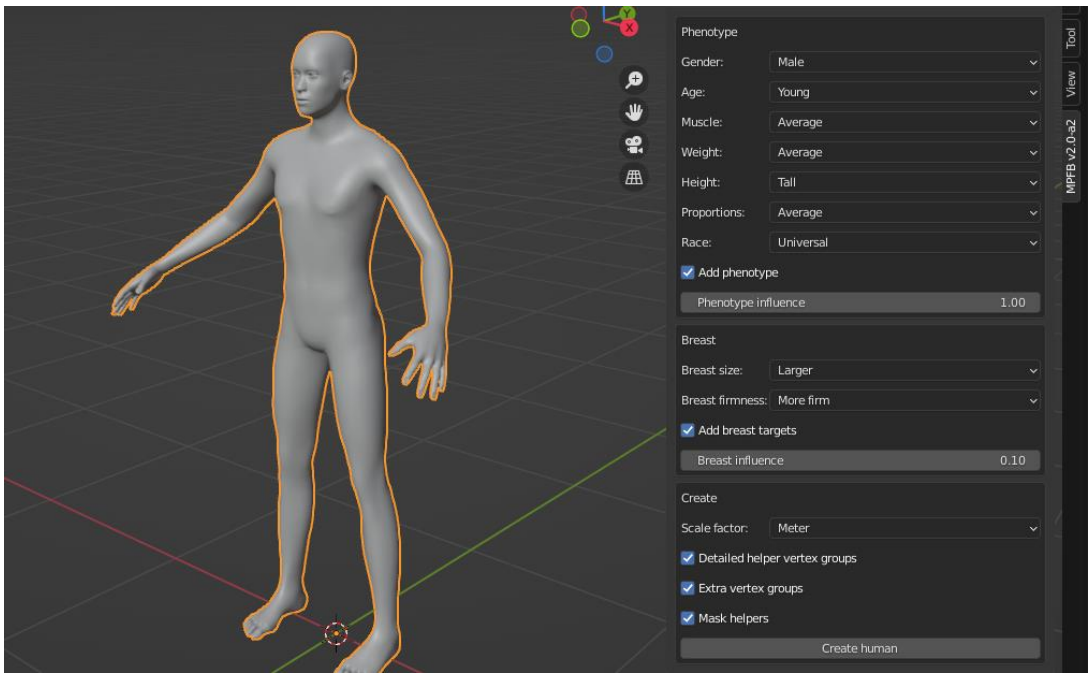
Grafičko sučelje izrađeno je korištenjem Qt alata [3]. Qt je besplatna platforma otvorenog koda koja služi za kreiranje grafičkih korisničkih sučelja i aplikacija koje se pokreću na više sustava (npr. Linux i Windows). Nudi vizualni uređivač koji ima spremne sve komponente. Potrebno ih je samo povlačenjem dovesti na željeno mjesto na prozoru. To omogućuje dobru integraciju između cijelog razvojnog tima jer ga mogu koristiti dizajneri i programeri. Jednostavan je za korištenje, ali ga to ne sprječava u izradi kompliciranih i lijepih sučelja. U ovom radu korišten je PySide6 [13] Python modul za rad s Qt alatima.

Izrađeno grafičko sučelje sastoji se od tri glavna dijela. Prvi dio prozora sučelja je prostor u kojemu korisnik odabire vrste anotacija koje želi dobiti. Polja za odabir anotacija u prozor su smještena korištenjem rasporeda u obliku koordinatne mreže (engl. *Grid layout*). Funkcionalnost odabira pojedinog polja postignuta je elementom potvrdnog okvira (engl. *Checkbox*). Desno od prostora za odabir anotacija nalaze se tri polja za unos podataka koji opisuju veličinu skupa podataka koji se kreira. Skup podataka u ovom sustavu određen je brojem različitih modela ljudi koji će se kreirati, brojem različitih scena u koje će se svaki model čovjeka smjestiti te brojem iscrtanih slika iz različitih kutova za svaku scenu. Izvršava se provjera unesenih podataka za veličinu skupa podataka i u slučaju da su unesene vrijednosti koje nisu prirodni brojevi, izbacuje se poruka da je unesena neispravna vrijednost. Na dnu prozora nalazi se polje za odabir lokacije spremanja kreiranog skupa podataka. Pritiskom na gumb „Promijeni“ otvara se preglednik datoteka kako bi

se lakše odabrala lokacija spremanja. Gumb „Generiraj“ započinje proces kreiranja skupa podataka. Kako bi se izbjegla situacija da korisnik ne zna je li se proces kreiranja uspješno pokrenuo u pozadini te tako dođe u iskušenje da više puta pritisne gumb za kreiranje, mijenja se poruka na gumbu za kreiranje te se onemogućuju svi ostali gumbovi u grafičkom sučelju. Da bi se postigla opisana funkcionalnost, potrebno je koristiti više-nitno procesiranje. U novoj niti poziva se funkcija za kreiranje podataka i na taj način procesiranje neće blokirati izvođenje grafičkog dijela i mijenjanje stanja gumbova.

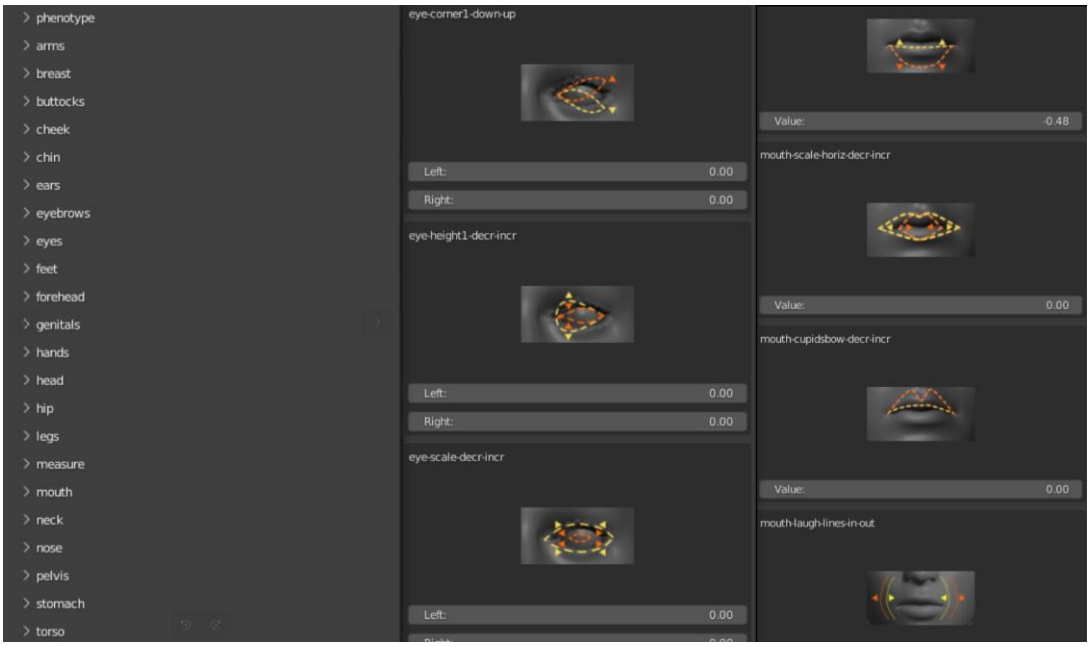
4.2. MakeHuman izrada modela čovjeka

Dodavanje proširenja u Blenderu je jednostavno. Potrebno je preuzeti MakeHuman komprimiranu datoteku s njihovih stranica te u Blenderu otići na Edit →Preferences → AddOns → Install. Nakon instaliranja, važno je isto to proširenje u tome prozoru označiti i omogućiti. Nakon što je dodano proširenje MakeHuman, moguće je putem sučelja kreirati model čovjeka. Kod kreiranja nude se opcije da se odabere spol modela, dob, količina mišića, masa, kakve su proporcije te rasa. Odabir rase i učinak fenotipa određuju koliko će jako rasna obilježja biti vidljiva. Sučelje kreiranja modela čovjeka vidljivo je na slici 5.3.



Slika 5.3. *Kreiranje modela čovjeka putem MakeHuman proširenja u Blenderu.*

Detalji na modelu koji se mogu mijenjati prikazani su na slici 5.4. Svakom od detalja moguće je promijeniti veličinu, poziciju, rotaciju i slično.



Slika 5.4. *Mijenjanje detalja modela.*

Kod za kreiranje osnovnog modela čovjeka prikazan je na slici 5.5.

Linija ***Kod***

```
1:      /*Kreiranje modela čovjeka */
2:      def createNewHuman():
3:          global skinIndex, skinValue, middleSkinValue, gender, age
4:          skinIndex = random.randint(0,len(skinToneScale)-1)
5:          skinValue = skinToneScale[skinIndex]
6:          middleSkinValue = middleTonesScale[skinIndex][random.randint(0,
        len(middleTonesScale[skinIndex])-1)]
7:          gender = genders[random.randint(0,len(genders)-1)]
8:          age = ages[random.randint(0,len(ages)-1)]
9:          muscle = muscles[random.randint(0,len(muscles)-1)]
10:         weight = weights[random.randint(0,len(weights)-1)]
11:         height = heights[random.randint(0,len(heights)-1)]
12:         proportion = proportions[random.randint(0,len(proportions)-1)]
13:         race = races[random.randint(0,len(races)-1)]
14:         bpy.context.scene.MPFB_NH_phenotype_gender = gender
15:         bpy.context.scene.MPFB_NH_phenotype_age = age
16:         bpy.context.scene.MPFB_NH_phenotype_muscle = muscle
17:         bpy.context.scene.MPFB_NH_phenotype_weight = weight
18:         bpy.context.scene.MPFB_NH_phenotype_height = height
19:         bpy.context.scene.MPFB_NH_phenotype_proportions = proportion
20:         bpy.context.scene.MPFB_NH_phenotype_race = race
21:         bpy.context.scene.MPFB_NH_phenotype_influence = random.uni-
        form(0,1)
22:         if(race == 'caucasian'):
```

```
23:             skinIndex = random.randint(0,len(skinToneScaleCaucasian)-1)
24:             skinIndex = random.randint(0,len(skinToneScaleCaucasian)-1)
25:             skinValue = skinToneScaleCaucasian[skinIndex]
26:             middleSkinValue = middleTonesScaleCaucasian[skinIndex][ran-
    dom.randint(0, len(middleTonesScaleCaucasian[skinIndex])-1)]
27:         elif(race == 'asian'):
28:             skinIndex = random.randint(0,len(skinToneScaleAsian)-1)
29:             skinValue = skinToneScaleAsian[skinIndex]
30:             middleSkinValue = middleTonesScaleAsian[skinIndex][ran-
    dom.randint(0, len(middleTonesScaleAsian[skinIndex])-1)]
31:         elif(race == 'african'):
32:             skinIndex = random.randint(0,len(skinToneScaleAfrican)-1)
33:             skinValue = skinToneScaleAfrican[skinIndex]
34:             middleSkinValue = middleTonesScaleAfrican[skinIndex][ran-
    dom.randint(0, len(middleTonesScaleAfrican[skinIndex])-1)]
35:         bpy.ops.mpfb.create_human()
```

Slika 5.5. *Kod za kreiranje modela čovjeka putem MakeHuman proširenja.*

U datoteci s postavkama za kreiranje podataka zapisani su podaci poput:

- Rezolucije slike
- Spola i dobi ljudi koji se generiraju (koristi se u kodu prikazanom iznad)
- Raspona veličina dijelova tijela (svaki dio tijela može se smanjiti ili povećati u odnosu na zadane vrijednosti).

Svi koraci i odabiri prikazani u sučelju na jednak način se odabiru putem klasa u Pythonu uzimajući u obzir vrijednosti navedene u datoteci s postavkama.

Slika 5.6. prikazuje programski kod koji mijenja oblik i veličinu nosa.

Linija ***Kod***

```
1:      /*Kod za uređivanje nosa  */
2:      def editNose():
3:          bpy.context.scene.nose_nose_septumangle_decr_incr = random.uni-
    form(-rangeNose, rangeNose)
4:          bpy.context.scene.nose_nose_width1_decr_incr = random.uniform(-
    rangeNose, rangeNose)
5:          bpy.context.scene.nose_nose_trans_backward_forward = random.uni-
    form(-rangeNose, rangeNose)
6:          bpy.context.scene.nose_nose_trans_down_up = random.uniform(-ran-
    geNose, rangeNose)
7:          bpy.context.scene.nose_nose_nostrils_width_decr_incr = random.u-
    niform(-rangeNose, rangeNose)
```

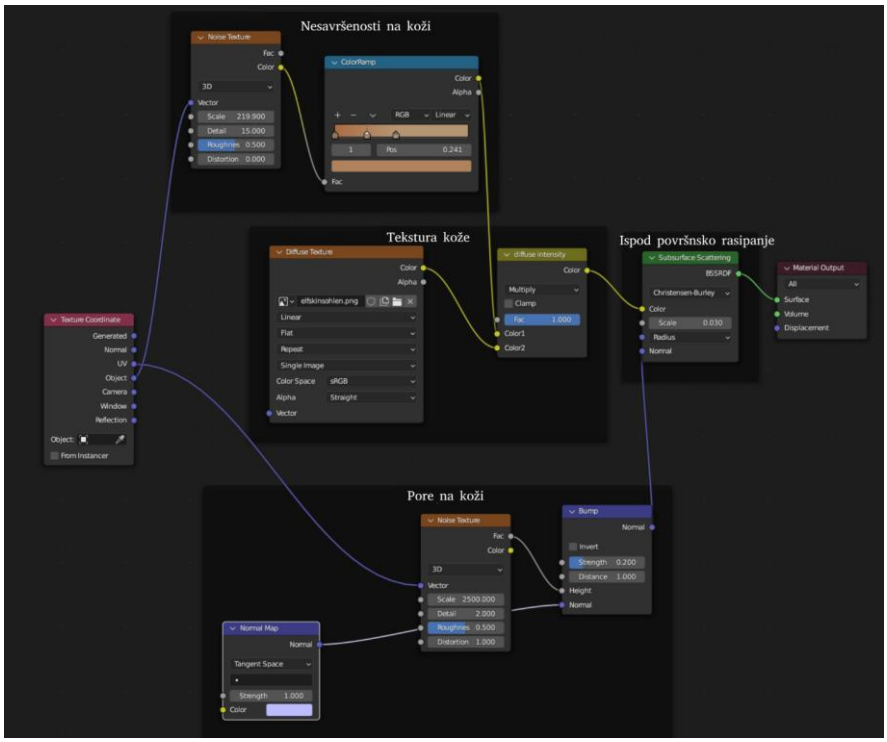
```
8:          bpy.context.scene.nose_nose_greek_decr_incr =random.uniform(-
        rangeNose, rangeNose)
9:          bpy.context.scene.nose_nose_nostrils_angle_down_up = random.uni-
        form(-rangeNose, rangeNose)
10:         bpy.context.scene.nose_nose_flaring_decr_incr = random.uniform(-
        rangeNose, rangeNose)
11:         bpy.context.scene.nose_nose_scale_depth_decr_incr = random.uni-
        form(-rangeNose, rangeNose)
12:         bpy.context.scene.nose_nose_scale_vert_decr_incr = random.uni-
        form(-rangeNose, rangeNose)
13:         bpy.context.scene.nose_nose_scale_horiz_decr_incr = random.uni-
        form(-rangeNose, rangeNose)
```

Slika 5.6. *Kod za uređivanje oblika i veličine nosa.*

Na sličan način mijenjaju se usta, glava, čelo, oči, obrve, brada, obrazi i uši modela.

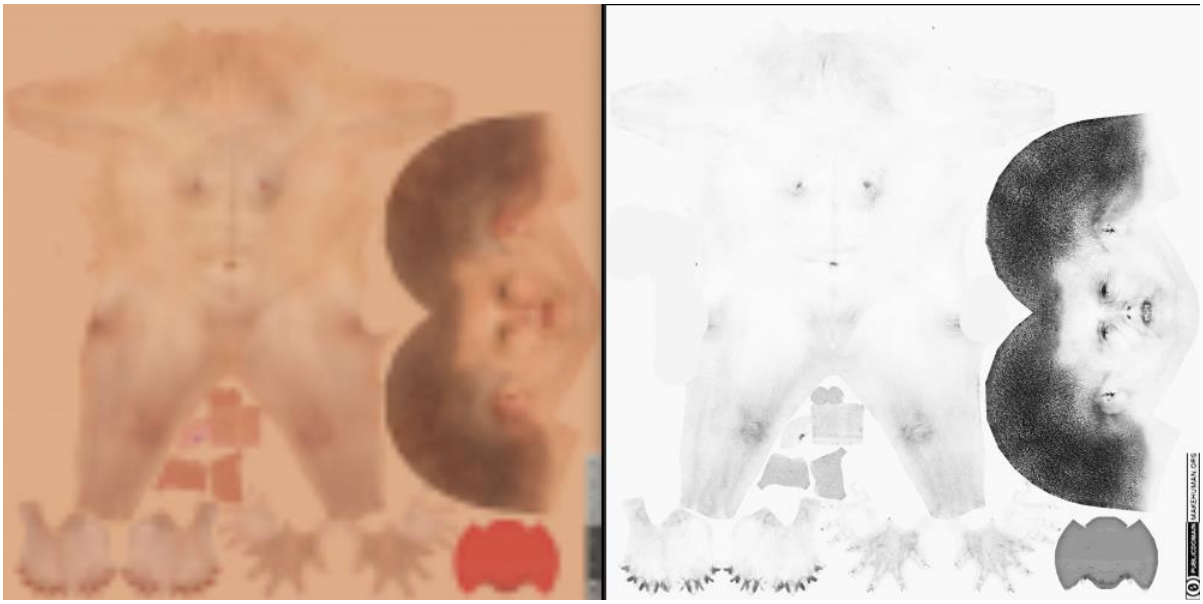
4.3. Uređivanje izrađenog modela

Kože koje nudi MakeHuman zajednica mogu biti realistične, ali to nije uvijek slučaj. Iako su vizualno realistične, pri detaljnijoj inspekciji modela vidljivo je da kože ne sadrži nikakve reljefe, što je neprirodno. Osim toga, setovi koža koji postoje su takvi kakvi su izrađeni i nije ih moguće mijenjati. To znači da se na boju kožu ne može utjecati niti se može izvući boja kao anotacija. Izradom posebnog grafa čvorova za teksturiranje, ostvarena je modifikacija boje kože, dodavanje detalja, nedostataka i reljefa na koži. Postavke čvorova za teksturiranje prikazane su slikom 5.7.

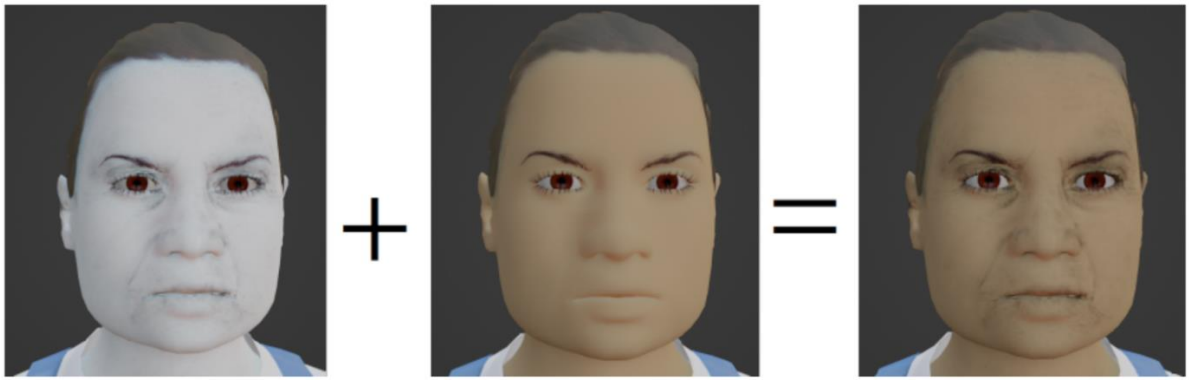


Slika 5.7. *Postavke čvorova za teksturiranje i modificiranje površine kože modela.*

Mape tekstura kože koje na sebi imaju nacrtane sve bore i detalje iskorištene su na način da su ručno uređene u programu za obradu slika. Te mape uređene su tako da ostanu crno-bijele, a da nisu jako tamne da ne utječu na osnovnu boju kože pri preklapanju. Originalne mape i crno bijele mape prikazane su slikom 5.8. Na osnovnu boju kože dodana je mapa teksture kože. Korištenjem funkcije pomnoži, ondje gdje je mapa kože bijela, ostaje odabrana nijansa kože, a tamniji dijelovi dodani su na osnovnu boju kože. Preklapanje te mape s bojom prikazano je na slici 5.9.



Slika 5.8. *Lijevo je mapa kože u boji, a desno je uređena mapa koja je posvijetljena i uklonjena joj je boja.*

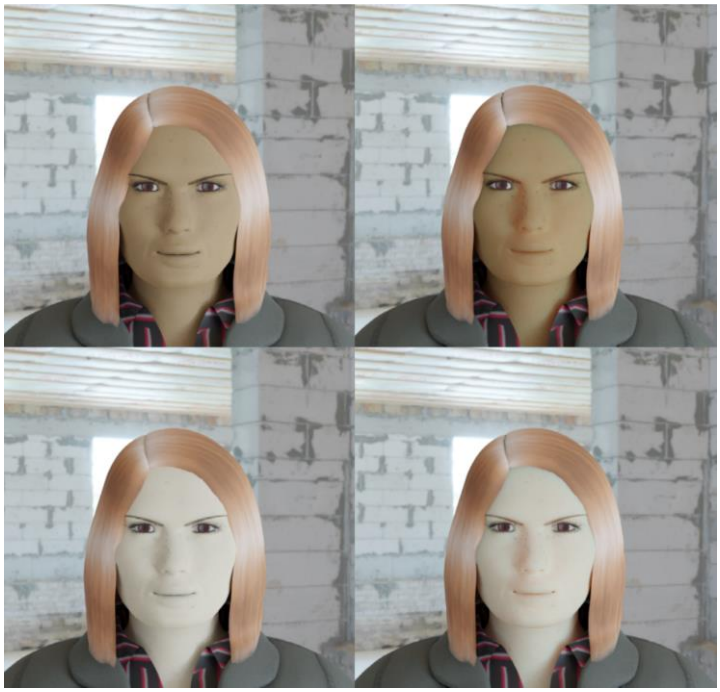


Slika 5.9. *Preklapanje tekstura bora na osnovnu boju.*

Da bi se postigla dodatna raznolikost i jedinstvenost modela ljudi, potrebno je dodati nasumične mrlje ili pjege na licu. To je ostvareno kombinacijom čvora teksture šuma i rampe boja. Rezultat dodavanja tih mrlja prikazan je slikom 5.10.

Slika 5.10. *Model ljudskog lica bez nesavršenosti na lijevoj strani i s mrljama na licu na desnoj strani.*

Kod iscrtavanja slika različitih 3D modela, potrebno je na različite načine prikazati površine. Svaka površina na svoj način upija svjetlost. Tako se, na primjer, razlikuje površina voska, voća, stakla, itd. Svaki materijal ima svoju razinu prozirnosti koja utječe na to koliko će se svjetlosti upiti. Ispod površinsko raspršivanje simulira svjetlost koja pada na površinu materijala, upija se i onda izlazi iz površine pod različitim kutom na različitom mjestu [14]. Ljudska koža je prozirna do jedne mjere i bez raspršivanja svjetlosti iscrtane slike kože ne izgledaju realistično. U Blenderu je dodan čvor za ispod površinsko raspršivanje svjetlosti, a rezultat njegove primjene vidljiv je na slici 5.11.



Slika 5.11. *Model ljudskog lica bez ispod površinskog raspršivanja svjetla na lijevoj strani i s ispod površinskim raspršivanjem svjetla na desnoj strani.*

Za odabir osnovne boje kože korištena je L'Oreal-ova tablica boja kože koja se sastoji od 66 nijansi [15]. Tablica je nastala zbog potrebe poznavanja različitih tipova kože kako bi se bolje upoznalo s biološkim mehanizmima koji utječu na kožu. HEX kodovi tih nijansi nalaze se u datoteci s ostalim postavkama podataka koji se generiraju. Kako se kod kreiranja modela u MakeHuman proširenju dodaje utjecaj fenotipa ovisno o rasi, tako je i tablica boja kože podijeljena po rasama.

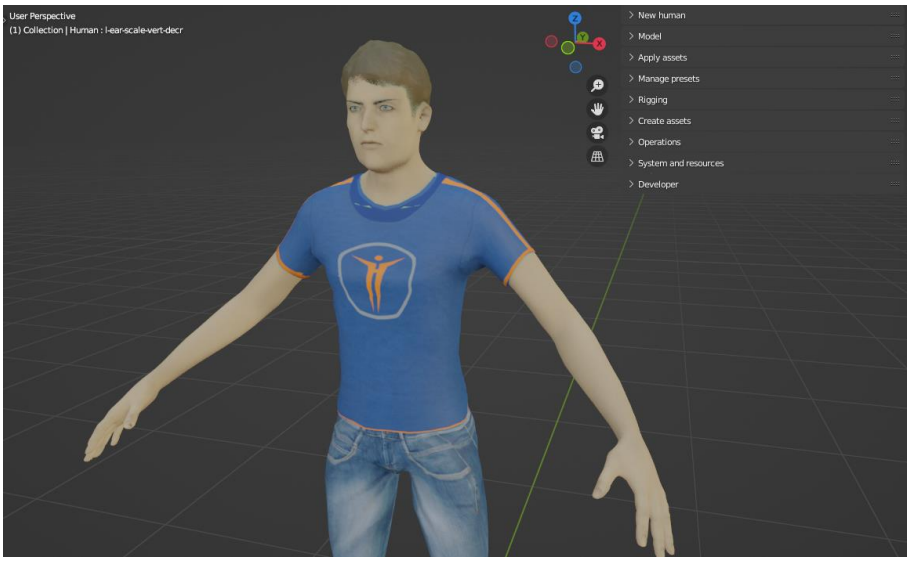
Nakon dodavanja teksture kože, na model se dodaju oči, obrve i trepavice. Dodavanje obrva prikazano je kodom na slici 5.12., a na sličan način na model se dodaju obrve, trepavice, odjeća i kosa.

Linija ***Kod***

```
1:      /*Dodavanje obrva*/
2:      def loadEyebrows():
3:          randomEyebrows = random.choice([x for x in
        os.listdir(os.path.join(assetsPath ,"data/eyebrows", gender)) if
        os.path.isdir(os.path.join(assetsPath, "data/eyebrows", gender,
        x))])
            print(randomEyebrows)
4:
5:          for file in os.listdir(os.path.join(assetsPath, "data/eyebrows",
        gender, randomEyebrows)):
6:              if file.endswith(".mhclo"):
7:                  bpy.ops.mpfb.load_library_clothes(filepath=
        os.path.join(assetsPath, "data/eyebrows", gender, randomEyebrows,
        file), object_type="Eyebrows", material_type="MAKESKIN")
```

Slika 5.12. *Programski kod za dodavanje obrva na model.*

Na model čovjeka je na kraju dodana odjeća i kosa, što prikazuje slika 5.13.



Slika 5.13. *Dodavanje kose i odjeće na model.*

4.4. Pomicanje postojećeg modela i raznolika pozadina

4.4.1. Mijenjanje poze modela

Pomicanjem dijelova kostura modela, čovjeku se mijenja poza, što je prikazano slikom 5.14. Svaki dio kostura ima svoj naziv. Kako bi se ispravno animirao lik u Blenderu, tijelu je potrebno dodati kostur. Proces tog dodavanja kostura na 3D model (engl. *Rigging*) omogućuje rotiranje pojedinih zglobova, odnosno pomicanje kostiju i dijelova tijela. Kostur se sastoji od baze, glavne kosti, i od ostalih kostiju koji se granaju iz baze. U Blenderu postoji nekoliko predefiniranih modela kostura i svaki od njih je različite kompleksnosti. U ovome radu, za animiranje se koristi Mixamo sustav te se stoga dodaje predefinirani Mixamo kostur koji se nudi kao opcija u novijim verzijama MakeHuman proširenja. Kada bi se model čovjeka radio manualno, bez korištenja alata poput MakeHumana, kostur bi se samostalno morao primijeniti na tijelo. Međutim, kod korištenja alata za izradu čovjeka, Mixamo kostur je već mapiran na svaki dio tijela, što omogućava automatizaciju tog procesa. Animacije ljudskih likova potrebno je preuzeti s Mixamo internetske baze i zatim ih uvesti u scenu. Mixamo proširenje u Blenderu nudi opciju postavljanja uvezene animacije u FBX (Filmbox) formatu kao izvornog kostura. Pozivanjem funkcije za dodavanjem animacije na kostur, model čovjeka automatizirano se animira po uzoru na uvezenu animaciju. Animacija se sastoji od niza kadrova te je tako moguće izraditi videozapis animacije. U srhu dobivanja jedne slike u nasumičnoj pozi, u ovome radu uzimat će se jedan nasumičan kadar iz jedne nasumično odabrane animacije.

Slika 5.14. *Pomicanje kostura modela i mijenjanje poze.*

Kako bi se lokacije zglobova mogle dohvaćati u izmijenjenom, animiranom položaju, pozu je potrebno postaviti kao zadanu pozu (engl. *Rest pose*) te primijeniti sve transformacije (translaciju, rotaciju i skalu). S obzirom da ovi koraci također pomiču i model koji se animira, koriste se dva kostura, jedan za animaciju modela, a jedan za dobivanje lokacija zglobova i ključnih točaka. Kako se u ovom sustavu Blender pokreće samo jednom, a kod dodavanja novih modela briše se scena, kod uvoza modela, nazivi objekata imat će izmijenjene sufikse. Kako bi se znalo koji je naziv animacije koja se zadnje uvezla, pamti se popis objekata na sceni prije uvoza i poslije uvoza. Razlika tih dvaju popisa je nova uvezena animacija. Postupak animiranja modela prikazan je na slici 5.15.

Linija ***Kod***

```
1:      /*Dodavanje animacije i primjena poze */
2:      yourFBXfilePath = os.path.join(assetsPath, "Animations",
        random.choice(os.listdir(os.path.join(assetsPath, "Animations"))))
3:      bpy.ops.import_scene.fbx( filepath = yourFBXfilePath )
4:      # Model koji se koristi za animaciju
5:      bpy.context.object.name = "Annimation_armature"
6:      for obj in bpy.context.scene.objects:
7:          if obj.name.startswith("Annimation_armature"):
8:              # Dohvacanje duzine animacije
9:              animation_length =
        obj.animation_data.action.curve_frame_range[1]
10:     # Odabir nasumicnog kadra
11:     bpy.context.scene.frame_current = random.randint(1,
        animation_length)
12:     for obj in bpy.data.armatures:
13:         list_1.append(obj)
14:     # Model koji se koristi za dohvaćanje lokacija
15:     bpy.ops.import_scene.fbx( filepath = yourFBXfilePath )
16:     for obj in bpy.data.armatures:
17:         list_2.append(obj)
18:     new_armatures_list = list(set(list_2).difference(list_1))
19:     bpy.ops.object.mode_set(mode='POSE')
20:     bpy.ops.pose.armature_apply()
21:     bpy.ops.object.mode_set(mode='OBJECT')
22:     bpy.ops.object.transform_apply()
23:     bpy.context.scene.MPFB_ADR_standard_rig = 'mixamo'
24:     bpy.ops.mpfb.add_standard_rig()
25:     bpy.ops.mr.make_rig()for obj in bpy.context.scene.objects:
26:         if obj.name.startswith("Annimation_armature"):
27:             bpy.context.scene.mix_source_armature = obj
28:     bpy.ops.mr.import_anim_to_rig()
```

Slika 5.15. *Pomicanje kostura modela i mijenjanje poze.*

4.4.2. Mijenjanje pozadine

Scena je ostvarena pomoću HDRI mapa. HDRI mapa je fotografija snimljena sa 360 stupnjeva i sadrži u sebi vizualnu snimku i informacije o osvjetljenju scene. Drugi naziv im je HDRI okruženje. Na običnim fotografijama, svaki piksel ima svoju vrijednost boje i svjetlosti koja je u rasponu od 0 do 255. Takve fotografije ne sadrže dodatne informacije o svijetu u kojemu su snimljene. S druge strane, HDRI mape prilikom snimanja za svaki piksel uzimaju vrijednost svjetla u rasponu tisuća ili čak milijuna, što je znatno veći raspon od 256 vrijednosti. To znači da sadrže puno više informacija o svjetlu na sceni od običnih fotografija. Ti dodatni rasponi svjetla koriste se za osvjetljavanje 3D modela koji su smješteni u okruženje HDRI mape [16]. Model čovjeka stavljen u HDRI okruženje prikazan je na slici 5.16.

Slika 5.16. *Model čovjeka u HDRI okruženju.*

4.5. Dobivanje anotacija

Predloženi sustav nudio bi korisniku nekoliko vrsta anotacija. Odabrane su najzanimljivije i najkorisnije, a to su:

- Nijansa boje kože
- Značajke na tijelu (ključni zglobovi i slično)
- Semantička segmentacija scene (odvojen čovjek od scene, kosa, obrve, oči, trepavice, odjeća i brada ako je ima)
- Dubinska slika.

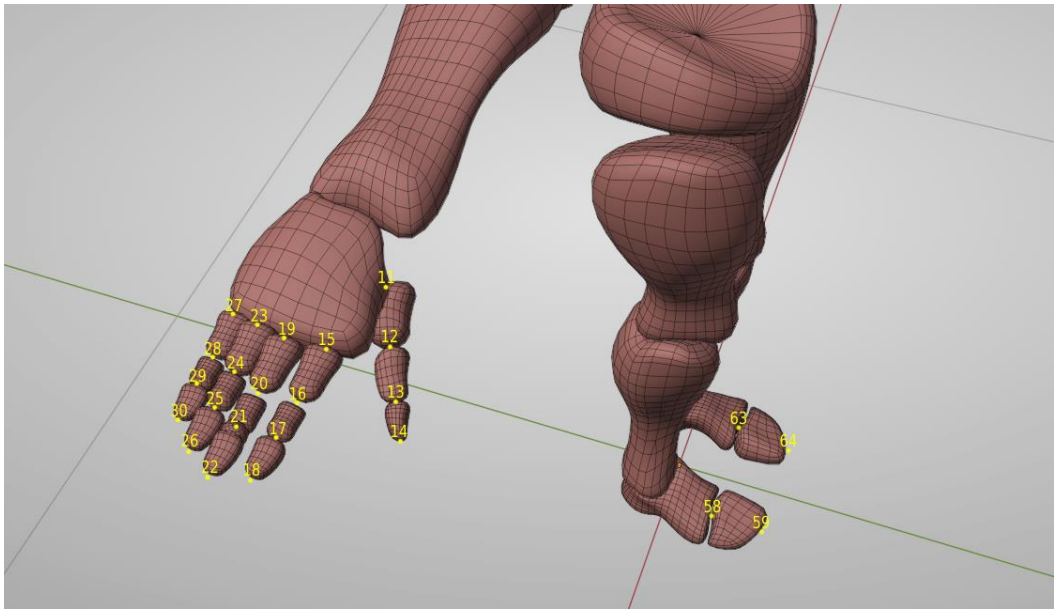
Anotacija nijanse boje kože zapisana je na jednostavan način. HEX kod boje nalazit će se u nazivu slike.

4.5.1. Značajke na tijelu

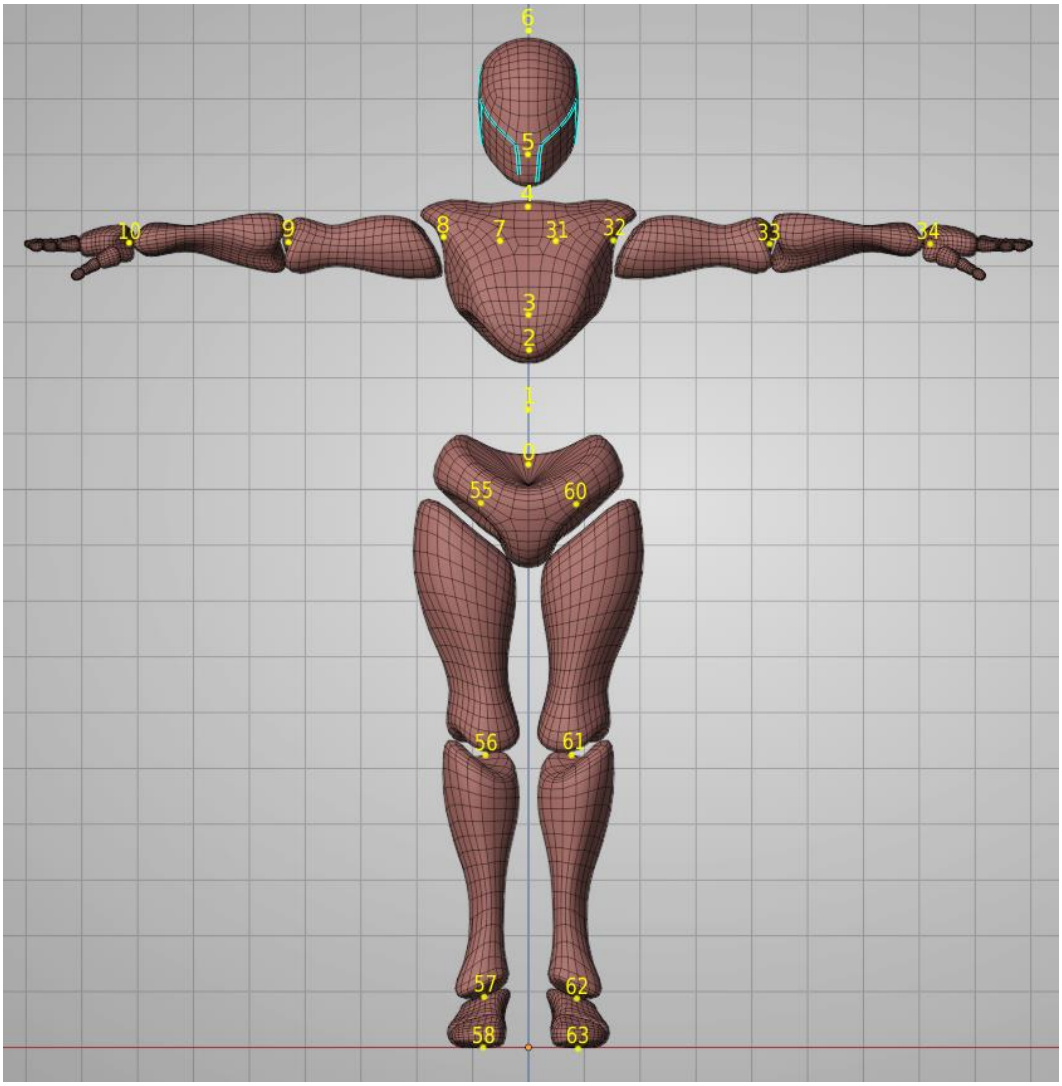
Kostur čovjeka koji se povezuje s modelom služi za lakše animiranje modela. Kao i kostur pravog čovjeka, kostur za animiranje sadrži zglobove koje je moguće pomicati i rotirati. Svaka kost sastoji se od svoje glave i repa te je dohvaćanjem lokacije određene glave, odnosno repa kosti moguće dobiti podatke o lokaciji zanimljivih točaka ili zglobova.

Lokacije pojedine točke u 3D prostoru potrebno je transformirati u prostor kamere kako bi 3D koordinate postale 2D koordinate piksela na iscrtanoj slici. Modul bpy_extras sadrži funkciju world_to_camera_view koja radi tu transformaciju. Važno je imati na umu da se koordinate (0, 0) koje vraća funkcija nalaze u donjem lijevom kutu slike, dok se u računalnom vidu te koordinate nalaze u gornjem lijevom kutu. To znači da je pri dohvaćanju 2D lokacija točaka potrebno invertirati y os.

Značajke koje ovaj sustav vraća zabilježene su u csv (engl., *Comma-separated values)* datoteku čiji se svaki redak sastoji od tri vrijednost. Prva vrijednost označava redni broj značajke, a sljedeće dvije vrijednosti predstavljaju x i y lokaciju značajke. Redni brojevi značajki prikazani su na slikama 5.17. i 5.18.



Slika 5.17. *Redni brojevi značajki prikazani na rukama i stopalima. Istim redom označene su značajke na drugoj ruci i stopalu.*



Slika 5.18. *Redni brojevi značajki prikazani na tijelu.*

4.5.2. Semantička segmentacija

Proširenje bpycv nudi izlaz u obliku segmentacije scene. U ovom slučaju koristi se segmentacija scene po instancama. Za razliku od semantičke segmentacije, segmentacija na instance promatra različite objekte iste klase kao različite instance, a ne kao jednu instancu. Međutim, kako je u ovom sustavu na sceni uvijek jedan čovjek, za svaki objekt na sceni postojat će samo jedno instanca. To znači da je izlaz jednak semantičkoj segmentaciji. Za pozadinu su korištene HDRI mape te je stoga nemoguće dohvatiti podatke o segmentaciji pozadine na ovaj način. Primjeri semantičke segmentacije koju nudi ovaj sustav prikazani su na slici 5.19.

Slika 5.19. *Semantička segmentacija scene. Prvi red slika su iscrtane RGB slike, a ispod svake slike nalazi se njena segmentacija.*

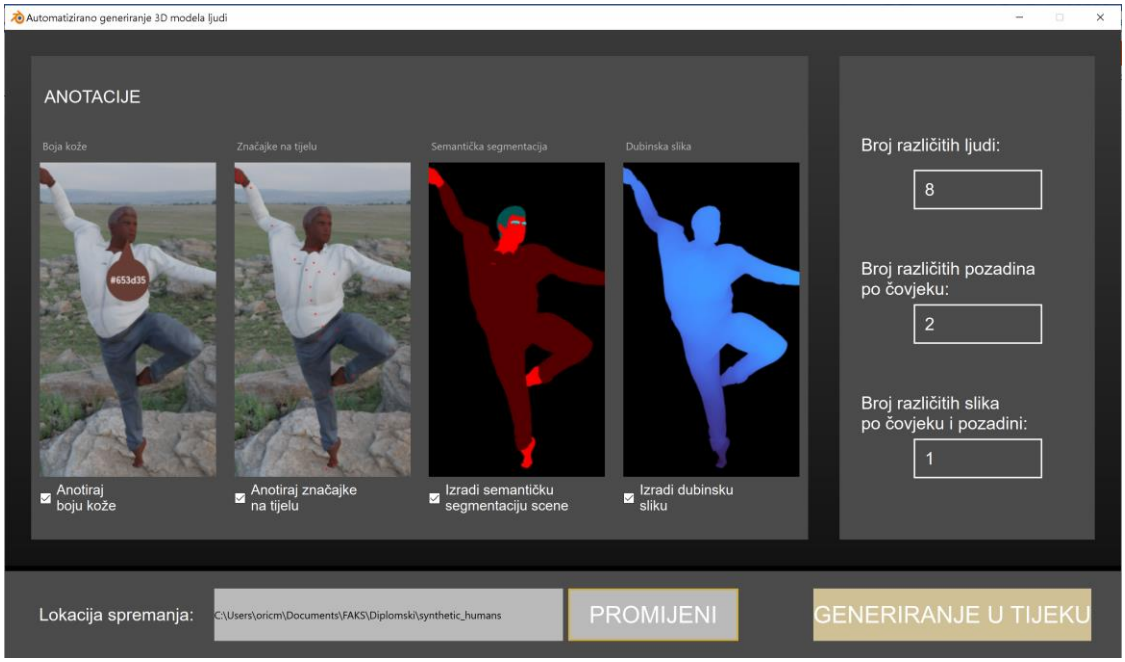
4.5.3. Dubinska slika

Dubinska slika je slika na kojoj vrijednost svakog piksela donosi informaciju o udaljenosti te točke od kamere. Što piksel ima veću vrijednost, odnosno, što je boja na tome mjestu svjetlija, to je ta točka bliža kameri. Kao i kod segmentacije, u ovom slučaju nemamo podatke o pozadini te se na izlazu iz bpycv-a ne vide informacije o dubini scene oko modela. Primjeri dubinskih slika nalaze se na slici 5.20.

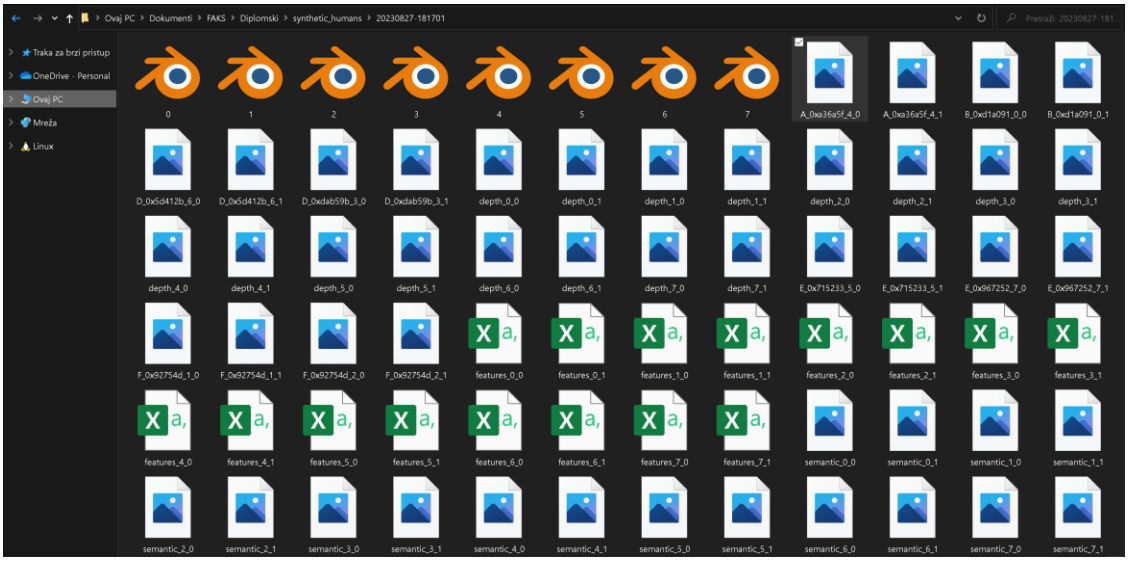
Slika 5.20. *Dubinske slike. Prvi red slika su iscrtane RGB slike, a ispod svake slike nalazi se njena dubinska slika.*

5. TESTIRANJE

U nastavku je prikazano nekoliko slika i pripadajućih anotacija koje je sustav dao kao izlaz. Testovi su provedeni na način da se rekreira način na koji bi korisnik koristio sustav. Pokrenuto je grafičko sučelje, odabrane su sve anotacije, unesene su vrijednosti veličine skupa podataka, odabrana lokacija spremanja te pokrenuto generiranje pritiskom na gumb. Pokrenuto sučelje prikazano je slikom 6.1. Sustav kreira mapu na odabranoj lokaciji te ondje sprema sve navedeno, a kreirana mapa vidljiva je na slici 6.2. Ime mape nastalo je korištenjem točnog vremena pokretanja.



Slika 6.1. *Pokrenuto generiranje u grafičkom sučelju.*



Slika 6.2. *Mapa s podacima koju je kreirao sustav.*

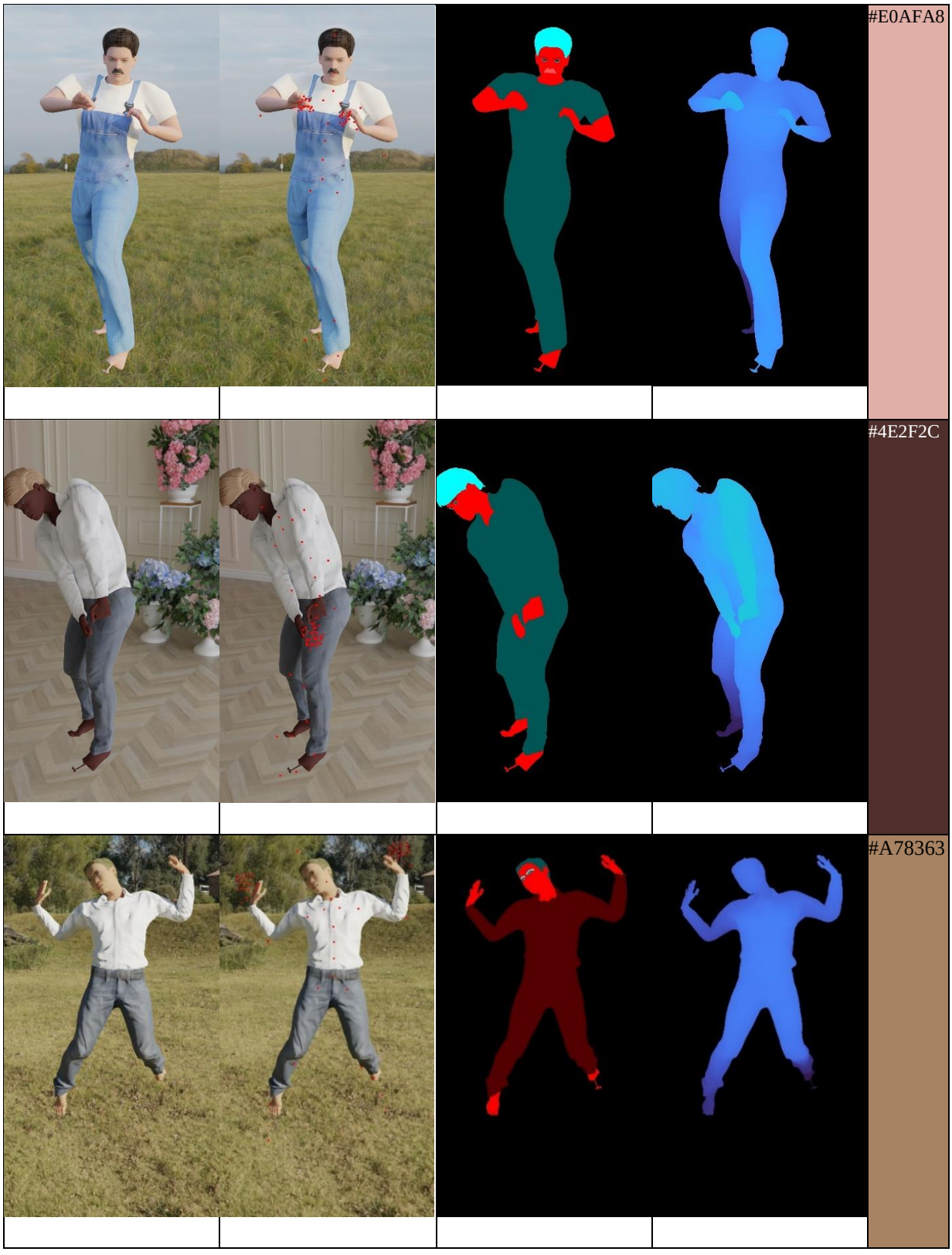
U datoteci s postavkama moguće je odabrati broj uzoraka koji stroj za iscrtavanje provodi. Što je broj uzoraka veći, to je iscrtana slika realističnija i veće kvalitete. Slika 6.3. nastala je iscrtavanjem s velikim brojem uzoraka.

Slika 6.3. *Slika nastala iscrtavanjem s velikim brojem uzoraka. Na modelu je vidljiva detaljna tekstura kože.*

Vrijeme potrebno za kreiranje slika ovisi o tehničkim specifikacijama sustava na kojemu se generiranje pokreće. Kako bi se skratilo vrijeme kreiranja slika, u testovima je broj uzoraka postavljen na 64. Tako je kvaliteta slika smanjena u odnosu na prošli primjer, ali razlika nije uočljiva bez da se sliku pretjerano uvećava. Tablica 6.1. prikazuje primjere iscrtanih slika s pripadajućim anotacijama.

Tablica 6.1. *Primjeri iscrtanih slika s pripadajućim anotacijama. Sadržaji u stupcima su: RGB iscrtana slika, slika s prikazanim značajkama, semantička segmentacija, dubinska slika, HEX broj boje kože uz vizualnu reprezentaciju boje.*

				#784D3D
				#A36A5F
				#DAB59B



Modeli ljudi su raznoliki i realistični. Poze su također raznolike i svaka slika donosi neku novu pozu. Značajke na tijelu nekada nisu precizne jer je kod kreiranja modela izmijenjena masa,

veličina, spol i to sve utječe na odstupanje modela od svoga kostura. Semantička segmentacija, dubinska slika i anotacija boje kože odgovaraju kreiranim modelima.

Tablica 6.2. prikazuje modele ljudi stavljene u različita okruženja.

Tablica 6.4. *Modeli ljudi stavljeni u različita okruženja.*

6. ZAKLJUČAK

Kod treniranja modela važno je dobro izabrati set podataka za treniranje. Prikupljanje stvarnih podataka je teško, troši puno resursa, a ponekad je nemoguće. Nakon prikupljanja potrebno je angažirati osobu ili cijeli tim koji će provesti proces anotacije. Sintetički podaci mogu zamijeniti stvarne i tako uštediti resurse. Za razliku od stvarnih, koje je potrebno ručno anotirati, anotacija sintetičkih podataka je automatiziran proces.

U današnje vrijeme postoje kvalitetni sintetički setovi podataka ljudi ili sustavi za kreiranje takvih podataka. Međutim, ti su sustavi i setovi zaštićeni od slobodne i besplatne uporabe. Stoga se ovaj rad posvetio izradi sustava za generiranje sintetičkih podataka koji prikazuju ljude.

Cijeli sustav generiranja je automatiziran i pokreće se putem grafičkog sučelja što ga čini lakim za korištenje. Kod kreiranja podataka moguće je odabrati željene postavke seta, a to uključuje veličinu, raznolikost i odabrane anotacije. Nakon odabira postavki, pokreće se generiranje koje traje određeno vrijeme ovisno o specifikacija računala na kojemu se sustav koristi. Nakon generiranja, iscrtane slike s anotacijama spremljene su na odabranu lokaciju.

Kvaliteta izrađenih modela može se procijeniti vizualnim putem tako da se provjeri koliko realistično izgledaju kreirani modeli ljudi. U ovom radu velik je fokus bio na teksturi kože jer ona uvelike utječe na to koliko će koža modela, a samim time i model, izgledati kao prava osoba. Pozadina i osvjetljenje modela realistično su prikazan korištenjem HDRI mapa.

U ovom sustavu korišteni su besplatno dostupni resursi te kvaliteta modela dijelom ovisi o njima. Nadogradnja ovog sustava bila bi proširenje biblioteka modela odjeće i kose koje se koriste u ovom sustavu.

S obzirom na to da je u ovom sustavu implementirano animiranje lika kroz više kadrova, a uzet je samo jedan kadar za iscrtavanje slike, sustav je moguće nadograditi dodavanjem funkcionalnosti spremanja videozapisa cijele animacije.

LITERATURA

[1] B. Foundation, »Blender«, [Mrežno]. Available: https://www.blender.org/.

[2] »The Public 3D Asset Library«, Poly Haven, [Mrežno]. Available: https://polyhaven.com/. [Pokušaj pristupa 4. 5. 2023.].

[3] Q. Group, »The Future of Digital Experiences«, Qt Group, [Mrežno]. Available: https://www.qt.io/. [Pokušaj pristupa 22. 6. 2023.].

[4] Microsoft, »Face Synthetics«, Microsoft, [Mrežno]. Available: https://microsoft.github.io/FaceSynthetics/. [Pokušaj pristupa 1. 4. 2023.].

[5] S. AI, »Better Models Deployed Faster With Synthetic Data«, Synthesis AI, [Mrežno]. Available: https://synthesis.ai/. [Pokušaj pristupa 1. 4. 2023.].

[6] A. Zewe, »In machine learning, synthetic data can offer real performance improvements«, MIT News Office, 2022.. [Mrežno]. Available: https://news.mit.edu/2022/synthetic-data-ai-improvements-1103. [Pokušaj pristupa 5. 4. 2023.].

[7] S. Hinterstoisser, »On Pre-trained Image Features and Synthetic Images for Deep Learning«, u *Computer Vision - ECCV 2018 Workshops*, Munich, 2018.

[8] B. Foundation, »Blender 3.5.1 Python API«, Blender, [Mrežno]. Available: https://docs.blender.org/api/current/info_overview.html. [Pokušaj pristupa 1. 4. 2023.].

[9] MakeHuman, »MakeHuman Documentation«, MakeHuman, [Mrežno]. Available: http://www.makehumancommunity.org/wiki/Documentation:Index. [Pokušaj pristupa 1. 4. 2023.].

[10] DIYer22, »Bpycv: Computer Vision and Deep Learning Utils for Blender«, Bpycv, [Mrežno]. Available: https://github.com/DIYer22/bpycv. [Pokušaj pristupa 20. 8. 2023.].

[11] Mixamo, Adobe, [Mrežno]. Available: https://www.mixamo.com/#/. [Pokušaj pristupa 10 8 2023].

[12] W. Maxwell, »What is Mixamo and How Can it be Used in Games«, CG Obsession, [Mrežno]. Available: https://cgobsession.com/what-is-mixamo-and-how-can-it-be-used-in-games/. [Pokušaj pristupa 10 8 2023].

[13] PySide, »PySide6«, PySide, [Mrežno]. Available: https://pypi.org/project/PySide6/. [Pokušaj pristupa 22. 6. 2023.].

[14] Pluralsight, »Understanding Subsurface Scattering to Render Translucency«, Pluralsight, [Mrežno]. Available: https://www.pluralsight.com/blog/film-games/understanding-subsurface-scattering-capturing-appearance-translucent-materials. [Pokušaj pristupa 20. 6. 2023.].

[15] L. Groupe, »Expert In Skin And Hair Types Around The World«, L'Oreal, [Mrežno]. Available: https://www.loreal.com/en/articles/science-and-technology/expert-inskin/. [Pokušaj pristupa 30 7 2023].

[16] R. K. Hegeler, »What is an HDRi Map? Detailed Explanation and Real-World Examples«, CGI.Backgrounds, [Mrežno]. Available: https://www.cgibackgrounds.com/blog/what-is-an-hdri. [Pokušaj pristupa 22. 6. 2023.].

[17] S. I. Nikolenko, »Synthetic Data for Deep Learning«, Springer, 2019..

SAŽETAK

Ovaj rad opisao je primjenu sintetičkih podataka za strojno učenje te predstavio nekoliko postojećih sustava za izradu sintetičkih podataka ljudi. Zbog nedostataka tih sustava, npr. nedostupnost korištenja sustava za slobodnu uporabu, predstavljen je novi sustav za kreiranje modela ljudi. Sustav kreira 3D modele ljudi u alatu za 3D modeliranje, Blender, korištenjem javno dostupnih proširenja. Svaki model čovjeka detaljno se uređuje kako bi bio što realističniji. Poseban naglasak stavljen je na uređivanje teksture ljudske kože. Poboljšanje izgleda izrađenih ljudi moguće je napraviti proširenjem baze modela odjeće i kose koji se dodaju na modele. Modificiraju se poza čovjeka i scena u kojoj se nalazi svaki čovjek, te se kao izlaz dobivaju slike scene s određenim anotacijama. Cijeli postupak automatiziran je korištenjem Pythona. Pokretanje i odabir opcija generiranja podataka vrši se putem grafičkog sučelja.

Ključne riječi: Blender, Python, sintetički podaci, 3D modeliranje

ABSTRACT

Automated human 3D model generation

This thesis describes the use of synthetic data in machine learning and introduces a few existing solutions for synthetic human data generation. The downfalls of those systems, such as the fact that those systems can not be freely used publicly, gave motivation for introducing a pipeline for generation of human models. Proposed pipeline creates 3D models of people in a 3D modelling tool Blender by using publicly available addons. Each model is edited in detail to achieve a realistc look. Focus is given on texturing the human skin. The look of models could be improved by extending the base of clothes and hair models. The pose of models and the scene they are in is modified and an anotated render is given as an output. The whole pipeline is automated using Python. Generation start and parameters selection are done in a graphical interface.

Key words: Blender, Python, synthetic data, 3D modelling

ŽIVOTOPIS

Mihaela Orić rođena je 24. rujna 1999. u Slavonskom Brodu. Odrasla je u Slavonskom Kobašu i ondje pohađala Osnovnu školu „Dr. Stjepan Ilijašević“. Nakon osnovne škole, pohađala je Klasičnu gimnaziju fra Marijana Lanosovića s pravom javnosti, Slavonski Brod. Nakon klasične gimnazije upisala je preddiplomski sveučilišni studij računarstva na Fakultetu elektrotehnike, računarstva i informacijskih tehnologija, Osijek. Nakon preddiplomskog studija upisuje diplomski studij računarstva, izborni blok Robotika i umjetna inteligencija.

_____

Potpis autora