

Simulacija i vizualizacija prirodnih procesa u interaktivnom 3D okruženju	Verzija: 1.0
Tehnička dokumentacija	Datum: 20/01/26

Simulacija i vizualizacija prirodnih procesa u interaktivnom 3D okruženju Tehnička dokumentacija Verzija 1.0

Studentski tim: Marin Banožić
Maksimilijan Hižman
Antonio Hohnjec
Dorijan Jančić
Lovro Ledić
Marko Lujo
Vito Vrbić

Nastavnik: prof. dr. sc. Željka Mihajlović

Simulacija i vizualizacija prirodnih procesa u interaktivnom 3D okruženju	Verzija: 1.0
Tehnička dokumentacija	Datum: 20/01/26

Sadržaj

1.	Opis razvijenog proizvoda	3
1.1	Dinamika Vode	3
1.2	Erozija Terena	3
1.3	Ponašanja Vegetacije	4
2.	Tehničke značajke	6
2.1	Dinamika Vode	6
2.1.1	Generacija valova	6
2.1.2	Sjenčanje valova	7
2.1.3	Slojevi	8
2.1.4	Teselacija	9
2.2	Erozija Terena	9
2.2.1	Perlinov šum	9
2.2.2	Simulacija kišne erozije	10
2.2.3	Bojanje visinske mape	10
2.3	Ponašanja Vegetacije	11
2.3.1	Trava	11
2.3.2	Krošnje	13
3.	Upute za korištenje	17
3.1	Dinamika Vode	17
3.2	Erozija Terena	17
3.3	Ponašanja Vegetacije	17
4.	Literatura	18

Simulacija i vizualizacija prirodnih procesa u interaktivnom 3D okruženju	Verzija: 1.0
Tehnička dokumentacija	Datum: 20/01/26

Tehnička dokumentacija

1. Opis razvijenog proizvoda

Razvijeni proizvod sastoji se od tri odvojena virtualna okruženja u kojima se fokus stavlja na tri različita prirodna procesa (ili skupinu istih). Virtualna okruženja omogućuju vizualni prikaz simulacije zasebnih prirodnih procesa te određenu razinu interakcije s istima.

1.1 Dinamika Vode

U virtualnom okruženju dinamike vode prikazuje se simulirani ocean. Površina oceana interaktivno reagira na osvjetljenje u sceni, dok je kretanje vode simulirano korištenjem brze Fourierove transformacije (FFT). Korisnik se može kretati isključivo pomoću leteće kamere, što omogućava slobodno istraživanje i pregledavanje površine oceana iz različitih kutova i visina.

Izgled ovog virtualnog okruženja vidljiv je na slici 1.1.



Slika 1.1: Prikaz površine oceana.

1.2 Erozija Terena

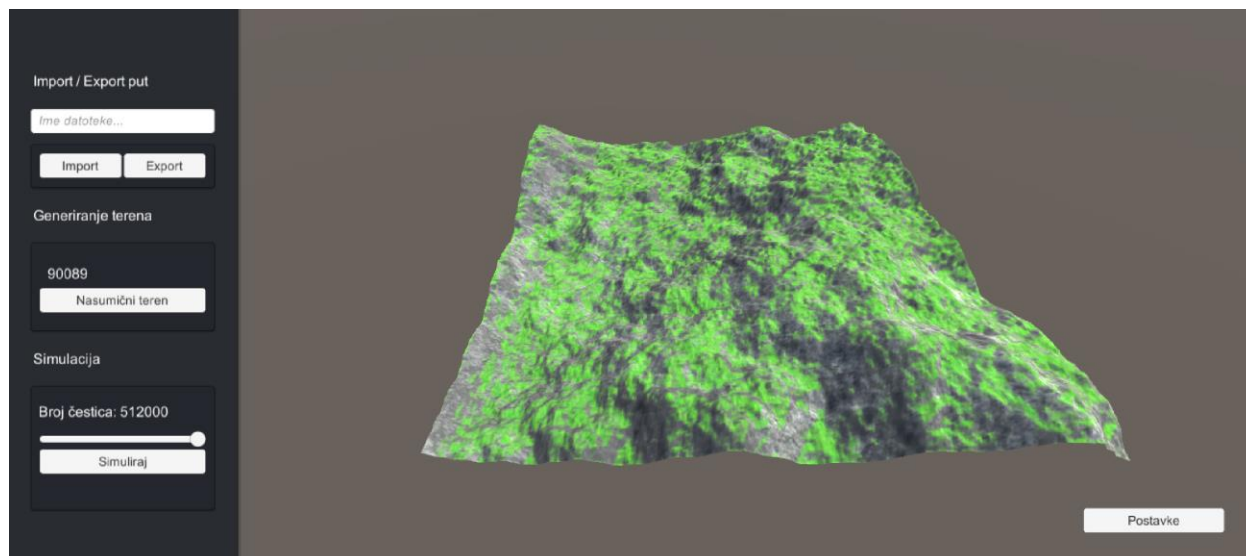
Virtualno okruženje erozije terena stavlja korisnika u pogled iz ptičje perspektive gledajući na kvadratnu površinu simuliranog terena.

Teren u virtualnom okruženju prvo je generiran metodom Perlinovog šuma, a korisniku je omogućeno pokretanje simulacije vodene erozije korištenjem grafičkog sučelja. Nakon izvršavanja simulacije, teren će promijeniti svoj izgled te će korisnik moći pregledati rezultate i spremi/učitati teren iz datoteke. Korisniku je omogućeno mijenjanje postavki simulacije čestica koje će utjecati na krajnji rezultat simulacije vodene erozije.

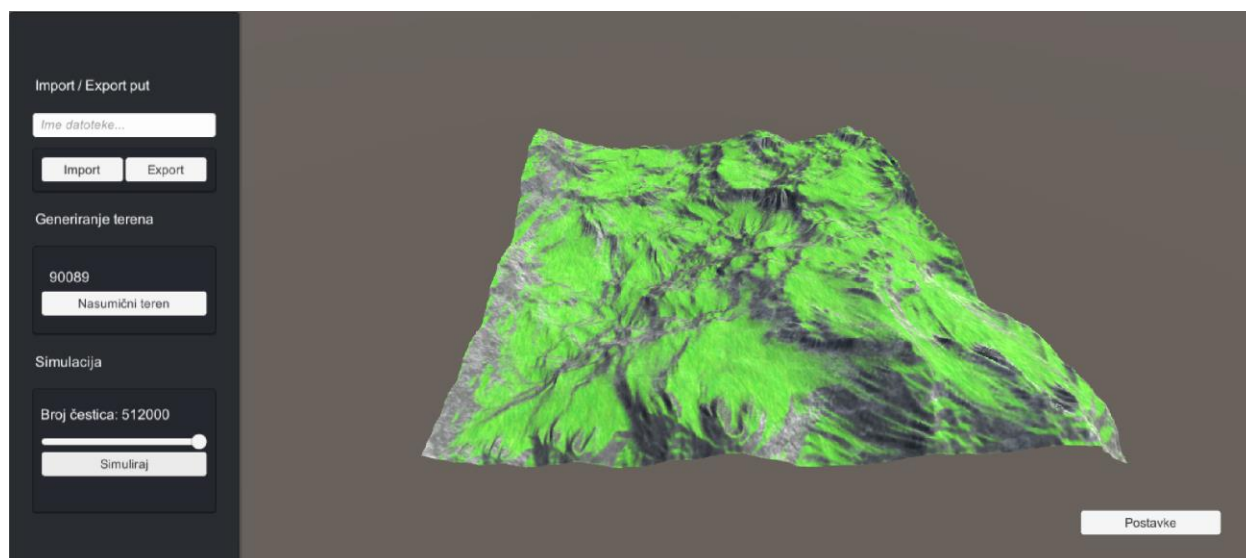
Izgled ovog virtualnog okruženja vidljiv je na slikama 1.2 i 1.3.

Važno je napomenuti da je simulacija implementirana sinkrono, a pokretanje s velikim brojem čestica može privremeno zamrznuti prozor, te je u tom slučaju potrebno pričekati izvršenje simulacije.

Simulacija i vizualizacija prirodnih procesa u interaktivnom 3D okruženju	Verzija: 1.0
Tehnička dokumentacija	Datum: 20/01/26



Slika 1.2: Virtualno okruženje s prikazom na teren prije izvođenja simulacije.



Slika 1.3: Virtualno okruženje s prikazom na teren nakon izvođenja dvije iteracije simulacije.

1.3 Ponašanja Vegetacije

U virtualnom okruženju ponašanja vegetacije implementirana je stilizirana interaktivna trava koja reagira na vjetar te na hod igrača. Implementirane su i stilizirane krošnje drveća i grmlja koje reagiraju na vjetar.

Interaktivni dio virtualnog okruženja tvori sivi stilizirani ljudski lik kojim je moguće upravljati pomoću miša i tipkovnice; on se pomoću tipkovnice može kretati po virtualnom terenu koji je prekriven travom, grmljem i drvećem, dok je kamera koja ga promatra orbitalna i ona se pokreće pomoću miša.

Izgled ovog virtualnog okruženja vidljiv je na slikama 1.4 i 1.5.

Simulacija i vizualizacija prirodnih procesa u interaktivnom 3D okruženju	Verzija: 1.0
Tehnička dokumentacija	Datum: 20/01/26



Slika 1.4: Virtualno okruženje u kojem se nalazi interaktivna trava, stabla, grmlja i statičan sivi lik



Slika 1.5: Virtualno okruženje u kojem se nalazi interaktivna trava, stablo i sivi lik koji trči

Simulacija i vizualizacija prirodnih procesa u interaktivnom 3D okruženju	Verzija: 1.0
Tehnička dokumentacija	Datum: 20/01/26

2. Tehničke značajke

U ovom poglavlju opisani su problemi koji postoje u simulaciji i vizualizaciji obrađenih prirodnih fenomena te načini kojima su ovi problemi riješeni ili zaobiđeni pri implementaciji prirodnih fenomena u svakom od tri virtualna okruženja.

Uz opis problema, u ovom poglavlju opisani su i algoritmi i načini koji su se koristili za implementaciju i vizualizaciju, a koji su najviše povezani s područjem računalne grafike. Opisani su i oni algoritmi i načini na koje se stavilo najviše pažnje pri realizaciji rješenja, a koji nisu usko povezani s područjem računalne grafike.

2.1 Dinamika Vode

Generiranje i simuliranje vode u računalnoj grafici predstavlja opsežan problem kojem se može pristupiti na razne načine. U ovom radu implementira se spektralna tehnika simulacije oceana koja se već dulje vrijeme koristi za iscrtavanje detaljnih i realističnih većih površina vode u videoigrama i filmovima. Površina oceana u ovoj tehnici modelira se kao 3-dimenzijska plošna mreža čiji se vrhovi pomiču u vremenu prema spektralnim amplitudama valova izračunatim pomoću brze Fourierove transformacije (FFT).

2.1.1 Generacija valova

Funkcija visine vala napravljena je kao zbroj kompleksnih sinusa sa vremenski ovisnim amplitudama na horizontalnoj poziciji $x^{[1]}$:

$$h(x, t) = \sum_k \tilde{h}(\mathbf{k}, t) \exp(i\mathbf{k} \cdot \mathbf{x})$$

U ovakvoj funkciji vala, vremenski se razvoj spektralne amplitude $\tilde{h}$ računa kao zbroj konjugata^[1]:

$$\tilde{h}(\mathbf{k}, t) = \tilde{h}_0(\mathbf{k}) \exp(i\omega(\mathbf{k})t) + \overline{\tilde{h}_0} \exp(-i\omega(\mathbf{k})t)$$

Vektor označen s $\mathbf{k}$ predstavlja valne vektore pojedinačnog vala. Početna spektralna amplituda, označena s $\tilde{h}_0$ generira se kao slučajna varijabla sljedećom formulom, gdje su ξ_r i ξ_i nezavisne slučajne varijable s normalnom distribucijom u intervalu $[0, 1]$. Gdje je S spektralna gustoća energije vala, a D je direkcijska funkcija ovisna o globalnoj rotaciji vektora $\theta^{[2]}$:

$$\tilde{h}_0(\mathbf{k}) = \frac{1}{\sqrt{2}} (\xi_r + i\xi_i) \sqrt{S(\mathbf{k})D(\theta)}$$

Za spektralnu funkciju implementiran je Tessendorov spektar koji je varijanta Phillipsovog spektra, dok se za direkcijsku komponentu koristi kvadrat kosinusa. Ove formule prikazane su u nastavku:

$$S(k) = A \frac{\exp\left(\frac{-g^2}{U^4 k^2}\right)}{k^4} \exp(-k^2 l^2)$$

$$D(\theta) = \cos^2(\theta - w)$$

U ovom virtualnom okruženju, implementirana je varijanta Stockhamovog FFT algoritma^{[3][4]}. Ovaj algoritam inicijalno izvodi 1-dimenzijsku transformaciju, a primjenom u dva koraka, jednom vertikalno i jednom horizontalno, postiže se 2-dimenzijska transformacija. Pseudokod algoritma dan je u nastavku:

```
FFT(i, input):
    stage = 0
    butterfly[stage][i] = input – sadrži 4D zapis
    SinkronizirajDretve()
```

Simulacija i vizualizacija prirodnih procesa u interaktivnom 3D okruženju	Verzija: 1.0
Tehnička dokumentacija	Datum: 20/01/26

```

za j = 0 do log(N) - 1:
    m = N / (2**(j + 1)) -- pola segmenta
    s = m * (i / m); -- početak segmenta
    b = (s + i) % N; -- početak segmenta u butterfly-u

    theta = 2π / N
    w = complex(cos(theta * b), sin(theta * b))

    a, b = butterfly[stage][b, b + m]

    butterfly[(stage + 1) % 2][i] =
        a + [complexmul(w, b[0, 1], complexmul(b[2, 3])]

    stage = (stage + 1) % 2
    SinkronizirajDretve()

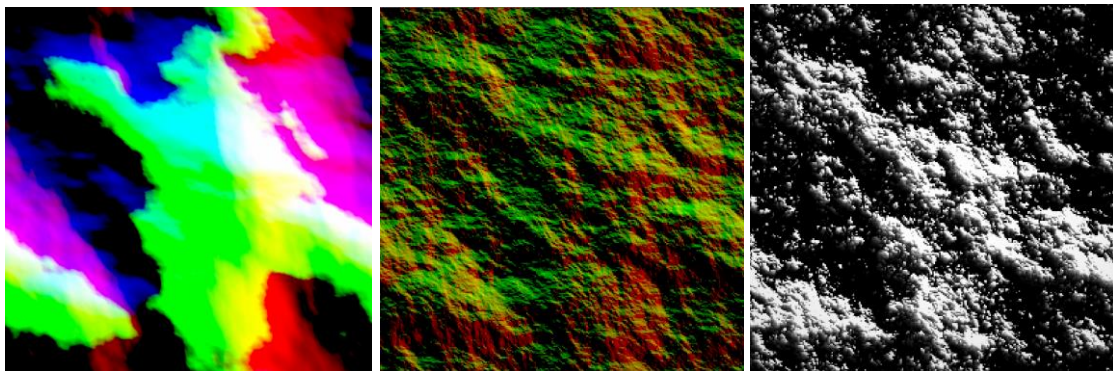
vrati butterfly[stage][i]

```

Dodavanjem horizontalnog pomaka uz vertikalni moguće je prikazati valove sa oštrijim rubovima što daje povećava realizam simuliranih valova. Ovo je prikazano na sljedećoj formuli:

$$D(x, t) = - \sum_k i \frac{k}{|k|} \tilde{h}(k, t) \exp(ik^n \cdot x)$$

Veliki horizontalni pomak može uzrokovati uvijanje vrhova valova u same sebe. Detekcija ovakvih uvijanja može se provoditi provjerom Jakobijana transformacije gdje neaktivne vrijednosti determinante Jakobijana upućuju na preklapanje vrhova valova. Na temelju detekcije negativnog Jakobijana može se generirati maska pjene koja se akumulira u teksturu i postupno prigušuje kako bi se simuliralo raspršivanje sjene. Za izračun normala koriste se derivacije visine po horizontalnim koordinatama. Derivacije svih pomaka mogu se dobiti primjenom FFT-a.



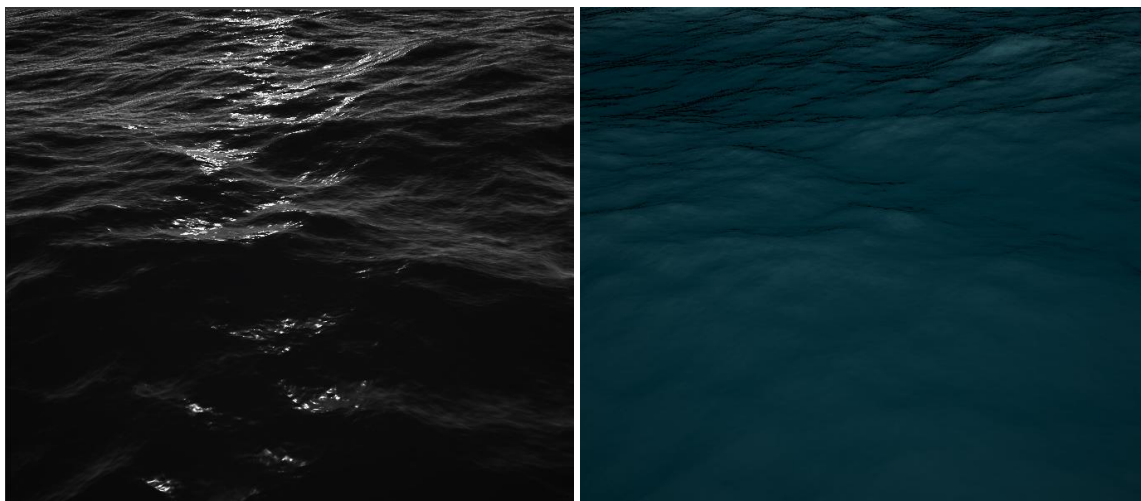
Slika 2.1: Generirane teksture za ocean: tekstura pomaka, tekstura normala i tekstura akumulirane pjene

2.1.2 Sjenčanje valova

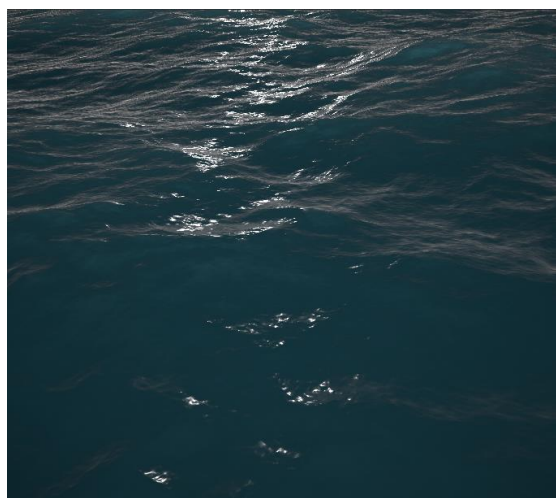
Animacija površine postiže se čitanjem teksture pomaka ili pomicanjem vrhova mreže u sjenčaru vrhova prema dobivenim vrijednostima pomaka.

Za sjenčanje valova koristi se fizikalno temeljen model^[5]. Model je kombinacija zrcalne i difuzne komponente. Modifikacija modela je da se koristi GGX funkcija distribucije normala umjesto Beckman-a, uz manje modifikacije za difuznu komponentu. Za pjenu se povećava hrapavost površine.

Simulacija i vizualizacija prirodnih procesa u interaktivnom 3D okruženju	Verzija: 1.0
Tehnička dokumentacija	Datum: 20/01/26



Slika 2.2: Zrcalna komponenta (lijevo) i difuzna komponenta (desno)



Slika 2.3: Kombinacija zrcalne i difuzne komponente

2.1.3 Slojevi

Za plohu vala definiramo, između ostalih, dva važna parametra: duljinu (L) i rezoluciju (N). Ovi parametri određuju raspon valnih brojeva i gustoću njihovog uzorkovanja, a samim time određuju i razinu detalja valova. Najmanji i najveći valni broj dani su relacijama:

$$k_{min} = \frac{2\pi}{L} \quad k_{max} = \frac{\sqrt{2}\pi N}{L}$$

Povećanjem isključivo rezolucije možemo povećati i raspon i gustoću uzorkovanja, no to zahtjeva znatno više računalnih resursa.

S druge strane, povećanjem duljine povećat će se gustoća uzorkovanja, ali će se smanjiti raspon valnih brojeva. Zato se često, za neku fiksnu rezoluciju, istovremeno generira više slojeva s različitim duljinama koji će komplementirati jedni druge i koji se u konačnici zbrajaju.

U okviru ovog virtualnog okruženja, korištena su tri sloja. Svaki sljedeći sloj ima duljinu manju od onog prethodnog. Kako bi spriječili preklapanje spektara valnih brojeva (što rezultira dvostrukim zbrajanjem istih valova),

Simulacija i vizualizacija prirodnih procesa u interaktivnom 3D okruženju	Verzija: 1.0
Tehnička dokumentacija	Datum: 20/01/26

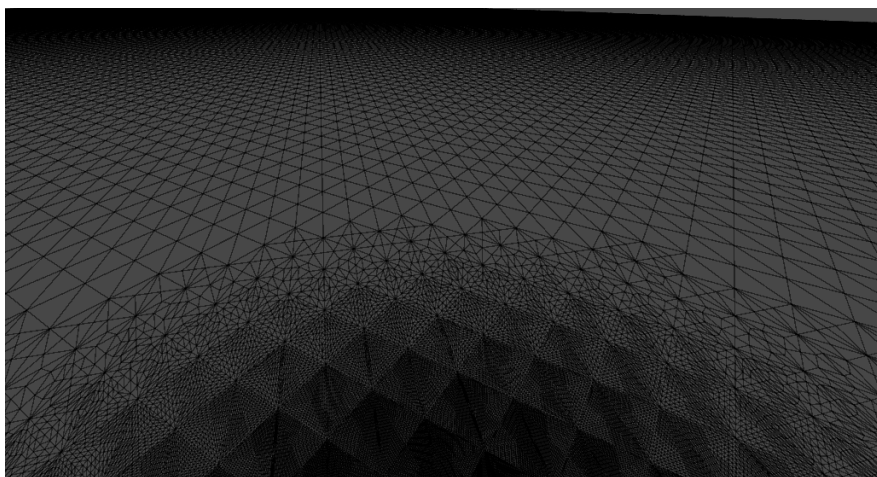
svakom sljedećem sloju postavljamo donju granicu valnog broja jednaku najvećem valnom broju prethodnog sloja, definiran u formuli poviše.^[6]

2.1.4 Teselacija

Kako bi postigli realističnost valova, potrebna je velika rezolucija plohe na kojoj su prikazani. No kako ne bismo gubili na performansama, u ovoj situaciji prikladno je koristiti tehniku teselacije kako bi se postigla velika rezolucija dijela plohe blizu promatraču i prihvatljivo mala rezolucija na većim udaljenostima. Faktor teselacije ovisi o duljini rubova trokuta i udaljenosti od promatrača:

$$\text{faktor_teselacije} = \text{duljina_ruba} \cdot \exp(-\text{udaljenost} \cdot 0.1)$$

Na slici 2.4 vidljiv je efekt teselacije nad plohom u ovisnosti o blizini kamere.



Slika 2.4: Teselacija plohe mora

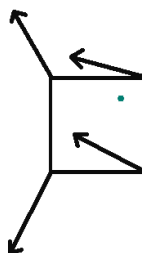
2.2 Eroziya Terena

2.2.1 Perlinov šum

Početno stanje terena generirano je algoritmom Perlinovog šuma. Iako sam algoritam nije implementiran u kodu (korišten je već-implementiran), za potpunost dokumentacije ukratko je opisan u nastavku.

Algoritam Perlinovog šuma generira kontinuirani pseudoslučajni niz (ili polje) brojeva, te se često koristi za proceduralno generiranje sadržaja u računalnoj grafici. Sam algoritam prima poziciju u n-dimenzionalnom prostoru, te ovisno o veličini, uzima 2^n rubnih točaka mreže prostora.

Za svaki rub generira se vektor gradijenta koji je pseudoslučajni vektor u ovisnosti o koordinati rubne točke.



Slika 2.5: Vektori gradijenta

Vektori gradijenta skalarno se množe s vektorom razlike točke ruba i ulazne točke, čime se dobije vrijednost ruba (uobičajeno skalirana na raspon $[-1,1]$).

Simulacija i vizualizacija prirodnih procesa u interaktivnom 3D okruženju	Verzija: 1.0
Tehnička dokumentacija	Datum: 20/01/26

Vrijednosti ruba zatim se interpoliraju (počevši od 0), u ovisnosti o udaljenosti njihove pozicije u prostoru od ulazne točke. Linearna interpolacija često izgleda preoštro, pa se u tim slučajevima uzimaju "glađe" funkcije višeg reda. Primjer je funkcija "smoothstep":

$$3x^2 - 2x^3, \text{ gdje je } x \text{ u rasponu } [0,1]$$

Zbroj ovako interpoliranih vrijednosti je završna vrijednost šuma.

Za malo prirodniji izgled u ovoj simulaciji korišteno je nekoliko slojeva šuma, gdje se na početni šum značajne veličine mreže dodaje nekoliko manjih šumova s proporcionalnom težinom.

2.2.2 Simulacija kišne erozije

Simulacija erozije u ovom virtualnom okruženju postiže se postupkom u kojem se na mapu visine (2-dimenzionalno polje brojeva gdje svaki broj predstavlja visinu terena u točki) značajan broj iteracija izvodi simulacija kapljice vode (ili neke druge tekućine). Kapljica se stvara na nasumičnoj poziciji te putuje nizvodno, usputno skupljajući te odlažući sediment^[7].

Jedna kapljica opisana je svojom pozicijom, brzinom, volumenom i postotkom volumena koji je sediment.

Nakon stvaranja na nasumičnoj poziciji, simulacija kapljice izvodi se iterativno (početna konfiguracija je 50 iteracija), gdje se svaku iteraciju događa sljedeće:

- kapljici je dodana brzina ovisno o normali terena na njezinoj poziciji
- kapljica se miče ovisno o svojoj brzini
- brzina se smanjuje ovisno o trenju
- računa se koncentracija ekvilibrija ovisno o brzini čestice (što je čestica brža, to je veća)
- ukoliko je postotak sedimenta kapljice manji od ekvilibrija, kapljica skuplja sediment s trenutne pozicije terena, smanjujući vrijednost na mapi visine.
- ako je postotak veći, kapljica gubi sediment te se vrijednost visinske mape povećava
- kapljica gubi nešto volumena isparavanjem.

Kako bi rezultati postali vidljivi, navedeni algoritam trebao bi se izvoditi nekoliko puta za jednu česticu, te nekoliko tisuća puta sveukupno.

2.2.3 Bojanje visinske mape

Bojanje mape postiže se jednostavnim sjenčarom. Ovaj sjenčar služi za dinamičko bojanje terena bilo kojeg oblika, ovisno o nagibu terena u danoj točki.

Sjenčar učitava dvije teksture (jednu za travu – *grass* i jednu za kamenje – *rock*), te dvije vrijednosti za mekani prijelaz između te dvije teksture (prag *SlopeThreshold* i parametar *SlopeSmoothing*).

Boja samog terena ovisi o nagibu terena, a nagib se može izračunati skalarnim umnoškom normale terena u nekoj točki sa vektorom koji gleda prema gore. Ovaj umnožak daje broj u intervalu [0,1] koji govori koliki je nagib terena (vrijednosti blizu 0 označavaju litice i strmi teren, a vrijednosti bliže 1 označavaju ravni teren).

Prema toj vrijednosti uzima se boja sa teksture trave ili teksture kamena. Ako je vrijednost nagiba dovoljno blizu praga (vrijednosti *SlopeThreshold* i *SlopeSmoothing*), djelomično će se prikazati obje teksture radi mekanog prijelaza (boju između dvije teksture određuje metoda *lerp*).

Simulacija i vizualizacija prirodnih procesa u interaktivnom 3D okruženju	Verzija: 1.0
Tehnička dokumentacija	Datum: 20/01/26

2.3 Ponašanja Vegetacije

2.3.1 Trava

Pri simulaciji i vizualizaciji trave u bilo kojem virtualnom okruženju, potrebno je razumjeti da trava kao pojam za simulirati nikada nije jedan objekt: trava je skupina ogromne količine odvojenih objekata travki od kojih svaka može na svoj način reagirati na stimulans. Reakcija svake travke na stimulans ovisi o udaljenosti i smjeru stimulansa, poziciji travke u svijetu, normalu površine na kojoj se travka nalazi i dodatne manje aspekte.

Korištenje 'GameObject' objekta unutar okruženja Unity kako bi se svaka travka simulirala moguće je za malenu količinu travki, nedovoljnu za simuliranje prostrane livade. Ali korištenje jednog 'GameObject' objekta koji vodi simulaciju i vizualizaciju svake travke omogućuje znatno veći broj travki za prikaz.

Pri implementaciji trave u ovom virtualnom okruženju korišten je sljedeći algoritam opisan pseudokodom koji se ponavlja svaki prikaz:

```

ako(kamera.pomaknuta):
    ako(kamera.trenutni_XZsegment!=kamera.prošli_XZsegment):

ažuriraj_aktivne_XZsegmente()

    spremnik_vidljivih_instanci.count = 0
    za(segment : aktivni_XZsegmenti):
        ako(segment.unutar_vidnog_polja(kamera)):
            segment.dodaj_vidljive_instance(spremnik_vidljivih_instanci)

prikaži_indirektno(spremnik_vidljivih_instanci)

```

Ažuriranje aktivnih segmenata provodi se unutar objekta klase 'ChunkHandler' koji prvo uklanja XZ segmente koji su na XZ osi previše udaljeni od kamere, a zatim dodaje segmente koji su sada dovoljno blizu kameri.

Kada se neki XZ segment u svijetu stvori, on generira instance trave koje se nalaze unutar njegovih granica. Sama instanca trave je struktura koja sadrži dva vektora (pozicija u svijetu i normala površine). Generiranje instanci trave rađeno je pomoću sljedećeg pseudokoda koji je razbijen na nekoliko prikaza kako se prikazivanje ne bi kočilo:

```

za(i od 0 do broj_instanci):
    pozicija = random_XZ_pozicija(i+chunk_seed)
    hit = baci_zraku(pozicija, dolje)
    ako(hit != null):
        polje_instanci.spremi_iduće(hit.pozicija, hit.normala_površine)

```

Dodavanje instanci travke u spremnik vidljivih travki radi se svaki prikaz u kojem se kamera pomakla, a svaki XZ segment svijeta to radi pomoću '.compute' sjenčara nad svojim spremnikom svih instanci. Ovaj sjenčar radi provjeru nalazi li se obuhvaćajuća sfera svake travke unutar vidnog polja kamere i u slučaju kada se nalazi stavlja travku na kraj spremnika vidljivih instanci. Ovaj sjenčar omogućuje i deterministički probabilističko uklanjanje udaljenije trave kako bi prikazivanje bilo brže.

Kada je spremnik vidljivih instanci napunjen, sve instance se prikazuju pomoću posebnog '.shader' sjenčara koji uz premještanje radi i simulaciju vjetra pomoću jednostavnog 2-dimenzionalnog šuma.

Uz ovakav 'normalan' prikaz trave, u virtualnom okruženju igrač može gaziti po travi s čime se travke savijaju pod njim, a to je opisano sljedećim pseudokodom:

Simulacija i vizualizacija prirodnih procesa u interaktivnom 3D okruženju	Verzija: 1.0
Tehnička dokumentacija	Datum: 20/01/26

```

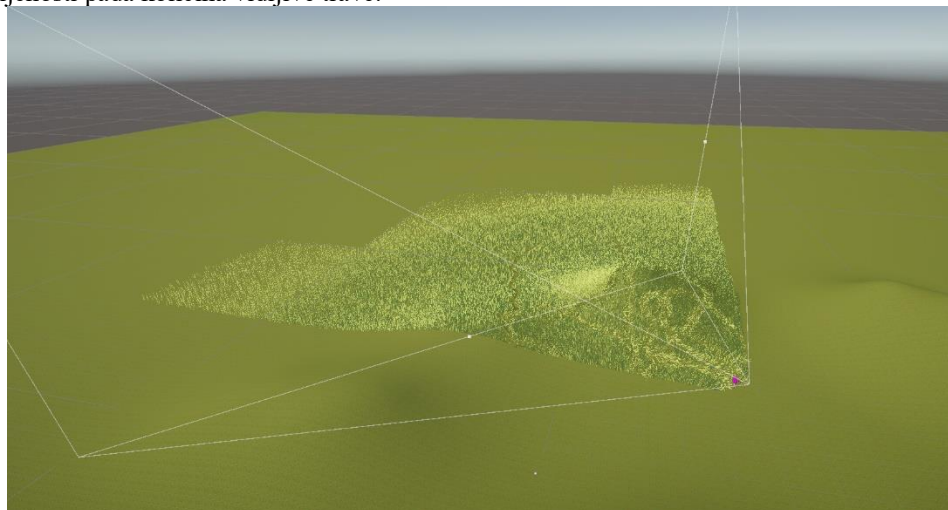
ako(igrač.pomaknut):
    spremnik_vidljivih_instanci.count = 0
    za(segment : aktivni_XZsegmenti):
        ako(segment.unutar_dosega_smetnje(smetnja)):
            segment.dodaj_smetnju(smetnja)
        ako(segment.unutar_vidnog_polja(kamera)):
            segment.dodaj_vidljive_instance(spremnik_vidljivih_instanci)

prikaži_indirektno(spremnik_vidljivih_instanci)

```

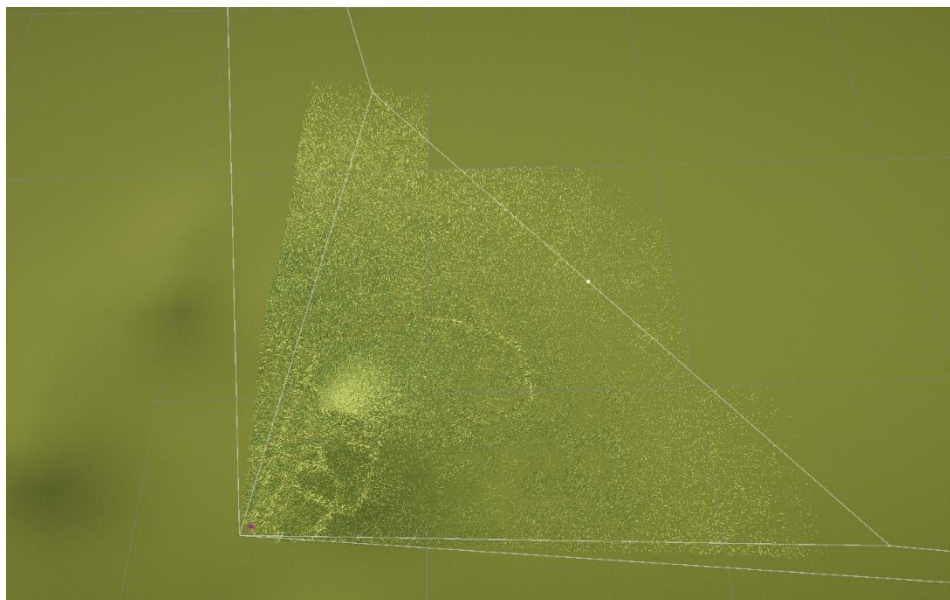
Dodavanje smetnje u spremnik svih instanci nekog XZ segmenta radi se pomoću 'compute' sjenčara u kojem se pomoću informacija smetnje računa novi smjer normale u kojoj se travka nalazi (često se travka polegne na ravninu tangente površine na kojoj se nalazi kao da ju je igrač pregazio). Taj novi smjer normale se spremi umjesto stare unutar spremnika svih instanci. Ovo računanje radi se za sve travke unutar spremnika, ali samo one na koje utječe smetnja imaju promijenjene normale nakon zvanja sjenčara.

Na slici 2.6 može se vidjeti kako su vidljive samo one instance trave koje se nalaze unutar krnjeg stošca kamere (vidnog polja). Na slici 2.7 vidi se ista scena, ali druga perspektiva na kojoj je moguće vidjeti XZ segmente svijeta, te kako s udaljenosti pada količina vidljive trave.



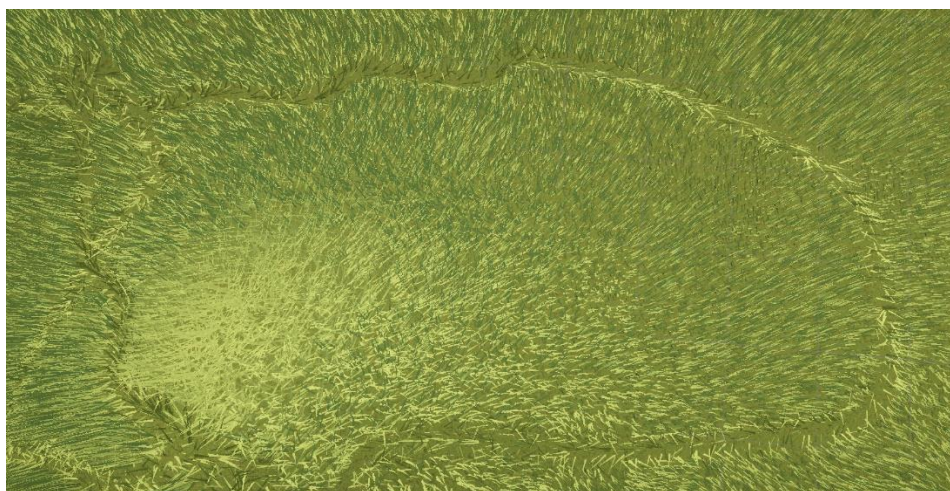
Slika 2.6: Instance trave i krnji stožac kamere, pogled sa strane

Simulacija i vizualizacija prirodnih procesa u interaktivnom 3D okruženju	Verzija: 1.0
Tehnička dokumentacija	Datum: 20/01/26



Slika 2.7: Instance trave i krnji stožac kamere, pogled odozgo

Na slici 2.8 može se vidjeti put kojeg je igrač napravio u livadi, te kako je trava po kojoj je igrač hodao sada drugačija od trave bez smetnji.



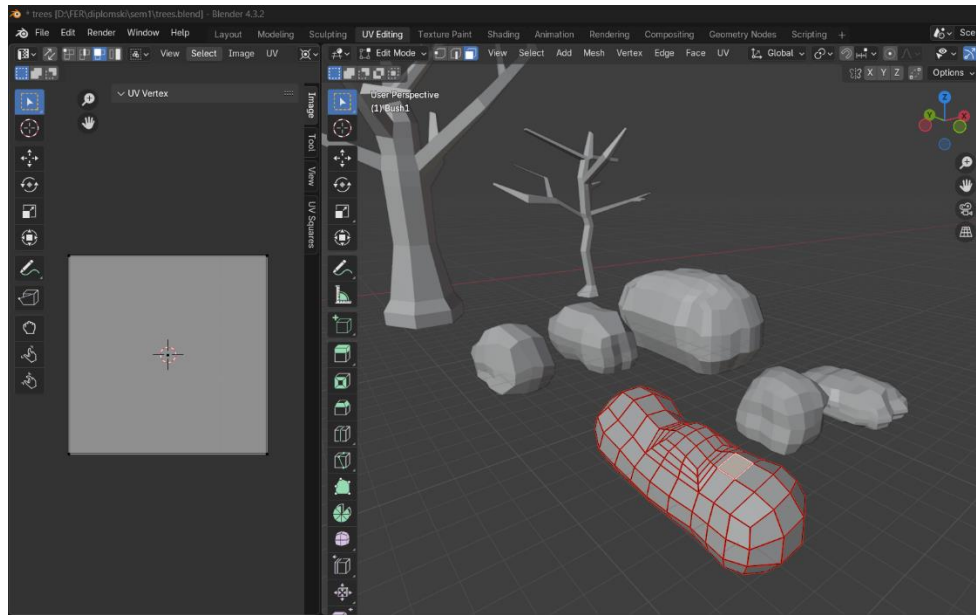
Slika 2.8: Trava i put napravljen hodanjem po istoj

2.3.2 Krošnje

Simulacija ponašanja krošnja drveća i grmlja je prije svega složen proces koji zahtjeva dobro razumijevanje samog koncepta modela koji će se koristiti u ovome projektu te će biti jednak za krošnje i grmlja.

U svrhu projekta, izrađeni su modeli krošnja i grmlja sa određenim specifikacijama te i samo drveće na koje će se modeli krošnji stavljati, vidljivo na slici 2.9.

Simulacija i vizualizacija prirodnih procesa u interaktivnom 3D okruženju	Verzija: 1.0
Tehnička dokumentacija	Datum: 20/01/26

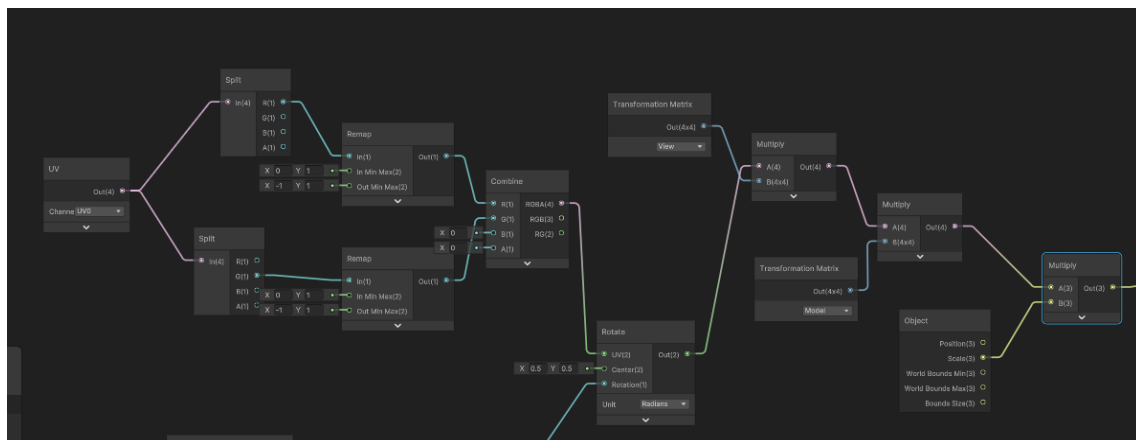


Slika 2.9: Modeli krošnji, grmlja, drveća

Ideja simulacije bazira se na tehnici panoa na mreži poligona modela u kojoj je osnovni primitiv četverokut. Modeli su se, iz ovog razloga, morali posebno konfigurirati u programu Blender na način da se najprije iz primitiva kocke napravi niz operacija podjele (engl. subdivision) i ekstruzije (engl. extrusion) u željeni oblik. Nakon, primjenom još jedne operacije podjele postiže se zaglađivanje (engl. smoothness) te se dobivaju prirodniji oblici za krošnje.

Ključna stvar je da podjelom dobivamo mrežu kvadrata koja se treba raspakirati u UV uređivaču. Na slici 2.9 označen je jedan model i njegova UV konfiguracija. Ovaj korak je je vrlo važan za daljnju funkcionalnost sjenčara u programu Unity.^[8]

Tehnika panoa je način 3D prikaza u grafici u kojemu se uzima 2D element (najčešće u obliku pravokutnika) te se on rotira prema kameri u stvarnome vremenu iscrtavanja. Karakteristično za tu tehniku je da se panou nikada neće vidjeti rub te se radi o stvarnoj geometriji koja je u potpunosti ravna, no dočarava dubinu zbog rotacije prema promatraču. Model krošnje u ovome projektu je nakupina puno manjih panoa koji su reprezentirani pomoću UV koordinata mreže modela, a sjenčaru je namijenjeno da bude zadužen da svaki pano bude ispravno usmjeren.



Slika 2.10: Dio grafa sjenčara za prikaz drveća

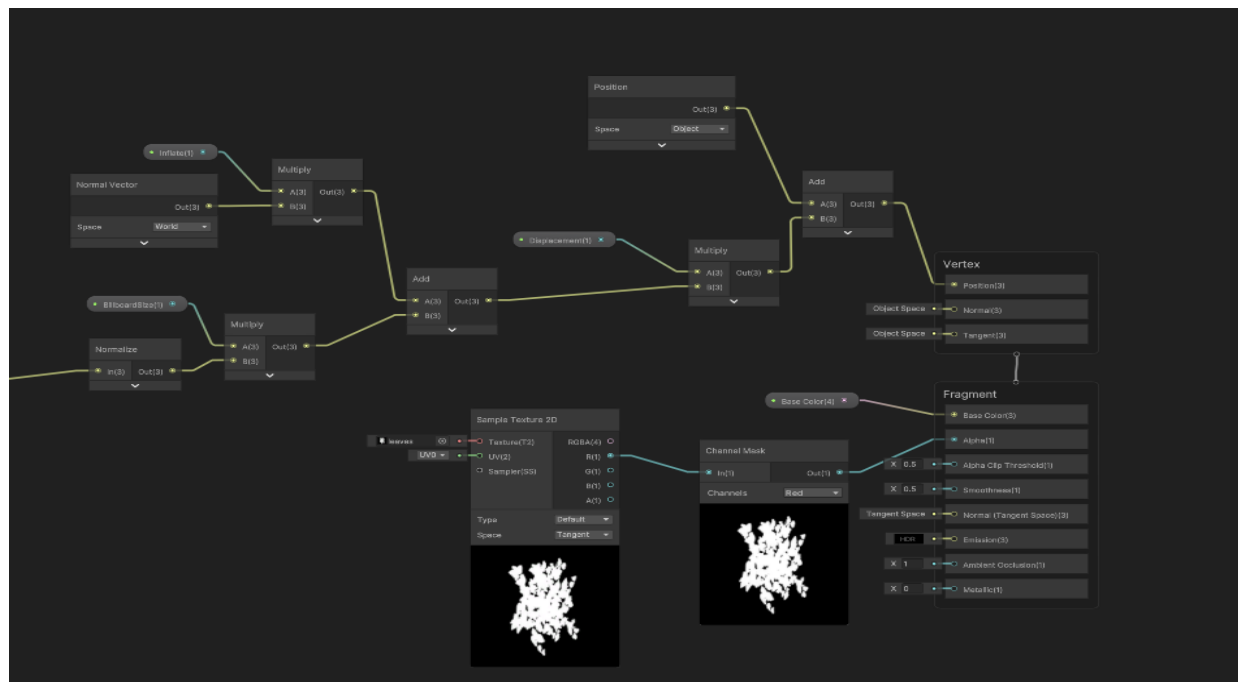
Slika 2.10 prikazuje dio sjenčara koji koristi UV koordinate mreže kao pomoćni izvor podataka za definiranje lokalnog

Simulacija i vizualizacija prirodnih procesa u interaktivnom 3D okruženju	Verzija: 1.0
Tehnička dokumentacija	Datum: 20/01/26

položaja pojedinih četverokutnih elemenata. UV koordinate se ne koriste za uzorkovanje teksture, već kao matematička reprezentacija ravnine panoa unutar koje se provodi rotacija. Kako bi se omogućila pravilna obrada, UV komponente se razdvajaju pomoću čvora Split te se svaka komponenta zasebno preslikava čvorom Remap iz raspona [0, 1] u raspon [-1, 1]. Time se UV prostor centriran oko ishodišta prilagođava daljnjim transformacijama.^[8]

Nakon preslikavanja, komponente se ponovno spajaju čvorom Combine u vektor koji predstavlja lokalnu koordinatu vrha unutar panoa. Tako dobiveni vektor se zatim rotira pomoću čvora Rotate oko središta panoa, definiranog vrijednošću (0.5, 0.5), čime se osigurava da se rotacija odvija simetrično oko sredine svakog četverokuta. Kut rotacije izražava se u radijanima, što omogućuje preciznu kontrolu nad orijentacijom panoa.^[8]

Rezultat rotacije se potom transformira pomoću čvorova Transformation Matrix kako bi se uskladio s koordinatnim sustavom pogleda i prostora modela. Time se osigurava da izračunata rotacija pravilno reagira na položaj i orijentaciju kamere. Na kraju se transformirani vektor kombinira s pozicijom objekta dobivenom iz Object čvora, čime se omogućuje da se rotacija panoa primjenjuje relativno u odnosu na sam model, neovisno o njegovoj transformaciji u sceni.^[8]



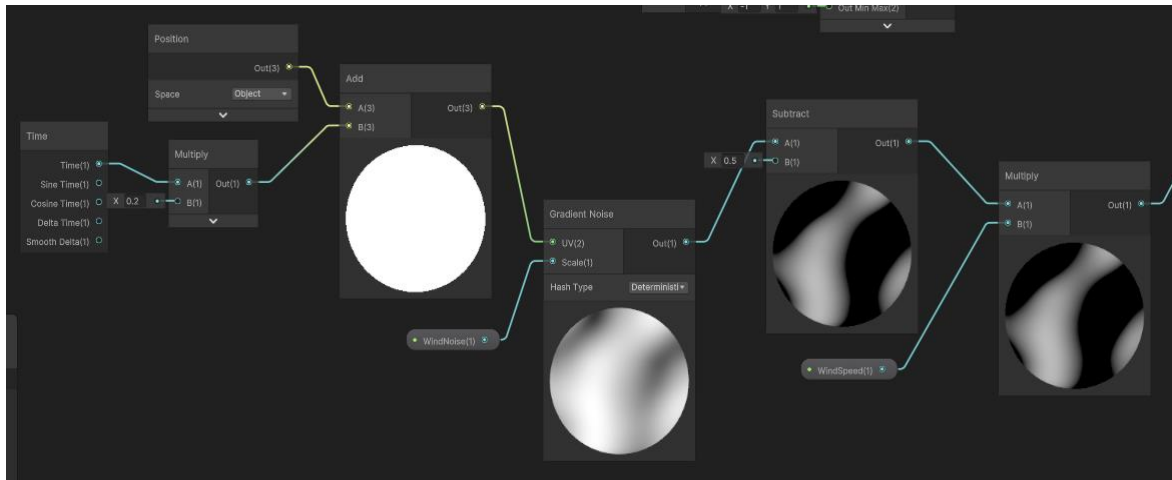
Slika 2.11: Dio grafa sjenčara za prikaz drveća

Slika 2.11 prikazuje dio sjenčara gdje se definira završna pozicija vrhova panoa te vizualna obrada fragmenata. Nakon određivanja orijentacije panoa, uvodi se dodatni pomak vrhova kako bi se razbio dojam ravne geometrije i postigla veća prostorna raznolikost. U tu svrhu koristi se vektor normale u svjetskom prostoru, koji se množi s parametrom Inflatija, čime se omogućuje kontrolirano proširivanje panoa u smjeru normale. Dodatno, normalizirani smjer rotacije panoa se skalira parametrom Billboard Size kako bi se regulirala njihova veličina i raspored unutar volumena.^[8]

Dobiveni vektori pomaka se zbrajaju i skaliraju parametrom Displacement, čime se omogućuje jedinstvena kontrola intenziteta ukupnog pomaka. Tako izračunati pomak se pribraja izvornoj poziciji vrha u objekt prostoru, čime se definira konačna pozicija koja se prosljeđuje u vršni dio sjenčara.^[8]

U fragment dijelu sjenčara uzorkuje se tekstura lišća^[9] s alfa kanalom, pri čemu se alfa vrijednost koristi za odrezivanje rubova panoa pomoću Alpha Clip Threshold parametra. Time se uklanja pravokutni oblik geometrije i postiže prirodniji izgled vegetacije. Osnovna boja definira se parametrom Base Color, dok su ostali materijalni parametri postavljeni na konstantne vrijednosti radi zadržavanja stiliziranog izgleda i jednostavnosti sjenčanja.^[8]

Simulacija i vizualizacija prirodnih procesa u interaktivnom 3D okruženju	Verzija: 1.0
Tehnička dokumentacija	Datum: 20/01/26



Slika 2.12: Dio grafa sjenčara za prikaz drveća

Na slici 2.12 vidi se dio sjenčara u kojemu se uvodi simulacija vjetra s ciljem dodatnog povećanja vizualne uvjerljivosti krošnji i grmlja. Budući da je vegetacija u prirodi rijetko potpuno statična, suptilno gibanje panoa doprinosi osjećaju živosti scene te smanjuje dojam umjetne pravilnosti koji može nastati kod statičnih modela.^[10]

Efekt vjetra ostvaruje se korištenjem vremenske komponente koja se dodaje položaju vrhova, čime se postiže kontinuirana promjena uzorka u vremenu. Na tako dobivene koordinate primjenjuje se gradijentni šum koji generira glatke, niskofrekventne varijacije, pogodne za simulaciju sporog i prirodnog gibanja. Dobiveni šum se dodatno skalira parametrima koji kontroliraju jačinu i brzinu vjetra, nakon čega se koristi kao dodatni pomak vrhova. Na taj način se postiže efekt blagog njihanja panoa, koji vizualno odgovara ponašanju lišća i manjih grana pod utjecajem vjetra.^[10]

Simulacija i vizualizacija prirodnih procesa u interaktivnom 3D okruženju	Verzija: 1.0
Tehnička dokumentacija	Datum: 20/01/26

3. Upute za korištenje

Sve izvršne datoteke nalaze se na web adresi: <https://github.com/vito-vrbic/dipproj>

3.1 Dinamika Vode

Za korištenje izvršne datoteke virtualnog okruženja dinamike vode, potrebno je otvoriti izvršnu datoteku nakon čega je moguće upravljati letećom kamerom pomoću sljedećih tipki:

- 'W' – Kretanje naprijed
- 'A' – Kretanje lijevo
- 'S' – Kretanje nazad
- 'D' – Kretanje desno
- 'E' – Kretanje gore
- 'Q' – Kretanje dolje
- 'Shift' – ubrzano kretanje kamere

3.2 Erozija Terena

Za korištenje izvršne datoteke virtualnog okruženja erozije terena potrebno je otvoriti izvršnu datoteku nakon čega je moguće upravljati simulacijom pomoću pokazivača i grafičkog sučelja gumba i sljedećih tipki:

- 'A' ili lijeva strelica – Okretanje kamere ulijevo
- 'D' ili desna strelica – Okretanje kamere udesno

Pritiskom na polja unosa imena datoteke ili početnog sjemena Perlinovog šuma slova i brojeve potrebno je unijeti tipkovnicom.

3.3 Ponašanja Vegetacije

Za korištenje izvršne datoteke virtualnog okruženja ponašanja vegetacije, potrebno je otvoriti izvršnu datoteku nakon čega je moguće upravljati likom i kamerom pomoću sljedećih tipki:

- 'W' – Kretanje naprijed
- 'A' – Kretanje lijevo
- 'S' – Kretanje nazad
- 'D' – Kretanje desno
- 'Shift' – Dok se tipka drži, lik trči
- 'Left Mouse' – Pritiskom jedanput pokazivač nestaje i kamera se kreće orbitalno micanjem miša
- 'Escape' – Ako pokazivač nije vidljiv, pritiskom ove tipke postaje vidljiv. Ako pokazivač je vidljiv, pritiskom ove tipke, aplikacija se gasi.

Simulacija i vizualizacija prirodnih procesa u interaktivnom 3D okruženju	Verzija: 1.0
Tehnička dokumentacija	Datum: 20/01/26

4. Literatura

- [1] Jerry Tessendorf, *Simulating Ocean Water*, 2004
- [2] Christopher J. Horvath, *Empirical Directional Wave Spectra for Computer Graphics*, ACM, 2015
- [3] Fynn-Jorin Flügge, *Realtime GPGPU FFT Ocean Water Simulation*, TUHH, 2017
- [4] gasgiant, FFT-Ocean, <https://github.com/gasgiant/FFT-Ocean>
- [5] Mark Mihelich, Tim Tcheblovkov, *Wakes, explosions and lighting: Interactive water simulation in 'Atlas'*, GDC, 2019
- [6] Gamper, T. *Ocean Surface Generation and Rendering*. Diplomski rad. TU Wien, 2018.
- [7] McDonald, N. *Simple Particle-Based Hydraulic Erosion*, <https://nickmed.me/2020/04/10/simple-particle-based-hydraulic-erosion/> (pristupljeno 18.1.2026.)
- [8] – Pontus Karlsson, *Fluffy stylized trees tutorial, using quadmesh-to-billboards shader in Unity*, YouTube video, <https://www.youtube.com/watch?v=iASMFba7Gel&t=625s>, pristupljeno 11. 11. 2025.
- [9] – Lil Puff Puff studios, *Leaf Texture Tutorial: Creation and Application*, YouTube video, https://www.youtube.com/watch?v=LQOfTOWxT_w, pristupljeno 13. 11. 2025.
- [10] – Unity Discussions, *URP wind shader graph*, Unity, <https://discussions.unity.com/t/urp-wind-shader-graph/857589>, pristupljeno: 28.11.2025.