

Razvoj aplikacije za kalibraciju stereovizijskog sustava primjenom modela konačnih elemenata

Šaban, Fran

Undergraduate thesis / Završni rad

2024

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Mechanical Engineering and Naval Architecture / Sveučilište u Zagrebu, Fakultet strojarstva i brodogradnje**

Permanent link / Trajna poveznica: <https://urn.nsk.hr/urn:nbn:hr:235:120430>

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

Download date / Datum preuzimanja: **2025-02-28**

Repository / Repozitorij:

[Repository of Faculty of Mechanical Engineering and Naval Architecture University of Zagreb](#)



FAKULTET STROJARSTVA I BRODOGRADNJE

SVEUČILIŠTE U ZAGREBU

ZAVRŠNI RAD

Fran Šaban

FAKULTET STROJARSTVA I BRODOGRADNJE

SVEUČILIŠTE U ZAGREBU

ZAVRŠNI RAD

Mentor:

Izv. prof. dr. sc. Zvonimir Tomićević

Student:

Fran Šaban

Zagreb, 2024.

Izjavljujem da sam ovaj rad izradio samostalno koristeći znanja stečena tijekom studija i navedenu literaturu.

Posebno se zahvaljujem svojem mentoru izv. prof. dr. sc. Zvonimiru Tomičeviću i Andriji Zaplatiću, mag. ing. mech. radi izrazite podrške i pomoći tijekom pisanja ovog rada.

Ovaj rad izrađen je u sklopu projekta „Promatranje, identifikacija i kvantifikacija mehanizama mehaničkog oštećenja na kompozitima ojačanim staklenim vlaknima primjenom različitih metoda nerazornog ispitivanja“ (UIP-2019-04-5460, Z. Tomičević).

Fran Šaban



| | |
|--|--------|
| Sveučilište u Zagrebu Fakultet strojarstva i brodogradnje | |
| Datum | Prilog |
| Klasa: 602 – 04 / 24 – 06 / 01 | |
| Ur.broj: 15 – 24 – | |

ZAVRŠNI ZADATAK

Student: **Fran Šaban** JMBAG: **0035226902**

Naslov rada na hrvatskom jeziku: **Razvoj aplikacije za kalibraciju stereovizijskog sustava primjenom modela konačnih elemenata**

Naslov rada na engleskom jeziku: **Development of the stereovision calibration application using the finite element model**

Opis zadatka:

Upotreba beskontaktnih optičkih mjernih metoda u eksperimentalnoj mehanici ubrzano raste uslijed njihove dostupnosti i jednostavnosti korištenja. Posebno se ističe metoda korelacije digitalne slike kao najčešće korištena beskontaktna optička metoda mjerenja cijelog polja pomaka i/ili deformacija na površini promatranih objekata. Ova metoda postaje neizostavna u situacijama gdje se promatrani objekt odlikuje složenom geometrijom. U slučaju kada promatrani objekt nije ravninske geometrije te kada se očekuju izvan ravninski pomaci, za postizanje točnih rezultata često se koristi stereovizijski sustav s dvije ili više kamere, što omogućuje simulaciju ljudskog vida i dubinsku percepciju. Međutim, točnost mjerenja u slučaju stereovizijskog sustava ovisi o kalibraciji koja definira odnos između trodimenzionalnog prostora i dvodimenzionalnih slika s kamere.

U ovom radu potrebno je razviti aplikaciju za kalibraciju stereovizijskog sustava koji se sastoji od dviju ili više digitalnih kamera primjenom modela konačnih elemenata. Tijekom postupka kalibracije određuje se projekcijska matrica koja omogućuje projekciju točaka 3D objekta na 2D ravninu slike sa kamere. Projekcijska matrica se sastoji od intrinzičnih i ekstrinzičnih parametara kamere koji se kalibracijom određuju za svaku kameru. Primjena mreže konačnih elemenata omogućuje izravno korištenje samog promatranog objekta za kalibraciju stereo optičkog sustava budući da 3D mreža odgovara stvarnoj geometriji objekta. Proces kalibracije je završen kada projicirana mreža konačnih elemenata precizno odgovara stvarnom objektu zabilježenom na slici sa kamere.

Tijekom rada potrebno je:

1. Upoznati se s principom rada *pinhole* modela kamere te postojećim principima kalibracije.
2. Razviti algoritam za kalibraciju stereovizijskog sustava primjenom metode konačnih elemenata u C++ programskom jeziku.
3. Izraditi aplikaciju sa sučeljem za kalibraciju stereovizijskog para slika.
4. Provesti validaciju algoritma kalibracije na umjetno izrađenim slikama.

U radu navesti korištenu literaturu i eventualno dobivenu pomoć.

Zadatak zadan:

Datum predaje rada:

Predviđeni datumi obrane:

30. 11. 2023.

1. rok: 22. i 23. 2. 2024.
2. rok (izvanredni): 11. 7. 2024.
3. rok: 19. i 20. 9. 2024.

1. rok: 26. 2. – 1. 3. 2024.
2. rok (izvanredni): 15. 7. 2024.
3. rok: 23. 9. – 27. 9. 2024.

Zadatak zadao:

Predsjednik Povjerenstva:

Izv. prof. dr. sc. Zvonimir Tomičević

Prof. dr. sc. Vladimir Soldo

SADRŽAJ

| | |
|---|-----|
| SADRŽAJ..... | I |
| POPIS SLIKA..... | III |
| POPIS OZNAKA..... | IV |
| SAŽETAK | V |
| SUMMARY | VI |
| 1. UVOD..... | 1 |
| 2. STEREOVIZIJSKI SUSTAV..... | 3 |
| 2.1. Osnove stereovizijskih sustava | 4 |
| 2.2. Pinhole model kamere..... | 5 |
| 2.3. Epipolarna geometrija..... | 9 |
| 3. PREGLED KALIBRACIJSKIH TEHNIKA STEREOVIZIJSKIH SUSTAVA | 11 |
| 3.1. Kalibracija na temelju 2D kalibracijskih objekata..... | 11 |
| 3.2. Kalibracija na temelju 3D kalibracijskih objekata..... | 13 |
| 3.3. Samokalibracija..... | 14 |
| 3.4. Kalibracija na temelju iščezavajućih točaka | 14 |
| 4. GLOBALNI PRISTUP KALIBRACIJI STEREOVIZIJSKIH SUSTAVA..... | 16 |
| 4.1. Predkalibracija | 17 |
| 4.1.1. DLT algoritam | 17 |
| 4.1.2. SVD algoritam | 21 |
| 4.2. Kalibracija..... | 23 |
| 5. APLIKACIJA CROCAB..... | 26 |
| 5.1. Ulazni podatci | 26 |
| 5.2. Predkalibracija | 27 |
| 5.2.1. Prikazivanje mreže konačnih elemenata | 28 |
| 5.2.2. Odabir točaka na mreži konačnih elemenata | 29 |
| 5.2.3. Odabir točaka na slikama..... | 30 |
| 5.3. Kalibracija..... | 32 |
| 5.3.1. Postavke prije kalibracijske petlje | 33 |
| 5.3.2. Kalibracijska petlja | 34 |
| 5.4. Dodatne funkcionalnosti..... | 38 |

| | | |
|--------|--|----|
| 5.4.1. | Učitavanje koordinata točaka na slikama | 38 |
| 5.4.2. | Odabir lokacije za ispis podataka o predkalibraciji i brisanje svih podataka | 38 |
| 6. | VALIDACIJA CROKAB APLIKACIJE NA UMJETNO GENERIRANIM SLIKAMA.... | 40 |
| 7. | ZAKLJUČAK..... | 43 |

POPIS SLIKA

| | |
|--|----|
| Slika 1. Binokularni vid [2] | 4 |
| Slika 2. Projekcija 3D točke M na 2D sliku kroz pinhole model [4]..... | 5 |
| Slika 3. Epipolarna geometrija stereovizijskog sustava [6] | 9 |
| Slika 4. Kalibracijski uzorak sa šahovnicom | 12 |
| Slika 5. Kalibracijski uzorak s mrežom točaka..... | 12 |
| Slika 6. Kalibracijski objekt u stilu otvorene knjige [9] | 13 |
| Slika 7. Iščekavajuća točka [10]..... | 15 |
| Slika 8. Mreža B-spline krivulja (zeleno) s njihovim pripadajućim kontrolnim točkama (crveno)[13] | 16 |
| Slika 9. Umjetno generirane slike s lijeve kamere (a) i desne kamere (b)..... | 26 |
| Slika 10. Izgled programskog sučelja | 27 |
| Slika 11. Mreža konačnih elemenata kalibracijskog objekta | 29 |
| Slika 12. Točke na mreži s redoslijedom odabiranja..... | 30 |
| Slika 13. Odabrane točke na slici..... | 31 |
| Slika 14. Mesh plottan na sliku s lijeve kamere..... | 32 |
| Slika 15. Ispis konzole tokom iteracijskog postupka..... | 35 |
| Slika 16. Reziduali u prvoj iteraciji | 36 |
| Slika 17. Reziduali u osmoj iteraciji | 36 |
| Slika 18. Sučelje za prikaz slika | 37 |
| Slika 19. Prozor za odabir točaka | 38 |
| Slika 20. Projicirana mreža konačnih elemenata na sliku sa lijeve (a) i desne (b) kamere | 40 |
| Slika 21. Prikaz reziduala prije (a) i nakon (b) kalibracije | 41 |
| Slika 22. Funkcija gustoće vjerojatnosti reziduala | 42 |

POPIS OZNAKA

| Oznaka | Jedinica | Opis oznake |
|-------------------------|----------|-------------------------------------|
| B | [-] | globalni vektor drugih članova |
| b | [-] | vektor drugih članova |
| C | [-] | globalna Hessian matrica |
| $c_{x,y}$ | [-] | koordinate optičkog centra kamera |
| E | [-] | esencijalna matrica |
| F | [-] | fundamentalna matrica |
| f | [-] | fokalna udaljenost kamera |
| $f^c(\{\mathbf{x}^c\})$ | [-] | slika s kamera |
| H | [-] | Hessian matrica |
| k | [-] | omjer homotetije |
| P | [-] | projekcijska matrica |
| R | [-] | matrica rotacija |
| s | [-] | parametar skaliranja veličine slike |
| T | [-] | matrica transformacija |
| $\delta\mathbf{P}$ | [-] | vektor promjene parametara kamera |
| P | [-] | kalibracijski reziduali |
| Π | [-] | globalna projekcijska matrica |

SAŽETAK

Uporaba optičkih metoda mjerenja pomaka postala je ključna pri mjerenju površinskih pomaka predmeta sa složenom geometrijom. Metoda koja je došla do velikog izražaja posljednjih godina je korelacija digitalne slike. Korištenje te metode uz primjenu stereovizijskih sustava omogućilo je precizno mjerenje pomaka. Kako bi stereovizijski sustav davao točne rezultate mjerenja, potrebno je odrediti parametre kamera koje obavljaju mjerenje. To se postiže kalibracijom stereovizijskog sustava. U ovom je radu opisan razvoj aplikacije za kalibraciju *CroCab* u kojoj se kalibracija stereovizijskog sustava postiže preko globalnog pristupa kalibraciji sa mrežom konačnih elemenata. Kalibracija je provedena nad umjetno generiranim slikama, te su prikazani rezultati i greška kalibracije.

Ključne riječi: optičke metode mjerenja pomaka, korelacija digitalne slike, stereovizijski sustavi, kalibracija stereovizijskih sustava, aplikacija za kalibraciju stereovizijskih sustava, globalni pristup kalibraciji

SUMMARY

The use of optical methods in strain measurement has become key when measuring strain of an object with complex geometry. A method that has come to great prominence in recent years is digital image correlation. The use of this method alongside the application of stereovision systems enable accurate and simple displacement measurement. In order for the stereovision system to provide accurate measurement results, it is necessary to determine the parameters of the cameras that perform the measurement. This is achieved by calibrating the stereovision system. This paper describes the development of the *CroCab* calibration application, in which the calibration of the stereo system is achieved through a global calibration approach with a finite element network. Calibration is performed on artificially generated images, and the results with the corresponding error are presented.

Keywords: optical strain measurement , digital image correlation, stereovision systems, calibration of stereovision systems, application for calibrating stereovision systems, global approach to calibration

1. UVOD

U području inženjeringa i znanstvenih istraživanja, precizno mjerenje pomaka ključno je za razumijevanje ponašanja materijala, integriteta konstrukcija i analizu performansi. Tradicionalne metode mjerenja napreznja često se oslanjaju na invazivne tehnike ili glomaznu instrumentaciju, što može ograničiti učinkovitost i točnost mjerenja pomaka na objektima. Optička metoda mjerenja pomaka koja je došla do velikog izražaja posljednjih godina je korelacija digitalne slike. U slučajevima kada je promatrani objekt složene (neravninske) geometrije i kada se očekuju izvan ravninski pomaci, slike objekta se često zabilježavaju primjenom stereovizijskih sustava sa dvije ili više kamere. Stereovizijski sustavi rade na principu sličnom ljudskom vidu gdje se uporabom dvije kamere omogućava dubinska percepcija. No, kako bi se dubinske informacije mogle iščitati iz slika, potrebno je definirati odnos između trodimenzionalnog prostora i dvodimenzionalnih slika sa kamera. Točno definiranje tog odnosa postiže se uporabom kalibracijskih tehnika.

Kalibracija sustava stereovizije igra ključnu ulogu u osiguravanju točnosti i pouzdanosti mjerenja pomaka. Usklađivanje značajki sa slika i značajki na kalibracijskom objektu omogućuje točnu transformaciju dvodimenzionalnih koordinata slika u trodimenzionalne koordinate okoline, osiguravajući precizno očitavanje površinskih pomaka na uzorku pod napreznjem. Kalibracija stereovizijskih sustava se inače postiže uporabom posebnih dvodimenzionalnih ili trodimenzionalnih kalibracijskih objekata. Ovaj rad objašnjava pristup kalibraciji gdje objekt na kojem se mjere pomaci može ujedno biti i kalibracijski objekt. To se postiže nanošenjem stohastičkog rastera na površinu predmeta koji služi kao kalibracijski uzorak, te kalibracijom na temelju očitavanja i uspoređivanja sivih vrijednosti piksela dvaju slika kalibracijskog uzorka.

U ovom radu potrebno je bilo razviti aplikaciju namijenjenu kalibraciji stereovizijskog sustava dvije ili više digitalne kamere primjenom modela konačnih elemenata. Cilj kalibracije je određivanje projekcijske matrice koja omogućuje preslikavanje 3D točaka poznate mreže konačnih elemenata na 2D ravninu slike sa kamere.

Rad se sastoji od sedam poglavlja. U drugom poglavlju definiran je stereovizijski sustav te je opisan princip koncepta stereovizije temeljen na *pinhole* modelu kamere. Nadalje, u sljedećem poglavlju prikazani su najčešći pristupi kalibraciji stereovizijskih sustava pomoću različitih kalibracijskih objekata. Globalni pristup kalibraciji stereovizijskih sustava, temeljen na metodi

konačnih elemenata, opisan je u četvrtom poglavlju. Zatim je opisana *CroCab* aplikacija za provedbu kalibracije razvijena u sklopu ovog rada. Na kraju provedena je kalibracija stereovizijskog sustava nad umjetno generiranim slikama.

Cilj ovog rada je bio prikazati strukturu i proces razvoja aplikacije za kalibraciju stereovizijskog sustava, te provjeriti točnost procesa kalibracije kamera na temelju dobivenih rezultata.

2. STEREOVIZIJSKI SUSTAV

Računalni stereovizijski sustav je sustav izvlačenja trodimenzionalnih informacija iz dvodimenzionalnih digitalnih slika. Koncept stereovizije ima korijene u ljudskoj percepciji. Razvoj stereovizijske tehnologije počinje sredinom 19. stoljeća sa Charlesom Wheatstoneom i njegovim izumom stereoskopa, uređaja koji demonstrira način na koji mozak koristi slike dobivene iz lijevog i desnog oka kako bi percipirao dubinu. Daljnji razvoj računalne stereovizije je obilježen slijedećim događajima:

- 1960. – 1970. počinju rana istraživanja na području stereovizije sa znanstvenicima poput Marvinu Minskya i Seymoura Paperta koji istražuju mogućnosti računalne percepcije,
- 1979. David Marr i Tomaso Poggio objavljuju rad „*A Computational Theory of Human Stereo Vision*“ koji postavlja teorijski temelj algoritama za usklađivanje stereo sustava (engl. *stereo matching*). Usklađivanje stereo sustava označuje proces pronalaska piksela na slici koji odgovaraju danoj 3D točki na predmetu ili u okolini,
- 1980-ih nastaju različite tehnike usklađivanja stereo sustava, uključujući metode značajka (engl. *feature-based methods*) ili metode površine (engl. *area-based methods*),
- 1990-ih istraživači razvijaju globalne optimizacijske metode poput dinamičkog programiranja, povećavajući preciznost ,
- 2000-ih izrada sve jačih računala omogućuje korištenje stereovizije u stvarnom vremenu. SGM (eng. *Semi-Global Matching*) algoritam, predstavljen od strane Heika Hirschmüllera 2005. godine postaje široko korišten zbog njegove efikasnosti i preciznosti,
- 2015. John-Eric Dufour, Benoît Beaubier, François Hild i Stéphane Roux objavljuju rad „*CAD-based displacement measurement with stereo-DIC*“ u kojemu je detaljno opisan globalni pristup korelacije digitalne slike.

Kako se tehnologija razvijala i napredovala tako su se razvijale i uporabne mogućnosti stereovizije pa se danas ona koristi u više svrha:

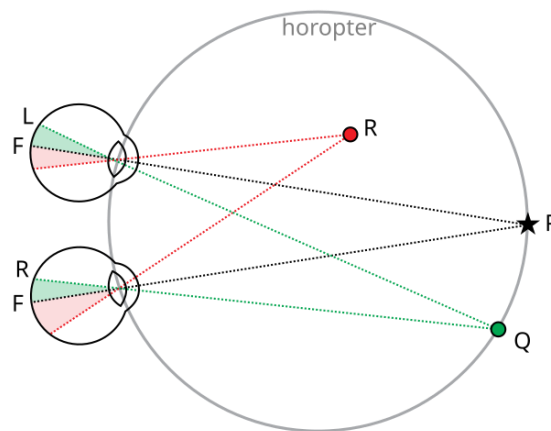
- u autonomnim vozilima za percepciju dubine i detekciju prepreka,
- robotici za navigacijske i manipulacijske zadatke,
- 3D rekonstrukciji u računalnoj grafici i virtualnoj realnosti,
- industrijskoj automatizaciji za kontrolu kvalitete i mjerenja,
- satelitskim slikama za mapiranje terena.

- karakterizaciji ponašanja materijala pri raznim uvjetima opterećenja pomoću metoda mjerenja pomaka

2.1. Osnove stereovizijskih sustava

Računalna stereovizija je proces dobivanja 3D informacija iz digitalnih slika. Usporedbom slika zabilježenih iz različitih kuteva, mogu se dobiti dubinske informacije promatrajući relativne pozicije predmeta u njima. Princip rada stereovizije temelji se na ljudskom binokularnom vidu, u kojem se proces procjene relativne udaljenosti predmeta naziva stereopsija. Ključan pojam koji omogućava taj proces kod ljudi i kod stereovizijskih sustava je disparitet [1].

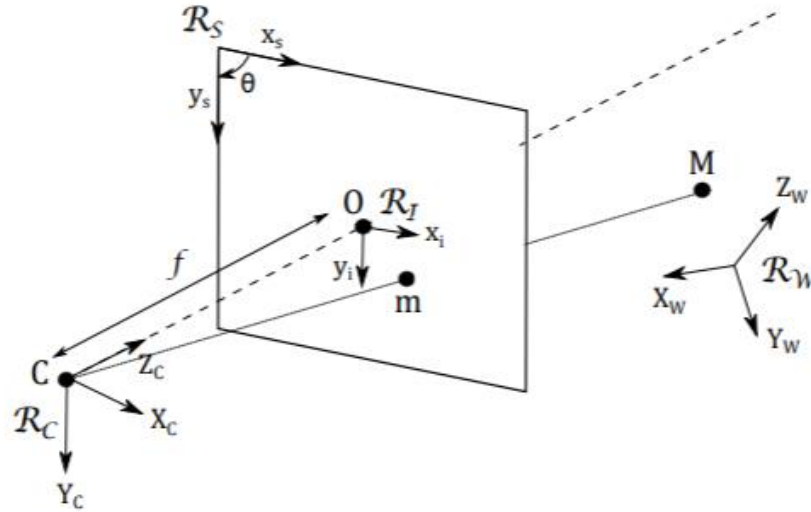
Disparitet je razlika između pozicije predmeta na slici viđenoj iz jednog i drugog oka zbog njihovog horizontalnog razmaka. Iako disparitet kod ljudskog vida ima generalno vertikalnu i horizontalnu komponentu, njegova horizontalna komponenta nosi informacije o dubini predmeta. Vertikalna komponenta nastaje samo kada je predmet bliži jednom oku nego drugom pa prema tome ona nosi informacije o relativnoj poziciji očiju.



Slika 1. Binokularni vid [2]

2.2. Pinhole model kamere

Pinhole model kamere opisuje idealizirani oblik kamere koji koristi jednu točku kao otvor, nazvanu *pinhole* kroz koju prolaze svjetlosne zrake iz okoline [4]. Te svjetlosne zrake projiciraju se na slikovnu ravninu kamere, stvarajući 2D reprezentaciju 3D scene.



Slika 2. Projekcija 3D točke M na 2D sliku kroz pinhole model [4]

pozicija 3D točke M definirana u globalnom koordinatnom sustavu okoline R_W sa koordinatama (X_W, Y_W, Z_W) , je opisana u lokalnom koordinatnom sustavu kamere s točkama (X_C, Y_C, Z_C) . Ova transformacija između koordinata okoline $\{\mathbf{X}_W\}$ i lokalnih koordinata $\{\mathbf{X}_C\}$ ovisi samo o tri translacije i tri rotacije. Ta transformacija se može opisati s (4×4) matricom transformacije $[\mathbf{T}]$

$$\{\mathbf{X}_C\} = [\mathbf{T}]\{\mathbf{X}_W\}, \quad (2.1)$$

to jest

$$\begin{bmatrix} X_C \\ Y_C \\ Z_C \\ 1 \end{bmatrix} = [\mathbf{T}] \begin{bmatrix} X_W \\ Y_W \\ Z_W \\ 1 \end{bmatrix}. \quad (2.2)$$

Matrica transformacije $[T]$ se može rastaviti na (3×3) rotacijsku matricu $[R]$ definiranu s tri Eulerova kuta rotacije i na vektor translacije $\{t\}$ definiran s tri translacije u smjeru svake osi

$$[T] = \begin{bmatrix} [R] & \{t\} \\ \{\mathbf{0}_{1 \times 3}\} & 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2.3)$$

Tri translacije i tri kuta rotacije predstavljaju šest ekstrinzičnih parametara koji definiraju transformaciju globalnih koordinata u lokalne koordinate. Stoga matrica $[T]$ predstavlja ekstrinzičnu matricu.

Druga transformacija predstavlja projekciju 3D točke M iz lokalnog koordinatnog sustava kamere R_c na 2D koordinatni sustav slike R_l . Projicirana 2D točka m je definirana homogenim koordinatama $(x_i, y_i, 1)$ i dobiva se pomoću projekcijske matrice $[P]$ veličine (3×4)

$$s \cdot \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = [P] \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix}. \quad (2.4)$$

Jedini parametar koji definira projekcijsku matricu $[P]$ je fokalna udaljenost f . Parametar skaliranja veličine slike je označen sa s . Posljednjom transformacijom dobivaju se 2D koordinate projicirane točke m u koordinatnom sustavu senzora R_s , označene s homogenim koordinatama $(x_s, y_s, 1)$, pomoću matrice $[A]$ veličine (3×3)

$$\begin{bmatrix} x_s \\ y_s \\ 1 \end{bmatrix} = [\mathbf{A}] \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix}, \quad (2.5)$$

Matrica $[\mathbf{A}]$ definirana je kao

$$[\mathbf{A}] = \begin{bmatrix} S_x & -S_x \cot \theta & -S_x (\hat{c}_x - \hat{c}_y \cot \theta) \\ 0 & \frac{S_y}{\sin \theta} & -\frac{S_y c_y}{\sin \theta} \\ 0 & 0 & 1 \end{bmatrix}, \quad (2.6)$$

gdje je θ kut između dvije osi x_i i y_i , S_x i S_y su faktori skaliranja u smjeru x_i i y_i , a parametri \hat{c}_x i \hat{c}_y odgovaraju poziciji optičkog centra u koordinatnom sustavu senzora. Uvođenjem pretpostavke da kut θ iznosi $\frac{\pi}{2}$, to jest da su osi x_i i y_i okomite jedna na drugu, jednadžba (2.6) se može pojednostaviti. Stoga, kombinacijom matrice $[\mathbf{P}]$ i $[\mathbf{A}]$ dobiva se intrinzična matrica $[\mathbf{K}]$ veličine (3×4)

$$[\mathbf{K}] = [\mathbf{A}][\mathbf{P}] = \begin{bmatrix} fS_x & 0 & -S_x \hat{c}_x \\ 0 & fS_y & -S_y \hat{c}_y \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}. \quad (2.7)$$

koja je definirana sa četiri parametra: horizontalne i vertikalne fokalne udaljenosti f_x i f_y , te horizontalna i vertikalna pozicija projekcije u odnosu na optički centar senzora c_x i c_y . Tri matrice transformacije $[\mathbf{A}]$, $[\mathbf{P}]$ i $[\mathbf{T}]$, mogu se spojiti u sveukupnu globalnu projekcijsku matricu $[\mathbf{H}]$

$$s \cdot \{\mathbf{x}_s\} = [\mathbf{A}][\mathbf{P}][\mathbf{T}]\{\mathbf{X}_w\} = [\mathbf{II}]\{\mathbf{X}_w\}, \quad (2.8)$$

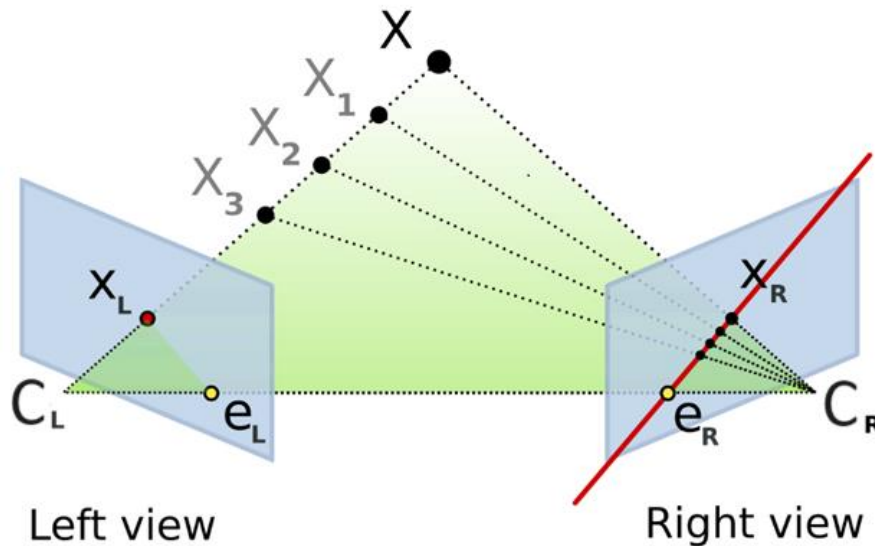
ili, u raspisanom obliku

$$s \cdot \begin{bmatrix} x_s \\ y_s \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} r_{11}f_x + r_{31}c_x & r_{12}f_x + r_{32}c_x & r_{13}f_x + r_{33}c_x & t_x f_x + t_z c_x \\ r_{21}f_y + r_{31}c_y & r_{22}f_y + r_{32}c_y & r_{23}f_y + r_{33}c_y & t_y f_y + t_z c_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix}}_{[\mathbf{II}]} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}. \quad (2.9)$$

Projekcijska matrica je prema tome definirana pomoću deset neovisnih parametara, šest ekstrinzičnih parametara i četiri intrinzična parametra.

2.3. Epipolarna geometrija

Epipolarna geometrija opisuje geometrijski odnos između dviju ili više kamera koje promatraju istu 3D scenu sa različitim pozicija. Navedeni geometrijski odnosi izvedeni su pod pretpostavkom da kamere mogu biti opisane modelom *pinhole* kamera.



Slika 3. Epipolarna geometrija stereovizijskog sustava [6]

Slika 3. prikazuje dva modela *pinhole* kamera koje gledaju u istu točku X . Kod stvarnih kamera, ravnina u kojoj se projicira slika se nalazi iza fokalnih centara slika pa se prema tome dobiva slika koja je simetrično transformirana oko fokalnog centra kamere. U ovom slučaju postavlja se virtualna ravnina ispred fokalnog centra kamere radi jednostavnosti zato što sliku onda nije potrebno dodatno transformirati. Točka X predstavlja točku interesa kod jedne i druge kamere, dok su X_L i X_R projekcije točke X na ravninu slike lijeve i desne kamere. Optički centri obje kamere projiciraju na ravninu slike suprotne kamere točke e_L i e_R koje se ujedno nazivaju i epipolarne točke.

Linija $C_L - X$ se na lijevoj kameri vidi kao točka zato što je ona kolinearna s normalom na ravninu slike lijeve kamere, te prolazi točkom C_L . Suprotno tome, desna kamera vidi tu liniju kao liniju $e_R - X_R$ u svojoj ravnini slike. Linija $e_R - X_R$ se naziva i epipolarna linija. Epipolarna linija je funkcija pozicije točke X u 3D prostoru. Kako se pozicija točke X mijenja, skup epipolarnih linija se generira u obje slike.

Moguće je definirati dva geometrijska ograničenja ukoliko je relativna pozicija kamera poznata. Uz pretpostavku da je projekcijska točka \mathbf{x}_L poznata i da je epipolarna linija $\mathbf{e}_R - \mathbf{x}_R$ poznata te da se točka \mathbf{X} projicira u desnu sliku na točku \mathbf{x}_R koja mora ležati na toj određenoj epipolarnoj liniji. To znači da se za svaku točku promatranu u jednoj slici ista točka mora promatrati u drugoj slici na poznatoj epipolarnoj liniji. To pruža epipolarno ograničenje: projekcija \mathbf{X} na ravnini desne kamere \mathbf{x}_R mora biti sadržana u epipolarnoj liniji $\mathbf{e}_R - \mathbf{x}_R$. Sve točke \mathbf{X} , npr. $\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3$ na liniji $C_L - \mathbf{X}_L$ će zadovoljiti to ograničenje. To znači da je moguće testirati odgovaraju li dvije točke istoj 3D točki.

Ako su točke \mathbf{x}_L i \mathbf{x}_R poznate, njihove projekcijske linije su također poznate. Ako dvije točke slike odgovaraju istoj 3D točki \mathbf{X} , projekcijske linije moraju se sjeći točno u \mathbf{X} . To znači da se \mathbf{X} može izračunati iz koordinata dviju točaka slike, postupkom koji se naziva triangulacija. To je ujedno i drugo epipolarno ograničenje.

Epipolarna ograničenja također se mogu opisati esencijalnom matricom ili fundamentalnom matricom između dvije kamere. Esencijalna matrica $[\mathbf{E}]$ je 3×3 matrica koja sadrži relacije između odgovarajućih točaka na stereo slikama. Ako su \mathbf{y}_L i \mathbf{y}_R normalizirane koordinate lijeve i desne slike, onda se esencijalna matrica $[\mathbf{E}]$ može opisati slijedećom jednačinom

$$(\mathbf{y}_L)^T [\mathbf{E}] \mathbf{y}_R = 0 \quad (2.10)$$

ako \mathbf{y}_L i \mathbf{y}_R odgovaraju istoj 3D točki u sceni. Fundamentalna matrica je prethodnik esencijalnoj zato što za stvaranje fundamentalne matrice nisu potrebne informacije o unutarnjim parametrima kamere. Relacija fundamentalne matrice $[\mathbf{F}]$ i esencijalne matrice $[\mathbf{E}]$ je dana jednačinom

$$[\mathbf{E}] = [\mathbf{K}_L]^T [\mathbf{F}] [\mathbf{K}_R] \quad (2.11)$$

gdje su $[\mathbf{K}_L]$ i $[\mathbf{K}_R]$ unutarnji parametri lijeve i desne kamere.

3. PREGLED KALIBRACIJSKIH TEHNIKA STEREOVIZIJSKIH SUSTAVA

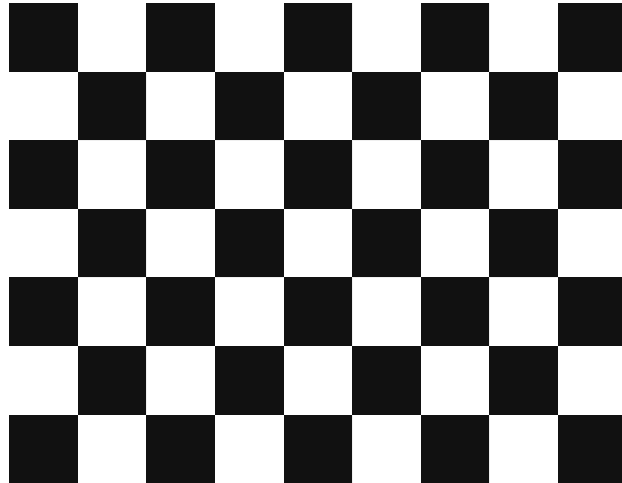
Kalibracija stereovizijskog sustava ključan je proces u računalnom vidu koji uključuje određivanje intrinzičnih i ekstrinzičnih parametara kamera. Postupak kalibracije je bitan za precizno optičko mjerenje pomaka u promatranim objektima i u 3D rekonstruiranim scenama gdje je točna informacija o dubini nužna za preciznu lokalizaciju objekata i razumijevanje scene.

Točna kalibracija temelj je za daljnja mjerenja pomoću različitih metoda, među kojima se ističe korelacija digitalne slike (eng. *Digital Image Correlation* - DIC). U DIC metodi, mjeri se cijelo polje površinskih pomaka uzoraka tijekom ispitivanja. Kalibracijom se određuje odnos između veličine piksela i stvarne fizikalne mjerne jedinice, tj. milimetara.

3.1. Kalibracija na temelju 2D kalibracijskih objekata

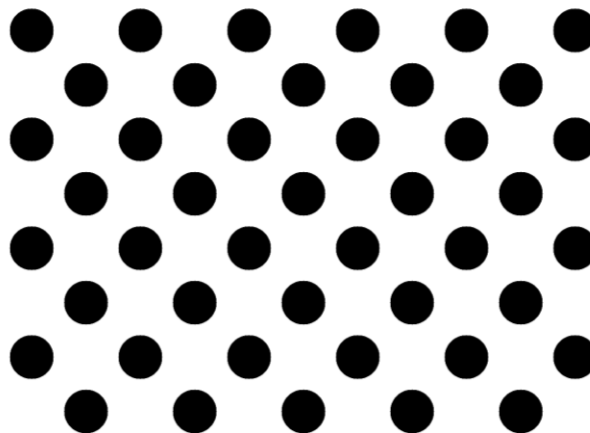
Ovaj pristup kalibraciji temelji se na korištenju pločastih uzoraka. Dvije najčešće korištene vrste uzoraka su mreža izmjenjujućih crnih i bijelih kvadratića (šahovnica) i mreža točaka.

Kalibracijski uzorak sa šahovnicom je najčešće korišten uzorak za kalibraciju u kojem rubovi mreže služe kao kontrolne točke, te se mogu precizno odrediti pomoću algoritama za detekciju rubova. Ovaj kalibracijski uzorak je jednostavan za izradu i prikladan za korištenje u uvjetima dobrog osvjetljenja. Međutim, u uvjetima lošeg osvjetljenja narušava se kontrast između rubova kvadratića te dolazi do njihovog zamućenja što onemogućuje točnu detekciju rubova.



Slika 4. Kalibracijski uzorak sa šahovnicom

Kalibracijski objekt sa mrežom točkaka sastoji se od crnih krugova sa poznatim razmakom među točkama. Središta krugova služe kao kalibracijske točke i mogu se detektirati sa preciznošću na razini ispod piksela. Takvi kalibracijski objekti imaju prednost pri kalibraciji termalnih kamera zbog njihove mogućnosti da smanje šum koji bi mogao dovoditi do nepreciznosti prilikom detekcije kutova [7].



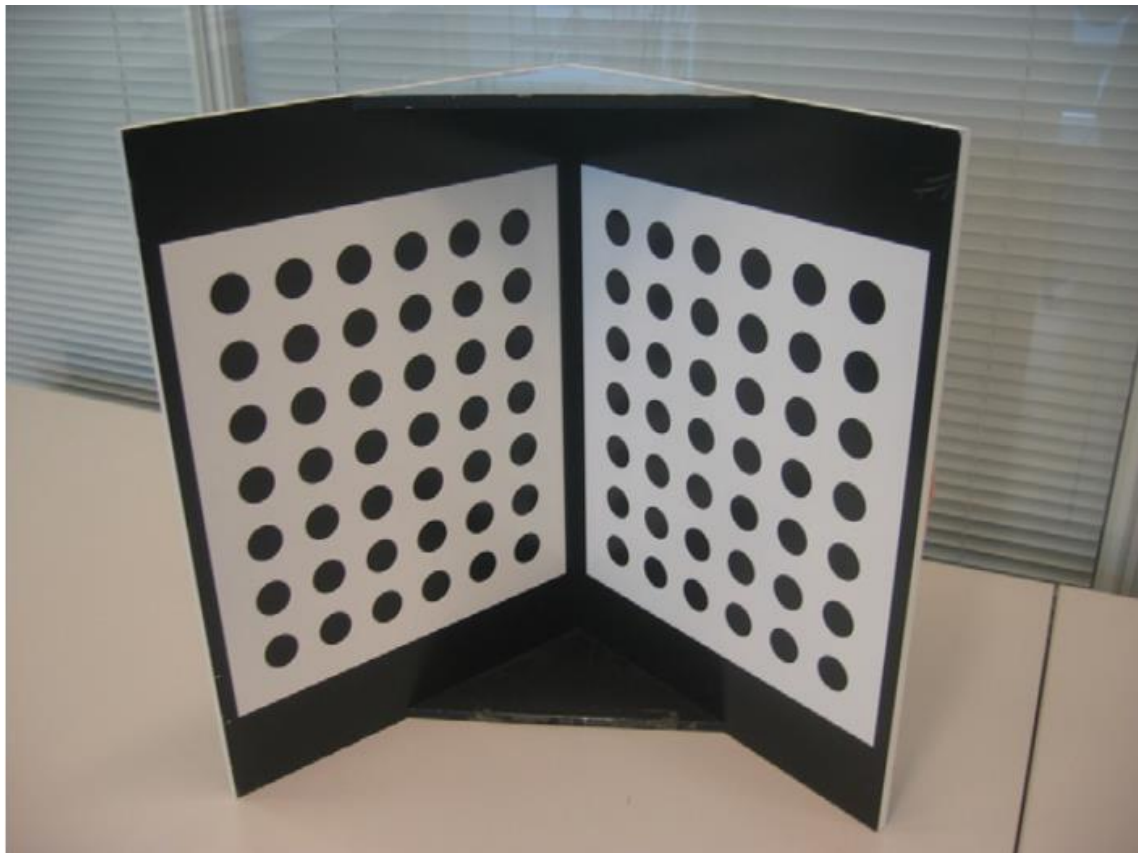
Slika 5. Kalibracijski uzorak s mrežom točkaka

Zhangova metoda [8] je najčešće korištena metoda za provođenje kalibracije pomoću 2D kalibracijskih objekata. Za kalibraciju potrebno je minimalno tri slike koje se mogu dobiti pomicanjem kamere ili uzorka.

U slučaju da su intrinzični parametri kamera poznati, kalibraciju je moguće napraviti i sa dvije slike. Zhangova metoda [8] je poznata po svojoj preciznosti, robusnosti i mogućnosti korekcija slika u slučaju distorzije leće. Integrirana je u raznim primjenama strojnog vida, te je implementirana u raznim programskim bibliotekama poput OpenCv-a.

3.2. Kalibracija na temelju 3D kalibracijskih objekata

Nanošenjem 2D kalibracijskih uzoraka na predmete sa složenijom geometrijom dobivaju se 3D kalibracijski predmeti. Neki od temeljnih oblika na koji se ti uzorci mogu nanjeti su sfere, kocke, kvadri i cilindri. Pozicioniranjem kalibracijskih uzoraka poput šahovnice ili mreže točaka u dvije prostorne ravnine, moguće je dobiti kalibracijski objekt koji slični otvorenoj knjizi (eng. *open-book*), prikazan na Slika 6. 3D kalibracijski objekti pružaju detaljnije dubinske informacije za razliku od 2D kalibracijskih objekata što omogućuje bolje kalibracijske rezultate.



Slika 6. Kalibracijski objekt u stilu otvorene knjige [9]

3.3. Samokalibracija

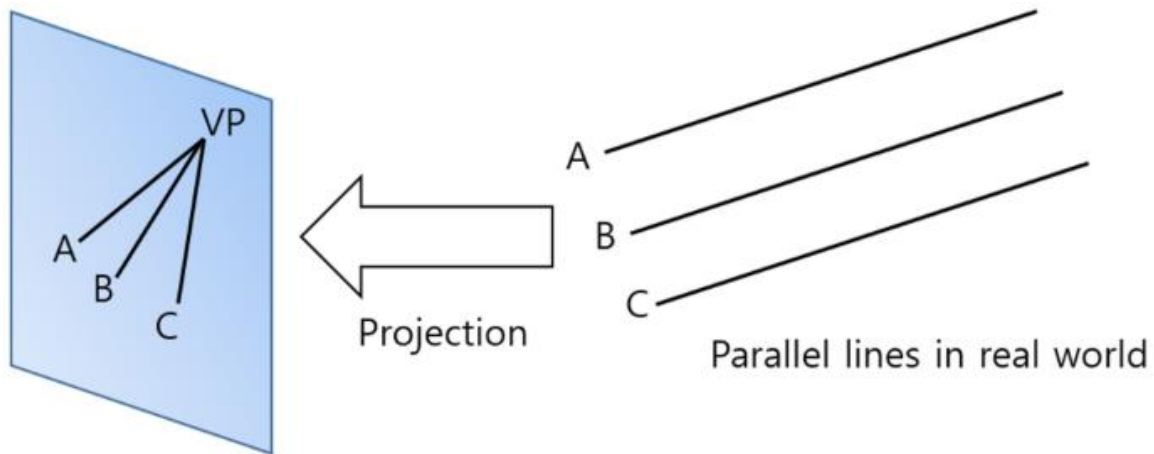
Samokalibracija, poznata i kao autokalibracija ili SfM (engl. *structure from motion*) je pristup kalibraciji u kojem se ne koriste specifični kalibracijski objekti ili uzorci. Umjesto toga, koriste se odnosi značajka slike kroz više pogleda iste statične scene kako bi se istovremeno odredili parametri kamere i 3D struktura scene. Glavni principi rada kod samokalibracije su osnovani na ograničenjima koja nastaju kada se na scenu gleda iz više pogleda. Ta ograničenja proizlaze iz epipolarne geometrije, Kruppavih jednadžbi koje povezuju intrinzične parametre kamere sa fundamentalnom matricom i rigidnosti scene koja nalaže da scena treba ostati nepomična za vrijeme slikanja.

Algoritmi samokalibracije često koriste iterativne postupke kako bi optimizirali parametre kamere i 3D strukture sve dok konvergencija nije postignuta. Samokalibracija se koristi u slučajevima gdje je potrebno uzeti u obzir puno ulaznih slika i u slučajevima gdje se kamere koje je potrebno kalibrirati nalaze na nepristupačnim mjestima u kojima tradicionalne metode kalibracije nisu moguće.

Nedostaci samokalibracije su teža procjena inicijalnih vrijednosti parametara kamere i 3D strukture pogotovo uz prisustvo šuma, nesigurnosti oko prave veličine predmeta što traži dodatne geometrijske informacije o slici, i komputacijska složenost procesa.

3.4. Kalibracija na temelju iščezavajućih točaka

Kalibracija na temelju iščezavajućih točaka koristi njihova geometrijska svojstva kako bi procijenila parametre kamere. Iščezavajuća točka predstavlja točku u ravnini slike gdje perspektivne projekcije paralelnih linija iz 3D prostora naizgled konvergiraju. U slučaju da su paralelne linije okomite na ravninu kamere, takva konstrukcija scene naziva se perspektiva s jednom točkom (eng. *one-point perspective*). Kod takve konstrukcije, točke konvergiraju u takozvanoj očnoj točki (eng. *eye point*).



Slika 7. Iščezavajuća točka [10]

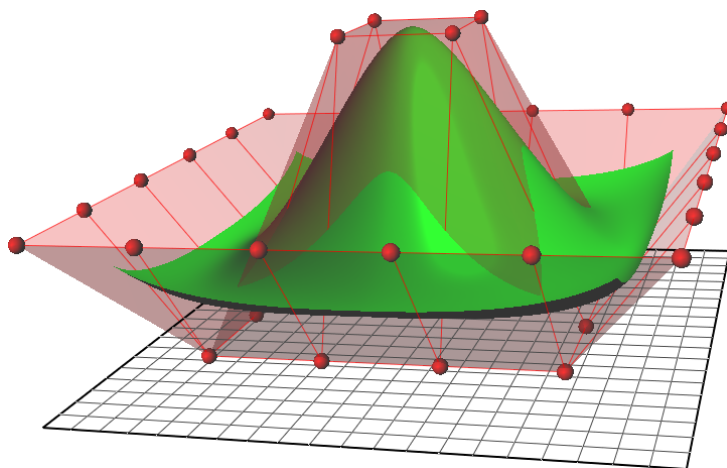
Kalibracija temeljena na iščezavajućim točkama nudi nekoliko prednosti. Kalibracija se može izvršiti koristeći jednu sliku ili nekoliko slika koje sadrže iščezavajuće točke, bez potrebe za specifičnim kalibracijskim uzorkom. Ova tehnika je posebno prikladna za urbane ili arhitektonske scene u kojima su paralelne linije, poput bridova kockama, česte. Još jedna prednost je robusnost na djelomična zaklanjanja, jer se iščezavajuće točke mogu procijeniti čak i ako su neke od paralelnih linija djelomično zaklonjene, što proces kalibracije čini otpornijim na prepreke.

Međutim, kalibracija temeljena na iščezavajućim točkama također ima neka ograničenja. Preciznost i primjenjivost ove tehnike oslanjaju se na prisutnost prikladnih iščezavajućih točaka u sceni, što znači da je ovisna o geometriji scene. Scene koje nemaju dominantne ortogonalne smjerove možda nisu prikladne za ovaj pristup. Osim toga, točnost detekcije iščezavajućih točaka ovisi o pouzdanosti algoritama za detekciju linija, što znači da je osjetljiva na detekciju linija. Šumne ili nepotpune detekcije linija mogu utjecati na rezultate kalibracije. Još jedno ograničenje je da se kalibracija temeljena na iščezavajućim točkama prvenstveno fokusira na procjenu orijentacije kamere i žarišne duljine, što znači da je ograničena u procjeni vanjskih parametara. Procjena potpunih vanjskih parametara, uključujući položaj kamere, može zahtijevati dodatne informacije ili tehnike.

4. GLOBALNI PRISTUP KALIBRACIJI STEREOVIZIJSKIH SUSTAVA

Prethodno je pokazano kako je za kalibraciju stereovizijskog sustava potreban kalibracijski objekt za koji su svi geometrijski odnosi strogo definirani i poznati. Međutim, ukoliko nije moguće koristiti isti, moguće je koristiti poznatu geometriju samog promatranog objekta za kalibraciju stereovizijskog sustava.

Navedeni pristup implementiran je u sklopu globalnog 2D i stereo DIC pristupa koji koristi mrežu konačnih elemenata za mjerenje cijelog polja pomaka. Glavna razlika između 2D i stereo DIC metode je potreba za provedbom postupka kalibracije u potonjoj mjernoj metodi. Globalni pristup kalibraciji razvijen je na temelju dva pristupa diskretizaciji promatranog područja, *NURBS* (eng. *Non-Uniform Rational B-Splines*) i mrežom konačnih elemenata. *NURBS* je 3D mreža *B-Spline* krivulja koje mogu predstavljati zaobljenja površine nekog elementa. Za razliku od mreže konačnih elemata koja modelira površinu predmeta na temelju poligona, *NURBS* mreža prikazana na modelira površinu predmeta putem krivulja. Svaka krivulja mreže je definirana putem kontrolnih točaka što je čini vrlo preciznom, no i vrlo složenom i teškom za korištenje pri kalibraciji kamera. Stoga se za potrebe ovoga rada koristi mreža konačnih elemenata kojom je opisana površinska (CAD) geometrija promatranog objekta preko trokutastih konačnih elemenata. Prednost primjene mreže konačnih elemenata je direktna povezanost sa računalnim simulacijama.



Slika 8. Mreža B-spline krivulja (zeleno) s njihovim pripadajućim kontrolnim točkama (crveno)[13]

Globalni pristup kalibraciji uključuje kalibraciju kamera usklađivanjem projekcijskih matrica kamera na temelju zabilježene slike i poznate CAD geometrije uzorka. Projekcijom mreže točaka na sliku, projekcijske matrice za svaku kameru se mogu optimizirati kroz iterativni postupak.

4.1. Predkalibracija

Predkalibracija u sklopu globalnog pristupa kalibraciji je procedura kojom se procjenjuju početne vrijednosti traženih intrinzičnih i ekstrinzičnih parametara kamera. Ovaj proces temelji se na poznavanju koordinata kontrolnih točaka na 3D objektu, tj. mreži konačnih elemenata, te 2D koordinata na slikama kamera.

4.1.1. DLT algoritam

DLT (eng. *Direct Linear Transform*) predkalibracijski algoritam računa početne ekstrinzične i intrinzične parametre kamere iz 3D koordinata u globalnom koordinatnom sustavu R_w i koordinata odabranih točaka u koordinatnom sustavu senzora R_s . Transformacija iz globalnih koordinata na slikovne koordinate, inače poznata kao i *PFT* (eng. *Projection Forward Transformation*) glasi

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = [\mathbf{II}] \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}. \quad (4.1)$$

Nepoznanica koju je potrebno odrediti iz izraza (4.1) je projekcijska matrica $[\mathbf{II}]$. Prvo se odabiru točke na mreži konačnih elemenata i točke na slikama, pri čemu odabrane točke na slikama moraju što bliže odgovarati poziciji odabranih točaka na mreži. Uz poznate koordinate piksela na senzoru kamere i 3D koordinate na mreži, mogu se dobiti elementi projekcijske matrice $[\mathbf{II}]$ prema jednadžbama

$$x_i = \frac{p_{11}X_i + p_{12}Y_i + p_{13}Z_i + p_{14}}{p_{31}X_i + p_{32}Y_i + p_{33}Z_i + p_{34}}, \quad (4.2)$$

$$y_i = \frac{p_{21}X_i + p_{22}Y_i + p_{23}Z_i + p_{24}}{p_{31}X_i + p_{32}Y_i + p_{33}Z_i + p_{34}}. \quad (4.3)$$

Matrica **[II]** će sadržavati dvanaest nepoznanica što znači da je potrebno odabrati barem šest točaka na slikama i mreži kako bi riješili sustav jednažbi. Raspisivanjem matrice **[II]**

$$\{\mathbf{x}_i\} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} \{\mathbf{X}_i\}, \quad (4.4)$$

možemo definirati tri vektora, **{A}**, **{B}** i **{C}** gdje svaki vektor predstavlja jedan redak parametara iz matrice **[II]**

$$\{\mathbf{A}\} = \begin{bmatrix} p_{11} \\ p_{12} \\ p_{13} \\ p_{14} \end{bmatrix}, \quad \{\mathbf{B}\} = \begin{bmatrix} p_{21} \\ p_{22} \\ p_{23} \\ p_{24} \end{bmatrix}, \quad \{\mathbf{C}\} = \begin{bmatrix} p_{31} \\ p_{32} \\ p_{33} \\ p_{34} \end{bmatrix}. \quad (4.5)$$

Jednažba (4.4) se sada može preoblikovati u

$$\{\mathbf{x}_i\} = \begin{bmatrix} \mathbf{A}^T \\ \mathbf{B}^T \\ \mathbf{C}^T \end{bmatrix} \{\mathbf{X}_i\}. \quad (4.6)$$

Za homogene projicirane koordinate, gornja jednažba će glasiti

$$\begin{bmatrix} u_i \\ v_i \\ w_i \end{bmatrix} = \begin{bmatrix} \{\mathbf{A}^T\}\{\mathbf{X}_i\} \\ \{\mathbf{B}^T\}\{\mathbf{X}_i\} \\ \{\mathbf{C}^T\}\{\mathbf{X}_i\} \end{bmatrix}. \quad (4.7)$$

Pretvaranje homogenih koordinata u euklidske provodi se dijeljenjem cijelog vektora projiciranih koordinata sa zadnjim elementom w_i , iz čega se dobiju x_i i y_i koordinate točke

$$x_i = \frac{u_i}{w_i} = \frac{\{\mathbf{A}^T\}\{\mathbf{X}_i\}}{\{\mathbf{C}^T\}\{\mathbf{X}_i\}}, \quad (4.8)$$

$$y_i = \frac{v_i}{w_i} = \frac{\{\mathbf{B}^T\}\{\mathbf{X}_i\}}{\{\mathbf{C}^T\}\{\mathbf{X}_i\}}. \quad (4.9)$$

Iz izraza (4.8) i (4.9) proizlazi sustav jednažbi

$$x_i = \frac{u_i}{w_i} = \frac{\{\mathbf{A}^T\}\{\mathbf{X}_i\}}{\{\mathbf{C}^T\}\{\mathbf{X}_i\}} \Rightarrow x_i\{\mathbf{C}^T\}\{\mathbf{X}_i\} - \{\mathbf{A}^T\}\{\mathbf{X}_i\} = 0, \quad (4.10)$$

$$y_i = \frac{u_i}{w_i} = \frac{\{\mathbf{A}^T\}\{\mathbf{X}_i\}}{\{\mathbf{C}^T\}\{\mathbf{X}_i\}} \Rightarrow y_i\{\mathbf{C}^T\}\{\mathbf{X}_i\} - \{\mathbf{B}^T\}\{\mathbf{X}_i\} = 0. \quad (4.11)$$

Vektore $\{\mathbf{A}\}$, $\{\mathbf{B}\}$ i $\{\mathbf{C}\}$ možemo skupiti u vektor parametara $\{\mathbf{p}\}$ koji će sadržavati sve nepoznate parametre projekcijske matrice koje tražimo

$$\{\mathbf{p}\} = \begin{bmatrix} \{\mathbf{A}\} \\ \{\mathbf{B}\} \\ \{\mathbf{C}\} \end{bmatrix}. \quad (4.12)$$

Sada se jednažbe (4.10) i (4.11) mogu zapisati kao

$$\{\mathbf{a}_{xi}^T\}\{\mathbf{p}\} = 0, \quad (4.13)$$

$$\{\mathbf{a}_{yi}^T\}\{\mathbf{p}\} = 0 \quad (4.14)$$

gdje vektori $\{\mathbf{a}_{xi}\}$ i $\{\mathbf{a}_{yi}\}$ sadrže informacije o izabranim globalnim koordinatama točaka i njihovim projiciranim koordinatama

$$\{\mathbf{a}_{xi}\} = \begin{bmatrix} -X_i \\ -Y_i \\ -Z_i \\ -1 \\ 0 \\ 0 \\ 0 \\ 0 \\ x_i X_i \\ x_i Y_i \\ x_i Z_i \\ x_i \end{bmatrix}, \{\mathbf{a}_{yi}\} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -X_i \\ -Y_i \\ -Z_i \\ -1 \\ y_i X_i \\ y_i Y_i \\ y_i Z_i \\ y_i \end{bmatrix}. \quad (4.15)$$

Vektori $\{\mathbf{a}_{xi}\}$ i $\{\mathbf{a}_{yi}\}$ za svaku od točaka se mogu skupiti u matricu $[\mathbf{M}]$ veličine $(2i \times 12)$, gdje je i broj točaka

$$\begin{bmatrix} \mathbf{a}_{x1}^T \\ \mathbf{a}_{y1}^T \\ \vdots \\ \mathbf{a}_{xi}^T \\ \mathbf{a}_{yi}^T \end{bmatrix} \{\mathbf{p}\} = [\mathbf{M}]\{\mathbf{p}\} = 0. \quad (4.16)$$

Rješavanje homogenih linearnih sistema jednadžbi oblika $\mathbf{Ax} = \mathbf{0}$ (poput jednadžbe (4.16)) je ekvivalentno pronalasku nultog prostora (eng. *null space*) matrice \mathbf{A} . Nulti prostor matrice je vektor koji, kada se pomnoži s tom istom matricom, daje nulu. U većini slučajeva, jednadžba (4.16) neće biti nula zato što ćemo imati više jednadžbi nego nepoznanica. U tom slučaju jednadžba (4.16) će imati oblik

$$[\mathbf{M}]\{\mathbf{p}\} = w. \quad (4.17)$$

Pronalazi se vektor $\{\mathbf{p}\}$ koji minimizira vrijednost Ω , gdje je Ω

$$\Omega = w^T w. \quad (4.18)$$

4.1.2. SVD algoritam

Vektor $\{\mathbf{p}\}$ koji zadovoljava uvjet (4.18), se može pronaći putem SVD (eng. *Singular Value Decomposition*) algoritma. Matrica $[\mathbf{M}]$ se rastavlja na tri posebne matrice: $(2i \times 12)$ matricu $[\mathbf{U}]$, (12×12) matricu $[\mathbf{S}]$ i (12×12) matricu $[\mathbf{V}^T]$. Matrica $[\mathbf{S}]$ je dijagonalna matrica koja na dijagonali sadrži jedinične vrijednosti poredane od najveće do najmanje, dok matrica $[\mathbf{V}^T]$ sadrži jedinične vektore. Prema tome, iz matrice $[\mathbf{V}^T]$ možemo odabrati jedinični vektor koji odgovara najmanjoj jedinstvenoj vrijednosti u matrici $[\mathbf{S}]$, i njega koristiti kao vektor $\{\mathbf{p}\}$. Vektor $\{\mathbf{p}\}$ se onda može preoblikovati u projekcijsku matricu $[\mathbf{\Pi}]$

$$\{\mathbf{p}\} = \begin{bmatrix} p_{11} \\ p_{12} \\ \vdots \\ p_{33} \\ p_{34} \end{bmatrix} \Rightarrow [\mathbf{\Pi}] = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix}. \quad (4.19)$$

Dekompozicija matrice $[\mathbf{M}]$ je jedino moguća ako je rang matrice 11, to jest, slučaj kada se odabere više od šest 3D i 2D točaka za procjenu ekstrinzičnih i intrinzičnih parametara. Prilikom biranja točaka isto je važno da se sve izabrane točke ne nalaze u istoj prostornoj ravnini. U slučaju da je koordinata X za sve 3D točke jednaka nuli, iz vektora prikazanih u izrazu (4.15) i izraza (4.16) je vidljivo da će tri stupca matrice $[\mathbf{M}]$ biti jednaki nuli, što će joj smanjiti rang i onemogućiti pronalaženje rješenja.

Dobivena matrica $[\mathbf{\Pi}]$ direktno ne sadrži željene ekstrinzične i intrinzične parametre nego je dobivena u obliku prikazanom u jednadžbi (2.9). Matricu $[\mathbf{\Pi}]$ možemo zapisati kao

$$[\mathbf{\Pi}] = [[\mathbf{H}] | \{\mathbf{h}\}] \quad (4.20)$$

gdje je matrica $[\mathbf{H}]$ umnožak gornje triangularne kalibracijske matrice $[\mathbf{K}]$ i rotacijske matrice $[\mathbf{R}]$. Vektor $\{\mathbf{h}\}$ je umnožak negativne matrice $[\mathbf{H}]$ i vektora koordinata projekcijskog centra $\{\mathbf{X}_o\}$.

QR dekompozicijom se iz matrice $[\mathbf{H}]$ dobivaju matrice $[\mathbf{K}]$ i $[\mathbf{R}]$. Potrebno je uzeti inverz matrice $[\mathbf{H}]$ kako bi matrice $[\mathbf{K}]$ i $[\mathbf{R}]$ zauzele mjesta jednaka rezultatnim matricama QR dekompozicije

$$[\mathbf{H}] = [\mathbf{K}][\mathbf{R}] \Rightarrow [\mathbf{H}]^{-1} = [\mathbf{R}]^T [\mathbf{K}]^{-1}. \quad (4.21)$$

Nakon provedene QR dekompozicije, matricu $[\mathbf{K}]$ je potrebno normalizirati dijeljenjem cijele matrice s njenim donjim desnim elementom k_{33} kako bi dobili oblik matrice prikazan u jednadžbi (2.7). Matricu $[\mathbf{K}]$ je isto potrebno zarotirati oko Z osi kako bi dobili dobre predznake na njenoj dijagonali. Matrica $[\mathbf{K}]$ sadrži potrebne intrinzične i ekstrinzične parametre.

4.2. Kalibracija

Nakon određivanja početnih rješenja projekcijskih matrica za obje kamere putem DLT algoritma, iz matrica $[\mathbf{K}]$ i $[\mathbf{R}]$ mogu se izvući intrinzični i ekstrinzični parametri za svaku kameru c . Za kalibraciju, pretpostavlja se konzervacija optičkog toka vrijednosti sive skale boja piksela na način da slike zabilježene lijevom $f^1(\{\mathbf{x}^1\})$ i desnom $f^2(\{\mathbf{x}^2\})$ kamerom imaju isti intenzitet sive skale piksela na promatranom području te zadovoljavaju jednadžbu [19]

$$f^1(\{\mathbf{x}^1\}) = f^2(\{\mathbf{x}^2\}). \quad (4.22)$$

Izraz (4.22) će jedino vrijediti idealnom slučaju, bez prisutnosti šuma. Međutim, u stvarnosti nije moguće postići isti intenzitet vrijednosti sive skale boja pojedinih piksela za obje kamere. Kako bi povećali broj točaka na mreži konačnih elemenata, mreža se može diskretizirati na Gaussove integracijske točke. Ta diskretizacija se dobiva postavljanjem točaka duž rubova i u unutašnjost konačnih elemenata primjenom algoritma koji interpolira vrijednosti koordinata točaka konačnog elementa. Mreža integracijskih točaka će biti finija od rezolucije slike, što znači da se unutar jednog piksela mogu nalaziti više integracijskih točaka. Sada se za svaku koordinatu projicirane integracijske točke mogu, putem bilinearne interpolacije, interpolirati sive vrijednosti piksela na tim koordinatama. Nadalje, postavlja se globalna formulacija minimizacijskog problema gdje se suma kvadratnih razlika

$$\rho^2(\{\mathbf{P}\}) = \left\| f^1(\{\mathbf{x}^1\}(\{\mathbf{X}\}, [\mathbf{\Pi}^1])) - f^2(\{\mathbf{x}^2\}(\{\mathbf{X}\}, [\mathbf{\Pi}^2])) \right\|^2 \quad (4.23)$$

minimizira nad promatranim područjem (eng. *Region Of Interest* - ROI), gdje ρ predstavlja kalibracijske rezidualne, vektor $\{\mathbf{P}\}$ sadrži intrinzične i ekstrinzične parametre kamere, $\{\mathbf{x}^c\} = (x^c, y^c)$ je vektor koordinata točaka na slikama s kamere, a $\{\mathbf{X}\} = (X, Y, Z)$ predstavlja vektor koordinata točaka na mreži konačnih elemenata. Reziduali se na početku svake iteracije računaju preko formule

$$\rho = \frac{1}{dyn} \sqrt{\frac{f_1^2(\mathbf{x}_1) - f_2^2(\mathbf{x}_2)}{Nc}} \quad (4.24)$$

gdje je dyn raspon vrijednosti sive skale boja od najmanje do najveće vrijednosti, a Nc broj kamere. Za ovaj integrirani pristup, procesom kalibracije određuju se intrinzični i ekstrinzični parametri i iz njih se grade projekcijske matrice $[\mathbf{\Pi}^{1,2}]$.

Taylorovim razvojem prvog reda jednadžbe (4.23) dobiva se linearni sustav

$$[\mathbf{C}]\{\delta\mathbf{P}\} = \{\mathbf{b}\} \quad (4.25)$$

gdje $[\mathbf{C}]$ predstavlja globalnu Hessian matricu, $\{\delta\mathbf{P}\}$ sadrži promjene parametara kamere s obzirom na početne pretpostavljene vrijednosti parametara, te $\{\mathbf{b}\}$ označuje vektor drugih članova. Polja osjetljivosti se za svaki parametar računaju uvođenjem malih varijacija parametara koje dovode do polja prividnih pomaka u 2D ravnini kamere za svaku 3D točku od $\{\mathbf{x}^c\}$ do $\{\mathbf{x}^c\} + \delta\{\mathbf{x}^c\}$

$$[\mathbf{S}_P^c] = \frac{\delta\{\mathbf{x}^c\}}{\delta\{\mathbf{P}^c\}} \quad (4.26)$$

Polje osjetljivosti $[\mathbf{S}_P^c]$ je 2D vektorsko polje koje opisuje kako promjena parametara utječe na položaj projiciranih točaka. Primjenom iterativne Newton-Raphson metode minimizira se funkcija (4.23) optimizacijom intrinzičnih i ekstrinzičnih parametara. Nakon računanja polja osjetljivosti, Hessian $[\mathbf{H}^c]$ matrica za svaku kameru se računa prema

$$[\mathbf{H}^c] = \sum_{ROI} \left[\nabla f^c(\{\mathbf{x}^c\}(\{\mathbf{X}\}, [\mathbf{\Pi}^c])) \cdot \{\mathbf{S}_P^c\} \right]^T \left[\nabla f^c(\{\mathbf{x}^c\}(\{\mathbf{X}\}, [\mathbf{\Pi}^c])) \cdot \{\mathbf{S}_P^c\} \right]. \quad (4.27)$$

Operator ∇ u izrazu (4.27) označuje gradijent intenziteta sivih vrijednosti na slikama u svakoj integracijskoj točki. Hessian matrice za svaku kameru se skupljaju u globalnu Hessian matricu $[\mathbf{C}]$

$$[\mathbf{C}] = \begin{bmatrix} [\mathbf{H}^1] & 0 \\ 0 & [\mathbf{H}^2] \end{bmatrix} \quad (4.28)$$

Vektori drugih članova $\{\mathbf{b}^c\}$ (eng. *second member*) za svaku kameru se mogu izraziti kao

$$\{\mathbf{b}^1\} = \sum_{ROI} \left[[\mathbf{S}_P^1] \cdot \nabla f^1(\{\mathbf{x}^1\}) \right]^T \cdot \left(f^1(\{\mathbf{x}^1\}(\{\mathbf{X}\}, [\mathbf{\Pi}^1])) - f^2(\{\mathbf{x}^2\}(\{\mathbf{X}\}, [\mathbf{\Pi}^2])) \right), \quad (4.29)$$

$$\{\mathbf{b}^2\} = \sum_{ROI} \left[[\mathbf{S}_P^2] \cdot \nabla f^2(\{\mathbf{x}^2\}) \right]^T \cdot \left(f^2(\{\mathbf{x}^2\}(\{\mathbf{X}\}, [\mathbf{\Pi}^2])) - f^1(\{\mathbf{x}^1\}(\{\mathbf{X}\}, [\mathbf{\Pi}^1])) \right) \quad (4.30)$$

i skupiti u globalni vektor drugih članova

$$\{\mathbf{B}\} = \begin{Bmatrix} \{\mathbf{b}^1\} \\ \{\mathbf{b}^2\} \end{Bmatrix} \quad (4.31)$$

Tijekom svake iteracije, korekcije traženih parametara za svaku kameru su izračunate pomoću izraza

$$\{\delta\mathbf{P}\} = [\mathbf{C}]^{-1}\{\mathbf{B}\}. \quad (4.32)$$

Kriterij konvergencije je dan u izrazu (4.33) i izražen je preko promjene kalibracijskih reziduala

$$\Delta\rho = \rho^n - \rho^{n-1} < 10^{-5}. \quad (4.33)$$

gdje je n broj trenutačne iteracije. Iteracije traju sve dok se kriterij konvergencije ne zadovolji.

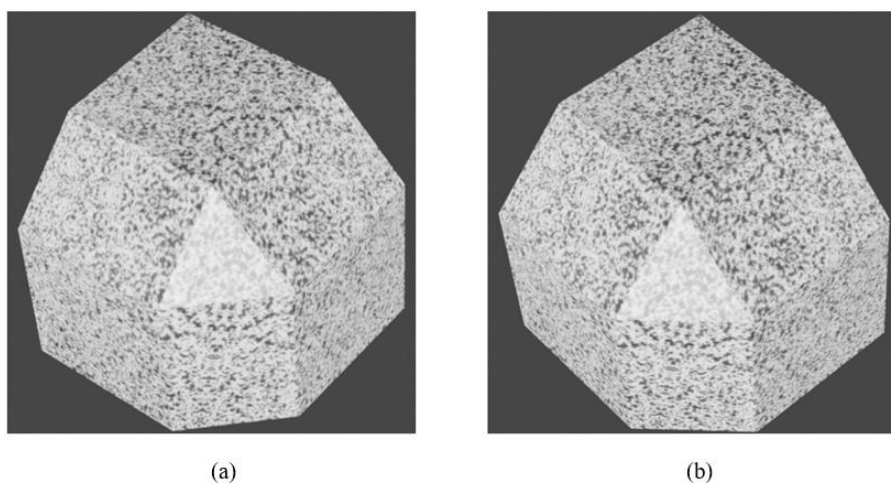
5. APLIKACIJA CROCAB

Aplikacija pod nazivom CroCab (eng. *Croatian Calibration*) sastoji se od dvije softverske cjeline, predkalibracijskog softvera napisanog u programskom jeziku Python i kalibracijskog dijela softvera napisanog u programskom jeziku C++. Python je odabran kao predkalibracijski softver zbog velike količine programskih biblioteka. Nadalje, programiranje u Pythonu je vrlo čitljivo, te je integracija raznih programskih biblioteka jednostavna. Dio predkalibracije nije računalno intenzivan za razliku od kalibracijskog dijela, pa je zato on napisan u C++-u kako bi se tijekom daljnjeg razvoja aplikacije omogućila vremenska i memorijska optimizacija kalibracijskog koda.

Za razvoj sučelja, korištena je biblioteka PyQt5 koja sadrži posebni program napravljen za brzo i efikasno slaganje programskog sučelja.

5.1. Ulazni podatci

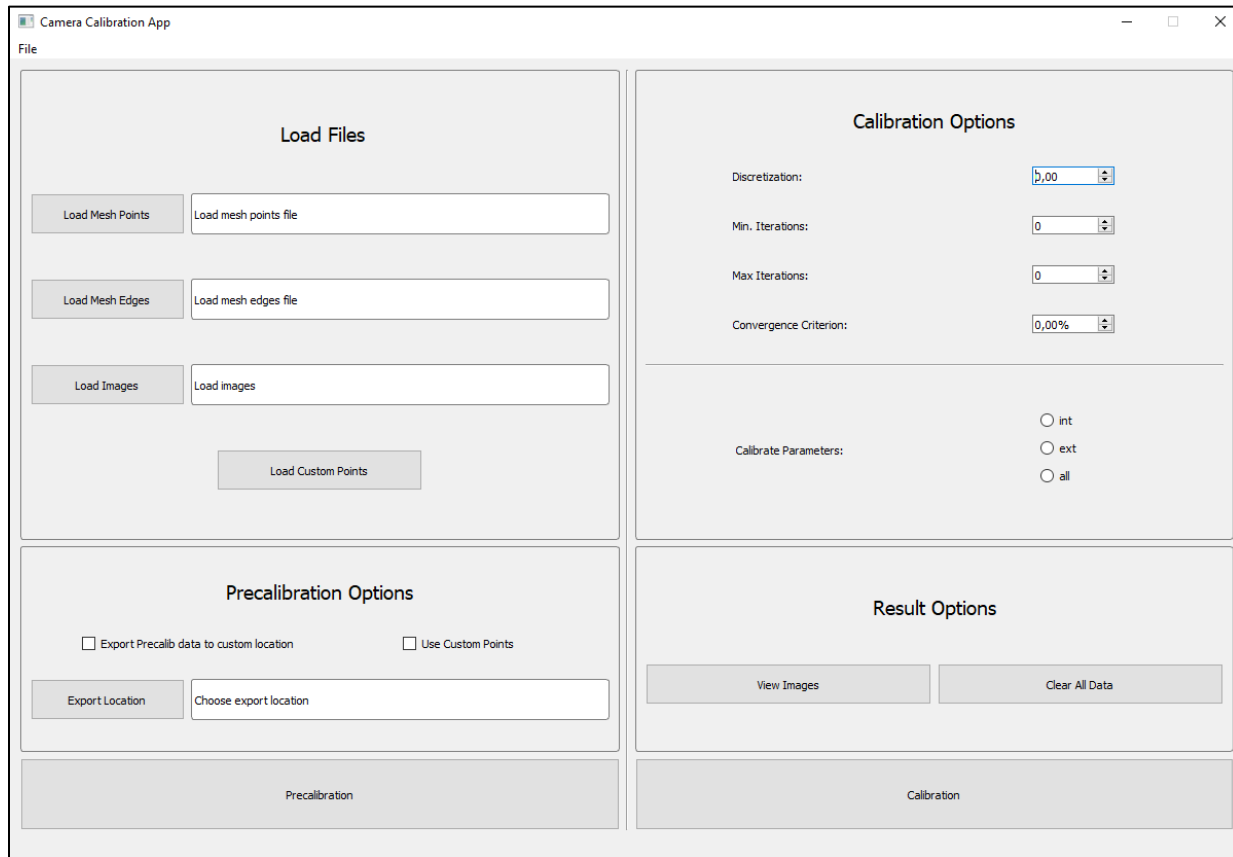
Ukupno postoji četiri ulazna podatka koja je potrebno dostaviti u dva posebna formata. Prva dva podatka su strukturno slični Abaqusovom ulaznom podatku koji opisuje mrežu konačnih elemenata. Abaqusov ulazni podatak sadrži informacije o koordinatama točaka i o njihovoj međusobnoj povezanosti, dok su u CroCab aplikaciji te informacije podjeljene u dva zasebna podatka. Uz podatke o mreži konačnih elemenata potrebno je učitati i slike za kalibraciju. Trenutačno aplikacija podržava učitavanje dvije slike istog kalibracijskog elementa. Kalibracijske slike su prikazane na Slici 10.



Slika 9. Umjetno generirane slike s lijeve kamere (a) i desne kamere (b)

5.2. Predkalibracija

Pokretanjem aplikacije pokazuje se programsko sučelje. Sučelje je podjeljeno na dva dijela, lijevi predkalibracijski dio i desni kalibracijski dio. Tijek rada teče odozgora prema dolje unutar zasebnog dijela.



Slika 10. Izgled programskog sučelja

Predkalibracija počinje sa učitavanjem potrebnih datoteka klikom na gumbе *Load Mesh Points*, *Load Mesh Edges* i *Load Images*. Klikom na svaki gumb otvara se prozor za odabir datoteka. Gumbi *Load Mesh Points* i *Load Mesh Edges* otvaraju prozor za odabir jedne datoteke, dok gumb *Load Images* otvara prozor za odabir dvije slike. Odabir više od jedne datoteke u prozoru odvija se držanjem gumba *Shift* ili *Ctrl* na tipkovnici i lijevim klikom miša.

5.2.1. Prikazivanje mreže konačnih elemenata

Za prikazivanje mreže konačnih elemenata i biranje točaka na njoj koristi se PyVista biblioteka za 3D vizualizaciju i analizu mreža. Kako bi PyVista mogla sagraditi mrežu, potrebno joj je dati podatke o koordinatama točaka mreže i podatke o međusobnoj povezanosti točaka. Proces učitavanja podataka odvija se klikom na spomentue gumbe. Koordinate mreže se prije njenog građenja moraju normalizirati procesom homotetije.

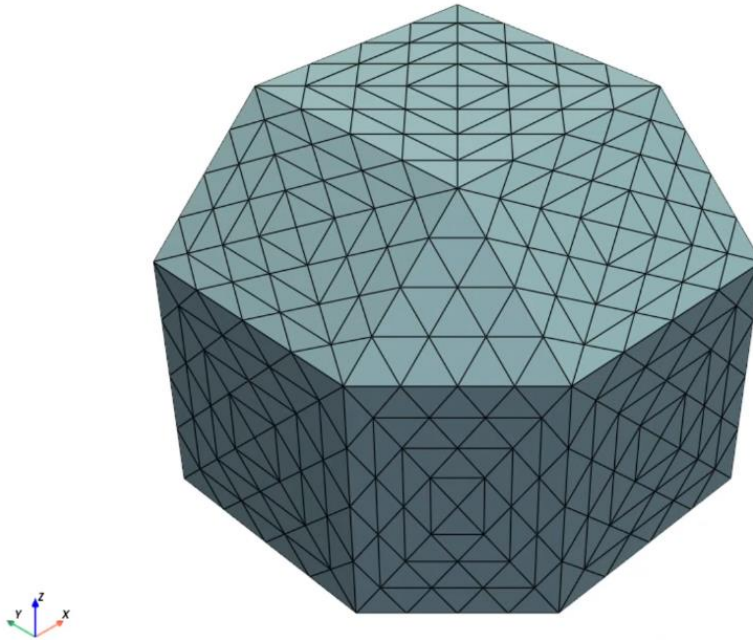
Homotetija mreže je geometrijska transformacija koja se primjenjuje na 3D mreže, a uključuje skaliranje mreže u odnosu na fiksnu točku, poznatu kao središte homotetije. Ova transformacija čuva oblik originalne mreže, mijenjajući jedino njezinu veličinu. U homotetičkoj transformaciji, svaka točka mreže pomiče se duž linije koja je povezuje sa središtem homotetije, bilo prema središtu ili od njega. Taj pomak je određen faktorom koji je proporcionalan njezinoj izvornoj udaljenosti od središta. Faktor k , nazvan omjer homotetije, određuje hoće li se mreža povećati (omjer > 1) ili smanjiti ($0 < \text{omjer} < 1$). U ovom slučaju taj se omjer dobiva preko formule

$$k = \frac{1}{\sqrt{\delta_1 \cdot \delta_2}}, \quad (5.1)$$

gdje su δ_1 i δ_2 dvije najveće razlike između x-koordinata, y-koordinata ili z-koordinata točaka

$$\delta = \max(x_{\max} - x_{\min}, y_{\max} - y_{\min}, z_{\max} - z_{\min}). \quad (5.2)$$

Svaka koordinata mreže konačnih elemenata se množi sa faktorom k kako bi se dobila skalirana mreža. Normalizacija mreže postupkom homotetije je važna zbog numeričke stabilnosti kalibracijskog algoritma.



Slika 11. Mreža konačnih elemenata kalibracijskog objekta

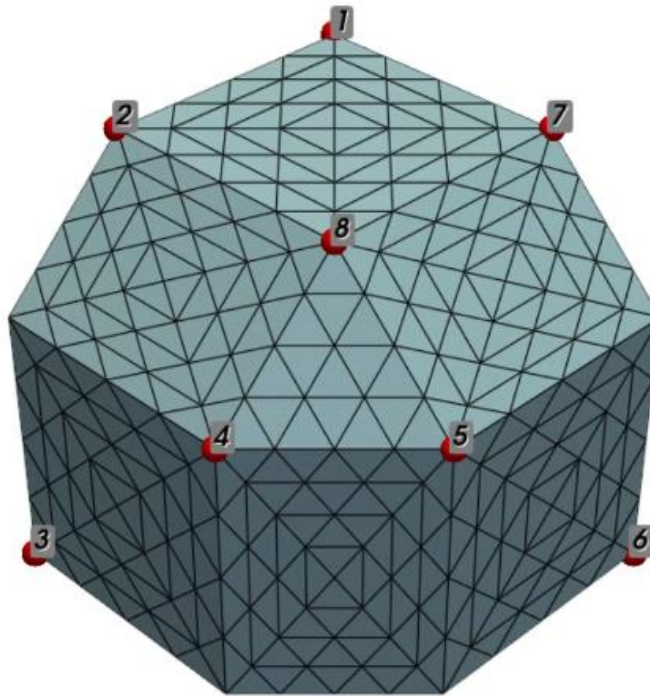
Mreža se gradi pomoću *plotter* objekta dostupnog u PyVisti. On uzima informacije učitavanjem podataka o koordinatama točaka i konačnim elementima iz tekstualnih datoteka. Početna pozicija kamere se namješta na način da je cijela mreža vidljiva i centrirana unutar prozora.

5.2.2. Odabir točaka na mreži konačnih elemenata

Ključan dio ovog algoritma je odabir točaka na mreži konačnih elemenata i pohrana koordinata odabranih točaka. Spremanje koordinata zadnje odabrane točke se postiže preko funkcije *enable_point_picking plotter* objekta, koja izvrši naredbu svaki put kada se klikne na točku. Prilikom odabira točke, točka se zacrveni i njene su koordinate spremljene u Python listu redom od prve do zadnje odabrane. Točke se odabiru putem desnog klika na mišu. Nakon odabira, prozor je moguće zatvoriti pritiskom na tipku *Esc* na tipkovnici.

Za postizanje rješenja jednadžbe (2.14.) iz koje se određuje projekcijska matrica kamera, potrebno je odabrati više od šest točaka na mreži konačnih elemenata. Zbog ograničenja da odabrane točke ne smiju biti u istoj ravnini, odabiru se točke na rubovima koje najbolje opisuju geometriju mreže, te time osiguravaju stabilnost i točnost rješenja. Nakon završetka odabira točaka, poziva se novi

plotter objekt koji u novom prozoru konstruira mrežu i ističe odabrane točke. Pored točaka se nalaze redni brojevi dodijeljeni prema redoslijedu biranja.



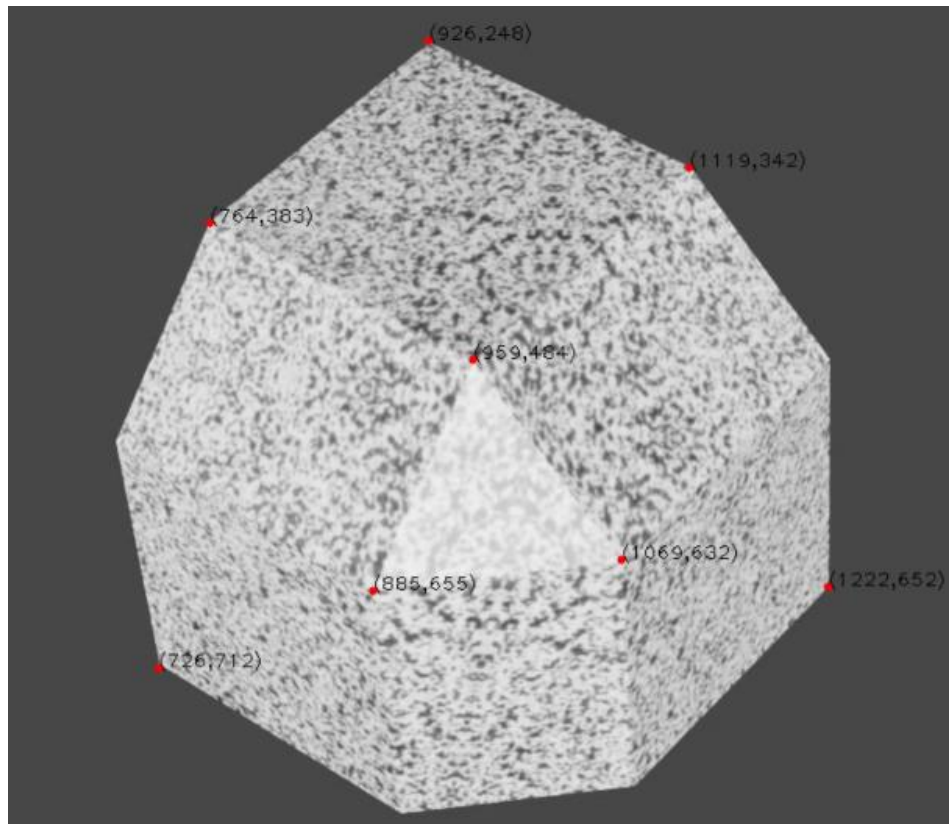
Slika 12. Točke na mreži s redoslijedom odabiranja

5.2.3. Odabir točaka na slikama

Nakon biranja točaka na mreži, točke se odabiru istim redoslijedom na lijevoj i desnoj slici. Slike s kamera prikazane na Slici 10. učitavaju se pomoću biblioteke OpenCV (eng. *Open Source Computer Vision Library*) koja je razvijena kako bi pružila zajedničku infrastrukturu za aplikacije računalnog vida i ubrzala korištenje strojne percepcije u komercijalnim proizvodima. Funkcionalnost koja je bitna za potrebe ove aplikacije je mogućnost biranja točaka na slikama u procesu predkalibracije i očitavanje sivih vrijednosti piksela u slikama kasnije u procesu kalibracije.

Funkcija koja omogućava biranje točaka na slici je *setMouseCallback*. Lijevim klikom miša na proizvoljnom mjestu na slici nacrtati će se crvena točka s njenim koordinatama.

Odabrane točke na slici moraju što bolje odgovarati odabranim točkama na mreži kako bi sagrađena projekcijska matrica bila što bliža potpuno kalibriranoj vrijednosti.



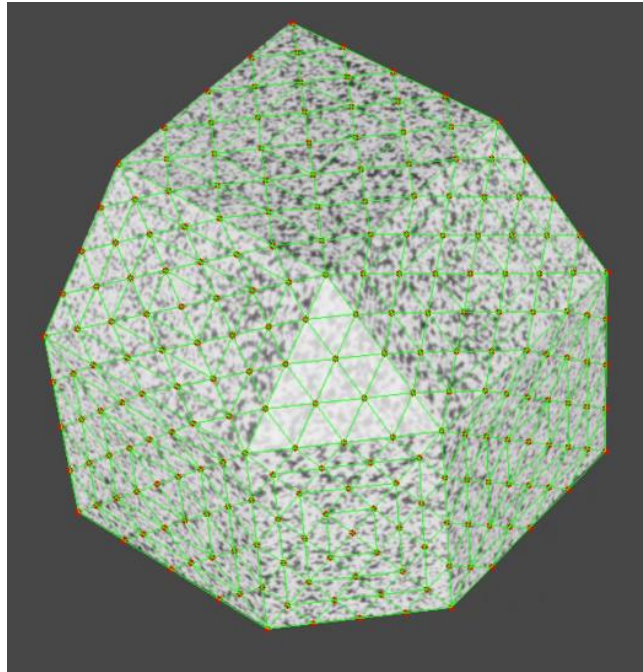
Slika 13. Odabrane točke na slici

Odabiranjem točaka na slikama, dobivaju se dvije liste koje sadrže koordinate točaka koje s odabranim točkama na meshu ulaze u funkciju DLT. Prvo se grade matrice $[\mathbf{M}^{1,2}]$ koje će za osam odabranih točaka imati dimenzije 16×12 . Za gradnju projekcijske matrice $[\mathbf{\Pi}^{1,2}]$ potrebno je provesti dekompoziciju matrice $[\mathbf{M}^{1,2}]$ kako bi se dobio vektore $\{\mathbf{p}^{1,2}\}$. SVD algoritam je sadržan unutar biblioteke *numpy*. Kako bi se provela dekompozicija 2D Python liste u kojoj su sadržani elementi matrica $[\mathbf{M}^{1,2}]$, potrebno je pretvoriti u strukturu podataka *numpy.array* nad kojom je omogućeno pozivanje funkcije *numpy.linalg.svd*($[\mathbf{M}^{1,2}]$). Dobiveni vektori $\{\mathbf{p}^{1,2}\}$ iste strukture *numpy.array* preoblikuju se u u matrice $[\mathbf{\Pi}^{1,2}]$ funkcijom *reshape*(4,3).

QR dekompozicijom putem funkcije *numpy.linalg.qr* dobiju se intrinzični i ekstrinzični parametri za svaku kameru u obliku matrica $[\mathbf{K}^{1,2}]$ i $[\mathbf{R}^{1,2}]$. Oni su potom korišteni su za projiciranje cijele mreže konačnih elemenata na slike, kako bi vidjeli jesu li projekcijske matrice dobro sagrađene,.

Prije projiciranja, 3D koordinate točaka na mreži moraju se pretvoriti u 2D koordinate koje je moguće crtati na slike. Funkcija *coordMeshToImg* radi tu pretvorbu putem parametara iz matrica $[\mathbf{\Pi}^{1,2}]$.

Mreža konačnih elemenata projicirana na sliku s lijeve kamere prikazana je na Slici 14.



Slika 14. Mreža konačnih elemenata projicirana na sliku s lijeve kamere

Kako bi se omogućio prijenos podataka između koda napisanog u Pythonu i koda napisanog u C++-u, vrijednosti dobivene u predkalibraciji se spremaju u *.json* datoteku. *Json* je kompaktni format koji služi za razmjenu podataka te se prvenstveno koristi kao komunikacija između servera i web aplikacije i u svrhe razmjene konfiguracijskih postavki. Zbog te druge stavke, koristan je i kao medij razmjene podataka između Pythona i C++-a zbog male količine podataka koji se razmjenjuju. Među tim podacima učitaju se i elementi mesha s njihovim koordinatama. U drugu, posebnu *.json* datoteku su spremljeni putevi do mapa koji sadržavaju slike koje je Python generirao, te putevi do mapa u koje će se spremati podatci dobiveni iz procesa kalibracije.

5.3. Kalibracija

Kalibracija projekcijskih matrica napisana je u C++-u zbog brzine programskog jezika i mogućnosti optimizacije algoritma u budućnosti. Programske biblioteke koje su korištene u

kalibracijskom dijelu su OpenCV(verzija napisana za C++) i Eigen, matematička biblioteka koja omogućava rukovanje sa strukturama podataka poput matrica i kvaterniona, te podržava složenije matematičke operacije nad tim strukturama. Kako bi kalibracija započela, potrebno je popuniti polja sa kalibracijskim postavkama i odrediti koji parametri kamere se žele kalibrirati.

5.3.1. Postavke prije kalibracijske petlje

Svi podatci dobiveni u pretkalibraciji se učitavaju iz *.json* datoteke. Mreža konačnih elemata u konačnici sadrži 220 elemenata, a slike su rezolucije 1920x1080 piksela. Mreža se potom diskretizira na Gaussove integracijske točke. Faktor koji utječe na finoću diskretizacije mreže unosi se u sučelju pod polje vrijednost diskretizacije. Što je vrijednost diskretizacije manja, to će se mreža konačnih elemenata diskretizirati na više integracijskih točaka.

Prije očitavanja vrijednosti u točkama potrebno je učitati slike uporabom OpenCV programske biblioteke koja se koristi i u predkalibraciji. Nakon učitavanja, slike se spremaju u matrice gdje svaki element matrice odgovara sivoj vrijednosti piksela. Siva vrijednost piksela je zapisana u obliku cijelog broja od 0 do 255. Za sliku rezolucije 1920x1080, matrica te slike će biti istih dimenzija. Pri kalibraciji, algoritam mora imati informacije o tome kako se promjene intrinzičnih i ekstrinzičnih parametara odražuju na promjenu sivih vrijednosti. Kako bi se to postiglo, za svaku sliku je potrebno izračunati gradijent u pozitivnom smjeru X i Y osi. Time se na mjestu svakog elementa zapisuje promjena u sivim vrijednostima umjesto apsolutnih sivih vrijednosti.

Prije ulaska u iteracijsku petlju potrebno je definirati postavke kalibracije. Te postavke se očitavaju iz sučelja i uz finoću diskretizacije mreže konačnih elemenata određuju minimalni i maksimalni broj iteracija kalibracijske petlje, kriterij konvergencije i parametre koji će se kalibrirati. Moguće je kalibrirati posebno intrinzične i ekstrinzične parametre ili sve zajedno. Prije građenja hessian matrice važno je normalizirati sve parametre kako bi se smanjio njen kondicijski broj (eng. *conditioning number*). Kondicijski broj mjeri koliko se izlazne varijable funkcije mijenjaju za malu promjenu ulaznih varijabli. Ako su parametri u hessian matrici neujednačeni što se tiče veličine, njen kondicijski broj će biti velik. Bez postupka normalizacije, proces kalibracije bi trajao puno duže ili uopće nebi mogao konvergirati. Faktor s kojim se ti parametri normaliziraju se naziva *scale image (SI)* i inicijalizacijska vrijednost mu je 1920, što generalno odgovara širini slike u pikselima.

5.3.2. Kalibracijska petlja

Na početku iteracije, mreža Gaussovih integracijskih točaka se projicira na sliku f_c sa svake kamere preko funkcije *coordMeshToImg*. Uz projicirane integracijske točke, interpoliraju se i matrice gradijenata slika, te se time za svaku sliku dobivaju dva vektora $\{\mathbf{g}_{xi}\}$ i $\{\mathbf{g}_{yi}\}$ koji sadrže interpolirane sive vrijednosti piksela. Ti se vektori onda zbrajaju u jedinstveni gradijent vektor za svaku sliku

$$\nabla f_i = \{\mathbf{g}_{xi}\} + \{\mathbf{g}_{yi}\} \cdot i. \quad (5.3)$$

Polja senzitivnosti $[\mathbf{S}_p^c]$ za svaku kameru mogu se izračunati preko izraza (4.26). Rezultat će biti matrica dimenzija $n_{point} \times n_{param}$ gdje je n_{point} broj integracijskih točaka, a n_{param} broj parametara koji se kalibriraju. Potom se računa globalna Hessian matrica preko jednadžbi (4.27) i (4.28), a nakon toga i globalni vektor sekundarnih članova preko jednadžbi (4.29), (4.30) i (4.31). Konačno, promjene parametara se računaju preko jednadžbe (4.32), te se nakon normalizacije dodaju parametrima odabranim prije početka iterativnog postupka. Prije ulaska u novi korak iteracije, projekcijske matrice se moraju opet sagraditi na temelju izmjenjenih parametara. Prilikom početka svake iteracije, u konzoli se ispisuju reziduali za sadašnju iteraciju, kondicijski broj hessian matrica, te promjena reziduala između sadašnje i prošle iteracije.

```
==== Iteration: 39 =====
Current residual: 0.0492484

Residual delta: -9.3893e-05
Exporting residuals to .csv

EXPORTING CSV
File name: projectedPHI1
Number of columns: 3

EXPORTING CSV
File name: residualVec
Number of columns: 1

Hessian1 conditioning number is: 149.194
Hessian2 conditioning number is: 178.518

Exporting projection matrices to .csv

EXPORTING CSV
File name: projMatrix1
Number of columns: 4

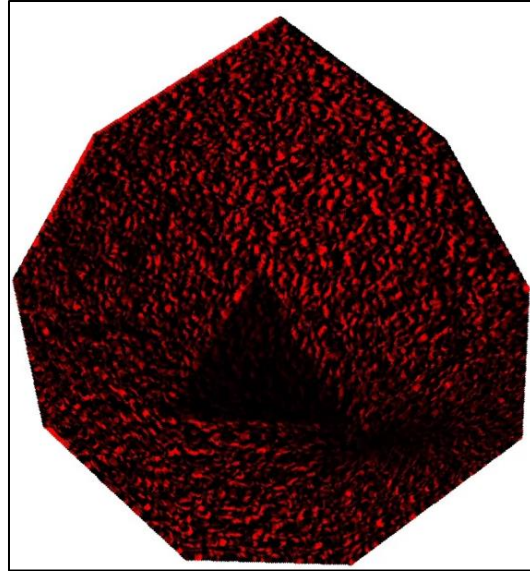
EXPORTING CSV
File name: projMatrix2
Number of columns: 4

dP1: -0.000212774 -1.00364e-05 0.000480156 -0.000167647 -0.000116909 0.000528312
dP2: -0.000299216 3.80217e-05 0.000591683 -0.000184458 -0.000120203 0.000426433
```

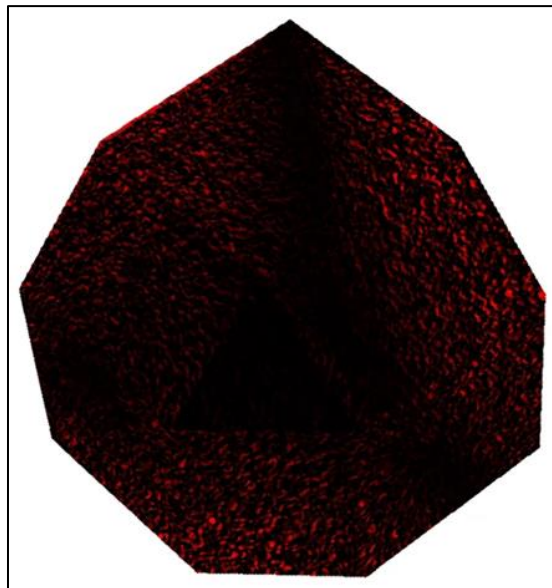
Slika 15. Ispis konzole tokom iteracijskog postupka

Uz ispis informacija vezano uz rezidualne, aplikacija obavijesti korisnika prilikom spremanja reziduala i projekcijskih matrica u *csv* datoteku.

Dok kalibracija prolazi kroz iterativni postupak, zaseban dio koda u Pythonu prati stvaranje novih *csv* datoteka. Nadalje, generiraju se nove slike sa projiciranom mrežom konačnih elemenata i slike na kojima su prikazani reziduali u svakoj Gaussovoj integracijskoj točki. Reziduali su prikazani na mreži točaka desne kamere i vidljivi su na Slikama 18. i 19.



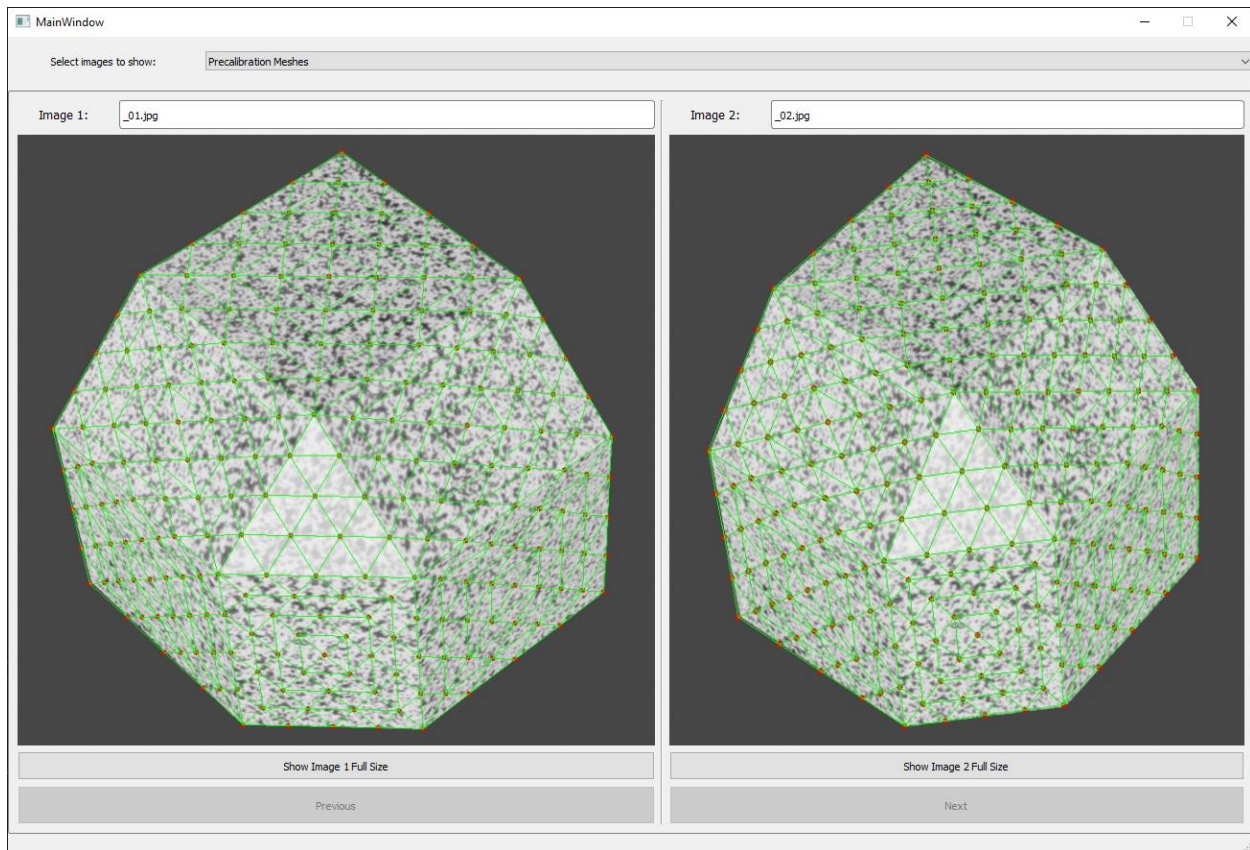
Slika 16. Reziduali u prvoj iteraciji



Slika 17. Reziduali u osmoj iteraciji

Veći reziduali (područja veće razlike sivih vrijednosti među slikama) su označeni crvenom bojom, dok područja podudaranja su označena sa sivom bojom. Reziduali se prikazuju putem OpenCV-a što ne omogućuje prikaz dodatnih alata za analizu poput skale intenziteta koja povezuje nijansu boje sa vrijednošću reziduala. Dalje u razvoju aplikacije planira se ugradnja više interaktivnog pristupa prikazivanja reziduala u obliku toplinske karte sa ugrađenom skalom intenziteta.

Aplikacija unutar sučelja ima ugrađen prozor u kojemu je moguće pogledati slike. Taj prozor se otvara klikom na gumb *View Images* na Slici 10 i prikazan je na Slici 20.



Slika 18. Sučelje za prikaz slika

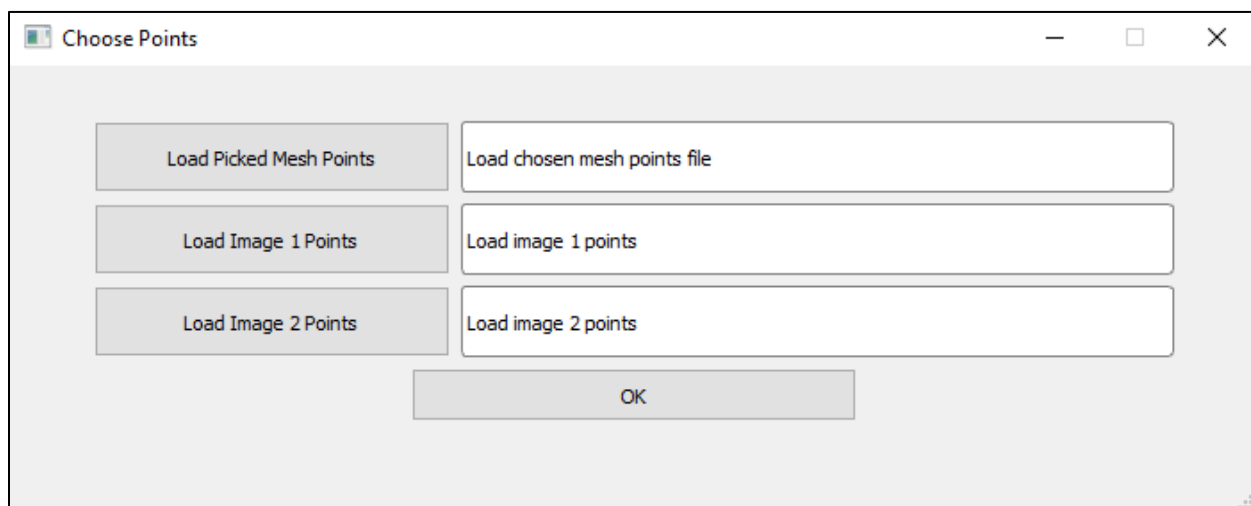
Prozor je podjeljen na dva dijela koja omogućuju prikazivanje slika sa i jedne i druge kamera istovremeno. Sa padajućim izbornikom *Select images to show*, korisnik bira koje slike želi prikazati. Za sada je moguće prikazati projicirane točke na slike nakon postupka predkalibracije, projicirane točke na slike tokom svakog koraka kalibracije i rezidualne tokom svakog koraka kalibracije. Klikom na gumb *Next* i *Previous* mijenja se korak iteracije iz kojega se prikazuju slike. Gumbi *Show Image 1 Full Size* i *Show Image 2 Full Size* otvaraju slike u *Windows* programu za prikaz slika.

5.4. Dodatne funkcionalnosti

Unutar sučelja integrirano je još par dodatnih funkcionalnosti kako bi rad sa aplikacijom bio lakši.

5.4.1. Učitavanje koordinata točaka na slikama

Prva funkcionalnost je mogućnost učitavanja koordinata točaka na slikama što zaobilazi prozore za odabir točaka. Klikom na Load Image Points sa Slike 11. korisniku se otvara novi prozor za odabir točaka prikazan na Slici 22.



Slika 19. Prozor za odabir točaka

Sva tri gumba otvaraju novi prozor za odabir tekstualne datoteke koja sadrži koordinate koje korisnik želi odabrati. Format tekstualne datoteke za točke na mreži konačnih elemenata treba biti sličan onome na Slici 8., bez prvog elementa u liniji koji označava indeks točke. Ostali elementi trebaju biti odvojeni zarezom. Koordinate slika trebaju biti postavljene na isti način, no samo sa dva elementa po redu koji označavaju x i y-koordinate na slici. U programu je još potrebno stisnuti na kućicu Use Custom Points na glavnom dijelu sučelja pod Precalibration options.

5.4.2. Odabir lokacije za ispis podataka o predkalibraciji i brisanje svih podataka

Odabir lokacije za ispis podataka se radi pomoću Export Location gumba na Slici 11. Podatci će se ispisati u *json* datoteku i sadržavati će intrinzične i ekstrinzične parametre za svaku od kamera, te sve informacije potrebne za konstrukciju mreže konačnih elemenata. Klikom na gumb otvara se

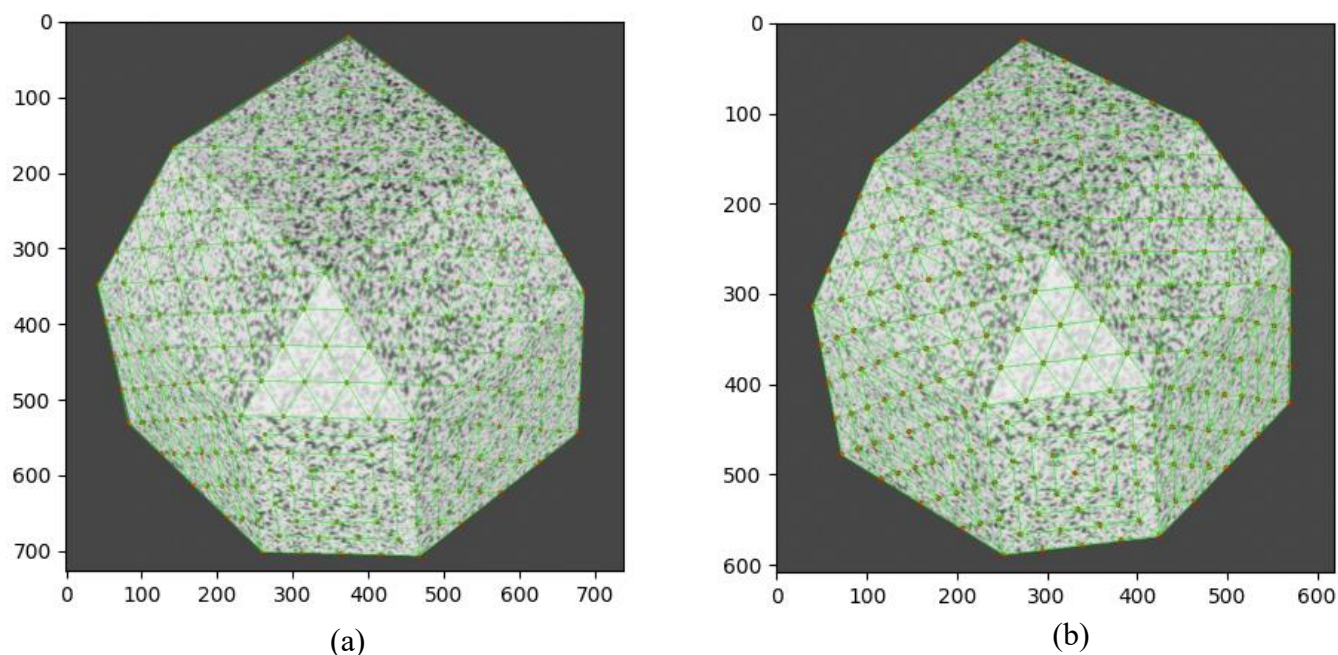
prozor za odabir lokacije na koju će se podatci ispisati. Uz odabir lokacije za ispis, potrebno je stisnuti na kućicu Export precalib data to custom location.

U slučaju ako je potrebno napraviti više uzastopnih kalibracija, dodan je i gumb čišćenja programa od prijašnjih kalibracijskih podataka radi uštede prostora i radi lakšeg prikazivanja novo generiranih slika, bez da korisnik mora više puta stisnuti gumb Next kako bi do njih došao. Taj gumb se nalazi u kalibracijskom dijelu programa unutar dijela Result Options na Slici 11.

6. VALIDACIJA CRO-CAB APLIKACIJE NA UMJETNO GENERIRANIM SLIKAMA

Validacija CroCab aplikacije se provela nad slikama koje su generirane u Blenderu, prikazanim na Slici 10. Umjetno generirane omogućavaju kontrolu svih parametara prilikom generiranja slika što ih čini prikladnim za validaciju kalibracijskog algoritma CroCab aplikacije.

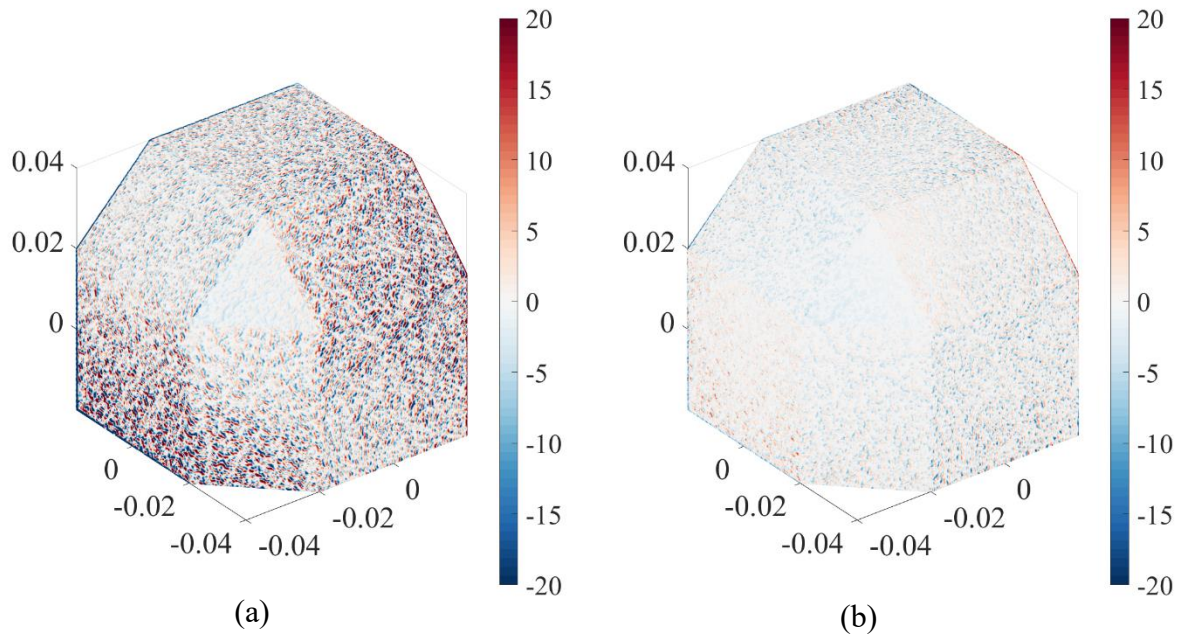
Mreža konačnih elemenata projicirana na slike nakon postupka predkalibracije prikazana je na Slici 20. Na slikama se vidi da su točke odabrane dovoljno precizno uz zadovoljavajuće preklapanje bridova mreže i slika.



Slika 20. Projicirana mreža konačnih elemenata na sliku sa lijeve (a) i desne (b) kamere

Aplikacija CroCab za sad ne podržava interaktivni prikaz reziduala stoga su polja kalibracijskih reziduala prikazani u programskom paketu MATLAB R2023a. Reziduali izraženi u postotku promjene dinamičkog raspona između slika prije i nakon postupka kalibracije prikazani su na Slici 21 pod (a) i (b). Veći dinamički raspon očituje se većim intenzitetom crvene i plave boje. Kritična mjesta na kojima su vidljive najveće vrijednosti kalibracijskih reziduala su rubovi kalibracijskog objekta te donja desna i lijeva strana (Slika 21.a). Nakon postupka kalibracije (Slika 21.b), vidljivo je da je ukupni intenzitet crvene i plave boje znatno smanjen što odgovara manjem dinamičkom

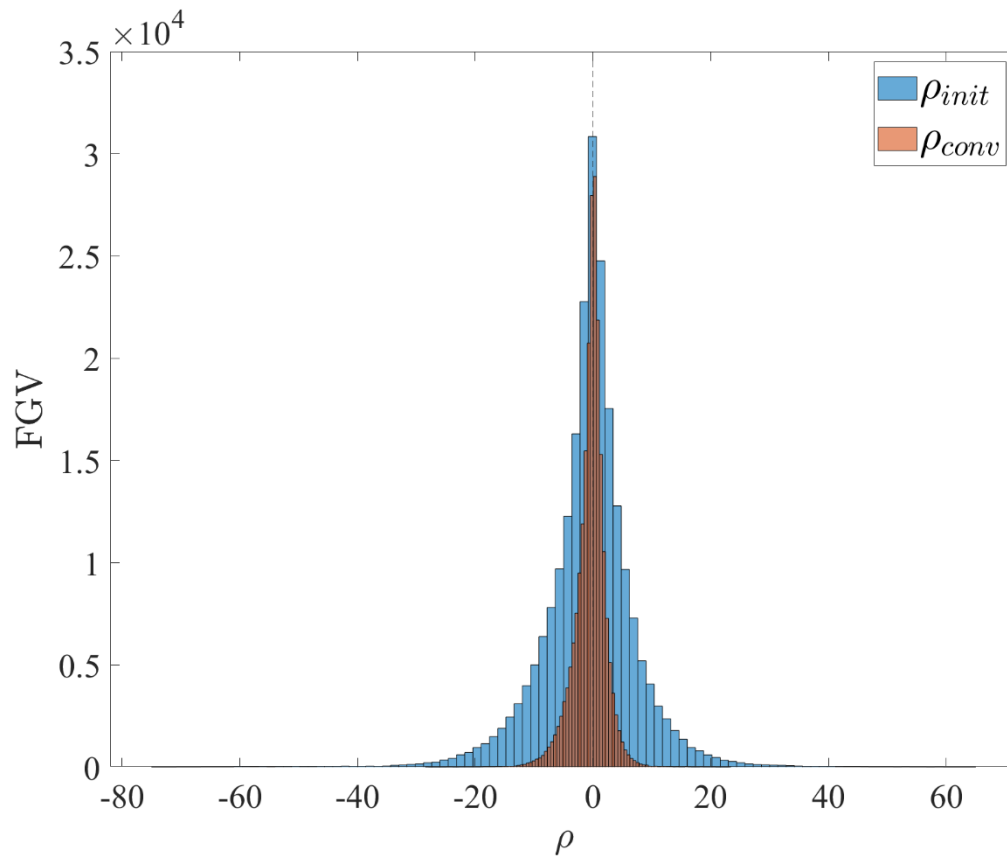
rasponu između slika. To ujedno pokazuje da je postupak kalibracije bio uspješan. Mjesta na kojima su još vidljive izražene vrijednosti reziduala su rubovi što može biti do neefikasnosti u kalibracijskom postupku ili do izbora slika za kalibraciju.



Slika 21. Prikaz reziduala prije (a) i nakon (b) kalibracije

Na temelju slika može se napraviti i histogram funkcije gustoće vjerojatnosti reziduala, prikazan na Slici 22. Na Y-osi se nalazi broj reziduala koji imaju određenu vrijednost, dok je na x-osi vrijednost reziduala izražena preko postotka promjene dinamičkog raspona. Plava boja prikazuje vrijednosti reziduala prije kalibracije (ρ_{init}), dok narančasta boja prikazuje vrijednosti reziduala nakon kalibracije za konvergenciju (ρ_{conv}). Iz priloženih histograma može se zaključiti kako su vrijednosti reziduala nakon provedene kalibracije znatno bliže nultoj vrijednosti u odnosu na početnu iteraciju (sužavanje histograma).

Vidljivo je da je količina reziduala sa vrijednostima blizu nule slična prije i poslije kalibracije, dok se najveći napredak očituje na rubovima histograma gdje je znatno smanjena količina reziduala sa visokim vrijednostima.



Slika 22. Funkcija gustoće vjerojatnosti reziduala

7. ZAKLJUČAK

U sklopu ovog završnog rada prikazan je razvoj aplikacije za globalni pristup kalibraciji stereovizijskog sustava pomoću poznate mreže konačnih elemenata te umjetno generiranih slika. Aplikacija je razvijena pomoću *Python* te *C++* programskih jezika.

U radu je ukratko prikazan povijesni razvoj koncepta stereovizije. Poznati postupci kalibracije stereovizijskih sustava temeljeni na *pinhole* modelu kamera detaljno su opisani. Izvedena je teorijska osnova navedenog modela te objašnjen cilj postupka kalibracije, tj. određivanje projekcijskih matrica kamera. Pomoću ovih matrica moguće je povezati 3D koordinate poznate geometrije objekta sa 2D točkama promatranog objekta zabilježenog na slikama. Kako bi se dobio bolji uvid u rad kalibracijskih algoritama, opisana je epipolarna geometrija stereovizijskih sustava zajedno sa njezinim geometrijskim ograničenjima. Napravljen je pregled različitih kalibracijskih tehnika poput kalibracije pomoću 2D i 3D uzoraka, samokalibracije, kalibracije pomoću iščezavajućih točaka sa naglaskom na korelaciju digitalne slike.

Globalni pristup kalibraciji stereovizijskih sustava, temeljen na metodi konačnih elemenata, detaljno je opisan te je izvedena teorijska osnova. Ovakav pristup temelj je provedbi mjerenja cijelog polja pomaka s globalnim stereo algoritmom korelacije digitalnih slika. Budući da se procesom kalibracije optimiziraju parametri kamera, potrebno je pronaći početno rješenje parametara kamera. U tu svrhu provodi se postupak predkalibracije gdje se odabiru točke na 3D mreži konačnih elemenata te odgovarajuće točke na 2D slikama. Pomoću *Direct Linear Transform* algoritma moguće je uz poznate 2D i 3D koordinate odabranih točaka odraditi početno rješenje projekcijskih matrica. Nadalje, definirana je problematika samog protokola kalibracije te postavljena funkcija cilja koja se rješava iterativnom Newton-Rhapsonovom metodom.

U posljednjim poglavljima opisan je protokol korištenja *CroCab* aplikacije za kalibraciju stereovizijskog sustava te su rezultati validirani nad umjetno generiranim slikama gdje su poznate vrijednosti projekcijskih matrica.

Aplikacija se nadalje može nadograditi mogućnošću kalibracije više od dvije kamere te implementacijom korekcija svijetline i kontrasta. Također, u budućem radu moguće je dodatno

implementirati protokole za vizualizaciju rezultata koji će omogućiti jednostavniju interpretaciju kalibracijskih rezultata.

Literatura

- [1] Computer stereo vision: https://en.wikipedia.org/wiki/Computer_stereo_vision; posjećeno 1. rujna, 2024.
- [2] Binocular vision: https://en.wikipedia.org/wiki/Binocular_vision; posjećeno 1. rujna, 2024
- [3] What's special about horizontal disparity:
<https://jov.arvojournals.org/article.aspx?articleid=2792993>; posjećeno 12. rujna, 2024.
- [4] M. Berny, „High-temperature tests for ceramic matrix composites : from full-field regularised measurements to thermomechanical parameter identification“, Mechanics of materials [physics.class-ph]. Université Paris-Saclay, str. 62-71, 2020.
- [5] Epipolar Geometry and the Fundamental Matrix:
<https://www.robots.ox.ac.uk/~vgg/hzbook/hzbook2/HZepipolar.pdf>; posjećeno 1 rujna, 2024
- [6] Epipolar Geometry: https://en.wikipedia.org/wiki/Epipolar_geometry; posjećeno 1. rujna, 2024.
- [7] Calibration Patterns: <https://www.mathworks.com/help/vision/ug/calibration-patterns.html>; posjećeno 2. rujna, 2024.
- [8] Z. Zhang, „Camera calibration with one-dimensional objects“, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 26, br. 7, str. 892-899, 2004.
- [9] V. Lepetit, F. Pascal, „Monocular Model-Based 3D Tracking of Rigid Objects: A Survey“, Foundations and Trends in Computer Graphics and Vision. 1. (2005).
- [10] J.O. Kim, J. Kim, D. Lee, „Calibrating a Three-Viewpoints Thermal Camera with Few Correspondences“, *J Sign Process Syst* 96, str. 191–202, (2024).
- [11] R. Hartley, A. Zisserman, „Multiple View Geometry in Computer Vision“, Cambridge University Press, 2003.
- [12] B. Triggs, P. McLauchlan, R. Hartley, A. Fitzgibbon, „Bundle Adjustment - A Modern Synthesis“, Vision Algorithms: Theory and Practice, str. 298-372, 2000.

- [13] R. Cipolla, T. Drummond, D. Robertson, „Camera calibration from vanishing points in images of architectural scenes“, British Machine Vision Conference (BMVC), str. 382-391, 1999.
- [14] M. Antunes, J. P. Barreto, „A global approach for the detection of vanishing points and mutually orthogonal vanishing directions“, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), str. 1336-1343, 2013.
- [15] Non-uniform rational B-spline: https://en.wikipedia.org/wiki/Non-uniform_rational_B-spline; posjećeno 13. rujna, 2024.
- [16] J. E. Dufour, B. Beaubier, F. Hild, *et al.*, „CAD-based Displacement Measurements with Stereo-DIC“, *Exp Mech* 55, str. 1657–1668, 2015.
- [17] V. Lepetit, F. Moreno-Noguer, P. Fua, „EPnP: An Accurate $O(n)$ Solution to the PnP Problem“, *Int J Comput Vis* 81, str. 155–166, 2009.
- [18] Direct Linear Transform: <https://www.baeldung.com/cs/direct-linear-transform>; posjećeno 15. prosinca, 2023.
- [19] A. Zaplatić, Z. Tomičević, X. Chang, I. Skozrit, S. Roux, F. Hild, „Hybrid infrared-visible multiview correlation to study damage in a woven composite complex-shaped specimen“, *Coupled Systems Mechanics*, vol. 12, no. 5, str. 445-459, 2023.