Praćenje pokreta i integracija u open-source programu Blender

Grgić, Anto

Undergraduate thesis / Završni rad

2021

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University North / Sveučilište Sjever**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:122:262232

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-04-25**

Repository / Repozitorij:

University North Digital Repository

zir.nsk.hr

dabar
DIGITALNI AKADEMSKI ARHIVI I REPOZITORIJI

Sveučilište
Sjever

Završni rad br. 714/MM/2021

Praćenje pokreta i integracija u open-source programu Blender

Anto Grgić, 2237/336

Varaždin, lipanj 2021. godine

Sveučilište
Sjever

Odjel za Multimediju, oblikovanje i primjenu

Završni rad br. 714/MM/2021

Praćenje pokreta i integracija u open-source programu Blender

Student

Anto Grgić, 2237/336

Mentor

doc.dr.sc. Andrija Bernik

Varaždin, lipanj 2021. godine

Sveučilište Sjever
Sveučilišni centar Varaždin
104. brigade 3, HR–42000 Varaždin

UNIVERSITY NORTH

Prijava završnog rada

Definiranje teme završnog rada i povjerenstva

ODJEL Odjel za multimediju

STUDIJ preddiplomski stručni studij Multimedija, oblikovanje i primjena

PRISTUPNIK Anto Grgić | MATIČNI BROJ 2237/336

DATUM 30.4.2021. | KOLEGIJ 3D modeliranje

NASLOV RADA Praćenje pokreta i integracija u open-source programu Blender

NASLOV RADA NA ENGL. JEZIKU Motion tracking and integration in open source program Blender

MENTOR doc.dr.sc. Andrija Bernik | ZVANJE Docent

ČLANOVI POVJERENSTVA

1. mr.sc. Dragan Matković, v. pred. - predsjednik
2. pred. Nikola Jozić - član
3. doc.dr.sc. Andrija Bernik - mentor
4. doc.art. Robert Geček - zamjenski član
5.

Zadatak završnog rada

BROJ 714/MM/2021

OPIS

Osnovni zadatak Završnog rada je opisati korake izvlačenja podataka iz sekvence slika (videomaterijala) kroz motion tracking u Blenderu. Gotovo svaki sustav za praćenje pokreta kreće s otkrivanjem pokreta. Motion tracking ima za cilj odvajanje potrebnih i odgovarajućih područja pokretnih objekata od pozadinske slike, odnosno izdvajanje relevantnih informacija iz slike (videomaterijal se sastoji od sekvence slika pri određenoj brzini). Relevantni podaci prikupljeni metodama opisanim u radu, nadalje se mogu integrirati u daljnje kreativne procese poput manipulacije dotičnog videomaterijala (stabilizacija, micanje predmeta, projekcija 3D modela), kao i dodavanje 3D scene ili likova u video što se često koristi u filmskoj industriji, no ne i isključivo toj industriji. Nakon opisa teorijskog dijela izvedbe, prikazati ću i opisati svoj primjer i vid integracije istog u poslovnoj sferi (skica).

U radu je potrebno:
1. Opisati sustave za praćenje pokreta i usporediti osnovni algoritam izvlačenja relevantnih podataka iz videomaterijala općenito i u programu Blender 2.92.
2. Detaljan opis alata, postupaka različitih metoda motion trackinga u Blenderu
3. Opisati primjer i definirati primjenu - analiza integracije

ZADATAK URUČEN 30.4.2021. | POTPIS MENTORA

REPUBLIKA HRVATSKA SVEUČILIŠTE SJEVER KOPRIVNICA SVEUČILIŠTE SJEVER

Sažetak

Predmet završnog rada je teorijski opis rada sustava za praćenje pokreta i praktična primjena praćenja pokreta. Cilj rada bio je istražiti, analizirati i primijeniti metode i principe rada sustava za motion tracking.

Gotovo svaki sustav za praćenje pokreta kreće s otkrivanjem značajki i pokreta na slici. Za kvalitetnu izvedbu praćenja pokreta (eng. *Motion tracking*), važno je prikupljanje relevantnih, konkretnih i točnih podataka iz videozapisa metodama opisanim u radu. Cilj motion trackinga je odvajanje i praćenje željenih i odgovarajućih područja pokretnih objekata od pozadinske slike, odnosno izdvajanje relevantnih informacija iz slike (videomaterijal se sastoji od sekvence slika pri određenoj brzini). Relevantni podaci prikupljeni metodama opisanima u radu, nadalje se mogu integrirati u daljnje kreativne procese poput manipulacije videomaterijala (stabilizacija, micanje predmeta, projekcija 3D modela), kao i dodavanje 3D scene ili likova u video što se često koristi u filmskoj industriji. U praktičnom dijelu zadatka, prikazat ću i opisati svoj primjer integracije praćenja pokreta. Rad je baziran u open source programu Blender 2.92.

U radu je potrebno:

1. Opisati sustave za praćenje pokreta, tj. algoritama za praćenje

2. Detaljan opis alata, postupaka i različitih metoda motion trackinga u Blenderu 2.92

3. Opisati primjer i definirati primjenu - analiza integracije

Ključne riječi: praćenje pokreta, videozapis, slika, 3D model, manipulacija, stabilizacija, Blender

Abstract

The subject of this undergraduate thesis is a theoretical description of the operation of motion tracking systems and the practical application of motion tracking. The aim of the paper was to research, analyze and apply the methods and principles of operation of motion tracking systems.

Almost every motion tracking system starts with the detection of features and movements in the image. In order to have quality motion tracking, it is important to collect relevant, concrete and accurate data from video content using methods described in the paper. The goal of motion tracking is to separate and track the desired and appropriate areas of moving objects from the background images, i.e. to extract relevant information from the images (the video material consists of a sequence of images at a certain speed). Relevant data collected by the methods described in the paper can be further integrated into creative processes such as video manipulation (stabilization, object moving, 3D model projection), as well as adding 3D scenes or characters to video, which is often used in the film industry. In the practical part of the paper, I will present and describe my example and the way of its integration. The paper is based in the open source program Blender 2.92.

The paper describes:

1. Motion tracking systems, i.e. tracking algorithms

2. Detailed description of tools, procedures and different methods of motion tracking in Blender 2.92

3. Practical example and defining the application - integration analysis

Keywords: motion tracking, video, image, 3D model, manipulation, stabilization, Blender

Popis korištenih kratica

*2***D**	*dvodimenzionalno*
	kratica koja označava drugu dimenziju
3D	*trodimenzionalno*
	kratica koja označava treću dimenziju
LOC	*eng. Location motion model*
LOCROT	*eng. Location rotation model*
LOCROTSCALE	*eng. Location rotation scale model*

Sadržaj

1. Uvod 1

2. Obrada slike i sustavi za praćenje pokreta 3

2.1. Osnovni sustavi za praćenje pokreta....3
2.2. Digitalna slika i pikseli 5
2.2.1. Digitalna slika 8
2.2.2. Dubina boje....9
2.3. Morfološka obrada slike 12
2.3.1. Uvod u morfološku obradu slike 12
2.3.2. Morfološke operacije Erode i Dilate....14

3. Prepoznavanje značajki....16

3.1.1. Značajka na slici 17
3.2. Algoritmi za prepoznavanje značajki....19
3.3. Kanade-Lucas-Tomasi algoritam za praćenje pokreta....21
3.3.1.The registration problem....21

4. Uvod u Motion tracking u Blenderu 23

4.1. Movie Clip Editor - radno okruženje za motion tracking 24
4.1.1. Clip View 26
4.1.2. Graph view....29

5. Motion tracking - praktični dio 30

6. Integracija podataka dobivenih motion trackingom - Orijentacija scene i rekonstrukcija..36

7. Zaključak....39

8. Literatura....40

9. Prilozi 42

10. Slike 43

1. Uvod

Praćenje pokreta (eng. *Motion tracking*), softverski je proces gdje se iz snimke izvlače podaci o kretanju kamere za vrijeme snimanja. Ova tehnika omogućuje umetanje računalno generirane grafike (eng. *Computer-generated imagery*, CGI) u snimke s ispravnim položajem, orijentacijom i kretanjem u odnosu na snimane predmete u kadru. Ukoliko je motion tracking odrađen kvalitetno, umetnuti CGI u snimanom kadru izgledat će profesionalno i uvjerljivo, upravo zbog usklađene perspektive.

U teoretskom dijelu rada opisat će se gradivni blokovi sustava za praćenje pokreta koji su bitni za razumijevanje načela rada motion trackinga poput:

- Opisa digitalne slike i piksela
- Morfološke obrade slike
- Osnovnih sustava za motion tracking
- Algoritama za prepoznavanje značajki i motion tracking

Vrsta tehnologije i obrade koja će biti opisana u praktičnom dijelu zadatka može se smjestiti u kategoriju “zabavne” industrije, no postupkom koji će biti opisan moguće je primijeniti motion tracking u raznim sustavima.

Današnja primjena ove tehnologije zasniva se u mnogim područjima poput prepoznavanja lica, 3D rekonstrukcije objekata, navigacije robota, biomehanike, inžinjeringa s bespilotnim letjelicama, virtutalne realnosti, robotike i mnogim drugim istraživačkim djelatnostima. No ova vrsta prikupljanja podataka iz video snimaka nije nastala u posljednjih nekoliko godina, već se koristi od druge polovice 20. stoljeća u različitim granama industrije kao što je medicina - u svrhu rehabilitacije kod pacijenata koji su pretrpjeli moždani udar ili neke druge smetnje u motoričkoj funkciji. Rehabilitacija je dinamičan proces liječenja koji omogućuje pacijentima da vrate svoje funkcionalne motoričke sposobnosti u što većem postotku, a da bi se postigao ovaj cilj, aktivnosti pacijenta za vrijeme rehabilitacije moraju se kontinuirano nadzirati i naknadno ispravljati, pri čemu je motion tracking od iznimne važnosti [1].

Osim u medicini, praćenje pokreta koristi se često u filmskoj industriji, otkuda polazi i većina popularnosti modernog načina praćenja pokreta. 3D generirana računalna grafika (CGI) je grana računalne znanosti u kojoj se dostupna tehnologija i računalna snaga koriste za izradu visoko

realističnih vizualizacija i animacija, virtualnih scena, stvaranje likova itd. [2]. Tako se računalno generirana grafika koristi u mnogim umjetničkim granama, video igrama, računalnoj animaciji, vizualnim efektima u filmovima (eng. *Visual Effects*), televizijskim programima, reklamama i slično. Ne tako davno, su vizualni efekti bili "rezervirani" za jake produkcijske kuće i svjetske studije računalne grafike - zbog cijene računalne snage i dostupne tehnologije; no u današnje vrijeme, s neprestanim razvojem tehnologije i računalne snage, mogućnosti vizualnih efekata, upotpunjenih i omogućenih praćenjem pokreta, dostupne su sve široj javnosti.

Cilj ovog rada je istražiti tehniku motion trackinga, koje probleme rješava i koja rješenja nudi te što je u praktičnom smislu moguće učiniti putem motion trackinga. U teorijskom dijelu će biti opisani temeljni dijelovi sustava za motion tracking, kako jedan takav sustav radi i gdje se u primjeni može naći. U praktičnom dijelu rada biti će opisan proces motion trackinga u radnom okruženju Blender 2.92 te će se zaključno dati odgovor gdje se i kako može primijeniti opisana tehnologija, koja je besplatna i široko dostupna.

2. Obrada slike i sustavi za praćenje pokreta

2.1. Osnovni sustavi za praćenje pokreta

Svi sustavi za praćenje i detekciju pokreta gotovo su jednaki u načelu, odnosno prate nekoliko koraka koji vode ka jednakom cilju - praćenje pokreta te spremanje dobivenih podataka za kasniju obradu. U sljedećem odlomku biti će opisan osnovni algoritam kod praćenja pokreta, opisan u knjizi "A Block Matching Technique for Object Tracking Based on Peripheral Increment Sign Correlation Image" [3].

Prvi korak za bilo kakvo praćenje pokreta jest dobavljanje snimke objekta ili scene na kojoj želimo izvesti praćenje pokreta. Drugi korak je faza predobrade slike (eng. *image preprocessing*) u kojoj se slika obrađuje kako bi računalu bila bolja za praćenje. Dijelovi faze predobrade slike su: *gray scaling* (postupak pretvaranja slike u boji u crno-bijelu sliku kojom računalo može manipulirati) i takozvano zaglađivanje sive boje, smanjenje razlučivosti slike, *frame difference* (razlika između kadra k_1 i k_2), morfološke operacije (uključuje eng. *dilate, erode, open, close* i dr.*)* i označavanje (eng. *labeling)*. Navedeni koraci predobrade primjenjuju se kako bi se smanjio šum slike kako bi se postigla veća točnost praćenja. Nakon obrade i manipulacije, slika je spremna za prepoznavanje značajki i praćenje istih. Slika 1. preuzeta je iz navedene knjige te prikazuje dijagram tijeka opisan u ovom odlomku.

Slika 1. Shematski prikaz predobrade slike i detekcije objekata, preuzeto iz [3]

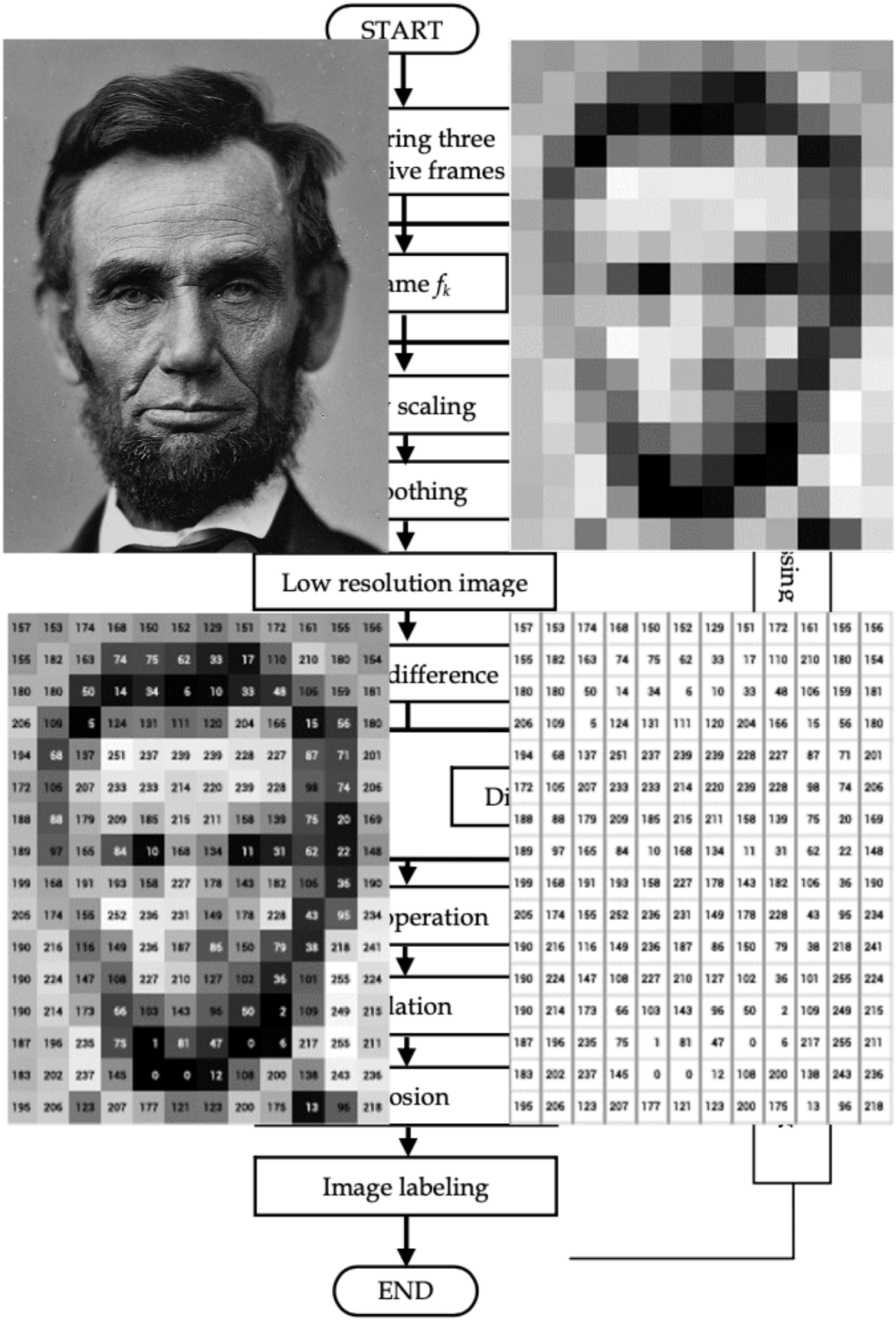
2.2. Digitalna slika i pikseli

Piksel (eng. *pixel, skraćeno od "picture element"*) je osnovni i najmanji deskriptivni element od kojeg se sastoji svaka digitalna slika. Svaki piksel ima određenu vrijednost svjetline. Pikseli su raspoređeni u mrežu, *raster*, odnosno matricu ili dvodimenzionalne nizove (eng. *array*). Digitalna slika je ograničena u razlučivosti prema broju piksela koji se koriste za prikazivanje iste te iz tog razloga svako povećavanje slike dovodi do njenog izobličavanja. [4]

Digitalnu sliku možemo zapisati kao f (*x,y*), pri čemu su *x* i *y* brojevi koji određuju poziciju svakog piksela na slici (f). Vrijednost *x* označava poziciju piksela u stupcima, a vrijednost *y* označava poziciju piksela u recima. Razlučivost slike je ukupan broj piksela *x* pomnoženo s ukupnim brojem piksela *y* [5]. Prema tome, digitalna slika, kako je računalo razumije, prikazana je na slici 2. i slici 3.

$$f(x,y) = \begin{bmatrix} f(1,1) & f(1,2) & \cdots & f(1,N) \\ f(1,1) & f(2,2) & \cdots & f(2,N) \\ \vdots & \vdots & \vdots & \vdots \\ f(N,1) & f(N,2) & \cdots & f(N,N) \end{bmatrix}$$

Slika 2. Prikaz zapisa digitalne slike na računalu, preuzeto iz [5]



Slika 3. Dijagram prikaza i zapisa piksela. Lijevo gore - slika Abrahama Lincolna; desno gore - pikselizirana slika; lijevo dolje - pikseli označeni brojevima od 0 do 255, predstavljajući njihovu svjetlinu; desno dolje - isti brojevi bez pozadinske slike, preuzeto sa [6]

Prikazano na slici 3. je digitalna slika kako je zapisuje i razumije računalo, dok na sljedećoj slici 4. vidimo prikaz rastera slike pri različitim razlučivostima, prikazanih kako ih mi vidimo na ekranu.

256x256 piksela

128x128 piksela

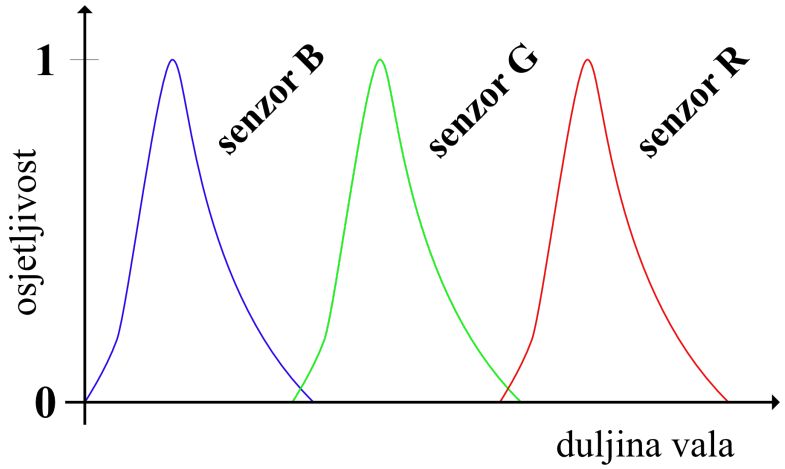
32x32 piksela

64x64 piksela

Slika 4. Prikaz digitalne slike sa smanjujućim brojem piksela po slici

2.2.1. Digitalna slika

Spectral band ili spektralna karakteristika osjetljivosti slikovnog senzora je raspon valnih duljina elektromagnetskog spektra na koje su senzori u fotoaparatima osjetljivi. Digitalne slike u boji sastoje se od tri krivulje (eng. *band*) koje su prikazane na grafu vidljivom na slici 8, odnosno tri primarne boje R (crvena), G (zelena) i B (plava). Svaka od ovih boja zabilježena je različitim senzorima osjetljivima isključivo na tu boju. Na slici 5 prikazan je primjer kromatskog grafa s tri različite krivulje, zabilježene s tri različita senzora. [5]



Slika 5. Spektralna karakteristika osjetljivosti slikovnog senzora s tri različita vala: R, G i B

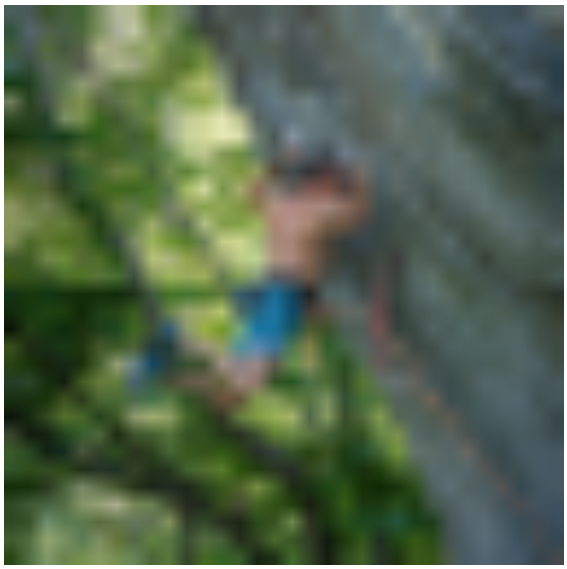
Većina algoritama za praćenje i otkrivanje značajki upotrebljava sive, odnosno monokromatske slike. Monokromatske izvedenice slika u boji sadrže gotovo isto informacija o slici kao slike u boji, no s monokromatskim slikama je lakše manipulirati. Isto tako, sve radnje koje se primjene na monokromatsku sliku, jednako tako se mogu primijeniti na svaku krivulju grafa iznad za sve tri primarne boje, na taj se način izvršava posao s trećinom procesorske snage. [5]

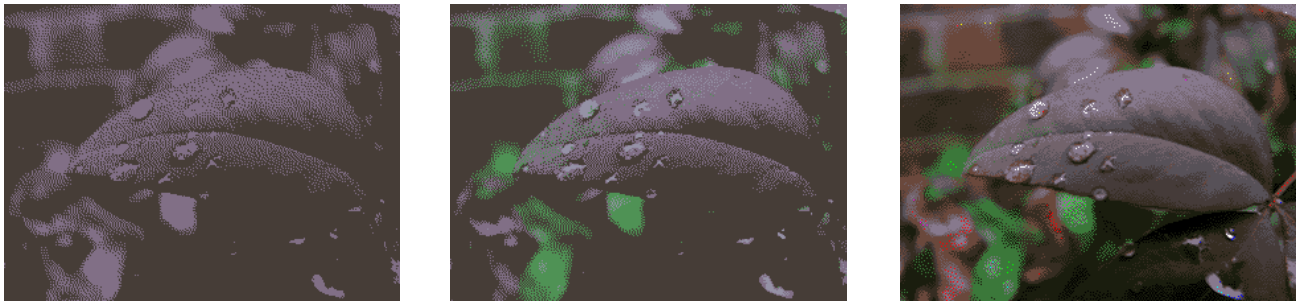
2.2.2. Dubina boje

Svaki piksel sadrži seriju brojeva koji opisuju njegovu boju ili intenzitet. Preciznost kojom piksel može specificirati boju naziva se dubina boje (eng. *color depth*).

Dubina boje, poznata kao i dubina bita, predstavlja broj bitova koji se koriste za svaku boju jednog piksela. Prema tome, dubina boje može se definirati kao preciznost kojom piksel prikazuje neku boju. Dubina boje, odnosno broj boja koje su pohranjene u slici jednak je 2^m pri čemu m označava broj bitova po potpikselu. Sukladno navedenome, 1-bitna slika može prikazati 2 boje: crnu i bijelu ili bilo koju boju u kombinaciji s crnom, 2-bitna slika može prikazati 4 boje, 4-bitna slika 16 boja, 8-bitna slika 256 boja itd. Slika koja sadrži 24 bita po pikselu naziva se slika u punoj boji (eng. *true color)*. Svaki piksel može poprimiti jednu od 16,777,126 boja (2^{24}) - 8 bita za crvenu, zelenu i plavu boju, odnosno 256 nijansi crvene, zelene i plave boje. Na slici 6 na sljedećoj stranici prikazana je slika pri pet različitih dubina boje. [6]

Ljudsko oko može razlikovati do 10 milijuna boja, što je manje od boja reproduciranih 24 bitnom slikom, što znači da bi 24 bitna slika trebala biti dovoljna da prikaže sve boje u vidljivom ljudskom spektru - no u praksi, zbog nesavršenosti monitora kod prikaza slike, ljudsko oko može vidjeti određenu razliku između 24-bitne slike i 30-bitne slike [6]. Što slika ima više bitova po pikselu, prostor koji slika zauzima je veći.





1 bit
2 boje

2 bita
4 boje

4 bita
16 boja

8 bita
256 boja

24 bita (true color)
16,777,126 boja

Slika 6. Slika pri pet različitih dubina boje, preuzeto sa [6]

Broj bitova, *b*, potrebnih za pohranu slike je veličine *N* x *N*, s 2^m sive boje, definirano je formulom:

$$b = N \times N \times m$$

Prema navedenoj formuli, za sliku od 1024 x 1024 piksela s 256 tonova sive boje (m = 8) potrebno je 8,388,607 bitova ili 1,048,576 bajtova (1 bajt = 8 bitova) ili 1048 kB ili 1 MB.

Smanjivanjem broja tonova sive kod nekih slika može se pogodovati boljom manipulacijom u kasnijem radu. Na sljedećoj stranici na slici 7 prikazan je primjer smanjivanja broja tonova sive (= 2^m) sa m = 8 (256 tonova sive) do m = 1 (crno-bijela). [5]

Slika 7. Prikaz smanjivanja broja tonova sive (= 2^m) sa m = 8 (256 tonova sive) do m = 1 (crno-bijela), preuzeto sa [5]

Iz slike 7 vidljivo je kako se smanjivanjem broja tonova sive boje kontrast slike povećava (m = 1), što je pogodnije za prepoznavanje i praćenje značajki. Također, količina prostora koju slika zauzima, a isto tako i snaga koju računalo treba za rukovanje slikom, znatno se smanjuje.

2.3. Morfološka obrada slike

2.3.1. Uvod u morfološku obradu slike

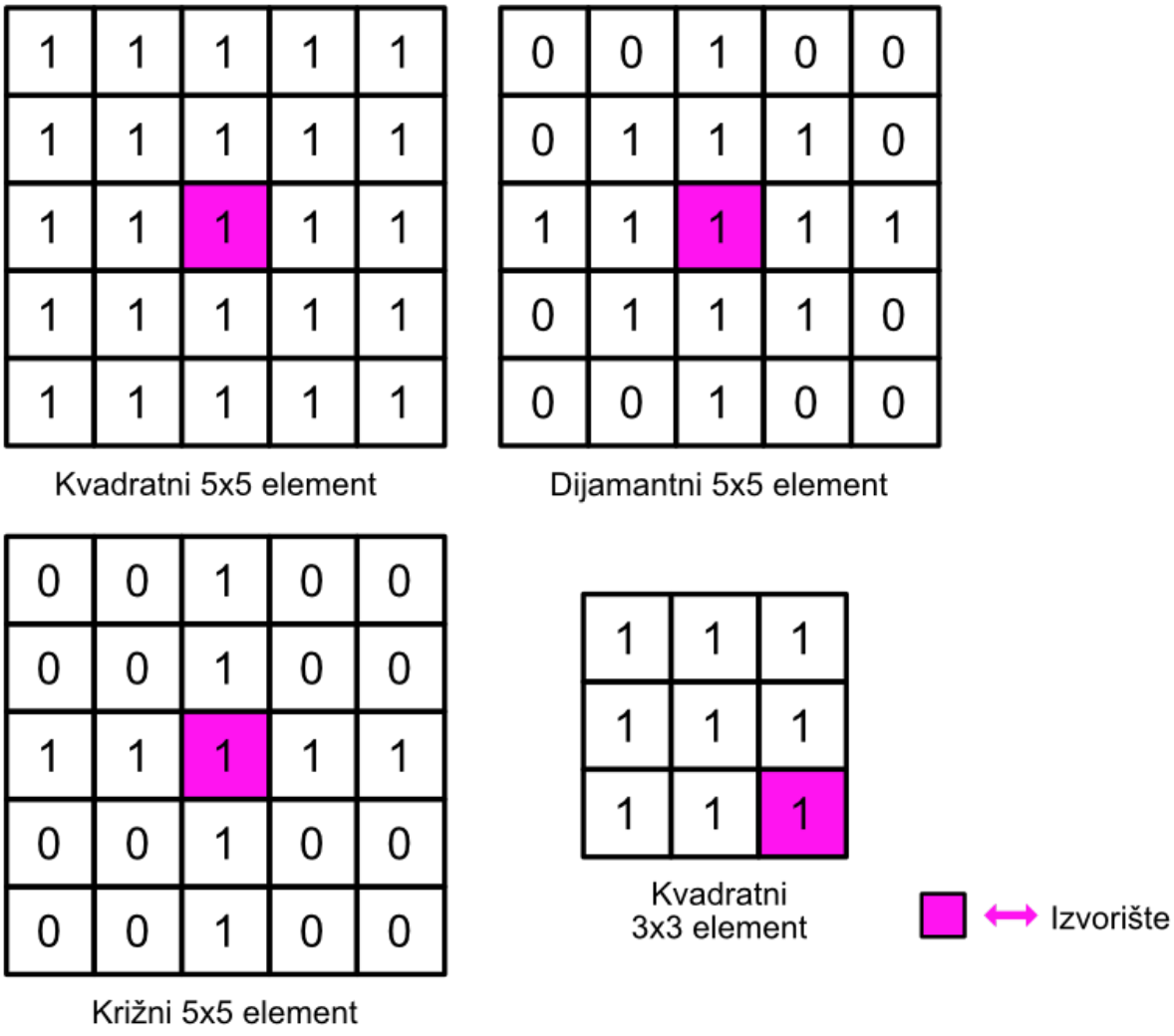
Binarne slike mogu sadržavati brojne nedostatke, kao što je digitalni šum (eng. *noise* ili *grain*). Digitalni šum je neželjeni nasumični signal, odnosno slučajna varijacija svjetline (ekspozicije) ili informacije o boji na slikama. [8] Cilj morfološke obrade slike je detektiranje i uklanjanje ovih nesavršenosti uzimajući u obzir oblik i strukturu slike. Te se operacije mogu proširiti na slike u nijansama sive. [9]

Morfološka obrada slike je širok skup nelinearnih operacija povezanih s oblicima ili morfologijom obilježja na slici. Morfološke operacije primjenjuju radnje na razini piksela na ulaznoj slici, stvarajući izlaznu sliku iste veličine, no povoljniju za daljnju obradu. U morfološkoj operaciji, vrijednost svakog piksela na izlaznoj slici temelji se na usporedbi odgovarajućeg piksela na ulaznoj slici sa susjednim pikselima. [10]

Ulazna slika se ispituje strukturnim elementom (prikazano na slikama 8 i 9), koji je zapravo mala binarna slika, odnosno matrica u kojoj su vrijednosti ili 0 ili 1. Strukturni element se uspoređuje sa svim mogućim mjestima na slici te se izlazna slika određuje primjenom pravila (operacija) na odgovarajući piksel i njegov susjedni piksel. Najosnovnije morfološke operacije su *dilation* i *erosion* ili *dilatacija (širenje)* i *erozija (sužavanje)*. Dilatacija dodaje piksele na granicama objekata na slici, dok erozija uklanja piksele na granicama predmeta. Dilatacija čini predmete vidljivijima i popunjava male rupe u predmetima, a erozija uklanja male predmete (piksele) tako da su ostali samo sadržajni objekti. U operacijama morfološke dilatacije i erozije, rezultat bilo kojeg datog piksela na izlaznoj slici određuje se primjenom pravila na odgovarajući piksel i njegovog susjednog piksela na ulaznoj slici. [10]

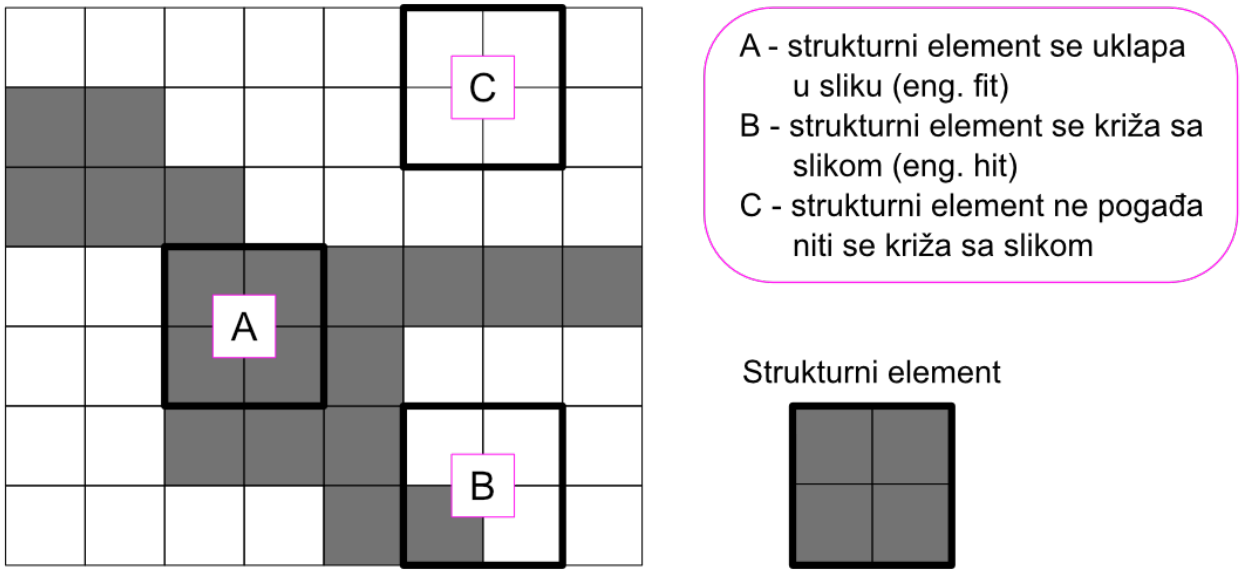
Morfološke operacije erozija i dilatacija ispituju sliku s malim oblikom ili predloškom koji se Dimenzija matrice određuje veličinu elementa strukturiranja, dok uzorak jedinica i nula specificira oblik elementa strukturiranja. [9]

Neki strukturni elementi prikazani su na slici 8:



Slika 8. Prikaz oblika strukturnih elemenata, preuzeto sa [9]

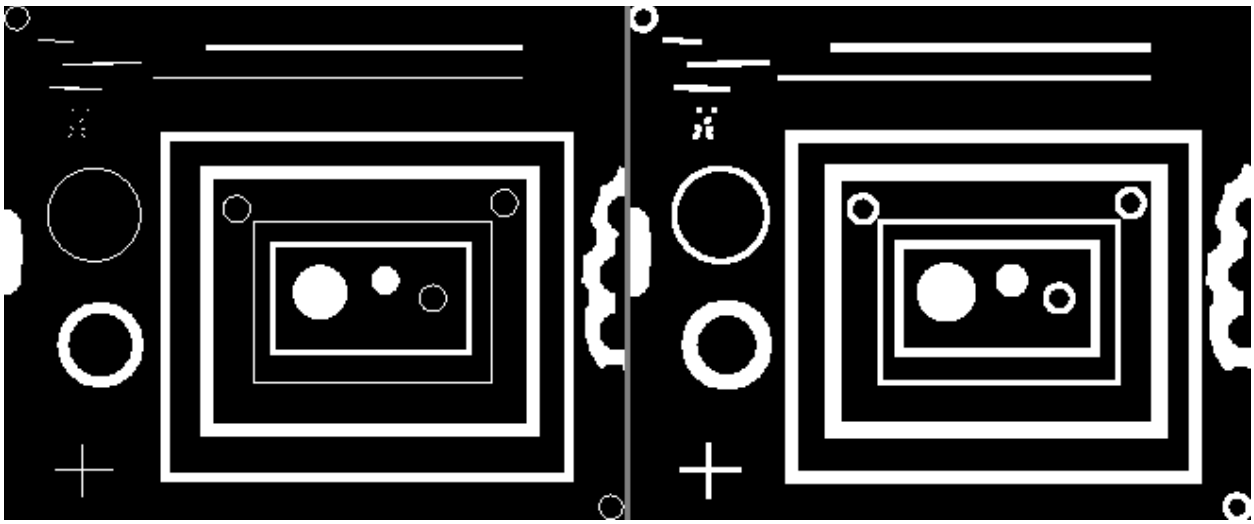
Element je pozicioniran na svim mogućim mjestima na slici i uspoređuje se s odgovarajućim susjedstvom piksela. Kaže se da element za strukturiranje odgovara na slici ako je za svaki od njegovih piksela postavljenih na 1, odgovarajući piksel slike također 1 (eng. *fit*). Slično tome, kaže se da element za strukturiranje pogađa ili presijeca sliku ako, barem za jedan od njegovih piksela postavljenih na 1 odgovarajući slikovni piksel je također 1 (eng. *hit*). Ovo ispitivanje slike prikazano je na slici 9.



Slika 9. Ispitivanje slike elementom za strukturiranje (bijeli pikseli imaju vrijednost 0, sivi pikseli imaju vrijednost 1), preuzeto sa [9]

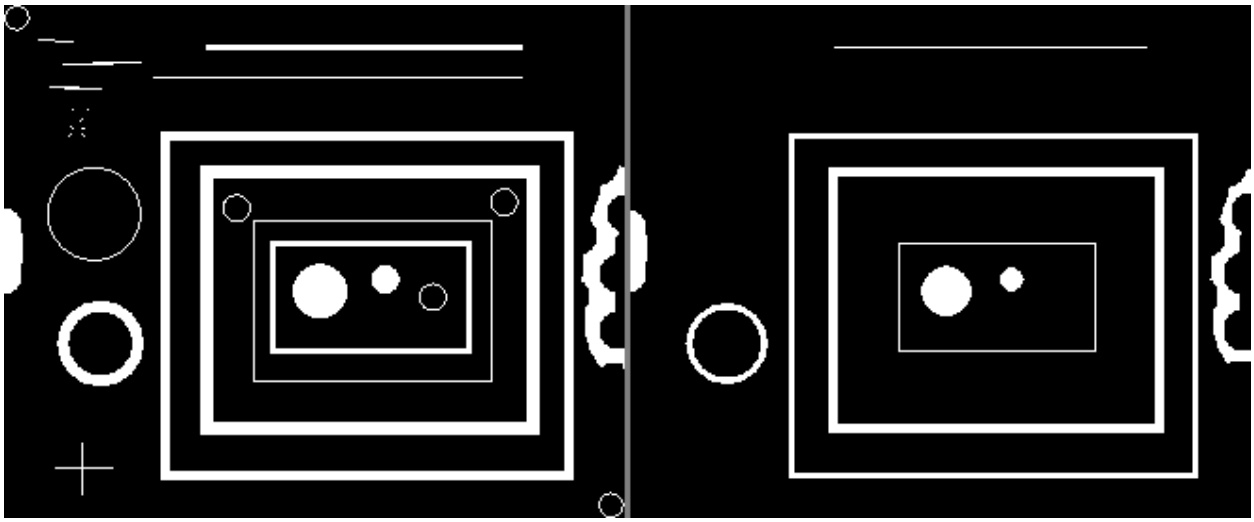
2.3.2. Morfološke operacije Erode i Dilate

Kada se element za strukturiranje postavi u binarnu sliku, svaki njegov piksel pridružen je odgovarajućem pikselu susjedstva ispod elementa za strukturiranje. Na slikama 10. i 11. na sljedećoj stranici prikazani su rezultati primjene morfoloških operacija dilate i erode na binarnoj slici.



Slika 10. Prikaz rezultata morfološke operacije dilate, preuzeto sa [10]

Morfološko širenje (dilate) čini predmete vidljivijima i popunjava male rupe u predmetima. Rupe zatvorene jednom regijom i praznine između različitih regija postaju sve manje, odnosno rubovi objekata se pojačavaju.



Slika 11. Prikaz rezultata morfološke operacije erode, preuzeto sa [10]

Morfološko sužavanje (erode) smanjuje sliku micanjem određenih piksela i s unutarnje i s vanjske granice regija. Rupe i praznine između različitih regija postaju veće, a sitni detalji se uklanjaju.

3. Prepoznavanje značajki

Kada se govori o prepoznavanju značajki i motion trackingu, govori se o području računalne znanosti unutar opsežnije kategorije - računalnog vida. Računalni vid (eng. *Computer vision*) je računalno znanstveno područje koje se bavi načinom na koji računala mogu obrađivati, analizirati i kategorizirati vizualne podatke (slike ili videozapise) na jednak/sličan način kao i ljudi. Koncept računalnog vida temelji se na podučavanju računala da obrađuju sliku na razini piksela i razumiju je. [11]

Neki od uobičajenih zadataka za koje se koriste sustavi računalnog vida:

• **Klasifikacija predmeta**. Sustav analizira dati vizualni sadržaj i razvrstava objekt na fotografiji / videozapisu u definiranu kategoriju. Na primjer, sustav može pronaći psa među svim objektima na slici.

• **Identifikacija predmeta.** Sustav analizira dati vizualni sadržaj i identificira određeni objekt na fotografiji / videozapisu. Na primjer, sustav može pronaći određenog psa među više pasa na slici.

• **Praćenje objekata.** Sustav analizira videozapis te pronalazi objekt (ili objekte) koji odgovara kriterijima pretraživanja i prati njegovo kretanje. [12]

Koncept računalnog vida ustvari uči računalo kako bi ono "razumjelo" što se na slici nalazi; tehnički, strojevi pokušavaju dohvatiti vizualne informacije, rukovati njima i interpretirati rezultate pomoću posebnih softverskih algoritama.

3.1.1. Značajka na slici

Značajka (eng. *feature*) na slici, kako je ljudi percipiraju, je izraženo svojstvo nekoga ili nečega. Ona se može, ovisno o njenoj kvaliteti i različitosti, jasno i nedvojbeno primijetiti kao jednoznačna u lokalnom okruženju.

Govoreći u smislu motion trackinga, za računalo se značajka definira kao informacija na slici koja se bitno razlikuje u svojim svojstvima (pikselima) od drugih lokalnih područja. Često su to uglovi, izražene točke, rubovi objekata, točke s velikim kontrastom i slično. [11]

Otkrivanje značajki

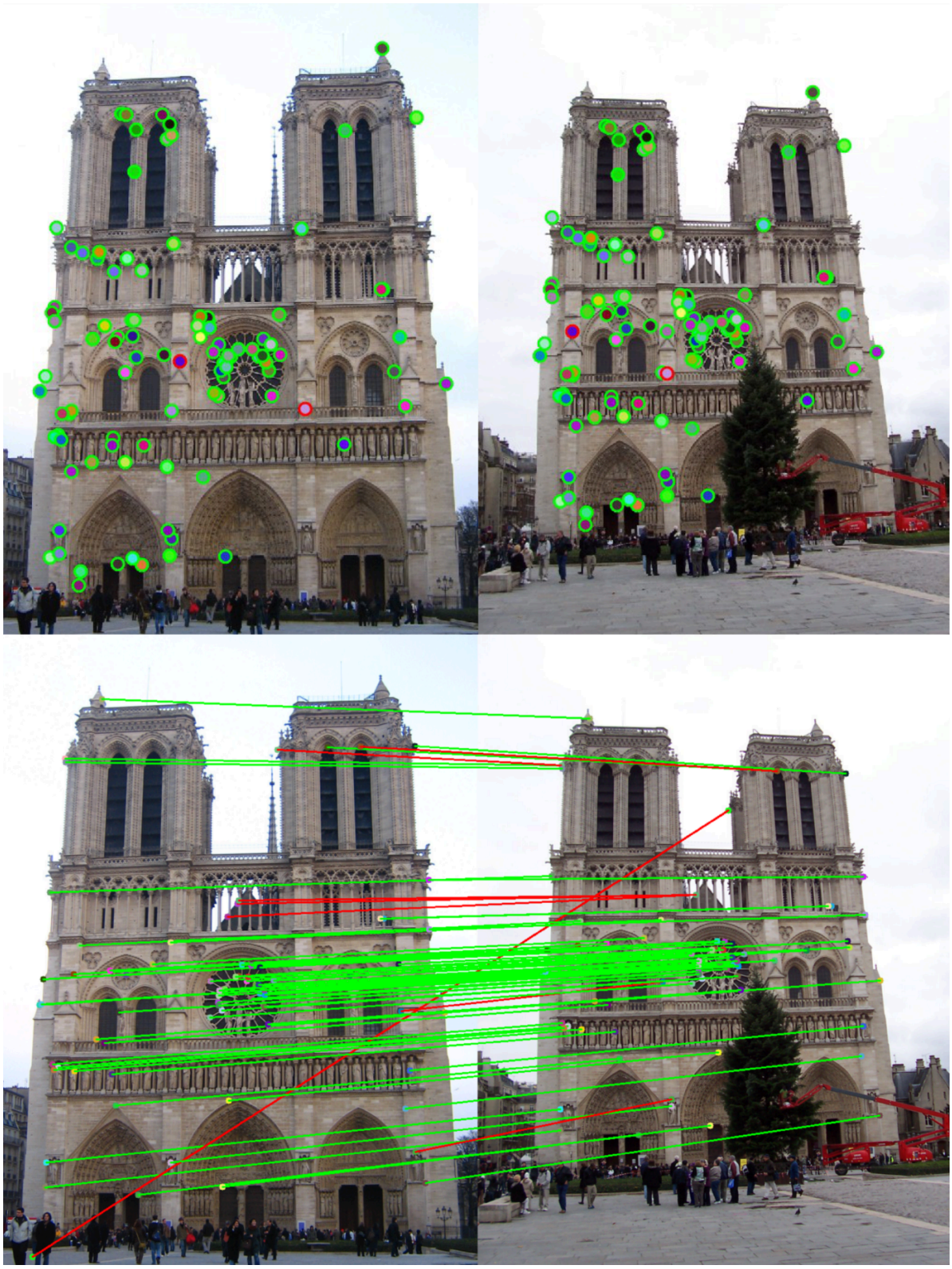
Kako bi se značajka u daljnjem radu mogla pratiti, najprije se mora otkriti i identificirati. Svaka značajka opisana je oko svog lokalnog radiusa i to idealno neosjetljivo na promjene svjetla, translacije, skaliranja i rotacije. [11]

Podudaranje značajki

Otkrivene značajke uspoređuju se sa slikama kako bi se identificirale slične/podudarne značajke. Za dvije slike možemo dobiti skup parova (Xi, Yi) ↔ (Xi`, Yi`), gdje je (Xi, Yi) značajka na jednoj slici, a (Xi`, Yi`) odgovarajuća značajka na drugoj slici. [11]

Svojstva značajki

Kako bi značajka za pretraživanje bila efikasna, bitno je da ima dobro definiran položaj u prostoru slike odnosno da je dobro lokalizirana. Isto tako, važno je da je značajka stabilna u lokalnim i globalnim poremećajima u domeni slike kao što su varijacija osvjetljenja/svjetline, skaliranje, rotacija i sl. tako da se točke interesa mogu pouzdano izračunati s visokim stupnjem ponovljivosti. [11]



Slika 12. Otkrivanje i podudaranje značajki na slici, preuzeto sa [11]

3.2. Algoritmi za prepoznavanje značajki

Postoje mnogi algoritmi za prepoznavanje značajki, te ovisno o problemu kojem pristupamo i načinu na koji je problem potrebno riješiti, poželjno je odabrati odgovarajući algoritam. Svaki algoritam za praćenje pokreta i prepoznavanje značajki razlikuje se u učinkovitosti, preciznosti, brzini izvedbe, polju primjene i slično.

U nastavku će biti navedeni neki popularni algoritmi za prepoznavanje i praćenje značajki, uz popratnu rečenicu osnovne ideje algoritma. Svaki od algoritama zasniva se na publikacijama istraživačkih znanstvenika, te se svaka od publikacija za daljnje istraživanje može pronaći u literaturi na kraju rada. Neki poznati algoritmi za prepoznavanje i praćenje značajki su:

- **SURF** (Speeded Up Robust Feature) **-** SURF metoda (ubrzane robusne značajke) brz je i robustan algoritam za lokalno prepoznavanje i praćenje. Glavni interes SURF pristupa leži u njegovom brzom proračunu operatora koji koriste okvirne filtre, omogućujući tako aplikacije u stvarnom vremenu poput praćenja i prepoznavanja objekata. [13]

- **SIFT** (Scale Invariant Feature Transform) - *Značajke su nepromjenjive prema skaliranju slike, premještanju i rotaciji, a djelomično su nepromjenjive prema promjenama osvjetljenja i afiniranoj ili 3D projekciji. Te značajke dijele slična svojstva s neuronima u temporal korteksu koji se koriste za prepoznavanje predmeta u vidu primata.* [14] Značajke u SIFT-u se učinkovito otkrivaju postupkom progresivnog filtriranja, korišenjem Gaussian zamućenja (eng. *Gaussian blur*) koji identificira stabilne stočke u prostoru slike. Ta tehnika naziva se eng. *Difference of Gaussian* (DoG).

- **BRIEF** (Binary Robust Independent Elementary Features) - BRIEF Koristi takozvane binarne nizove, koji se računaju pomoću jednostavnih testova razlike intenziteta. U radu kažu da algoritam daje slične ili bolje performanse prepoznavanja od SURF algoritma, dok je vrijeme prepoznavanja samo frakcija onoga kod SURF-a [15]

• **FAST** (Features from accelerated segment test) - FAST je metoda otkrivanja kutova koja se može koristiti za izdvajanje značajki. FAST kutni detektor izvorno su razvili Edward Rosten i Tom Drummond, a objavljen je 2006. Najveća prednost FAST kutnog detektora je njegova učinkovitost. Kako mu i samo ime kaže, doista je brži od mnogih drugih dobro poznatih metoda ekstrakcije značajki, poput razlike Gaussiansa (DoG) koju koriste detektori poput SIFT. FAST kutni detektor pogodan je za primjenu i obradu videozapisa u stvarnom vremenu upravo zbog brzih performansi. [16]

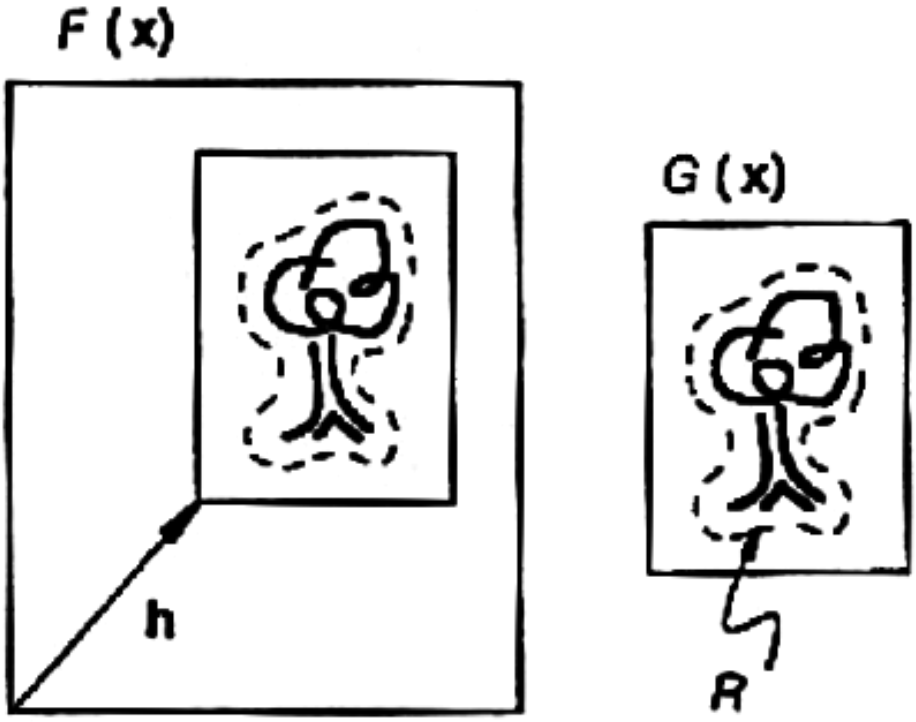
3.3. Kanade-Lucas-Tomasi algoritam za praćenje pokreta

Kanade-Lucas-Tomasi algoritam za praćenje pokreta

Kanade-Lucas-Tomasi (KLT) jedan je od algoritama za prepoznavanje i praćenje značajki, koji se koristi u mnogo sustava za motion tracking, a ujedno i u radnom okruženju Blendera. Postoji mnoštvo tradiocionalnih algoritama i sustava za prepoznavanje i praćenje značajki, no većinom su skupi. Godine 1981., Bruce D. Lucas i Takeo Kanade na temelju višegodišnjeg istraživanja publiciraju rad "An Iterative Image Registration Technique with an Application to Stereo Vision". [umetni] U ovom poglavlju prikazat će se principi na kojima rad počiva te rješenja za potencijalne probleme u motion trackingu. Deset godina nakon toga, godine 1991. Carlo Tomasi i Takeo Kanade objavljuju rad "Detection and Tracking of Point Features".

3.3.1. The registration problem

Kanade – Lucas – Tomasi (KLT) pristup je izdvajanju značajki. KLT koristi podatke o prostornom intenzitetu gradijenta (eng. *Spatial intensity gradient*) za usmjeravanje potrage za položajem koji daje najbolje podudaranje. Dakle, u načelu cilj KLT algoritma je, pristupom prostornog ograničavanja pretrage, odnosno lokalnog pretraživanja, ubrzati proces traženja i podudaranja značajki te smanjiti računalnu snagu koja je potrebna kako bi se izveo taj proces. Ovaj algoritam brži je od tradicionalnih tehnika za ispitivanje upravo zbog mnogo manje potencijalnih podudaranja između slika. [17] Na slici 13 na sljedećoj stranici vidljiv je slikovni opis problema, preuzet iz službenog rada [17].



Slika 13. “The image registration problem”, preuzeto iz [17]

Cilj KLT algoritma je pronaći vektor h na slici 13, koji minimizira područje pretrage i samim time ubrzava proces trackinga. Način na koji algoritam to postiže opisan je u radu [17].

4. Uvod u Motion tracking u Blenderu

Praćenje pokreta (*eng. motion tracking*) je metoda prikupljanja informacija/podataka o položaju kamere tijekom snimanja iz videozapisa. Koristi se za praćenje kretanja objekata i/ili kamere putem smještenih točaka za praćenje (eng. *tracking points*) kroz postavljena ograničenja, kako bi se dobiveni podaci praćenja kasnije koristili za integraciju na 3D objekte/ 3D scene koji su ili izrađeni u Blenderu ili uvezeni u aplikaciju.

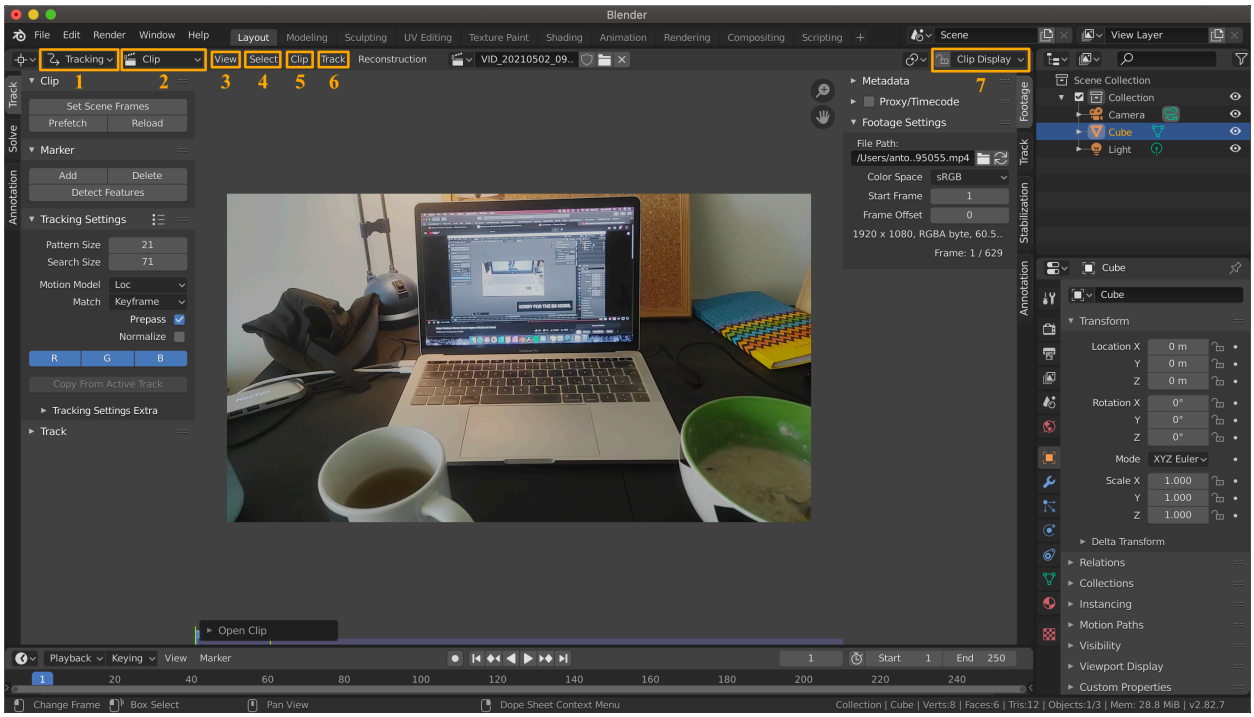
Motion tracker unutar Blendera uključuje nekoliko moćnih alata za 2D praćenje i 3D rekonstrukciju pokreta, uključujući praćenje kamere i praćenje predmeta kao i neke posebne značajke poput stabilizacije snimke.

Motion tracking u Blenderu, u svom najjednostavnijem opisu, sastoji se od 3 koraka:

1. 2D tracking videomaterijala (sekvence slika)
2. Postavljanje svojstava fotoaparata (žarišna duljina, keoficijenti izobličenja, **k1,k2) - kalibracija kamere**
3. Camera solving (rješavanje kamere), orijentacija scene i rekonstrukcija

4.1. Movie Clip Editor - radno okruženje za motion tracking

Tracking i masking u Blenderu izvode se pomoću eng. Movie Clip Editora.



Slika 14. Radno sučelje Movie Clip Editor - s dodanim brojevima na opcijama

MODE

U zaglavlju kod riječi Tracking, označeno brojem 1 na slici 14 nalazi se padajući izbornik gdje je moguće birati između 2 načina rada: tracking i masking. U ovom radu biti će opisan samo tracking mode.

EDITOR VIEW

Kod riječi Clip, označeno brojem 2 na slici 14 nalazi se padajući izbornik sa glavnim dijaloškim okvirima Movie Clip Editora: Clip, Graph i Dopesheet. Sve 3 funkcije kasnije će se koristiti kod praktičnog dijela rada. Clip View opisan je u daljnjem dijelu rada, poglavlje **4.1.1.**

VIEW

Kod riječi View, označeno brojem 3 na slici 14 nalazi se padajući izbornik s operatorima za kontrolu prikazivanja sadržaja u Movie Clip Editoru. Ovdje se nalaze opcije poput: View Out, View Fit, Zoom Out, Zoom in, Vertical split, Horziontal split itd.

SELECT

U padajućem izboru Select, označeno brojem 4 na slici 14 nalaze se opcije za selekciju u Movie Clip Editoru kao što su: Box select, Circle select, (De)Select all i Inverse.

CLIP

Pod brojem 5 na slici 14 označen je padajući izbornik Clip gdje se nalaze opcije: Open clip, Prefetch Frames, Reload Clip. Klikom na prefetch frames učitavamo podatke iz klipa uvezenog u Blender u cache memoriji, te se tako ubrzava i uglađuje pretpregled klipa u Movie Clip Editoru.

TRACK

Izbornik koji će se najviše koristiti u Movie Clip Editoru, upravo je Track izbornik, označen brojem 6 na slici 14. U ovom padajućem izborniku nalaze se svi najvažniji operatori za manipulaciju Trackerima. Neki od operatora su: Join Tracks, Clean tracks, Copy tracks, Track Forwards, Track Backwards, Move Marker. Za većinu ovih operatora, oni iskusniji koriste kratice na tipkovnici.

CLIP DISPLAY

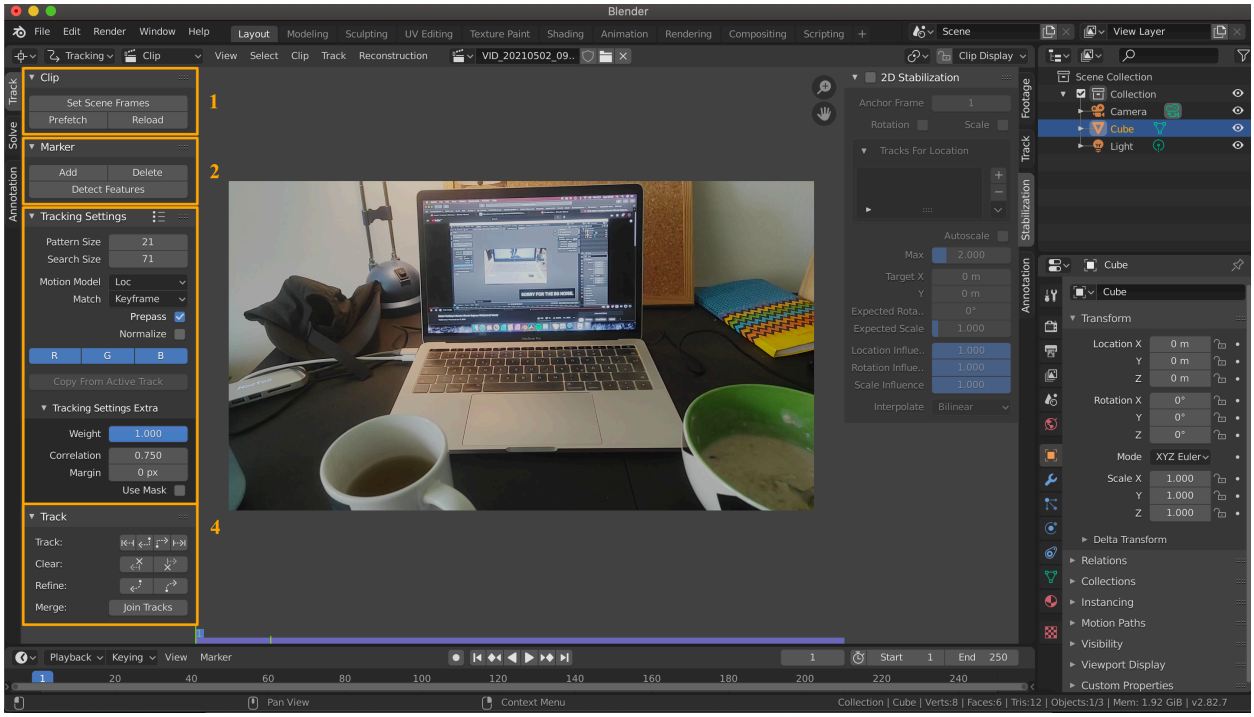
Označeno brojem 7 na slici 14 na desnoj strani zaglavlja nalazi se padajući izbornik Clip Display u kojem se nalaze postavke koje korisniku omogućuju promjenu prikaza samog ubačenog klipa u Editoru.

Channels

R, G, B kanali koriste se za pretpregled slike. Blenderov algoritam praćenja zapravo radi u sivim tonovima (eng. *Grayscale*). Isključivanjem ili uključivanjem R, G i B kanala korisniku omogućava potencijalno bolje prepoznavanje distinktivnih značajki na slici, te samim time i bolji odabir trackera.

4.1.1. Clip View

Glavni dijaloški okvir u kojem se odvija tracking odnosno praćenje točaka je Clip View. Slika 15. prikazuje izgled Clip view-a.



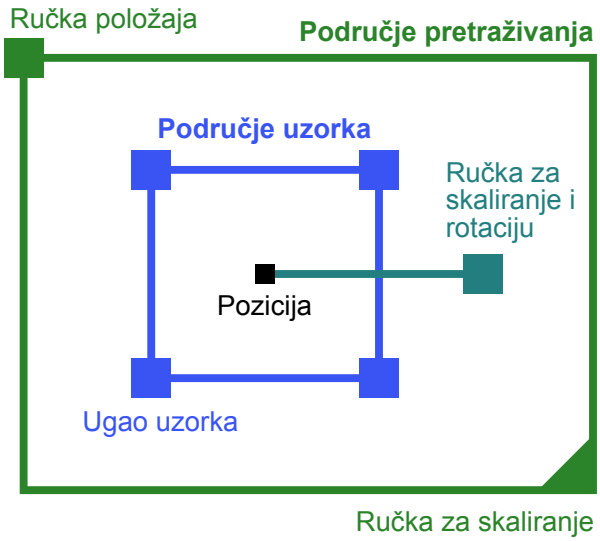
Slika 15. Clip view

CLIP

U prvom pravokutniku, označen brojem 1 na slici 15 nalaze se opcije za *Clip*. "Set Scene Frames" bitna je funkcija - ona postavlja završni *frame,* odnosno završnu sliku u sekvenci i završnu sliku u *timelineu* (vremenskoj crti) na jednak broj. Ova funkcija nam je važna jer se bez nje mogu nastati nepredviđene i neželjene pogreške pri daljnjoj obradi. Nadalje, ispod se nalazi funkcija "Prefetch" koja učitava *frameove* u *cache* memoriju, te samim time ubrzava rad s videosekvencom. "Reload Clip" jednostavno ponovno učitava klip.

MARKER

Na slici 16 vidljiv je shematski prikaz markera (eng. *tracker*) koji se postavlja za praćenje.



Slika 16. Shematski prikaz markera/trackera u Blenderu

"Add" - dodaje marker na sliku

"Delete" - briše odabrani marker

"Detect features" - funkcija u kojoj program sam kroz algoritme (KLT) prepoznaje distinktivne značajke na slici za koje smatra da bi mogli biti praćeni kroz sekvencu. Velika prednost ove opcije je ušteda vremena. Markere koje Blender postavi na sliku moguće je naknadno manipulirati pojedinačno, kao i skupno nakon što se u lijevom kutu otvori skočni prozor s opcijama.

TRACKING SETTINGS

Dijaloški okvir "Tracking settings" ili "Postavke praćenja", sadrže sve važne postavke za 2D algoritam praćenja. Svaka od postavljenih točaka za praćenje bit će postavljena s obzirom na postavke koje u ovom okviru postavimo.

"Pattern size, Search Size" - veličina trackera i veličina prostora za pretraživanje između frameova.

"Motion Model" - važna postavka koja definira model praćenja. Za svako kretanje kamere kojom je snimana scena, trebamo razmisliti kako se kretanje odvijalo. Svaki model ima svoj slučaj upotrebe te je potrebno odabrati model primjeren za snimku na kojoj se radi tracking. Što

je model složeniji (od Location do Affine), vrijeme trackinga po točki se produljuje, i to treba uzeti u obzir kod biranja modela praćenja. U praksi, često treba probati više modela te usporediti rezulate i odabrati najprikladniji za određenu snimku. Ponuđeni modeli za praćenje su:

- Location - samo promjena lokacije kamere, odnosno translacija konstantno paralelna na snimanu scenu. Dakle, pomicanje kamere samo po jednoj osi, kao što je to slučaj kad je kamera postavljena na dolly na tračnicama.
- Location & Rotation - promjena lokacije kamere po jednoj osi + rotacija kamere.
- Location & Scale - promjena lokacije kamere po jednoj osi + prilazak prema snimanom objektu/sceni ili odmicanje od istog.
- Location & Rotation & Scale - kombinacija Location, Rotation i Scale modela.
- Affine - koristi se kod snimaka ili frameova koji imaju promjenu perspektive.
- Perspective - koristi se za slobodno kretanje kamere u svim dimenzijama.
- "Match" - kontrola kojasadrži dvije opcije: Keyframe i Previous frame. Keyframe kreira tracker na prvom frameu i dalje koristi njega za usporedbu, dok Previous frame za svaki sljedeći frame koristi prethodni. Previous frame je sporiji, ali često i precizniji način rada.

"Prepass" - svaki tracker se dva puta provjerava, odatle i naziv prepass.

"Normalize" - trackeri će biti normalizirani, odnosno algoritam će uzeti prosječni intenzitet svjetlosti na trackeru, što ga ćini donekle nepromjenjivim na promjene svjetlosti na slici. U praksi se ova opcija pokazala učinkovito za većinu trackera.

"Copy from Active track" - kopira opcije s odabranog trackera na novi tracker

"Tracking settings extra - Weight" - težina ili važnost trackera može se mijenjati ovom opcijom. Kada se koristi nekoliko trackera, moguće je svakom trackeru dodijeliti smanjenu težinu i samim time kontrolirati njegov utjecaj na rezultat solvinga.

"Tracking settings extra - Correlation" - ova vrijednost, od 0 do 1, predstavlja minimalnu korelaciju između referentnog uzorka i podudarnog uzorka da bi se tracker smatrao uspješnim za taj frame.

"Tracking settings extra - Margin" - koristi se za postavljanje margine oko ruba scene preko koje tracker neće ići.

TRACK

Track motion meni koristi se za izvršavanje trackinga, s opcijama izvršavanja frame po frame, ili izvršavanje od početnog framea do zadnjeg.

4.1.2. Graph view

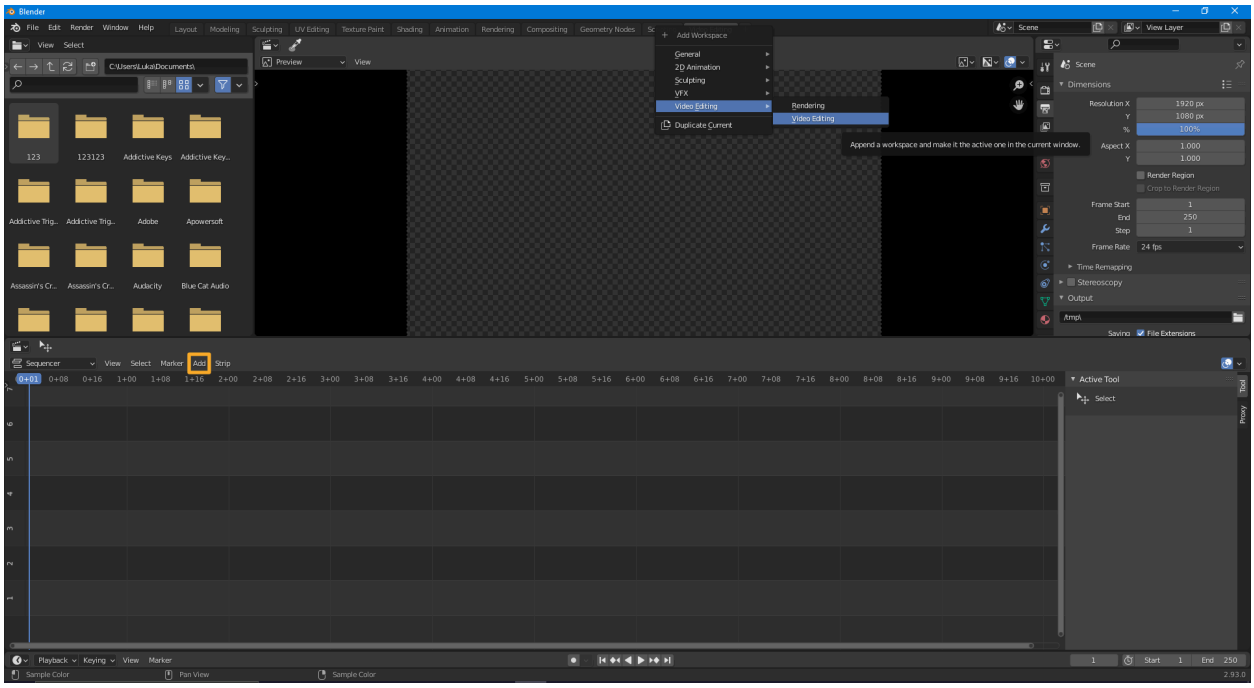
Drugi važan način pregleda u Movie Clip Editoru je Graph view. Ovaj pregled prikazuje graf s crvenim i zelenim krivuljama koje predstavljaju brzinu svakog trackera u danom okviru. Zelena linija je okomito kretanje, crvena vodoravno. Ono što želimo vidjeti, kod dobro odrađenog trackinga, jest da se krivulje na grafu većinom dodiruju i slijede sličan potez. Ukoliko neka od linija znatno odskače od drugih, to znači da je ta linija, koja predstavlja jedan od trackera, loš i slabo upotrebljiv. Svaku od takvih linija možemo ili izbrisati, ukoliko imamo dovoljno drugih trackera, ili prepraviti ili zamijeniti drugim kako bismo dobili bolji rezultat.

Prikaz Graph view-a vidljiv je na slici 23 na stranici 43.

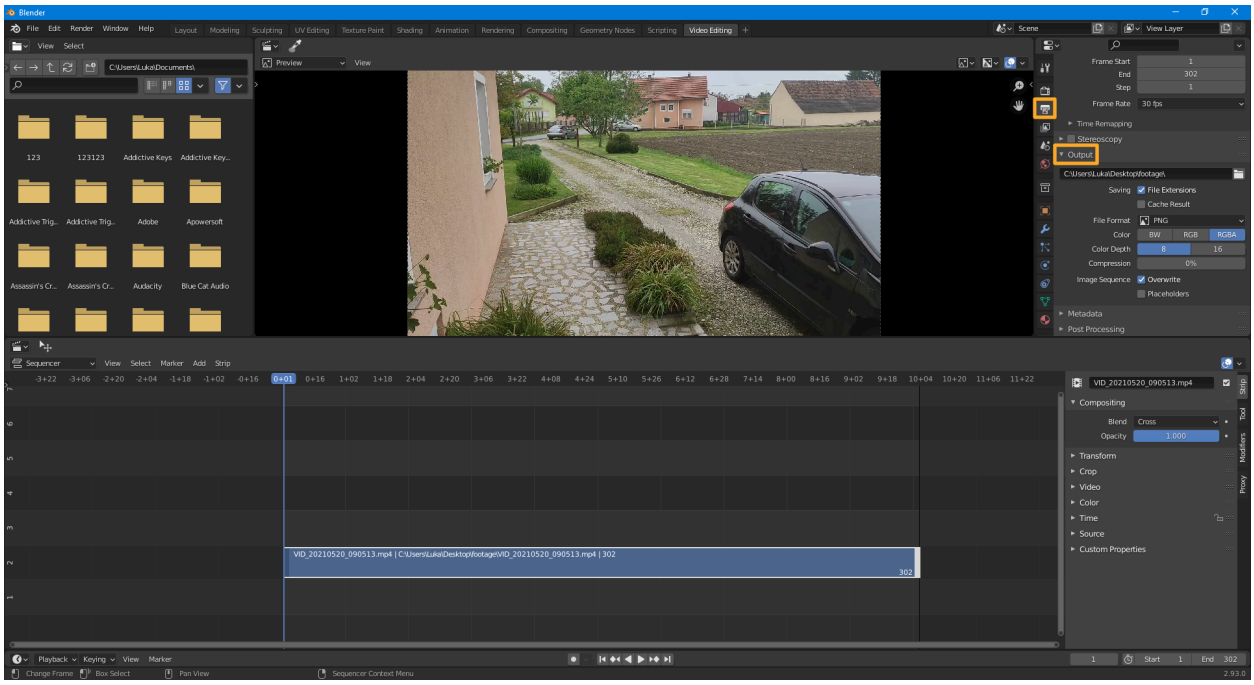
5. Motion tracking - praktični dio

Importing footage

Prvi korak, ili predkorak je umetnuti videomaterijal u Blender i pretvoriti ga u sekvencu slika. To se radi unutar Video Editing radnog okruženja, prikazanog na slici 17. Unutar ovog radnog okruženja, klikom na Add > Movie, označeno narančastim pravokutnikom na slici 17.



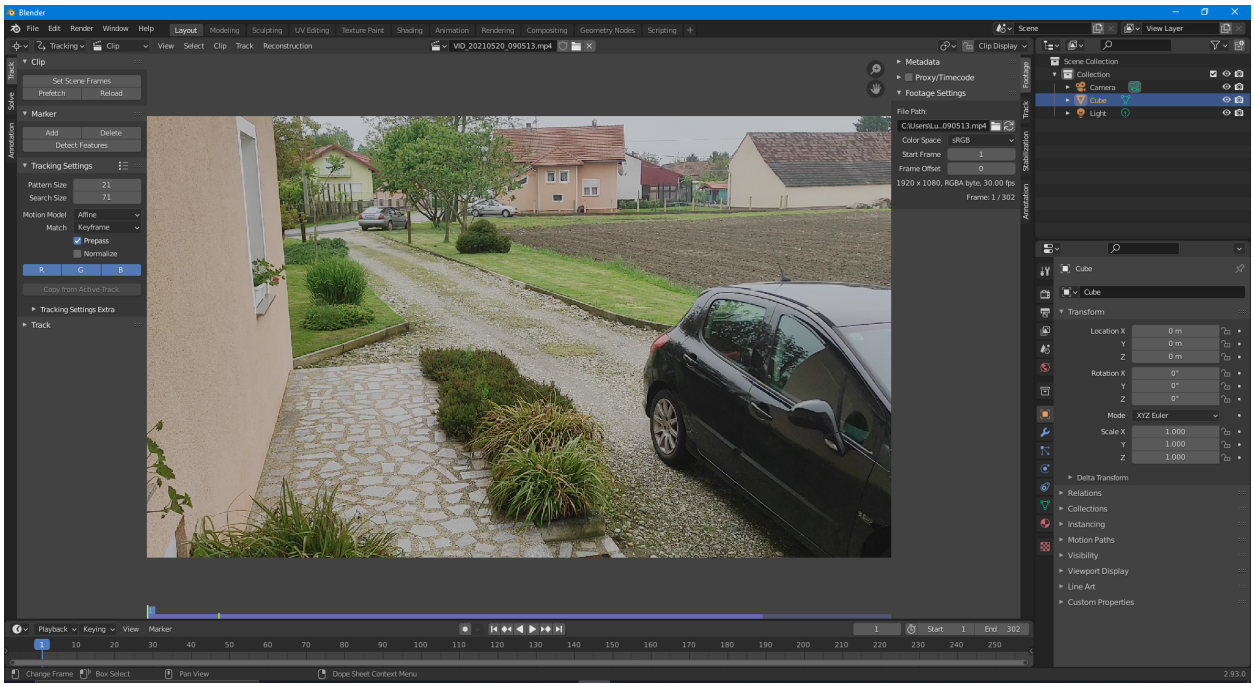
Slika 17. Video editing radno okruženje



Slika 18. Output properties - export slika

Na slici 18. označen je Output Properties prozor u kojem određujemo lokaciju exporta sekvence slika, željeni format slike (PNG) i kompresiju postavljamo na 0%. Nakon postavljenih postavki, odabirom na Render > Render Image u zaglavlju programa renderiramo videomaterijal u sekvencu slika.

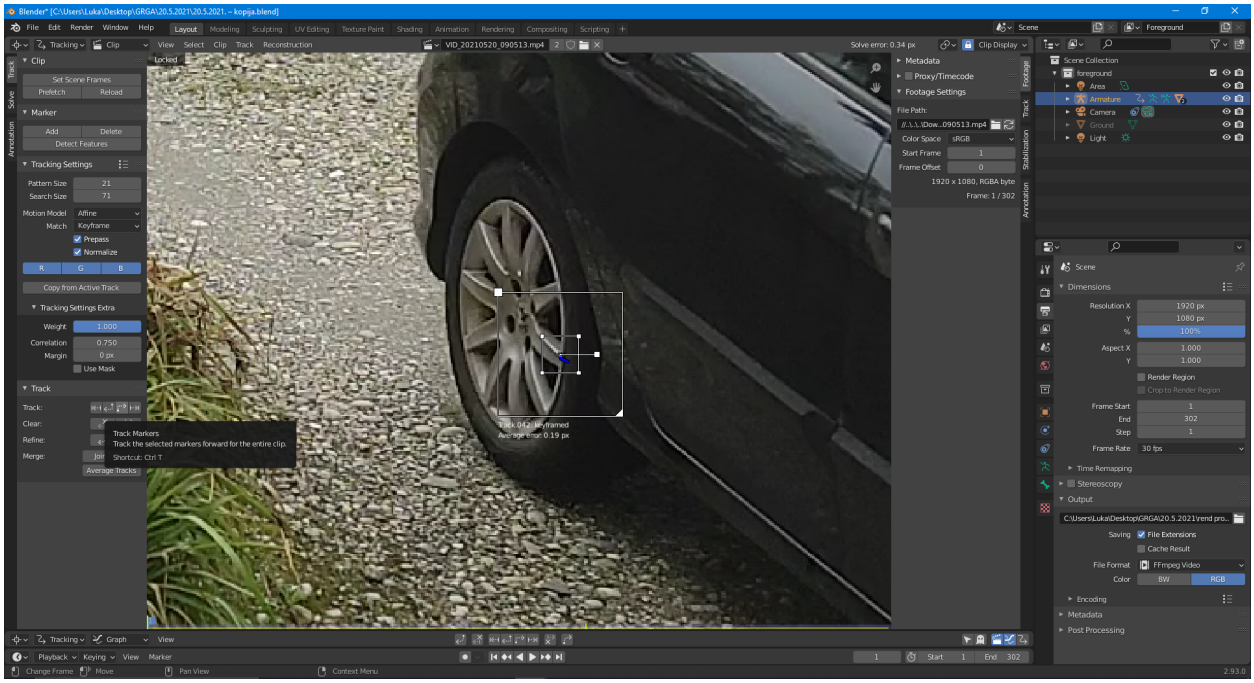
Nakon odrađenog predkoraka pretvaranja videomaterijala u sekvencu slika, istu tu sekvencu uvozimo u Movie Clip Editor klikom na Open i odabiranjem svih izvezenih slika. Nakon toga klikom na Prefetch i Set Scene Frames pripremamo sekvencu slika za obradu. Sada motion tracking može početi. Učitana sekvenca slika vidljiva je na slici 19.



Slika 19. Otvorena sekvenca slika u Movie Clip Editoru, spremna za tracking

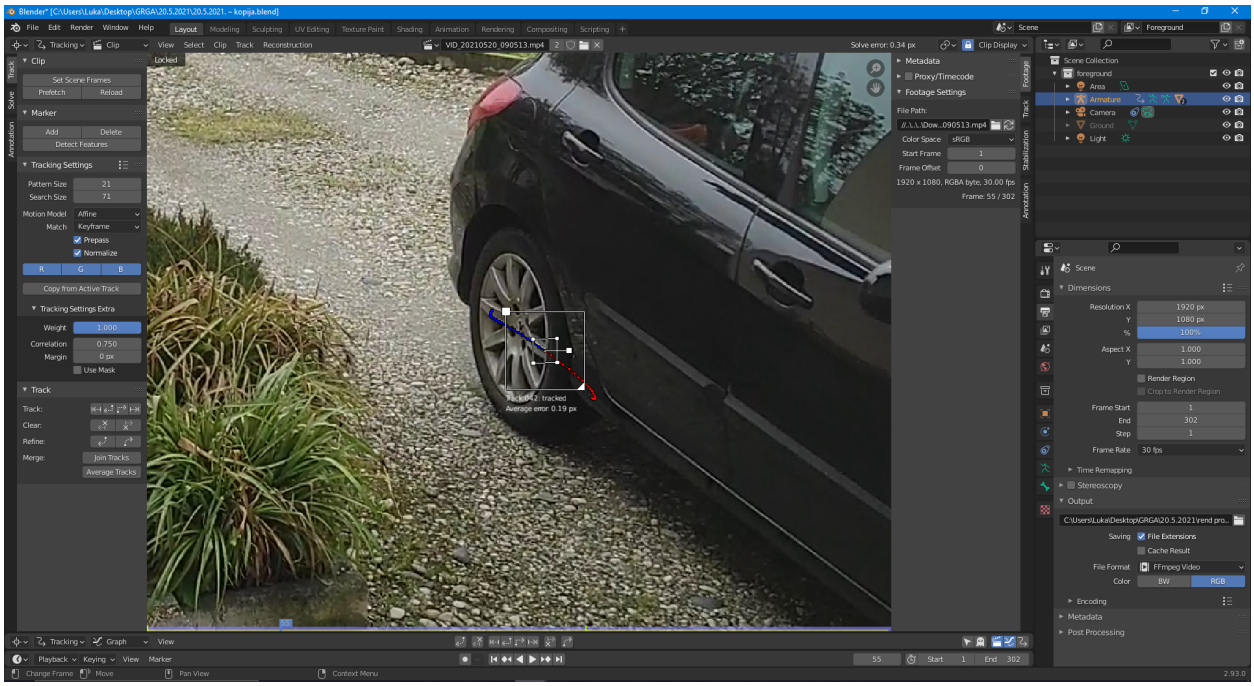
Tracking

Prvi pravi korak motion trackinga je postavljanje tracking pointsa pomoću CTRL + lijevi klik na točku gdje želimo postaviti tracker. U lijevom stupcu pored slike, postavljaju se postavke trackera. U ovom slučaju Location Model Affine se pokazao kao dobar izbor. Sve odabrane postavke kao i postavljen tracker vidljivi su na slici 20.



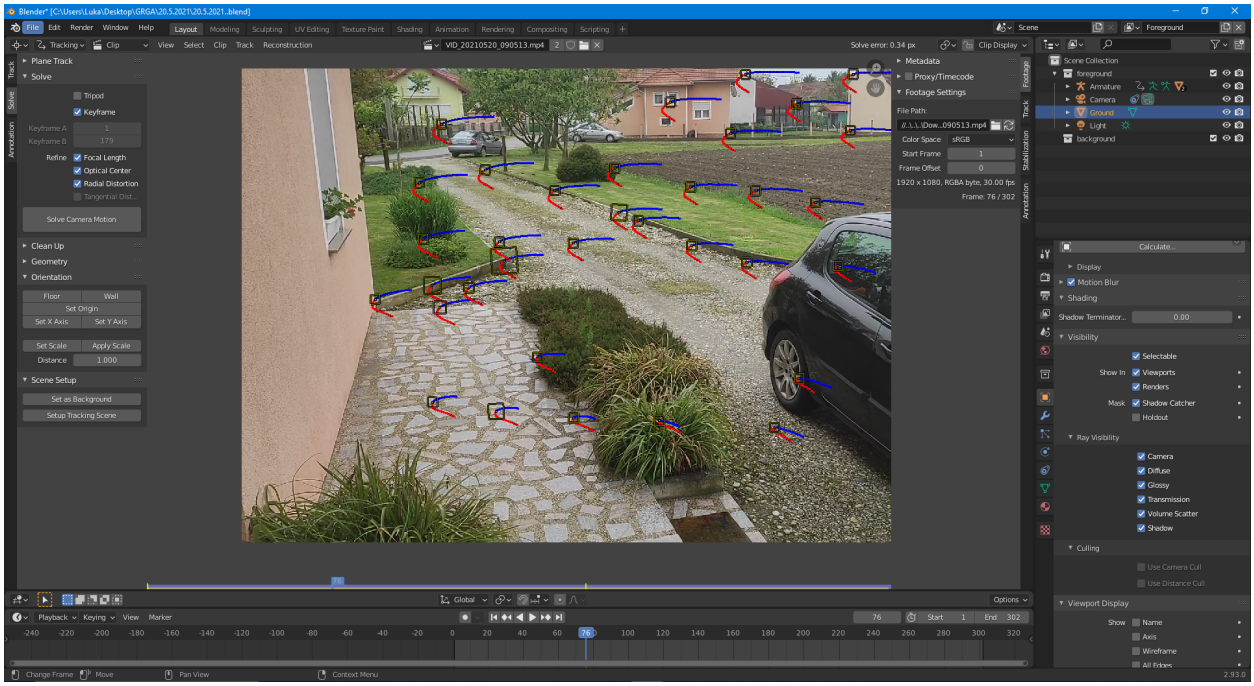
Slika 20. Postavljen tracking marker

Nakon postavljanja trackera, pritiskom na Track marker ili tipkovnicom CTRL + T algoritam kreće s praćenjem markera od prve do zadnje slike. Ukoliko u bilo kojem trenutku marker nije u mogućnosti pronaći željenu značajku, algoritam će stati. Ukoliko dođe do stajanja praćenja, postavke ili poziciju markera možemo podesiti na slici na kojoj je stao i nastaviti s praćenjem putem CTRL + T. Ovo radimo dok ne dođemo do zadnje slike u sceni.



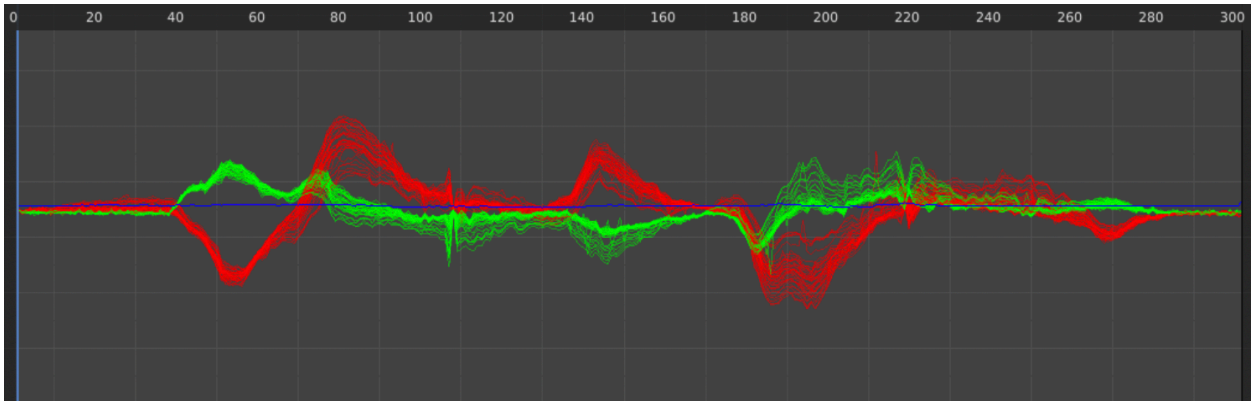
Slika 21. Marker s odrađenim trackingom

Za kvalitetan i iskoristiv track, potrebno je najmanje 8 markera, dakle opisani postupak dodavanja markera i tracking odrađujemo za još najmanje 7 točaka. Poželjno je da su markeri razmješteni na distinktivna mjesta na slici, te da svi markeri nisu na istoj plohi (zid ili pod). Nakon postavljanja najmanje 8 markera i uspješnog praćenja istih, rezultat će biti nešto slično kao na slici 22.



Slika 22. Uspješno trackani markeri na slici

Kako bismo vidjeli podatke o markerima na grafu, ulazimo u Graph view, prikazano na slici 23 na sljedećoj stranici. Ove krivulje su korisne kako bi se vidjelo kreću li se određeni markeri za praćenje drugačije od prosjeka. Crvene i zelene crte na grafikonu prikazuju brzinu trackera u datom okviru. Zelena je okomito kretanje (y os), crvena vodoravno (x os). Linija koja odskače od ostatka krivulje obično znači pogrešku praćenja. Svaki marker (linija na grafu) koji odskače od ostatka potrebno je urediti i ponovo odraditi praćenje, ili ako imamo dovoljno markera na sceni, jednostavno obrisati.

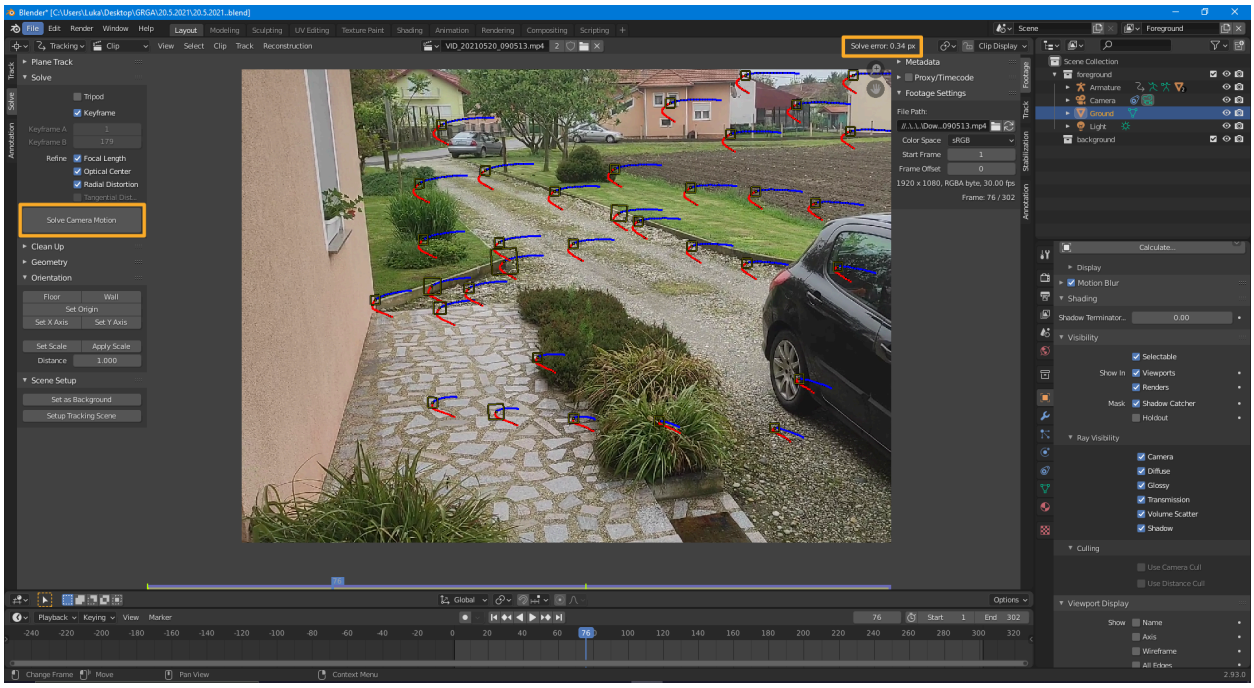


Slika 23. Graph view sa svim odabranim trackerima

Camera solving

Nakon uspješnog trackinga 8 ili više točaka, prelazimo na korak rješavanja/kalkuliranja kamere (eng. *Camera Motion Solving*). Ovaj korak odvija se unutar Solve panela s lijeve strane sučelja (pogledaj sliku 24). U ovom koraku biti će vidljivo jesu li trackeri dovoljno dobri za daljnju obradu ili ne. Camera Solver izračunava prosječnu pogrešku (eng. *Solve error*) svakog trackera u pikselima. U osvnovi, pogreška ispod 0,3 znači precizan track, dok je za vrijednosti iznad 1 piksel poželjno doraditi trackere. [https://docs.blender.org/manual/en/latest/movie_clip/tracking/clip/editing/track.html].

Na slici 24 narančastim pravokutnikom označena je tipka Solve Camera Motion, kao i Solve Error od 0.34 piksela, što je izvrsno. Grafički prikaz ove brojke vidljiv je u Graph View-u, reprezentiran plavom linijom na sredini (vidljivo na slici 23). Što je plava linija ravnija po x osi, to je solve error manji.



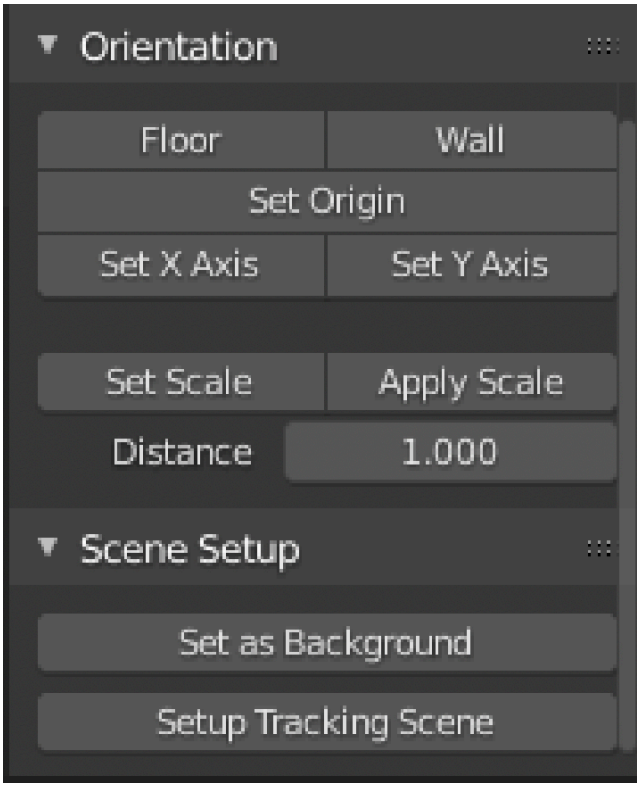
Slika 24. Solve error

Ukoliko je solve error manji od 0.5, faza trackinga je uspješno završena. Sljedeći korak je integracija ovih podataka u kreativan proces. Primjer primjene s dobivenim podacima nalazi se u sljedećem poglavlju.

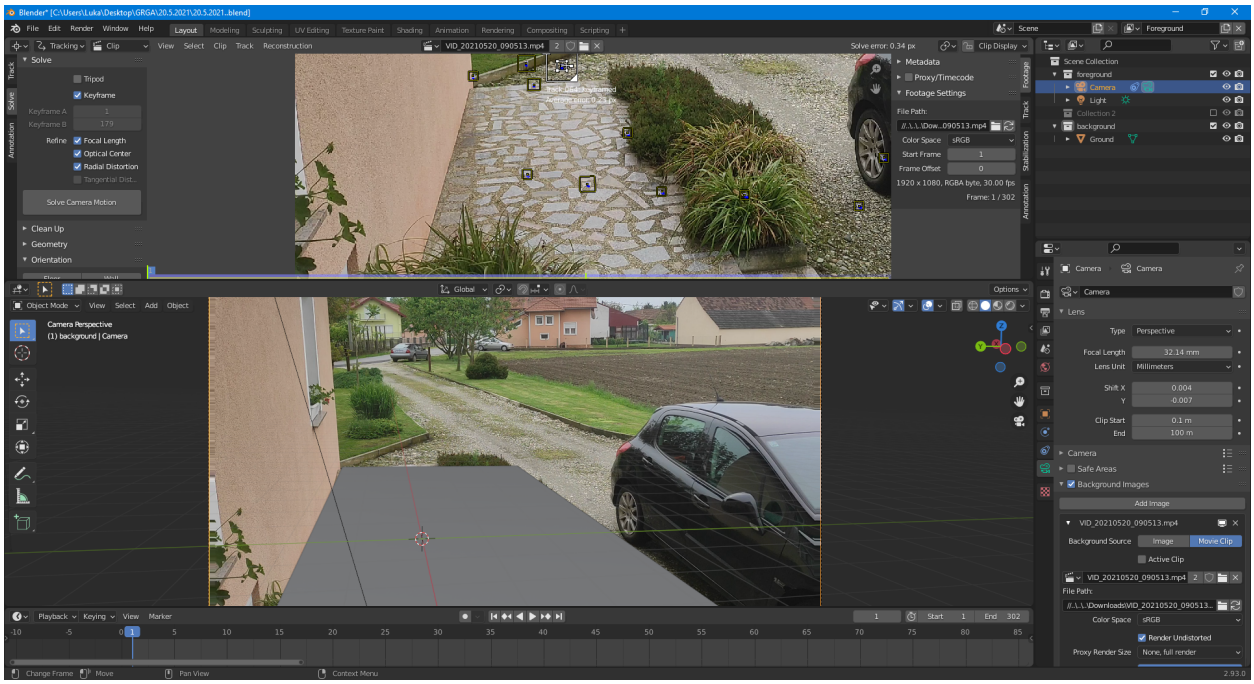
6. Integracija podataka dobivenih motion trackingom - Orijentacija scene i rekonstrukcija

Podaci dobiveni trackingom, u ovom dijelu integrirat će se u scenu. Na snimku biti će umetnut CGI, odnosno figura lika koji pleše. Ova figura preuzeta je s besplatne Adobe stranice https://www.mixamo.com.

Kako bi figura izgledala kao da zaista pleše ne sceni s usklađenom perspektivom, u Movie Clip Editor, ispod Solve Camera Motion, pritiskom na Set as Background i Setup Tracking Scene postavljanmo virtualnu scenu s usklađenom perspektivom. Slika 25 prikazuje tipke i opcije za rekonstrukciju scene, dok slika 26 prikazuje sam ishod na sceni.

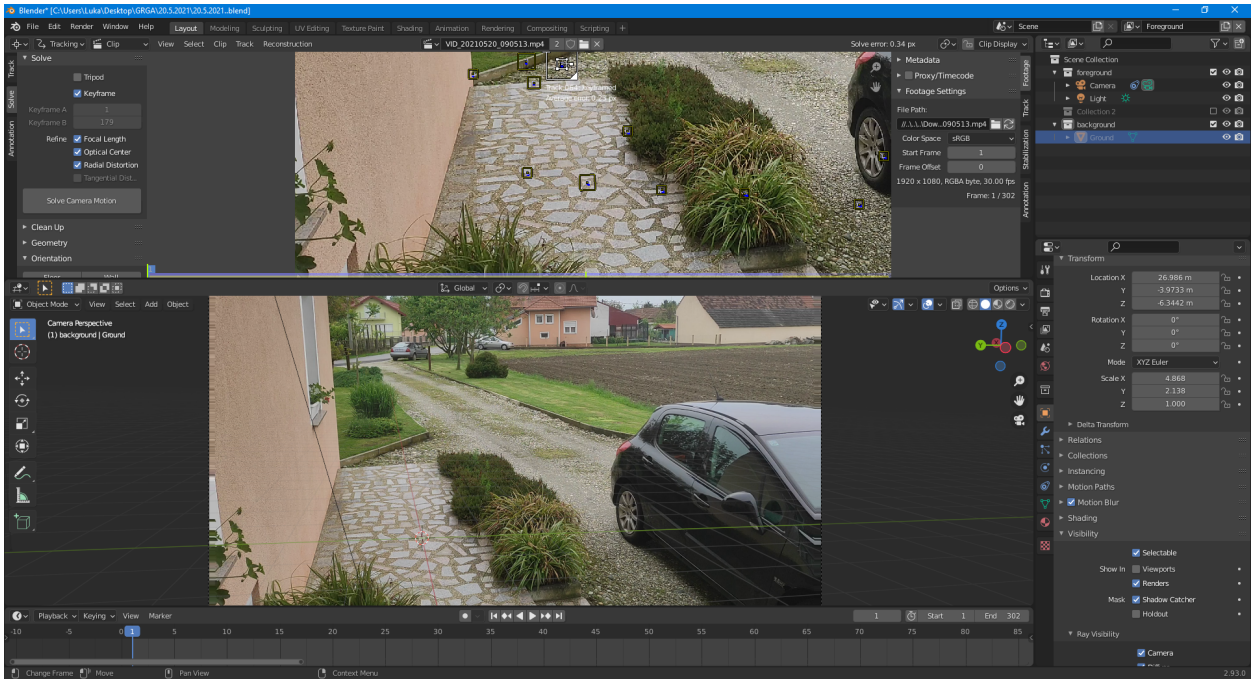


Slika 25. Opcije za rekonstrukciju scene



Slika 26. Setup Tracking Scene

Opcijama Set X Axis i Set Y Axis, podešava se virtualna podlogu te se postiže rezultat nalik ovome na slika 25. Dodana podloga služit će zapravo kao nevidljiva podloga koja će hvatati sjenu lika koji će naknadno biti dodan. Kako bi podloga bila nevidljiva, potrebno je u opcijama na desnoj strani sučelja, pod Object properties > Viewport Display > odkvačiti Viewports i odabrati Shadow Catcher.



Slika 27. Shadow Catcher

U gornjem desnom kutu potrebno je Layer Ground prebaciti u Foreground kolekciju, kako bi Blender ispravno generirao i hvatao sjene na podlozi.

Sljedeći korak je dodavanje lika u scenu, namještanje umjetno dodanog svjetla kako bi odgovarao svjetlu na slici i compositing. Nakon dodavanja navedenog, rezultat je vidljiv na slici 28.

Slika 28. Integracija motion trackinga - lik koji pleše

Animaciju lika na sceni moguće je vidjeti na ovom linku: https://youtu.be/YMWjLkG4L6w

7. Zaključak

Praćenje pokreta proces je prikupljanja podataka o kretanju kamere za vrijeme snimanja. Time se dobivaju podaci koji mogu biti relevantni za široku primjenu u realnom svijetu; počevši od medicine pa sve do virtualne stvarnosti. Motion tracking je u današnje vrijeme široko dostupan, kao što je to u open source programu Blender, te svatko s prosječnim računalom može odraditi kvalitetan motion tracking kod kuće. Ono što se dalje s podacima dobivenima iz praćenja radi, ostaje na korisniku, no mogućnosti primjene i integracije su velike.

U Varaždinu, 17. rujan 2021.

Potpis

8. Literatura

[1] Huiyu Zhou, Huosheng Hu: Human motion tracking for rehabilitation—A survey, Brunel University, Ujedinjeno Kraljevstvo, 2007.

[2] https://en.wikipedia.org/wiki/Computer-generated_imagery, dostupno 1.4.2021.

[3] Budi Sugandi, Hyoungseop Kim., Joo Kooi Tan and Seiji Ishikawa: A Block Matching Technique for Object Tracking Based on Peripheral Increment Sign Correlation Image, Kyushu Institute of Technology, Japan, 2013.

[4] https://hr.wikipedia.org/wiki/Rasterska_slika, dostupno 1.4.2021.

[5] Maria Petrou, Costac Petrou: Image processing: The Fundamentals, Ujedinjeno Kraljevstvo 2010.

[6] https://openframeworks.cc/ofBook/chapters/image_processing_computer_vision.html, dostupno 25.4.2021.

[7] https://en.wikipedia.org/wiki/Color_depth, dostupno 27.4.2021.

[8] https://en.wikipedia.org/wiki/Image_noise, dostupno 27.4.2021.

[9] https://www.cs.auckland.ac.nz/courses/compsci773s1c/lectures/ImageProcessing-html/topic4.htm, dostupno 15.4.2021.

[10] https://www.mathworks.com/help/images/morphological-dilation-and-erosion.html, dostupno 15.4.2021.

[11] https://xd.adobe.com/ideas/principles/emerging-technology/what-is-computer-vision-how-does-it-work/, dostupno 15.4.2021.

[12] https://medium.com/data-breach/introduction-to-feature-detection-and-matching-65e27179885d, dostupno 1.5.2021.

[13] https://medium.com/data-breach/introduction-to-surf-speeded-up-robust-features-c7396d6e7c4e, dostupno 1.5.2021.

[14] https://www.cs.ubc.ca/~lowe/papers/iccv99.pdf, dostupno 10.5.2021.

[15] https://link.springer.com/chapter/10.1007/978-3-642-15561-1_56, dostupno 10.5.2021.

[16] https://en.wikipedia.org/wiki/Features_from_accelerated_segment_test, dostupno 10.5.2021.

[17] https://www.ri.cmu.edu/pub_files/pub3/lucas_bruce_d_1981_2/lucas_bruce_d_1981_2.pdf, dostupno 10.5.2021.

9. Prilozi

https://youtu.be/YMWjLkG4L6w - Integracija motion trackinga - autorski rad

10. Slike

Slika 1. Shematski prikaz predobrade slike i detekcije objekata, preuzeto iz [3]

Slika 2. Prikaz zapisa digitalne slike na računalu, preuzeto iz [5]

Slika 3. Dijagram prikaza i zapisa piksela, preuzeto sa [6]

Slika 4. Prikaz digitalne slike sa smanjujućim brojem piksela po slici

Slika 5. Spektralna karakteristika osjetljivosti slikovnog senzora s tri različita vala: R, G i B

Slika 6. Slika pri pet različitih dubina boje, preuzeto sa [6]

Slika 7. Prikaz smanjivanja broja tonova sive, preuzeto sa [5]

Slika 8. Prikaz oblika strukturnih elemenata, preuzeto sa [9]

Slika 9. Ispitivanje slike elementom za strukturiranje, preuzeto sa [9]

Slika 10. Prikaz rezultata morfološke operacije dilate, preuzeto sa [10]

Slika 11. Prikaz rezultata morfološke operacije erode, preuzeto sa [10]

Slika 12. Otkrivanje i podudaranje značajki na slici, preuzeto sa [11]

Slika 13. “The image registration problem”, preuzeto iz [17]

Slika 14. Radno sučelje Movie Clip Editor

Slika 15. Clip view

Slika 16. Shematski prikaz markera/trackera u Blenderu

Slika 17. Video editing radno okruženje

Slika 18. Output properties - export slika

Slika 19. Otvorena sekvenca slika u Movie Clip Editoru, spremna za tracking

Slika 20. Postavljen tracking marker

Slika 21. Marker s odrađenim trackingom

Slika 22. Uspješno trackani markeri na slici

Slika 23. Graph view sa svim odabranim trackerima

Slika 24. Solve error

Slika 25. Opcije za rekonstrukciju scene

Slika 26. Setup Tracking Scene

Slika 27. Shadow Catcher

Slika 28. Integracija motion trackinga - lik koji pleše

Sveučilište
Sjever

SVEUČILIŠTE
SJEVER

IZJAVA O AUTORSTVU
I
SUGLASNOST ZA JAVNU OBJAVU

Završni/diplomski rad isključivo je autorsko djelo studenta koji je isti izradio te student odgovara za istinitost, izvornost i ispravnost teksta rada. U radu se ne smiju koristiti dijelovi tuđih radova (knjiga, članaka, doktorskih disertacija, magistarskih radova, izvora s interneta, i drugih izvora) bez navođenja izvora i autora navedenih radova. Svi dijelovi tuđih radova moraju biti pravilno navedeni i citirani. Dijelovi tuđih radova koji nisu pravilno citirani, smatraju se plagijatom, odnosno nezakonitim prisvajanjem tuđeg znanstvenog ili stručnoga rada. Sukladno navedenom studenti su dužni potpisati izjavu o autorstvu rada.

Ja, Ante Grgić *(ime i prezime)* pod punom moralnom, materijalnom i kaznenom odgovornošću, izjavljujem da sam isključivi autor/ica završnog/diplomskog (*obrisati nepotrebno*) rada pod naslovom Praćenje pokreta i integracija u open-source programu Blender *(upisati naslov)* te da u navedenom radu nisu na nedozvoljeni način (bez pravilnog citiranja) korišteni dijelovi tuđih radova.

Student/ica:
(upisati ime i prezime)

(vlastoručni potpis)

Sukladno Zakonu o znanstvenoj djelatnost i visokom obrazovanju završne/diplomske radove sveučilišta su dužna trajno objaviti na javnoj internetskoj bazi sveučilišne knjižnice u sastavu sveučilišta te kopirati u javnu internetsku bazu završnih/diplomskih radova Nacionalne i sveučilišne knjižnice. Završni radovi istovrsnih umjetničkih studija koji se realiziraju kroz umjetnička ostvarenja objavljuju se na odgovarajući način.

Ja, Ante Grgić *(ime i prezime)* neopozivo izjavljujem da sam suglasan/na s javnom objavom završnog/diplomskog (*obrisati nepotrebno*) rada pod naslovom Praćenje pokreta i integracija u open-source programu Blender *(upisati naslov)* čiji sam autor/ica.

Student/ica:
(upisati ime i prezime)

(vlastoručni potpis)