

# Usporedba komercijalnih alata za dobivanje 3D pokreta iz dvodimenzionalne snimke

---

Car, Saša

Master's thesis / Diplomski rad

2014

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Graphic Arts / Sveučilište u Zagrebu, Grafički fakultet**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:216:441519>

Rights / Prava: [In copyright](#) / [Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2024-11-15**



Repository / Repozitorij:

[Faculty of Graphic Arts Repository](#)



SVEUČILIŠTE U ZAGREBU  
GRAFIČKI FAKULTET

SAŠA CAR

**USPOREDBA KOMERCIJALNIH ALATA  
ZA DOBIVANJE 3D POKRETA IZ  
DVODIMENZIONALNE SNIMKE**

DIPLOMSKI RAD

Zagreb, 2014.



Sveučilište u Zagrebu  
Grafički fakultet

SAŠA CAR

# **USPOREDBA KOMERCIJALNIH ALATA ZA DOBIVANJE 3D POKRETA IZ DVODIMENZIONALNE SNIMKE**

DIPLOMSKI RAD

Mentor:

dr.sc.Lidija Mandić

Student:

Saša Car

Zagreb, 2014.

*Zahvaljujem se mentorici doc.dr.sc. Lidiji Mandić na kritikama i sugestijama pri izradi diplomskog rada. Veliko hvala prijatelju Aleksandru Ribaru koji mi je pomogao pri snimanju kadrova te mi nesebično odgovarao na moja pitanja.*

*Zahvaljujem se svojoj sestri Tei i roditeljima koji su bili strpljivi i podržavali me prilikom pisanja ovoga rada, ali i tokom cjelokupnog studiranja. Njima ujedno i posvećujem ovaj rad.*

## Sažetak

U ovom diplomskom radu uspoređeni su komercijalni alati za dobivanje 3D pokreta iz dvodimenzionalne snimke. Ubacivanje modeliranih 3D objekata u video snimku je vrlo rasprostranjeno u filmskoj industriji i video igrama ali je to ujedno i složen i zahtjevan posao koji zahtjeva mnogo znanja u različitim alatima koji omogućavaju izradu gotovog videa sa 3D modelima. Programi koji su korišteni su *Boujou*, *Pftrack* i *SynthEyes* koji su jedni od razvijenijih programa u ovom području, a program u kojem su modelirani 3D modeli je *NewTek Lightwave*. Ovo je specifična grana industrije vizualnih efekata koja zahtjeva složena i zahtjevna geometrijska izračunavanja te svaki od navedenih softvera daje različite rezultate u različitim uvjetima. Svrha ovog rada je definirati korisnost svakog od tih softvera pri određenom setu snimljenih kadrova koji reprezentiraju najčešće situacije gdje se taj softver koristi. Snimljeni video isječci se ubacuju u jedan od navedenih programa za dobivanje pokreta iz dvodimenzionalne snimke, obrađuju se i nakon obrade se ubacuje 3D model. Obrada snimljenih video isječaka u svakom od ovih programa je drugačija te su upravo u tom koraku obrade istraženi na koji način određeni program daje iste rezultate. Određivane su pozicije 3D modela tzv. „error pixel“ koji svaki od navedenih programa izračunava drugačije. Kada je ispod vrijednosti 1 on je zanemariv, a ako nije, pozicija 3D modela nije unutar snimke. Osim „error pixela“ i pozicije modela mjerena je i brzina pojedinih programa.

Ključne riječi:

3D model, Boujou, NewTek Lightwave, error pixel, PFTrack, SynthEyes

## Summary

This diploma research compared commercial tools for obtaining 3D movement from two-dimensional recording. Inserting modeled 3D objects into video recording is widely applied in film industry and video gaming, but at the same time it is complex and demanding work that requires a lot of knowledge in various tools that make creation of video with 3D models possible. Programs used are Boujou, Pfrack and SynthEyes which are one of most developed programs in this field. 3D models were done in NewTek Lightwave. This is a specific branch of visual effect industry that requires complex and demanding geometric calculations, where any of previously named software's produces different results under different conditions. Point of this research is to define use of every one of this software's in defined set of recorded frames that represent most frequent situations where that software is used. Recorded video clips are imported in one of named tools for obtaining movement from two-dimensional record, processed and a 3D model is imported. Processing of recorded video clips in each of these programs is different, therefore right this step of processing is examined in order to see how a certain program provides the same results. Positions of a 3D model, so called "error pixel", are determined which each of these programs calculates in a different manner. When the value is under 1, it is negligible. In such case when it's above 1, position of the 3D model is not within recording. Except "error pixel" and position of the model, the speed of each program was also measured.

Key words:

3D model, Boujou, NewTek Lightwave, error pixel, PFTrack, SynthEyes

## Sadržaj:

<b>1. UVOD</b> .....	1
<b>2. TEORIJSKI DIO</b> .....	2
<b>2.1. 3D računalna grafika</b> .....	2
<b>2.2. Povijest</b> .....	2
<b>2.3. Modeliranje</b> .....	2
<b>2.4. Modeliranje teksture i postavljanje materijala</b> .....	3
<b>2.5. Renderiranje</b> .....	4
<b>3. Match - moving</b> .....	6
<b>3.1. Tracking – praćenje pokreta kamere</b> .....	6
<b>4. Korišteni programi</b> .....	8
<b>4.1. SynthEyes</b> .....	8
<b>4.2. ThePixelFarm PFTrack</b> .....	11
<b>4.3. 2D3 Boujou</b> .....	15
<b>4.4. NewTek Lightwave 3D</b> .....	17
<b>5. Lightwave alati za manipuliranje</b> .....	19
<b>6. Opis SynthEyes programa</b> .....	22
<b>7. Opis PFTrack programa</b> .....	23
<b>8. Opis Boujou programa</b> .....	25
<b>3. EKSPERIMENTALNI DIO</b> .....	27
<b>3.1. Snimanje žive snimke</b> .....	27
<b>3.2. Praćenje kamere ili tracking</b> .....	27
<b>3.3. Ubacivanje snimke u PFTrack program za praćenje</b> .....	27
<b>3.4. Ubacivanje snimke u SynthEyes program za praćenje</b> .....	29
<b>3.5. Ubacivanje snimke u Boujou program za praćenje</b> .....	30
<b>4. Stvaranje 3D grafike</b> .....	31
<b>4.1. Modeliranje</b> .....	31
<b>4.2. Stvaranje virtualne scene</b> .....	31
<b>4.3. Ubacivanje i integriranje modela u scenu</b> .....	32
<b>4. REZULTATI ISTRAŽIVANJA</b> .....	35
<b>4.1. Test 1</b> .....	35
<b>4.2. Test 2</b> .....	35
<b>4.3. Test 3</b> .....	36
<b>4.4. Test 4</b> .....	36
<b>4.5. Test 5</b> .....	36

4.6. Test 6 .....	37
4.7. Test 7 .....	37
4.8. Test 8 .....	37
4.9. Test 9 .....	38
5. RASPRAVA .....	39
6. ZAKLJUČAK .....	41
7. LITERATURA .....	42
7.1. Popis slika .....	43
7.2. Popis tablica .....	45



## 1. UVOD

Ubacivanje i postavljanje virtualnih 3D elemenata u živu snimku zasigurno je jedna od najzahtjevnijih grana 3D industrije. Dobivanje realne slike u 3D grani multimedije je veoma zahtjevno te iziskuje mnogo vremena, znanja i vještina, no komponiranje virtualnih 3D elemenata u živu snimku je pravi izazov jer se mora savršeno reproducirati niz elemenata kako bi dobili željen rezultat.

Programi koji se koriste u ovom radu su *Boujou*, *PFTrack* i *SynthEyes* koji su danas jedni od razvijenijih programa u filmskoj industriji. Među njima postoji velika razlika u cijeni, sučelju i načinu rada međutim svi oni imaju jedan zajednički cilj, dobiti dobru virtualnu 3D scenu neke snimke na koju bi se kasnije mogli ubacivati 3D elementi. Nakon ubacivanja 3D modela u scenu, vrlo je važno da se 3D model drži na sceni zalijepljen jer je upravo to rezultat koji se želi postići praćenjem kamere. Osim statičnog modela na sceni, u ovom radu uspoređivat će se tzv. „*error pixel*“ koji prikazuje odstupanja pojedinog modela od virtualne scene. Ukoliko je njegova vrijednost ispod 1, model bi trebao biti statičan na snimci, a ako je vrijednost „*error pixela*“ veća od 1, model bi se trebao pomicati po snimci, a mjerenja će pokazati je li to zaista tako i u praksi. Osim navedenih testova mjerit će se i brzina izračuna praćenja kamere pojedinog programa.

## 2. TEORIJSKI DIO

### 2.1. 3D računalna grafika

3D računalna grafika za razliku od 2D računalne grafike koristi trodimenzionalne geometrijske informacije spremljene u računalu u svrhu izračunavanja kalkulacija i prikazivanja istih kako bi se dobila dvodimenzionalna slika. Dvodimenzionalna grafika kao i 3D računalna grafika temelji se na vektorskim algoritmima koji tijekom rada daju vektorsku sliku, a nakon završetka, kao produkt iz 3D programa nastaje rasterska slika. [1]

Računalna 3D grafika se često naziva i „3D modeli“ ili „modeliranje“ ali je razlika između ta dva pojma velika. Modeli su samo brojčana reprezentacija povezanih točaka koji tek nakon procesa renderiranja tj. izračunavanja modela postaju grafika. Upravo taj proces prikazivanja virtualnog 3D modela naziva se 3D renderiranje. [1]

### 2.2. Povijest

William Fetter je bio računalni grafički direktor kojem se pripisuje termin 'računalna grafika' zbog njegova rada u kompaniji Boeing 1961. godine. Napravio je prvi računalni model ljudskog tijela te se od tada uvodi termin „računalna grafika“. [2]

Prvi primjer računalne animacije je pokretni prikaz ljudskog lica i ruke, napravljen od strane Eda Catmulla i Freda Parkea na Sveučilištu u Kaliforniji. Osim što je bio prvi veliki film koji je koristio 3D animaciju, korišteni su i počeci digitalnog komponiranja za postavljanje likova ispred pozadine. [2]

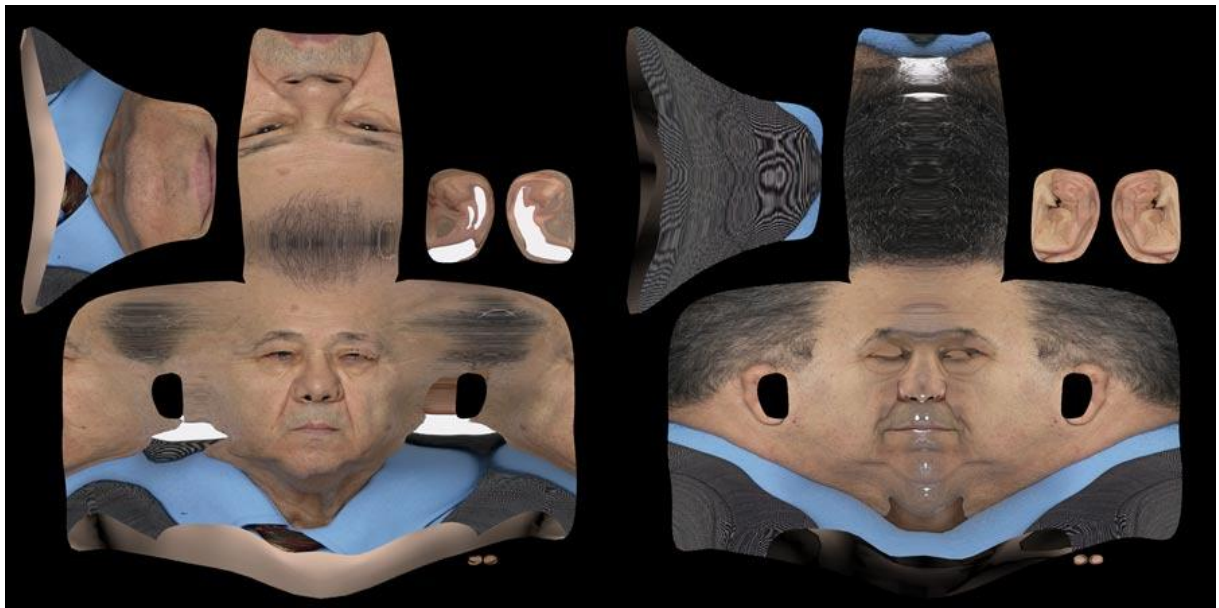
### 2.3. Modeliranje

Modeliranje je postupak kreiranja oblika željenog objekta. Tijekom tog postupka, korisnik koristi nacрте ili slike i mnoge alate kako bi kreirao određeni objekt. Alati koji se koriste su CAD alati iz kojih su nastali alati koji se koriste u 3D modeliranju. Danas se koristi i 3D skeniranje modela koje znatno olakšava i ubrzava proces modeliranja međutim zbog prevelike količine geometrije koju proces generira, modeli su često neupotrebljivi za 3D animaciju te najčešću primjenu nalaze u filmskoj industriji. [3]

## 2.4. Modeliranje teksture i postavljanje materijala

Modeliranje teksture je dodavanje boje ili slike prethodno izrađenom 3D modelu. Korištenjem raznih metoda (sferno, plošno, kubično i cilindrično mapiranje) može se postaviti željena slika na 3D model. Zbog kompleksnosti samih modela, koji najčešće nisu pravilnog oblika, dolazi do problema postavljanja teksture jer sama slika ne prati konture osnovnih geometrijskih oblika. Iz tog razloga razvijene su teksture koje se prilagođavaju 3D modelu te ne može doći do greške tijekom modeliranja teksture. Iako je ovaj način dobar, konačan rezultat ne daje realnu sliku pa je napravljen sustav UV mapiranja. [4]

UV mapiranje je proces spajanja 2D slike sa 3D modelom. U i V koordinate predstavljaju koordinatne osi 2D teksture. To je danas najkompliciraniji, ali ujedno i najefikasniji način mapiranja jer smanjuje potrebu za modeliranjem sitnih detalja što dovodi do smanjenja broja poligona na sceni i znatno štedi memoriju računala. Taj proces je kompleksan jer se 3D model prvo mora pretvoriti u dvodimenzionalnu mapu koja se zatim otvara u nekom od crtačkih programa kao što je *Photoshop*, gdje se koristi kao pozadina za crtanje teksture. [4]



*Slika 1: Izgled UV mape*

Izvor: [http://www.ten24.info/?page\\_id=445](http://www.ten24.info/?page_id=445), dostupno 20.07.2014. [16]

## 2.5. Renderiranje

Renderiranje je proces kojim se gotovi 3D modeli prebacuju u 2D sliku. Ovisno o kompleksnosti geometrije, materijalima, osvjetljenju i korištenom hardveru ovisi duljina renderiranja koje može potrajati i danima. Moderni *renderi* koriste dvije tehnike za izračun realne dvodimenzionalne slike, a to su *raytracing* i *global illumination*.

- ❖ *Raytracing* - omogućava preciznu simulaciju refleksija, refrakcija, sjena i kompleksne iluminacije. To je tehnika koja koristi matematičke algoritme i to na način da prati putanju zrake svjetlosti od njenog izvora do kamere i tako oponaša interakciju sa virtualnim objektima (slika 2). Često se naziva i direktno osvjetljenje jer ne uzima u obzir druge izvore svjetla. Nedostatak *raytracing*-a je velika količina računalnih izračuna kako bi se dobio realan prikaz. Koristi se u izradi statičnih slika, u filmu i televiziji. [5]



Slika 2: Sloj raytracing-a

Izvor: <http://hrcak.srce.hr/file/149149>, dostupno 20.07.2014 [17]

- ❖ *Global illumination* – je opći naziv za skup algoritama koji služe za dodavanje realističnijeg osvjetljenja jer simulira sekundarne izvore svjetla. Prilikom izračunavanja uzima se u obzir postotak svjetlosti koji upada na materijal i koji se emitira natrag u scenu. Ovisno o svojstvu hrapavosti, ta svjetlost se raspršuje, a njezina količina i boja ovisi o boji materijala i svojstvu difuznosti. Taj proces se nastavlja na sljedećem materijalu koji prima reflektiranu zraku itd. Postoji konačan broj refleksija koji program

računa, jer se količina reflektiranog svjetla smanji nakon određenog broja refleksije te nema smisla nastaviti nepotrebne kalkulacije. Taj broj kontrolira korisnik i jedan je od načina smanjenja vremena potrebnog za izračun jer je uz refleksije i prozirnost, globalna iluminacija najzahtjevnija komponenta renderiranja (slika 3). Globalna iluminacija uzima u obzir i ostale indirektno izvore svjetla poput neba koje emitira plavi dio spektra zbog raspršivanja sunčevog svjetla u atmosferi te se ponaša kao veliki polukružni izvor svjetla iz svih smjerova. Time se dobiva realnija slika eksterijera jer su sjene ispunjene svjetlom koje dolazi iz tog izvora te ostalim sekundarnim izvorima poput okolne geometrije. [5]



*Slika 3: Sloj globalne iluminacije*

Izvor: <http://hrcak.srce.hr/file/149149>, dostupno 20.07.2014 [17]

### 3. Match - moving

Praćenje kamere ili „*match-moving*“ je filmska tehnika koja omogućuje umetanje računalne grafike u *live – action* snimku s pravilnim položajem, razmjerom, orijentacijom i kretanjem u odnosu na snimljene objekte u kadru. *Match moving* se ponekad zamjenjuje sa „*motion capture*“ tehnikom koja se bazira na kretanju objekata, češće glumaca, a ne pokret kamere. *Motion capture* zahtjeva posebne kamere i senzore te kontrolirano okružje dok *match-moving* koristi snimke snimljene u normalnom nekontroliranom okružju s običnom kamerom. *Match-moving* se prvenstveno koristi za praćenje pokreta kamere kroz snimku tako da se identičan potez virtualne kamere može reproducirati u 3D programu za animiranje. Kada se novi animirani elementi sklapaju natrag u izvorni *live-action* kadar, oni će se pojaviti u savršeno usklađenoj perspektivi. *Match-moving* se bazira većinom na softveru, pristupačan je, služi kao alat za virtualne efekte te se koristi u mnogim televizijskim emisijama uživo, za efekte virtualne žute linije u Američkom nogometu i slično. [6]

Kod kompjuterski generiranog robota koji hoda preko ulice ili stoji na zgradi dok se kamera kreće, nema potrebe za programom za precizno 3D praćenje. Podudarajući pokrete kamere besprijekorno i dovoljno stabilno tijekom duljih sekvenci je teže, a sprječava probleme sa podrhtavanjem i zanošenjem koji bi inače narušili izgled objekta koji se nalazi točno u snimci. Uglavnom, u najboljem slučaju, *match-moving* omogućuje više slobode u smislu lokacije, čime štedi vrijeme i novac tako što se ne moraju postavljati velike setove ili platforme za kontrolu pokreta.

#### 3.1. Tracking – praćenje pokreta kamere

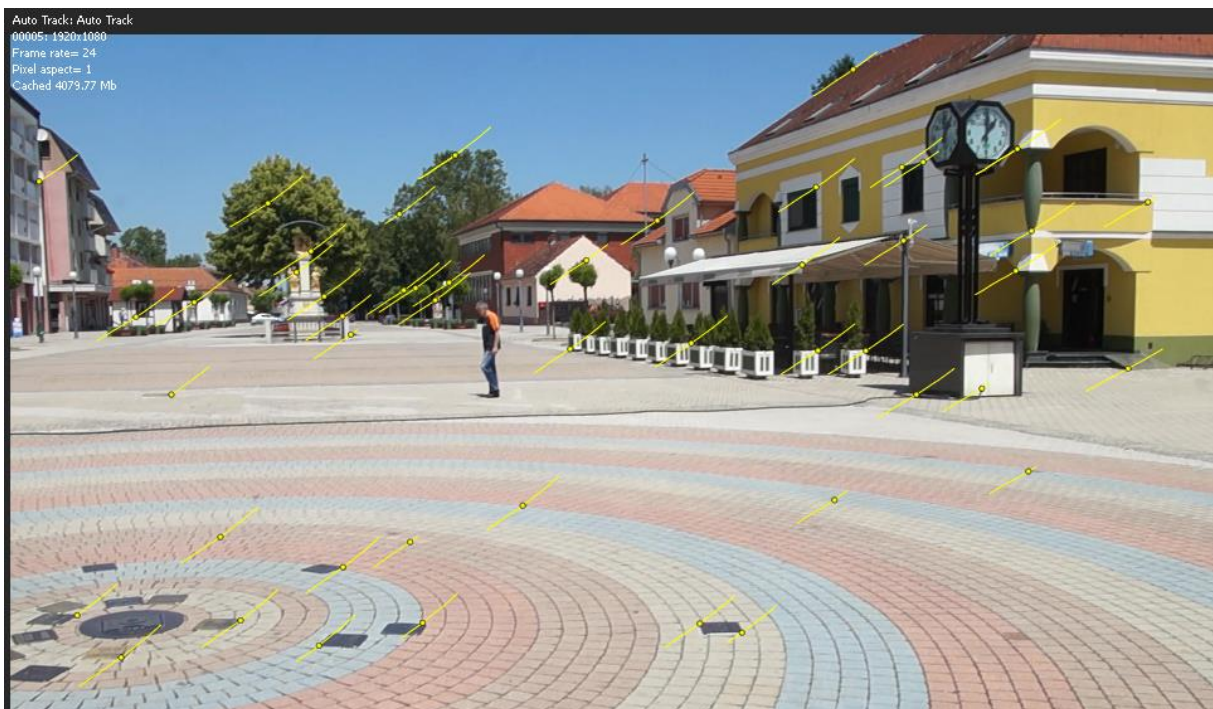
Prije nego se ubaci 3D model u snimku, potrebno je izvršiti proces praćenja pokreta. Bez tog procesa 3D model koji se dodaje bi prekrivao uvijek iste koordinate, a s time i poziciju na ekranu, a promjenom kuta i pozicije kamere na živoj snimci bilo bi očito da taj element ne pripada podlozi. [5]

Proces se sastoji od početnog zadavanja točaka koje će program pratiti kroz cijelu snimku. Iz tog oblaka točaka, čije su kretnje spremljene kroz cijeli video, program geometrijskim algoritmima poput paralakse kreira scenu gdje svakoj točki izračunava kretanje u 3D prostoru i njezin položaj. [5]



U trećem koraku program izračunava položaj kamere i uz pravilan postupak dobivaju se gotovo savršeni pokreti kamere, tako da se može vidjeti način na koji je držana kamera, ali u virtualnom trodimenzionalnom prostoru. [5]

Vrlo je važno da se dijelovi koji su na snimci pokretni (ljudi, automobili) maskiraju izvan izračuna ukoliko ih ne želimo pratiti (tzv. *garbage matte*), jer njihova paralaksa nije statična, te dolazi do velikih grešaka u pokretima krajnje virtualne kamere (slika 4). Svi programi na tržištu imaju ovakav proces praćenja kretnji, neovisno o tome da li koriste linearan proces prolaska kroz ove korake ili, u posljednje vrijeme, čvorišni.[5]

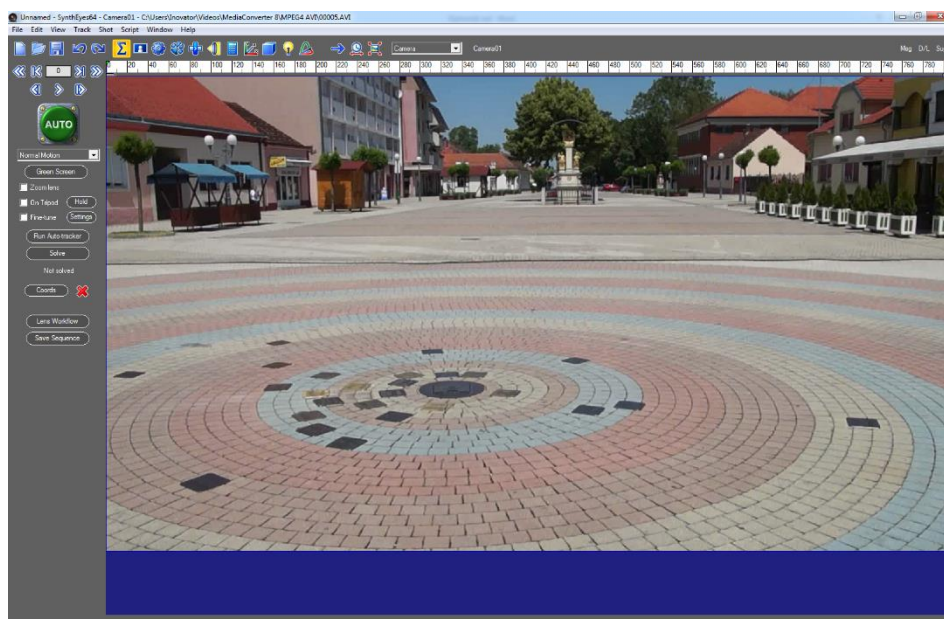


Slika 4: Izgled jednog kadra u procesu praćenja kamere

## 4. Korišteni programi

### 4.1. SynthEyes

SynthEyes je program za praćenje kamere i stabilizaciju slike koji nudi visoke performanse i mnogobrojne mogućnosti (slika 5). Koristi se u filmu, televiziji, reklamama, u glazbenoj video post produkciji, za popravljavanje drhtave slike, snimanje lica i tijela i još mnogo toga, a dostupan je za Windows, Mac i Linux platformu. Nudi kompletan *high-end* skup značajki uključujući praćenje, stabilizaciju i *motion capture*. Podržava praćenje kamere, praćenje objekata, praćenje objekata od referentnih mreža, praćenje kamere i objekata zajedno, praćenje više snimaka odjednom, stereoskopske snimke i još mnogo toga. Također, podržava snimke sa bilo kojom rezolucijom – DV, HD, film, IMAX, sa 8, 16 i 32 bitnim podacima, a mogu se koristiti na snimkama sa tisućama *frame*-ova. Izvanredno pojednostavljuje i ubrzava praćenje za snimanje na *green-screen*-u. [7]



Slika 5: Izgled učitane video snimke u SynthEyes program za praćenje

*SynthEyes* također ima mogućnost stabilizacije snimke gdje koristi svoje dvodimenzionalne i trodimenzionalne mogućnosti praćenja za generiranje stabilne slike. Ako se radi sa filmskim slikama, baš za televiziju, stabilizacijski sistem također pomaže kako bi se izbjegle neke uobičajene pogreške tijekom rada kao što su 3D praćenje, renderiranje i rad sa efektima. Ukoliko se radi sa dvije kamere, *SynthEyes* ne samo da može dodati 3D efekte već može pomoći podudaranje dvije snimke zajedno kako bi se smanjilo naprezanje očiju. Ovim programom može se vidjeti isječak snimke iz *live-action* snimke i odrediti kretanje prave



kamere, odrediti koje je bilo vidno polje, i koje su razne mogućnosti u 3D-u, tako da se mogu stvarati kompjuterski generirane slike koje se točno uklapaju u snimku. [7]

Značajke SynthEyes programa:[8]

❖ Vrste snimaka

- Praćenje kamere
- Praćenje objekta koji se kreće
- Stereoskopsko praćenje
- Potpuno automatsko praćenje za jednostavne snimke
- *Motion Capture* – za lice ili čitavog tijela pomoću više kamera
- Pregled snimaka – više nepokretnih snimaka snimljenih s različitih mjesta, s mogućnošću različitih formata slika i *zoomom*
- Ručni ili automatski izračun iskrivljenja objektivna

❖ Predodradba snimke

- Izbor R G B kanala i obratno
- Stabilizacija slike kojom upravlja korisnik integrirana sa 3D tracking-om
- Podešavanje boje i zasićenja
- Spremanje snimke na disk
- Visoko propusni filter za različite rasvjete kao što su bljesak i eksplozija
- Uklanjanje izobličenja leće – animirati za zoom sa izobličenjem
- Fleksibilno predodradbe snimke za lakše praćenje i skladištenje za filmove većih rezolucija

❖ Razdvajanje objekata

- Automatski odabir sistema za praćenje samo unutar *green-screen-a*
- Postavlja regije za svaki objekt koji se kreće
- Koristi *rotoscop* alfa-kanal
- Pohrana *roto-spline* i *green-screen* na disk

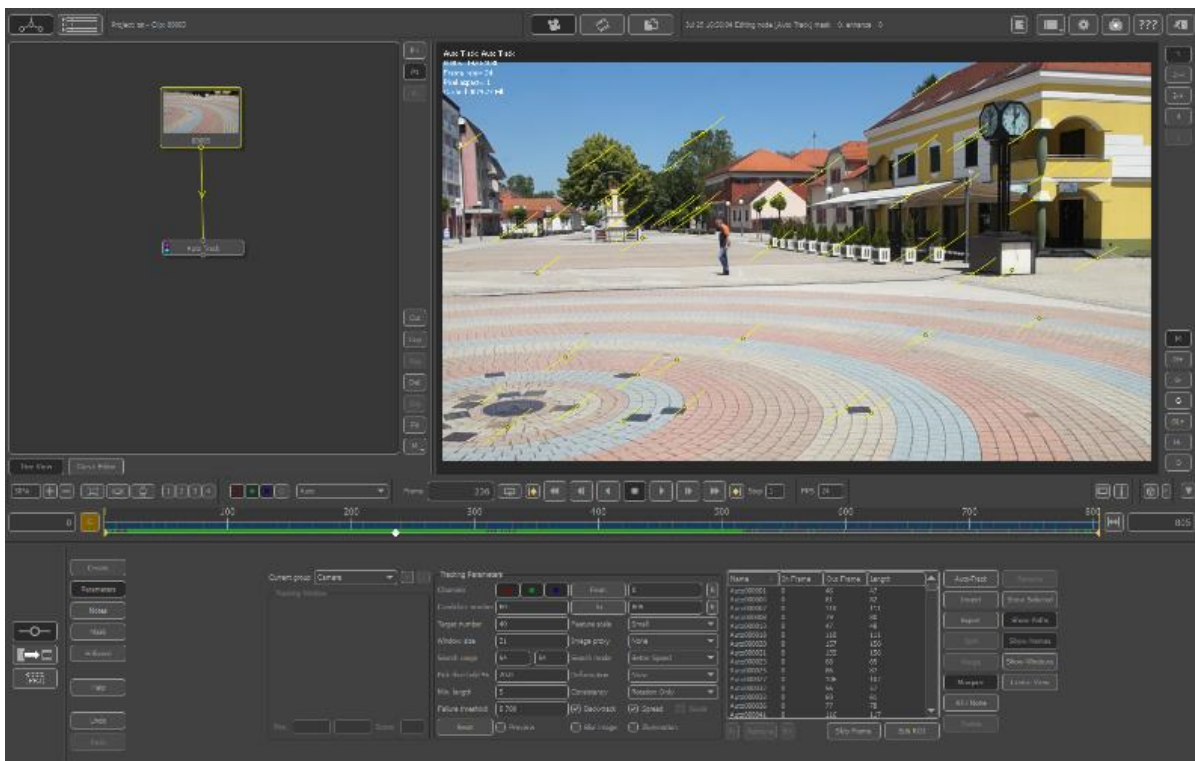
## ❖ Praćenje

- 3D i 2D planarni tracking podsustav, sa *auto - mask*, *roto - masking*, *alfa - masking*, *in-plane masks*, *layer ordering*
- *SimulTrack* prozor za simultano bezvremensko praćenje više sistema za praćenje
- Stereoskopsko automatsko praćenje s automatskim podudaranjem
- Pomiješano nadzirano i automatsko praćenje na istom kadru
- Podudaranje, bijela točka, crna točka, simetrija, *planar trackeri*
- *Offset trackeri*
- Dvosmjerno praćenje
- " Radar " zaslon za brzo identificiranje problematičnih točaka praćenja ili frame-ova
- Poseban način praćenja za ručno snimljene snimke
- Opcija da se ponovno prate automatski točke praćenja s nadziranim algoritmima za povećanje točnosti

## 4.2. ThePixelFarm PFTrack

*PFTrack* je od nedavno čvorišni program za praćenje pokreta na dvodimenzionalnoj snimci. Osim izračuna kretanja kamere *PFTrack* omogućuje razne vrste praćenje pokreta što uključuje i praćenje položaja pokretnih objekata na snimci, te generiranja informacija o njihovoj rotaciji i translaciji. [5]

Posljednja verzija *PFTracka* je iznova napisana tako da je u potpunosti 64 bitna, te podržava ubrzanje rada uz pomoć grafičkih kartica. Implementacija čvorišnog načina izrade je omogućila nelinearnost pri procesu izrade točke za praćenje što znači da se u svakom trenu može vratiti unatrag po čvorovima i promijeniti parametre već izvršenih radnji. [5]



Slika 6: Izgled učitane video snimke u *PFTrack* program za praćenje

Praćenje geometrije sada omogućuje i korištenje tzv. mekane geometrije, stoga su mogući i napredni procesi praćenja poput praćenja izraza lica i usana na glumcu. *PFTrack*, kao i većina današnjih programa za praćenje, podržava stereoskopski način rada, koji se koristi u izradi 3D filmova. [5]

*PFTrack* također podržava generiranje geometrije iz praćenih elemenata na slici, tako da se npr. površina poda može definirati kao jedna ploha i spremi je kao 3D model s teksturom preuzetom iz praćene snimke. Njegova velika prednost je mogućnost spremanja gotove 3D scene u nativnom formatu 3D programa, tako da je na korisniku da spremljenu scenu otvori u 3D programu bez potrebe za dodatnim *pluginovima* i uvoženjem podataka pozicija u zasebne objekte poput kamere. [5]

Značajke *PFTrack* programa: [9]

❖ Modeliranje slike

- Sa punom rezolucijom *Texture extraction*, generira detaljnu geometriju koja odgovara sceni, *shadow boxing* i niz rekonstrukcija
- Može se koristiti za konstruiranje 3D poligonalne modele koji odgovaraju elementima koje je snimila kamera
- Mogućnost izvoza geometrije za sve 3D programe
- Modeliranje bazirano na točkama
- Potpun izvod tekstura i mapiranja
- Nezavisna *motion* grupa

❖ Stereoskopsko praćenje

- Kada pratite stereoskopsku kameru, automatske postavke i postavke od strane korisnika istodobno prate lijevo i desno oko slike. Snimač ima potpun pristup podacima koji definiraju konvergenciju, interokularnu udaljenost itd.
- Podrška za dvije kamere lijevog i desnog oka
- Automatski izračun okulare udaljenosti i konvergencijske točke

❖ Procesiranje slike

- *PFTrack* ima alat „*Optical Flow*“ koji izračunava gustoću optičkog polja opisujući vidljivo kretanje objekta u odnosu na ravninu kamere
- Opsežne mogućnosti obrade slike uključuju ispravljanje *shutter*-a za popravljavanje pogrešaka slike koje potječu iz senzora kamere, prijenosnog klipa za obrezivanje, mijenjanja veličine i rotacijom izvora slike, uz optički tok za izračun i izvoz gustoće pokreta polja baziranog na vektorima

#### ❖ *Mocap Solver*

- Može se koristiti za kalibriranje pokreta pojedinih točaka za praćenje koje se vide iz dvije ili više kamera. To se često koristi kako bi se pratili pokreti glumačkog tijela ili lica gdje su točke koje se prate označene fizičkim markerima. Za razliku od standardnih praćenja pokreta, *Mocap Solver* čvor ne pretpostavlja da se objekt kreće u krutom načinu

#### ❖ Praćenje geometrije

- Može se koristiti za praćenje kamere ili objekta u pokretu pomoću trokutaste mreže umjesto praćenja točaka, što sprječava mnoge zamke koje muče konvencionalno praćenje kao što su zamucivanje, naglašavanje i bljeskovi
- Može se koristiti za praćenje deformiranih objekata kao lice koje govori. To se može postići izradom jedne ili više deformiranih skupina za praćenje, dodjeljivanjem trokuta u mrežu tih skupina i specificirati kako se skupine mogu transformirati u odnosu na ostatak mreže
- *PFFTrack* po pikselu *Z* dubine podržava maske kako bi odredio relativni poredak dubine objekata u sceni, kreirajući crno bijelu mapu dubina i trokutastu geometriju mreže
- Brza zamjena glave i lica koristeći uvezenu geometriju
- Praćenje keyframe-ova za veću točnost

#### ❖ *Node-based* graf rada

- Stablo praćenja omogućava korisniku da eksperimentira sa više tehnika u nelinearnom i nerazornom okruženju kako bi se postiglo najtočnije rješenje
- Kontrolira protok podataka kao što su čvorovi povezani da obavljaju različite zadatke u *PFFTracku* kao što su obrada slike, modeliranje lica i izvoz podataka. Čvorovi mogu biti beskrajno razgranati dopuštajući korištenje raznih tehnika kako bi postigli najtočniji rezultat

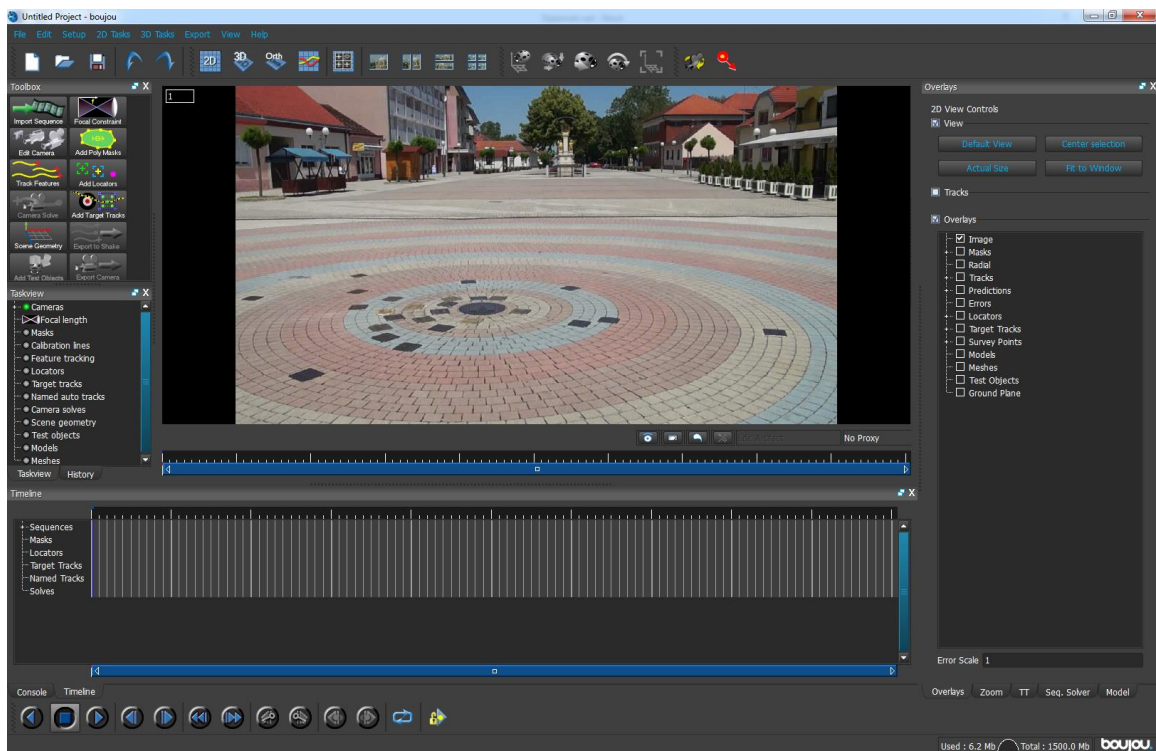
- ❖ Analiza optičkog protoka
  - Proračunski algoritam za optički tok
  - Višestruko poništavanje koraka
  - Mogućnost da se ignoriraju specifične korisnik/auto postavke iz fotoaparata
  
- ❖ Procjena žarišne duljine
  - Intuitivan 3D alat za praćenje
  - Izračun točne žarišne duljine bez podataka kamere

### 4.3. 2D3 Boujou

*2D3 Boujou* je popularan sistem za praćenje kamere i objekata prvenstveno poznat po svojim automatskim sposobnostima praćenja te je veoma jednostavan za korištenje. Pokrenut je 2001. godine, te je bio prvi potpuno automatizirani kalibracijski sistem za praćenje kamere koji je koristio napredne prilagodljive algoritme razvijene od strane znanstvenika. [10]

*Boujou* olakšava dodavanje 3D objekata u izravnu snimku u 3D animacijski paket stvarajući virtualnu kameru koja se poklapa sa fizičkom koja je zapravo i snimila scenu. Sve to, *Boujou* čini automatskim očitavanjem značajka u sceni duž cijele dužine kadra. Te značajke su automatski povezane jedna sa drugom. [10]

Također, *Boujou* ima potpuno opremljen graf urednik koji omogućuje manipulaciju i prilagodbu podataka po želji (slika 7). Mogu se uređivati podatci *frame* po *frame* što je osnova svakog programa ili se može ići preko granica, izgladivati, mijenjati vrijednosti, kopirati, lijepiti i brisati podatke po potrebi.[10]



Slika 7: Izgled učitane video snimke u Boujou program za praćenje

*Boujou* omogućava uvoz jednog ili više referentnih frame-ova ili sekvenci u glavnu snimku.

To omogućava:

- Dodavanje paralakse informacije
- Izradu prave 3D strukture za čvorišnu sekvencu gdje uopće nema 3D informacije u sekvenci

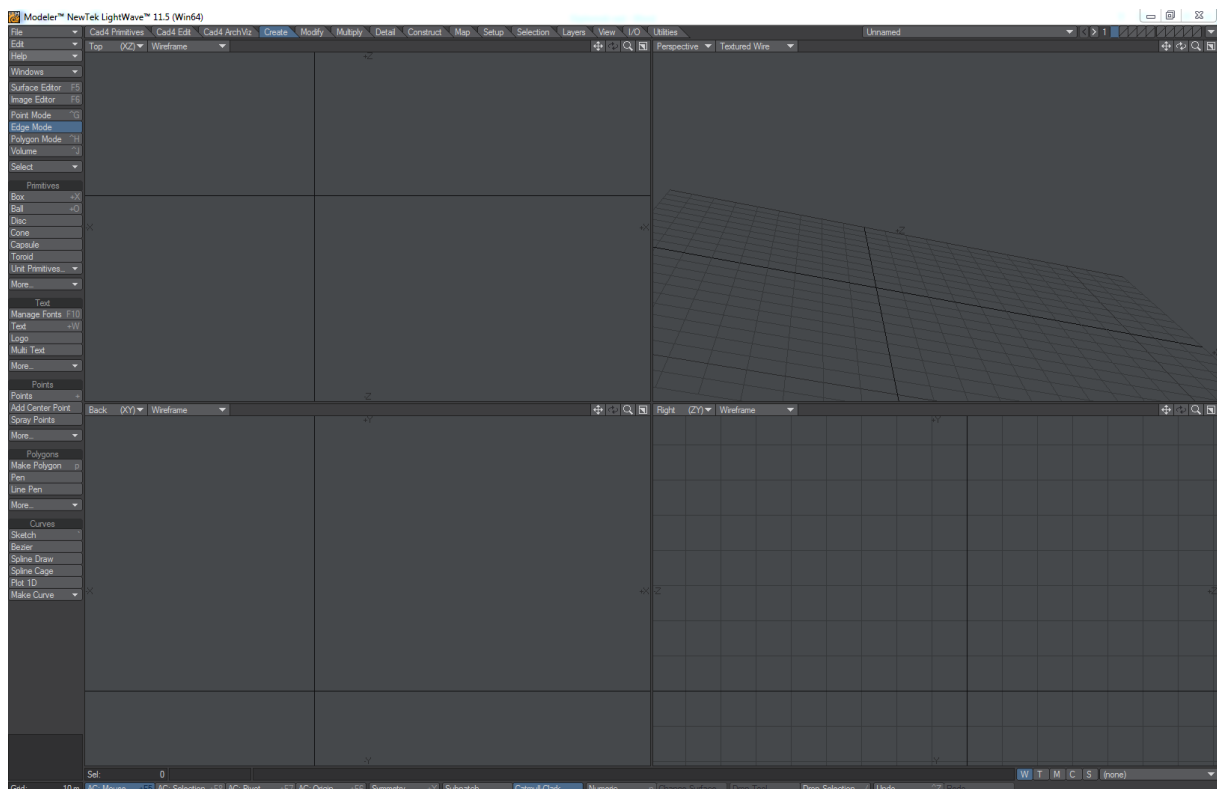
Ova značajka je posebno korisna za snimke niske paralakse ali može poboljšati točnost za različite vrste sekvenca. Može se uvesti više od jedne sekvence u projekt, može se odabrati da li se želi kreirati kamera za svaku sekvencu ili koristiti postojeću kameru. Kada se uvozi više od jedne sekvence, može se odabrati sekvencu na kojoj će se obavljati uobičajeni zadaci kao što je praćenje kamere i uvoz maske zasnivane na slici. [11]



#### 4.4. NewTek Lightwave 3D

*Lightwave 3D* je program koji na najbrži način pretvara trodimenzionalne scene u dvodimenzionalnu sliku ili animaciju ( proces poznat pod imenom *rendering*).

*Allen Hastings* je 1988. godine napravio program za animiranje i renderiranje koji se zvao *Videoscape*, dok je njegov prijatelj *Stuart Ferguson* napravio 3D program za modeliranje zvan *Modeler*. *NewTek* je planirao spojiti *Videoscape* i *Modeler* u svoj komplet za editiranje videa zvan *Video Toaster*. Na početku se trebao zvati *NewTek 3D* animacijski sustav *Amiga*, ali je ipak *Hastings* došao do danas popularnog imena „*Lightwave 3D*“. 1990. godine je *Video Toaster* paket pušten u prodaju, uključujući i *Lightwave 3D*. Četiri godine kasnije, 1994. Godine *Lightwave 3D* je pušten u prodaju kao samostalan program i to u verziji 9.3 koji je radio na *Windows* i *Mac OS X* platformi (slika 8). *NewTek* i *Lightwave 3D* su 1993. godine osvojili nagradu *Emmy*, te je 2004. godine *NewTek* je dobio nagradu *Emmy* za tehnologiju. [12]



Slika 8: Početni izgled *Lightwave-a*

Ne samo što je najbrži nego je njegova snaga renderiranja u svakoj novijoj verziji jača jer se nadodaju te poboljšavaju sposobnosti kao što su *radiosity*, *caustics*, *hypervoxels*, generiranje kose i krzna, meke refleksije i refrakcije. [12]

**Caustics:** pojava koja se javlja kada se svjetlost reflektira od zaobljenih predmeta ili se prelama kroz transparentne supstance tako da se fokusira na maloj površini. [13]

**Radiosity:** to je pojava reflektiranja svjetla od difuznih površina koja dovodi do toga da sama površina postaje indirektan izvor svjetla. [14]

**Hypervoxels:** to je nova tehnologija renderiranja koja nevjerovatno pojednostavljuje stvaranje ekstremno detaljnih trodimenzionalnih površina. Najveću primjenu ima kod kreiranja objekata kao što su vatra, dim, prašina, oblaci ili neki zadržali materijali. U nekoliko načina rada ova tehnika omogućava stvaranje skoro svakog efekta i nivoa detalja.

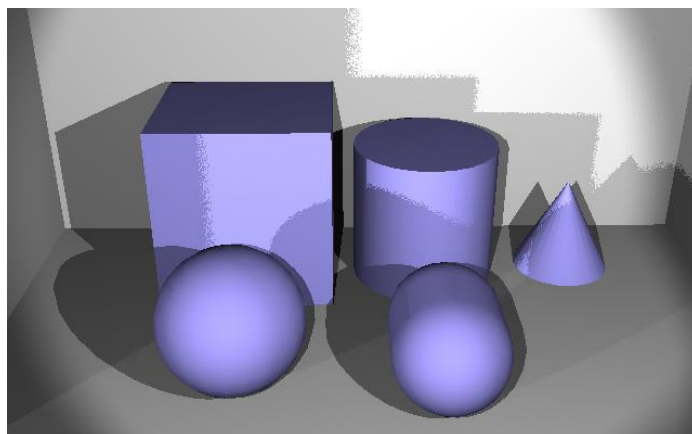
**Skytracer 2:** omogućava kreiranje realnih, sofisticiranih atmosferskih pojava. Sunce, mjesec, oblaci, rasipanje i apsorpcija svjetla u zraku, automatsko podešavanje vremena i pozicije koje može koristiti za simulaciju izlaska i zalaska sunca ili pomračenja, bilo na Zemlji ili na nekom drugom planetu.

Naravno, nije samo renderiranje ono što izdvaja *Lightwave* od drugih sličnih programa na tržištu. Jedno od njegovih najmoćnijih oružja je upravo modeliranje. Alata za modeliranje ima mnogo te je pomoću njih lako i brzo moguće napraviti željeni oblik. *Lightwave 3D* podržava višeslojne objekte sa virtualno neograničenim brojem slojeva, rad sa objektima u obliku poligona ili trodimenzionalnih površina definiranim krivuljama, veoma napredne opcije mapiranja tekstura i još mnogo toga. [12]

*Lightwave 3D* se koristio za stvaranje posebnih efekata za *Babylon 5*, *Zvijezdane staze*, *Voyager Space* te za razne znanstveno fantastične televizijske serije. Također se koristio u produkciji *Titanica*, *Star Wars* filmova, *Sin City* i drugih svjetski poznatih filmskih naslova. [12]

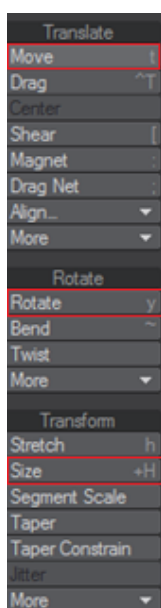
## 5. Lightwave alati za manipuliranje

Za potrebe ovog rada nije potrebno modelirati komplicirane modele već su dovoljni osnovni oblici koje ima sam program jer je cilj prikazati kako se ubacuje model u određeni program za praćenje te kako ga određeni program prepoznaje (slika 9). Modeli su izrađeni poligonalnim modeliranjem. Svaki model se sastoji od određenog broja poligona, od kojih svaki poligon ima svoje lice, također ima linije tj. rubove koji se nazivaju *edgevi*, a mjesta na kojima se oni sijeku su točke.



Slika 9: Alati za manipuliranje

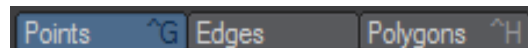
Neke od osnovnih funkcija za manipulaciju objektima su pomicanje, skaliranje i rotacija. Njih se može primijeniti na cijeli model ili na određeni dio modela. Ovi alati se nalaze na glavnoj alatnoj traci pod izbornikom „Modify“ (slika10).



Slika 10: Manipulatori -pomicanje, skaliranje i rotacija

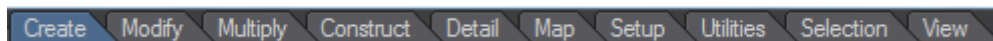
Također, pod tim istim izbornikom postoje još dodatne funkcije za manipulaciju kao što su *Magnet* alat koji služi za skalpiranje modela u specifičan oblik, *Twist* alat služi za okretanje točaka ili poligona, *Taper* je alat koji služi za skaliranje, *Bend* alat služi za savijanje modela. Naravno, postoji još mnogo alata za manipuliranje ali ovo su oni koji se najčešće koriste. Također, na glavnoj alatnoj traci postoje mnogi izbornici sa alatima i funkcijama kojima se može služiti. Izbornici i alati se mogu posložiti prema potrebama odnosno, onako kako najbolje odgovara. [15]

Selekcija pojedinih dijelova modela vrši se pomoću tipke „space“ koja selektira točke, rubove ili stranice. Znači, ako se želi označiti točku, rub ili stranicu na modelu, samo pritiskom „space“ tipke dolazi se na određenu funkciju koja se želi primijeniti na model (slika 11). [15]



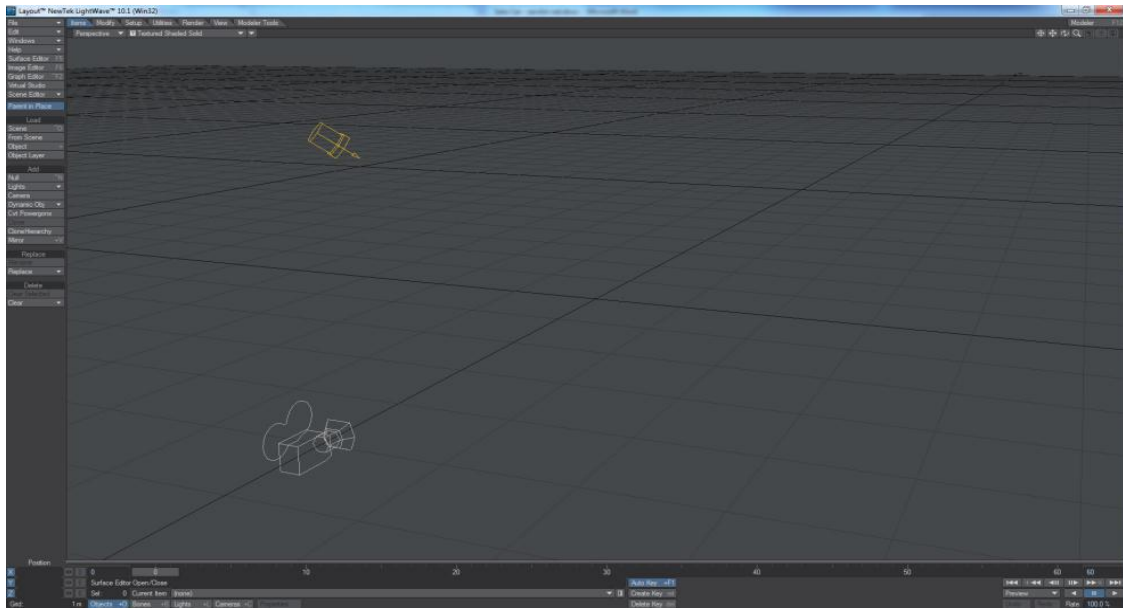
Slika 11: Alat za selektiranje

Glavni izbornik, odnosno alatna traka je zapravo navigacija za određene funkcije i alate koji se želi koristiti. U tim izbornicima su smješteni oni alati koji opisuju određeni izbornik. Npr, u izborniku *Create* se nalaze alati koji omogućuju da se nešto napravi, tj. modelira. Tako vrijedi za sve izbornike. Svaki izbornik nosi naziv koji je logičan te sa time čini *Lightwave* jako pristupačnim i jednostavnim za korištenje (slika 12). [15]



Slika 12: Alatna traka

Za neke modele koji su kompleksniji, npr. modeliranje automobila ili bilo kakvog drugog vozila ili kuće po točnim dimenzijama trebalo bi imati nacrt kako bi se modeliralo po točnim mjerama jer su to modeli koji zahtijevaju preciznost budući da se svaki segment modeliranja veže na onaj sljedeći. Potrebno je imati nacrt, tlocrt i bokocrt. Te tri slike se zatim stavljaju unutar radnoga prostora *Lightwave*-a pazeći da se međusobno podudaraju i dodiruju po jednoj stranici svoje dužine. Pravilno postavljene nacrti osnova su za kreiranje objekata.



Slika 13: Izgled Lightwave Layout-a

Za potrebe ovog istraživanja nije potreban nacrt jer su modelirani modeli jednostavniji koji ne zahtijevaju određenu preciznost. Nakon što se modelira model u Lightwave Modeler-u, da bi se renderirala scena, cijeli objekt se mora prebaciti u *Lightwave Layout* koji je direktno povezan sa *Lightwave Modeler-om*. *Layout* je zaseban program koji služi za renderiranje dok je *Modeler* samo za modeliranje. Ostali 3D programi nemaju ovakav princip rada već u jednom programu imaju sve. *Layout* izgleda slično kao i *Modeler* samo što nema funkcije za modeliranje i ostale alate kao *Modeler* već je namijenjen za renderiranje i animiranje (slika 13).

## 6. Opis SynthEyes programa

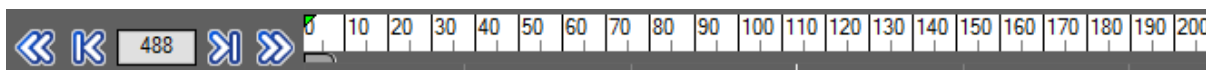
*SynthEyes* program se sastoji od alatne trake na kojoj se nalaze prečice do najkorištenijih funkcija koje program nudi (slika 14).



Slika 14: Alatna traka SynthEyes programa

Pritiskom na određeni alat mogu se mijenjati parametri kamere, svijetla, dodavati 3D modeli i niz drugih mogućnosti, a sve dodatne funkcije određenih parametara mogu se vidjeti sa lijeve strane glavnog prozora programa.

Ispod alatne trake nalazi se vremenska traka (slika 15) na kojoj se nalaze funkcije za puštanje snimke, pauziranje, ubrzavanje, vraćanje ili ručno upisivanje željenog frame-a.



Slika 15: Vremenska traka

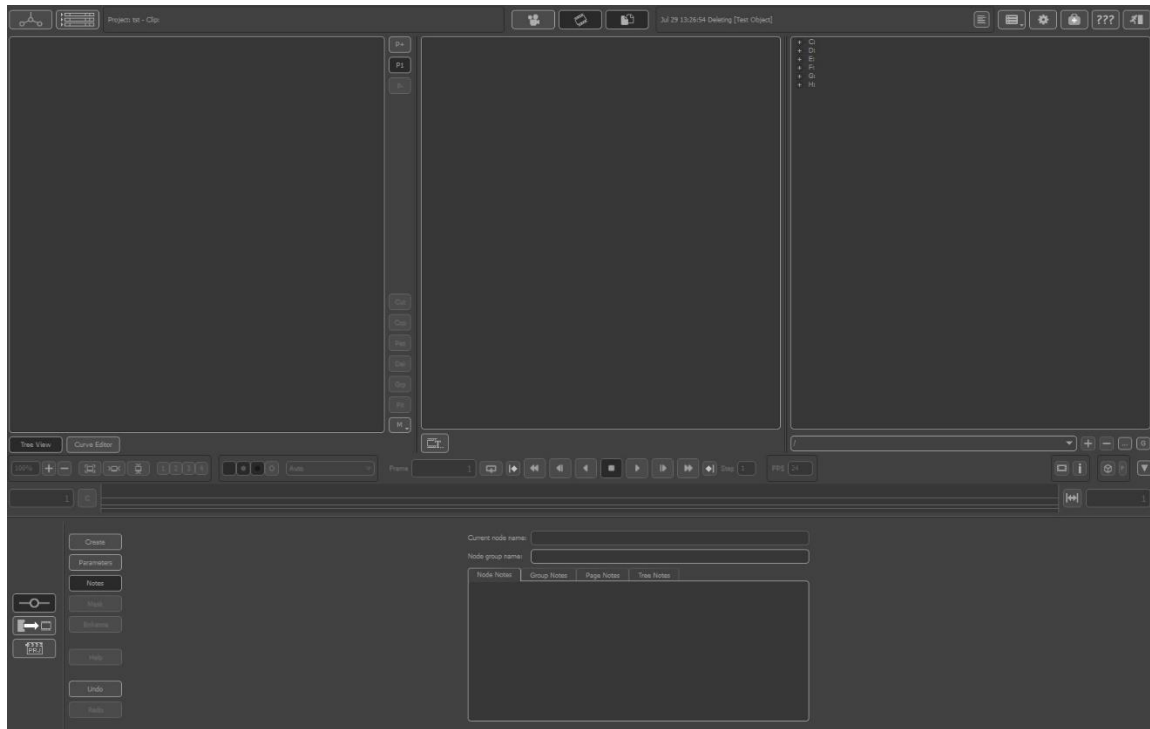
Sam program je vrlo jednostavan za korištenje i vrlo je brz, za razliku od ostalih programa za praćenje kojima je potrebno puno duže vremena da se pokrenu. Osim što se ističe brzinom, cijenom je povoljniji od drugih programa što je prikazano u tablici 1 i to za čak nekoliko desetaka puta.

Tablica 1: Cijene programa za praćenje

Naziv programa	verzija	cijena
SynthEyes	2011	€ 230
2d3 boujou	5.0	€ 7,500
PFTrack	4.1	€ 1,249

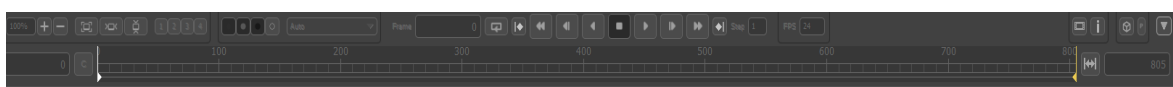
## 7. Opis PFTrack programa

Sučelje *PFTrack* programa je podijeljeno u nekoliko glavnih prozora. Sa desne strane nalazi se prozor iz kojeg se može učitati željena datoteka, srednji prozor prikazuje datoteke koje se nalaze u trenutnom projektu dok lijevi prozor prikazuje što se trenutno nalazi na sceni na kojoj se radi (slika 16).



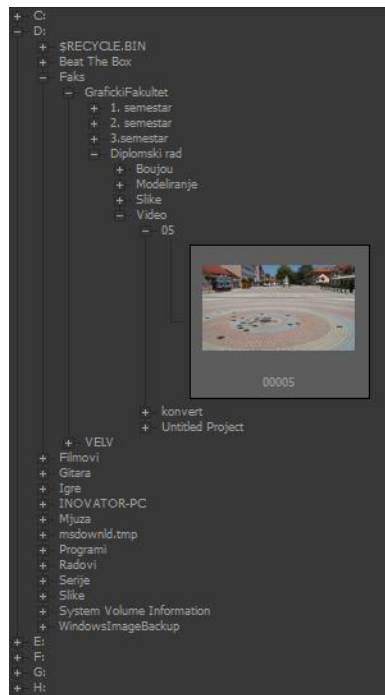
Slika 16: Izgled PFTrack programa

Alatna traka se nalazi sa na donjoj lijevoj strani programa koja ima nekoliko osnovnih funkcija, kao što su kreiranje novog projekta, umetanje postojećeg projekta te upravljanje čvorovima. Odabirom na kreiranje novog projekta, otvara se dodatni prozor u kojem se upisuje ime projekta, gdje se želi pohraniti, odabir željenog broja *frame*-ova te odabir kamere kojom je snimana scena kako bi sam program što točnije izračunao praćenje kamere. Vremenska traka se nalazi iznad alatne trake i sadrži sljedeće elemente: vremensku traku, tipke za premotavanje snimke, tipku za sinkronizaciju, gumb za prikaz snimljenih podataka. (slika 17). Iznad vremenske trake se nalaze tri prozora koji se mogu uključivati i isključivati po želji korisnika.



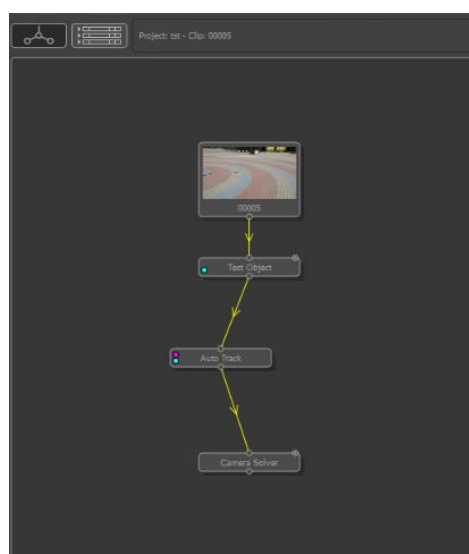
Slika 17: Vremenska traka PFTrack programa

U prozoru sa desne strane odabire se željena snimka koja se nalazi na vašem računalu (slika 18).



Slika 18: Izbornik za učitavanje datoteka u program

Nakon toga, snimka se premješta u srednji prozor koji prikazuje trenutne snimke u projektu dok je lijevi prozor glavni prozor za upravljanje snimkom gdje se u samu snimku ubacuju željeni čvorovi kao što su: test objekt, automatsko praćenje, izračunavanje kamere i mnogi drugi (slika 19). Nakon što se odabere željeni čvor, otvaraju se dodatne opcije uključenih čvorova. Čvorovi se također mogu dodavati pritiskom na prvi gumb alatne trake.



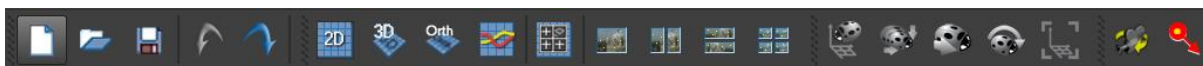
Slika 19: Izgled ubačene snimke u scenu sa dodanim čvorovima



## 8. Opis Boujou programa

*Boujou* izgleda slično kao i ostali programi za praćenje kamere. Sastoji se od nekoliko glavnih dijelova:

### 1. Početne trake

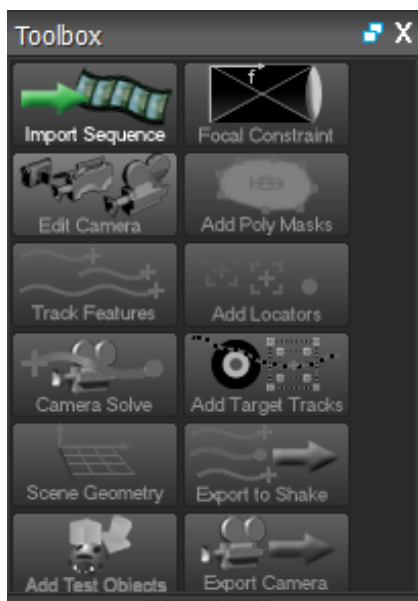


Slika 20: Početna traka Boujou-a

Na početnoj traci se nalazi niz akcija koje omogućavaju korisniku da na brz i jednostavan način dođe do željene funkcije. Na ovoj traci se nalaze alati kao što su izrada novog projekta, otvaranje postojećeg projekta, spremanje projekta, vraćanje koraka unazad/naprijed, alati za postavljanje prozora u 2D ili 3D pogled, na 2 ili više prozora itd. (slika 20).

### 2. Trake sa alatima

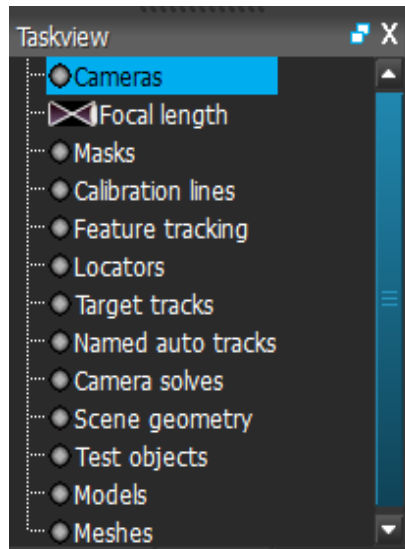
Na traci sa alatima se nalazi niz ikona koje omogućavaju umetanje snimke u projekt, uređivanje kamere, značajke praćenja, geometriju scene, dodavanje objekata za testiranje, dodavanje lokatora te izvoz snimke (slika 21).



Slika 21: Alatna traka Boujou programa

### 3. Izbornika sa zadacima

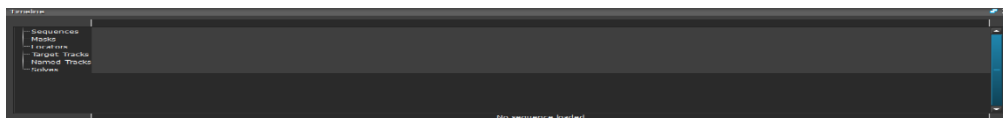
U izborniku sa zadacima nalaze se detalji snimke, geometrija scene, testni objekti, modeli, lokatori i niz drugih značajki. Prilikom aktiviranja svake značajke, upali se zeleni kružić koji signalizira da je uključena određena opcija. (slika 22).



Slika 22: Popis zadaća koje nudi Boujou

### 4. Vremenske trake

Nalazi se na dnu programa te prikazuje duljinu snimke kroz koju se može jednostavno „šetati“ koristeći ponuđene funkcije (slika 23). Vremenska traka radi na istom principu kako je to opisano u PFTrack programu.



Slika 23: Vremenska traka Boujou-a

## 3. EKSPERIMENTALNI DIO

### 3.1. Snimanje žive snimke

Snimanje kadrova izvršeno je na području grada Ludbrega na tri lokacije. Na svakoj lokaciji snimljena su tri kadra sa različitim tehnikama snimanja. Dva kadra su snimljena na stativu i to na način da se kamera kretala u svim smjerovima: gore, dolje, lijevo, desno. Drugi kadar je snimljen na način da se kamera pomicala naprijed-nazad s time da je u snimci uvijek isti kadar koji zapravo izgleda da se udaljava tj. približava. Treći kadar je snimljen ručno, bez stativa s ciljem da se testira kako određeni program prati kameru koja se drži u ruci i trese se jer je taj način praćenja zahtjevniji od praćenja kamere na stativu.

Kadrovi su snimljeni kamerom Sony SLT-A57, leće Konica Minolta AF 17-35mm F/2.8-4, trajanja oko 25 sekundi. Stolno računalo na kojem je izvršeno istraživanje sastoji se od sljedećih komponenti: četvero-jezgri procesor Intel Core i5-2380P na 3,1GHz, 16GB DDR3 radne memorije te sa grafičkom karticom Nvidia GeForce GTS 450 sa 2GB memorije.

### 3.2. Praćenje kamere ili tracking

Za potrebe ovog istraživanja korišten je program *NewTek Lightwave* 3D verzija 11.5, 64 bitna, te tri programa za praćenje kamere, a to su: *SynthEyes*, *Boujou* i *PFTrack*. Navedeni programi za praćenje kamere se koriste u profesionalne svrhe te se koriste u filmskoj industriji za dodavanje vizualnih efekata međutim, iako se koriste za iste namjene u specifičnim situacijama pojedini programi daju bolje rezultate. Važno je napomenuti da su prilikom ovog istraživanja tj. testiranja navedenih programa, sve opcije bile po tvorničkim postavkama kako bi se dobili najvjerodostojniji rezultati.

### 3.3. Ubacivanje snimke u PFTrack program za praćenje

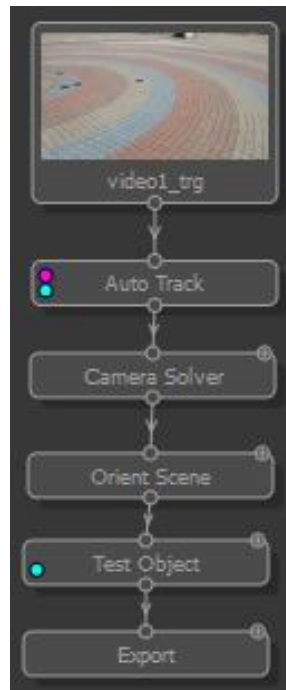
Prije samog ubacivanja snimke u program, bilo je potrebno konvertirati snimke koje su .mts formata u .jpeg sekvence jer se upravo taj format koristi u filmskoj industriji te ga svi programi podržavaju. Nakon toga, snimka je spremna za ubacivanje u program za praćenje. Nakon otvaranja *PFTrack* programa, odabirom „*Project Panel*“ otvara se novi prozor u kojem se kreira novi projekt. U tom prozoru dodaje se ime projekta, destinaciju na koju se sprema, broj frame-ova te vrstu kamere sa kojom je sniman kadar kako bi dobili što točnije rezultate.

Nakon toga ubacuje se željena snimka sa računala u program koju lijevim klikom miša povučemo u lijevi prozor kako bi proces praćenja kamere mogao početi (slika 24).



Slika 24: Ubacivanje video sekvence u program

Desnim klikom miša na učitani snimku, otvara se novi prozor sa velikim brojem čvorova koje program nudi. Za potrebe ovog istraživanja korišteni su čvorovi za „*auto track*“, „*camera solve*“, „*orient scene*“, „*test object*“ i „*export*“ (slika 25).

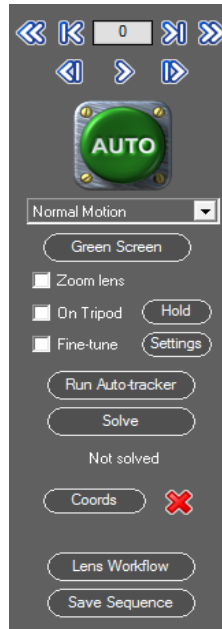


Slika 25: Dodavanje čvorova u video

Čvor „*auto track*“ služi za proračun praćenja kamere, „*camera solve*“ stvara 3D kameru prema pravoj kameri kojom je snimano, „*Orient scene*“ služi za označavanje koordinatnih osi koje zapravo predstavljaju podlogu na kojoj će biti budući 3D model, „*test object*“ služi za ubacivanje testnih objekata kako bi provjerili je li praćenje kamere bilo uspješno ili ne. Nakon provedbe navedenih zadataka, potrebno je eksportirati scenu u format koji nam je potreban, a to je u ovom slučaju *.lws* jer će se ova scena otvarati u *Lightwave-u*.

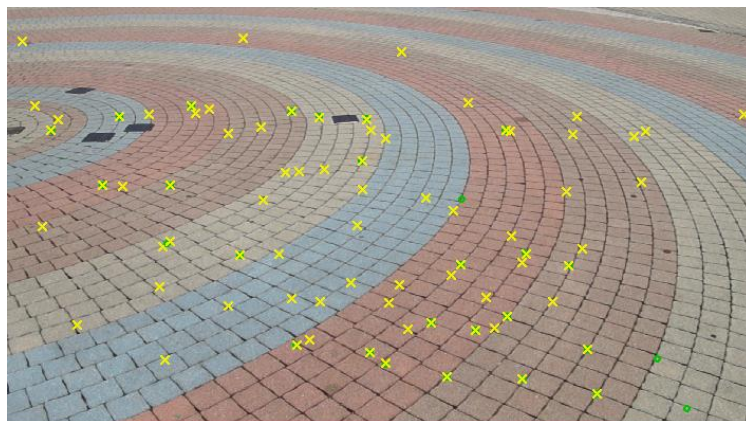
### 3.4. Ubacivanje snimke u SynthEyes program za praćenje

Ubacivanjem snimljenih sekvenci u *SynthEyes* program, otvara se prozor u kojem se mogu podesiti razne funkcije kao što su početak i kraj frame-a koji se želi pratiti, odabir kamere kojom je snimano, podešavanje boje snimke, rezanje snimke, podešavanje leća i sl. (slika 26).



Slika 26: Glavni prozor za praćenje

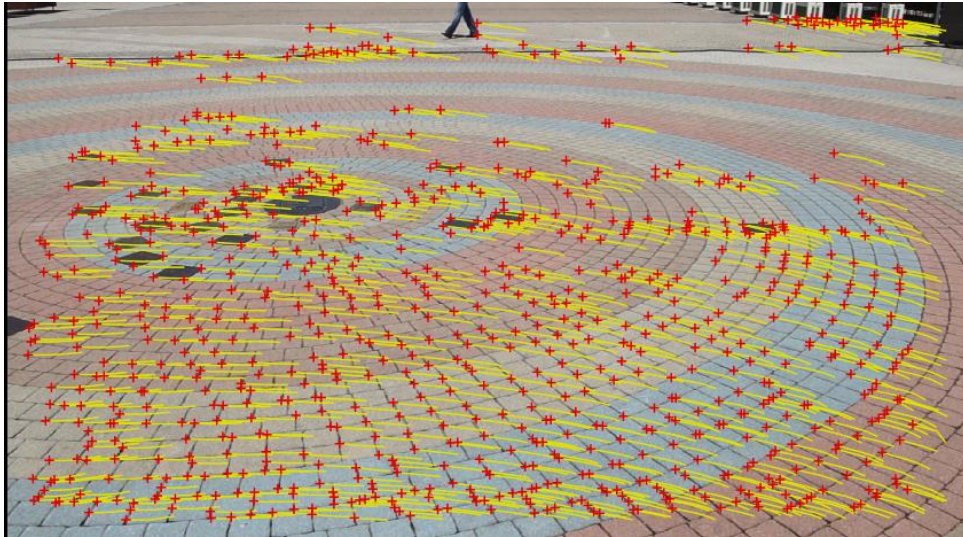
Nakon željenih podešavanja, otvara se snimka u glavnom prozoru programa u kojem slijedi praćenje kamere pritiskom na tipku „*auto*“ koja se nalazi sa lijeve strane početnog prozora. Pri završetku praćenja kamere potrebno je uključiti opciju „*Camera Solve*“ koja se nalazi u istim opcijama kao i funkcija „*auto*“. „*Camera solve*“ stvara točke 3D kamere koje prate smjer prave kamere (slika 27). Nakon toga potrebno je zadati koordinatne osi, x, y i z kojom se određuje tlo određene snimke. Nakon postavljanja osi, datoteka se može eksportirati u *.lws* format koji je potreban da bi se učitala snimka u *Lightwave-u*.



Slika 27: Izgled snimke nakon izvršenog procesa "praćenja kamere"

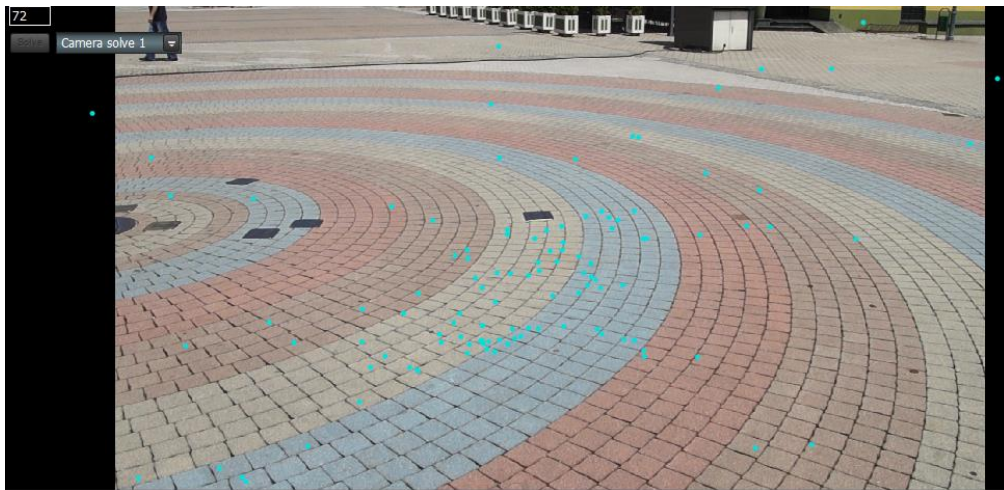
### 3.5. Ubacivanje snimke u Boujou program za praćenje

Nakon otvaranja programa *Boujou*, odabirom „*Import sequence*“ učitava se video. Odmah u početnom prozoru nakon otvaranja scene odabire se od kojeg do kojeg frame-a se želi pratiti kameru te nakon odabira se primijeni („*apply*“). Nakon što je slika učitana može se početi sa tracking-om klikom na tipku „*Track features*“ koja pokreće proces praćenja kamere (slika 28).



Slika 28: Izgled praćenja kamere u Boujou programu

Nakon završetka praćenja kamere, klikom na tipku „*Camera solve*“ uklanjaju se nepotrebne točke praćenja te se određuje pozicija kamere u 3D prostoru (slika 29).



Slika 29: Izgled 3D kamere

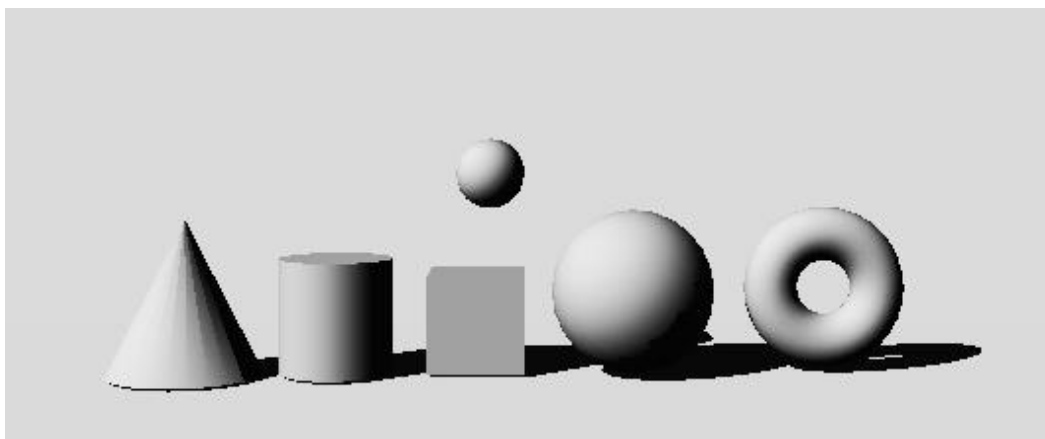
Na kraju se zadaju koordinatne osi kako bi se odredila pozicija mreže, tj. podloge koja označava x, y i z osi. Nakon spomenutih postavki cijela scena se eksporta u *.lws* format kako bi ga mogli kasnije učitati u *Lightwave* programu.



## 4. Stvaranje 3D grafike

### 4.1. Modeliranje

Za potrebe ovog istraživanja modelirani su jednostavni primitivni oblici koji će prikazati da li se u testiranim scenama oni drže na snimci ili lebde po zraku (slika 30). Sami modeli nisu dodatno obrađivani i nemaju nikakve efekte koji bi zasigurno doprinjeli realnosti konačnog rezultata međutim naglasak je na tome da se taj model drži na sceni, tj. da bude fiksiran.



Slika 30: Izgled 3D modela ubačenog u scenu

### 4.2. Stvaranje virtualne scene

Nakon kalkulacija kamere za sve snimke, u 3D program uvezena su kretanja spremljena iz *SynthEyes*, *Boujou* i *PFTrack* programa za praćenje. Uvezene datoteke imaju informacije o položaju i rotaciji kamere tako da je u *Lightwave Layout* programu vidljiva 3D pozicija istih. Također, pošto je u programima za praćenje pozicioniran grid tj. mreža koja predstavlja tlo snimke, istu nije bilo potrebno postavljati u 3D programu već je dovoljno smjestiti željeni 3D element u snimku te vidjeti da li će on biti zalijepljen za plohu ili će u slučaju lošeg „*tracka*“ tj. praćenja kamere lebjeti u zraku.

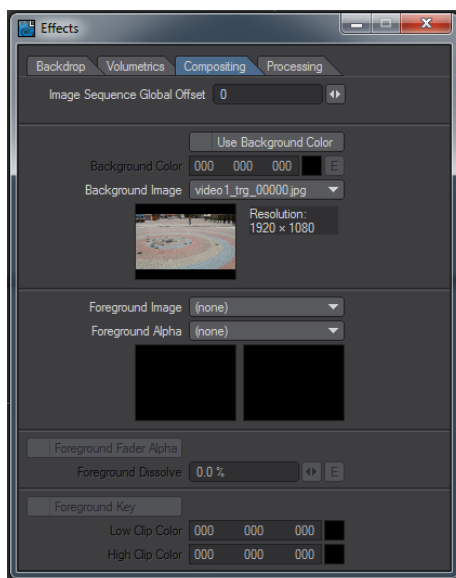
### 4.3. Ubacivanje i integriranje modela u scenu

Nakon otvaranja *Lightwave Layout-a* potrebno je ubaciti željenu scenu kako bi se mogao ubaciti 3D model. Nakon što se scena otvori vidljive su sitne plave točkice koje predstavljaju kretanje i poziciju 3D kamere (slika 31).



Slika 31: Prikaz kretanja 3D kamere

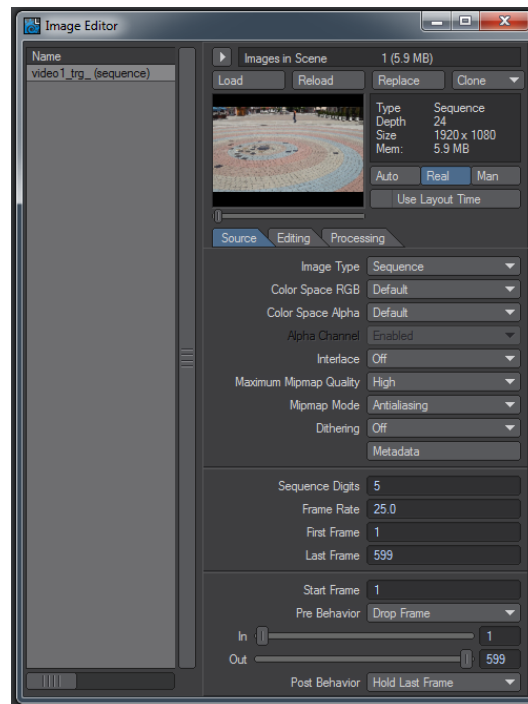
Nakon ubacivanja snimke potrebno je postaviti pozadinu tj. video sekvencu koja je prethodno snimljena da bi se realnije vidjela scena na kojoj se radi. Klikom na *Items – Windows – Compositing options*, pod opciju *Background Image* mora se staviti sekvenca videa na kojem se trenutno radi (slika 32).



Slika 32: Umetanje sekvence u 3D snimku



U *Image editoru* potrebno je podesiti sljedeće postavke kako bi programu signalizirali da se radi o video sekvenci, a ne o statičnoj slici: *Image Type: Sequence*, *Start Frame: 1* (slika 33).



Slika 33: Podešavanje postavki u "Image editor-u"

Pritiskom na tipku 6 prikaže se „*Camera View*“ te se vidi snimka i mreža koja je postavljena na scenu prethodno u programu za praćenje kamere (slika 34).



Slika 34: Prikaz video sekvence i mreže u 3D programu

Pošto je cijela scena postavljena, može se dodati 3D objekt i pokrenuti proces renderiranja kako bi se dobio konačan rezultat (35). Vrlo je važno pratiti da li se 3D model koji je postavljen na scenu pomiče ili je statičan. Ukoliko je statičan, tj. ni u jednom trenu se ne pomiče, proces praćenja kamere je uspješno napravljen.



*Slika 35: Umetnut 3D model na snimci*

Nakon što je postavljen 3D model na scenu, cijelu snimku je potrebno renderirati u *.png* sekvence. Prije početka samog renderiranja, potrebno je podesiti nekoliko postavki u *NewTek Lightwave Layout* programu za renderiranje. Na alatnoj traci pod izbornik „*Render*“ odabere se opcija „*Render Globals*“ – „*Output*“ te označi kvačica na RGB i odabere format eksporta „*LW\_PNG\_32*“ kako bi se dobile *.png* sekvence koje se kasnije spajaju u *After effects* programu. Na isti način su rađeni testovi sa svim scenama i sa sva tri programa.

## 4. REZULTATI ISTRAŽIVANJA

Svi snimljeni kadrovi su dužine 25 sekundi kako bi se dobili vjerodostojni rezultati jer su 3 videa snimana na 3 različite lokacije. Prvi snimak je napravljen na gradskom trgu u Ludbregu gdje se u kadru mogu vidjeti ljudi koji prolaze, drugi kadar je snimljen na livadi kako bi se pokazala kompleksnost ubacivanja 3D modela na scenu dok je treći kadar snimljen na cesti gdje prolaze automobili. U svakom od testova ispitivano je vrijeme praćenja kamere te „*error pixel*“ koji je kod svakog programa drugačiji te se na kraju svih potrebnih radnji znalo da li se određeni 3D model drži na sceni ili lebdi u zraku. Za potrebe ovog istraživanja svi programi su testirani sa automatskim postavkama.

### 4.1. Test 1

Lokacija: Trg sv. Trojstva u Ludbregu

Vrsta snimanja: u svim smjerovima na stativu

Nakon ubacivanja video sekvence „trg“ u svaki od programa dobiveni su sljedeći rezultati:

Tablica 2: Rezultati 1. video isječka

Program	Vrijeme računanja	Error Pixel	Uspješnost praćenja
SynthEyes	1:21 min	1,891	-
PFTrack	2:32 min	0,960	-
Boujou	4:44 min	0,965	-

### 4.2. Test 2

Lokacija: Trg sv. Trojstva u Ludbregu

Vrsta snimanja: naprijed - nazad na stativu

Tablica 3: Rezultati 2. video isječka

Program	Vrijeme računanja	Error Pixel	Uspješnost praćenja
SynthEyes	1:22 min	2,171	-
PFTrack	2:32 min	0,974	-
Boujou	4:30 min	0,905	-

### 4.3. Test 3

Lokacija: Trg sv. Trojstva u Ludbregu

Vrsta snimanja: u svim smjerovima bez stativa

Tablica 4: Rezultati 3. video isječka

Program	Vrijeme računanja	Error Pixel	Uspješnost praćenja
SynthEyes	1:10 min	3,345	+
PFTrack	2:30 min	0,946	-
Boujou	4:30 min	1,393	+

### 4.4. Test 4

Lokacija: Livada

Vrsta snimanja: u svim smjerovima na stativu

Tablica 5: Rezultati 4. video isječka

Program	Vrijeme računanja	Error Pixel	Uspješnost praćenja
SynthEyes	1:22 min	0,811	+
PFTrack	2:26 min	0,854	-
Boujou	4:12 min	1,177	-

### 4.5. Test 5

Lokacija: Livada

Vrsta snimanja: naprijed - nazad na stativu

Tablica 6: Rezultati 5. video isječka

Program	Vrijeme računanja	Error Pixel	Uspješnost praćenja
SynthEyes	1:03 min	0,936	+
PFTrack	2:20 min	0,801	+
Boujou	4:09 min	0,828	+

#### 4.6. Test 6

Lokacija: Livada

Vrsta snimanja: u svim smjerovima bez stativa

Tablica 7: Rezultati 6. video isječka

Program	Vrijeme računanja	Error Pixel	Uspješnost praćenja
SynthEyes	1:14 min	1,394	+
PFTrack	2:20 min	0,835	-
Boujou	3:58 min	1,566	-

#### 4.7. Test 7

Lokacija: Cesta

Vrsta snimanja: u svim smjerovima na stativu

Tablica 8: Rezultati 7. video isječka

Program	Vrijeme računanja	Error Pixel	Uspješnost praćenja
SynthEyes	1:09 min	1,087	+
PFTrack	2:20 min	0,936	-
Boujou	3:43 min	0,895	+

#### 4.8. Test 8

Lokacija: Cesta

Vrsta snimanja: naprijed - nazad na stativu

Tablica 9: Rezultati 8. video isječka

Program	Vrijeme računanja	Error Pixel	Uspješnost praćenja
SynthEyes	1:07 min	1,612	+
PFTrack	2:36 min	0,978	+
Boujou	3:48 min	1,114	+

## 4.9. Test 9

Lokacija: Cesta

Vrsta snimanja: u svim smjerovima bez stativa

Tablica 10: Rezultati 9. video isječka

<b>Program</b>	<b>Vrijeme računanja</b>	<b>Error Pixel</b>	<b>Uspješnost praćenja</b>
SynthEyes	1:08 min	3,672	-
PFTrack	2:20 min	0,956	+
Boujou	4:01 min	1,444	+

## 5. RASPRAVA

Testovi su se vršili na snimkama koje su snimljene na području grada Ludbrega i to na tri različite lokacije. Na svakoj lokaciji su se snimila tri kadra sa različitim tehnikama snimanja. Prvi kadar je snimljen na stativu i to tako da je kamera snimala u svim smjerovima, drugi kadar je također snimljen na stativu, ali smjer kamere je bio naprijed – nazad dok je treći kadar snimljen bez stativa, ručno. Nakon snimljenih kadrova iste je bilo potrebno ubaciti u programe za praćenje kamere u kojima su vršeni testovi. U testovima se mjerilo koliko vremena je potrebno određenom programu da obavi zadatak praćenja kamere, kolika je prosječna vrijednost greške te da li se nakon ubacivanja modela on drži na snimci ili lebdi po zraku. Nakon izvršenih mjerenja došlo je do sljedećih zaključaka:

Kod snimke koja je snimljena na stativu u svim smjerovima kod nekih programa došlo je do relativno malog „*error pixels*“ što znači da je većina testova bila ispod vrijednosti 1. Kod nekih programa model je ipak lebдио po zraku tj. nije bio statičan na samoj snimci. Razlog tome je što program izračunava prosječnu vrijednost točaka za praćenje te one točke koje su dosta odskakale dosta utječu na ishod samog testa. Na prvoj snimci (tablica 1) nijedan program nije dobro izračunao praćenje kamere iako je greška kod nekih testova bila vrlo mala. Na drugoj snimci samo je *SynthEyes* dobro izračunao praćenje kamere te je model bio statičan dok je na trećoj snimci *PFTrack* jedini zakazao u praćenju kamere.

Na snimkama koje su snimljene na stativu u smjeru naprijed – nazad uspješnost praćenja na prvoj snimci je bila loša te je *SynthEyes* program imao najveću grešku dok su *Boujou* i *PFTrack* imali grešku do vrijednosti 1 međutim zbog točaka praćenja koje su iznad te vrijednosti program je krivo proračunao poziciju 3D kamere. Na preostale dvije snimke uspješnost praćenja svih programa je bila odlična te su svi 3D elementi izgledali statično na snimci. Iako je kod tih proračuna kod nekih programa greška bila iznad dopuštene, programi su dobro proračunali poziciju i kretanje 3D kamere.

Na trećoj snimci koja je snimljena bez stativa, ručno, uspješnost praćenja kamere kod nekih programa je bila vrlo dobra dok je kod nekih bila vrlo loša. Kod *SynthEyes* programa nastale su velike greške praćenja kamere i to preko 3 slikovna elementa, ali je svejedno model bio statičan. U ovom testu *Boujou* se također pokazao vrlo dobrim uz relativno malu prosječnu vrijednost greške dok je *PFTrack* kao i u ostalim testovima imao veoma loše rezultate.

Brzina testiranih programa je vrlo različita, *SynthEyes* je mnogo brži od ostalih programa, a to se može objasniti tako što on po svojim automatskim postavkama ima zadani mali broj točaka za praćenje. Ukoliko bi se zadali drugačiji parametri prije samog procesa praćenja taj postupak bi zasigurno bio mnogo sporiji. *PFTrack* je također veoma brz i ima jako malu grešku, modeli nisu statični i stvaranje virtualne 3D kamere nije točno. Uz dodatne postavke taj proces bi se naravno mogao poboljšati. *Boujou* ima malu grešku kod svih izračuna, ali je i najsporiji u svim testovima zbog velikog broja točaka za praćenje.



## 6. ZAKLJUČAK

Nakon provedenih testova može se reći da greška koja nastaje prilikom izračuna ne utječe direktno na to da li će neki 3D element biti statičan ili ne, jer su testovi pokazali da je i sa velikom greškom moguće postići dobru 3D virtualnu kameru. To se može objasniti i time da ovisi na kojem dijelu snimke se nalazi greška iznad vrijednosti 1 i da li je na tom dijelu 3D element. Ukoliko se točka praćenja koja ima grešku veću od 1 nalazi u kadru u nekom kutu, ona neće utjecati na 3D model koji se nalazi na sredini kadra, te će praćenje kamere zapravo biti dobro. Također, prosječna vrijednost greške se može smanjiti tako da se obrišu točke praćenja koje odstupaju od 1 te se na njihovo mjesto mogu ručno postaviti nove točke.

Što se tiče brzine, programi se međusobno veoma razlikuju dok se *SynthEyes* posebno istaknuo brzinom kalkulacija dok su *Boujou* i *PFTrack* bili znatno sporiji. Naravno uz dodatne postavke tj. zadavanje određenog broja točaka za praćenje koje program mora izračunati, razlika u brzini se može povećati ili smanjiti.

Kada bi odredili koji je od testiranih programa najbolji, moglo bi se reći da svaki od njih ima svoju prednost međutim što se tiče brzine *SynthEyes* je daleko brži od ostalih programa dok *Boujou* ima najviše točnih praćenja kamere, a *PFTrack* se nije ni u jednom testu pokazao boljim od drugih, međutim to ne znači da se željeni rezultati ne mogu postići i njime.

Važno je napomenuti da su svi testovi rađeni sa automatskim postavkama kako bi se pokazalo koliko je potrebno određenom programu da uz nikakve dodatne postavke napravi najbolji, najbrži i najtočniji izračun praćenja kamere.

## 7. LITERATURA

1. \*\*\* <http://goodlearning.com/basic-creative-skills/graphics/2d-and-3d-graphics-the-difference/>, srpanj 2014.
2. \*\*\* [http://courses.washington.edu/eatreun/html/history/h\\_nw.html](http://courses.washington.edu/eatreun/html/history/h_nw.html), srpanj 2014.
3. \*\*\* <http://3d.about.com/od/3d-101-The-Basics/a/Anatomy-Of-A-3d-Model.htm>, kolovoz 2014.
4. \*\*\* [http://en.wikibooks.org/wiki/Blender\\_3D:\\_Noob\\_to\\_Pro/UV\\_Map\\_Basics](http://en.wikibooks.org/wiki/Blender_3D:_Noob_to_Pro/UV_Map_Basics), srpanj 2014.
5. \*\*\* [http://hrcak.srce.hr/index.php?show=clanak&id\\_clanak\\_jezik=149150](http://hrcak.srce.hr/index.php?show=clanak&id_clanak_jezik=149150) [ribar]
6. \*\*\* <http://www.thepixelart.com/breakdown-best-matchmoving-and-tracking-applications/>, kolovoz 2014.
7. \*\*\* <http://syntheyes.software.informer.com/>, srpanj 2014.
8. \*\*\* <http://www.ssontech.com/synsumm.html>, srpanj 2014.
9. \*\*\* <http://www.thepixelfarm.co.uk/education.php?learningZone>, kolovoz 2014.
10. \*\*\* [http://muscleeyes.com/docs/Sasha\\_Mirpour\\_3D\\_Camera\\_Tracking.pdf](http://muscleeyes.com/docs/Sasha_Mirpour_3D_Camera_Tracking.pdf), srpanj 2014.
11. \*\*\* <http://www.metrics.co.uk/boujou/product-range/boujou-5/>, srpanj 2014.
12. \*\*\* <http://www.tiskarstvo.net/printing%26design2013/dl/Tiskarstvo%202013%20zbornik%20radova%20web.pdf>, srpanj 2014.
13. \*\*\* [http://www.happy-digital.com/hdcaustics\\_docs/hd\\_caustics.html](http://www.happy-digital.com/hdcaustics_docs/hd_caustics.html), kolovoz 2014.
14. \*\*\* <http://web.cs.wpi.edu/~matt/courses/cs563/talks/radiosity.html>, kolovoz 2014.
15. \*\*\* <http://lightwave65.blogspot.com/2007/11/using-numeric-panel.html>, kolovoz 2014.
16. \*\*\* [http://www.ten24.info/?page\\_id=445](http://www.ten24.info/?page_id=445), srpanj 2014.
17. \*\*\* <http://hrcak.srce.hr/file/149149>, srpanj 2014.

## 7.1. Popis slika

Slika 1: Izgled UV mape

Slika 2: Sloj raytracing-a

Slika 3: Sloj globalne iluminacije

Slika 4: Izgled jednog kadra u procesu praćenja kamere

Slika 5: Izgled učitane video snimke u SynthEyes program za praćenje

Slika 6: Izgled učitane video snimke u PFTrack program za praćenje

Slika 7: Izgled učitane video snimke u Boujou program za praćenje

Slika 8: Početni izgled Lightwave-a

Slika 9: Alati za manipuliranje

Slika 10: Manipulatori -pomicanje, skaliranje i rotacija

Slika 11: Alat za selektiranje

Slika 12: Alatna traka

Slika 13: Izgled Lightwave Layout-a

Slika 14: Alatna traka SynthEyes programa

Slika 15: Vremenska traka

Slika 16: Izgled PFTrack programa

Slika 17: Vremenska traka PFTrack programa

Slika 18: Izbornik za učitavanje datoteka u program

Slika 19: Izgled ubačene snimke u scenu sa dodanim čvorovima

Slika 20: Početna traka Boujou-a

Slika 21: Alatna traka Boujou programa

Slika 22: Popis zadaća koje nudi Boujou

Slika 23: Vremenska traka Boujou-a

Slika 24: Ubacivanje video sekvence u program

Slika 25: Dodavanje čvorova u video sekvencu

Slika 26: Glavni prozor za praćenje kamere

Slika 27: Izgled snimke nakon izvršenog procesa "praćenja kamere"

Slika 28: Izgled praćenja kamere u Boujou programu

Slika 29: Izgled 3D kamere

Slika 30: Izgled 3D modela ubačenog u scenu

Slika 31: Prikaz kretanja 3D kamere

Slika 32: Umetanje sekvence u 3D snimku

Slika 33: Podešavanje postavki u "Image editor-u"

Slika 34: Prikaz video sekvence i mreže u 3D programu

Slika 35: Umetnut 3D model na snimci

## 7.2. Popis tablica

Tablica 1: Cijene programa za praćenje

Tablica 2: Rezultati 1. video isječka

Tablica 3: Rezultati 2. video isječka

Tablica 4: Rezultati 3. video isječka

Tablica 5: Rezultati 4. video isječka

Tablica 6: Rezultati 5. video isječka

Tablica 7: Rezultati 6. video isječka

Tablica 8: Rezultati 7. video isječka

Tablica 9: Rezultati 8. video isječka

Tablica 10: Rezultati 9. video isječka