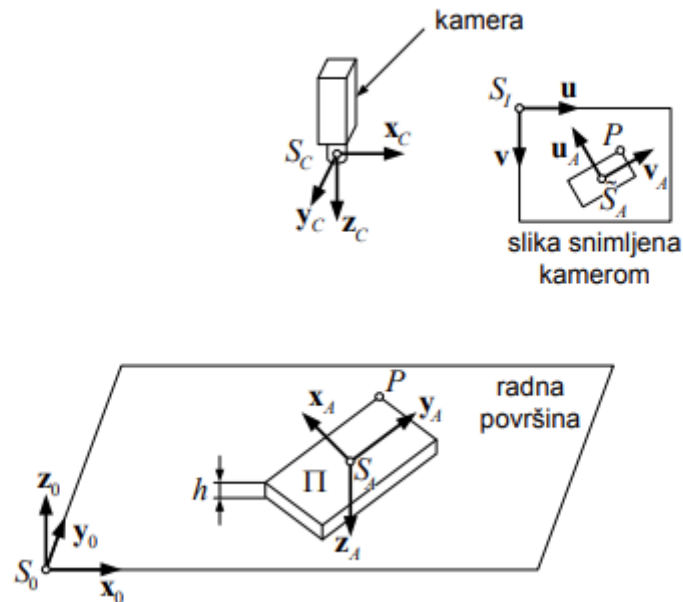


## Vježba 3: Houghova transformacija

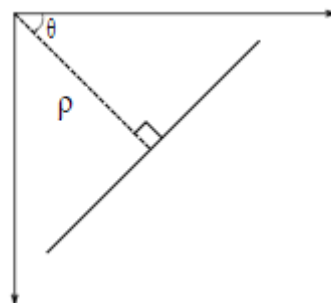
Kao cilj ove vježbe potrebno je prepoznati pravac na slici uslikanoj s kalibriranom kamerom koristeći Houghovu transformaciju. Iz parametara pravca prepoznatog na slici  $\theta$  i  $\rho$  ( $\theta, \rho$ ) potrebno je naći stvarnu udaljenost i kut pravca.

### Opće postavke:



Slika 1 Objekt na radnoj površini iznad kojeg je montirana kamera.

Pošto smo kameru postavili na takav način da z-osi koordinatnog sustava predmeta ( $S_0$ ) i koordinatnog sustava kamere ( $S_c$ ) budu istog smjera možemo reći da koordinatni sustav predmeta leži u istoj ravnini kao i koordinatni sustav kamere te tako efektivno ukloniti treću dimenziju pri programiranju potrebnih funkcija.



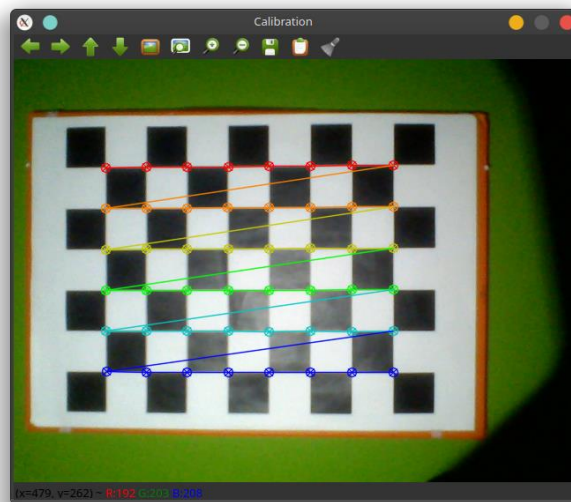
Slika 2 Prikaz podataka dobivenih Houghovom transformacijom za pravce

### Zadatak:

Potrebno je izračunati Houghovu transformaciju najdominantnijeg pravca danog predmeta te odrediti koliko je taj pravac udaljen od ishodišta koordinatnog sustava  $S_0$  u milimetrima te koja mu je rotacija.

**Postupak:****1. Kalibracija kamere :**

Kako bi mogli pouzdano pronaći udaljenost pravca i orijentaciju pravca s obzirom na koordinatni sustav u kojem se nalazi predmet, potrebno je prvo provesti kalibraciju kamere.

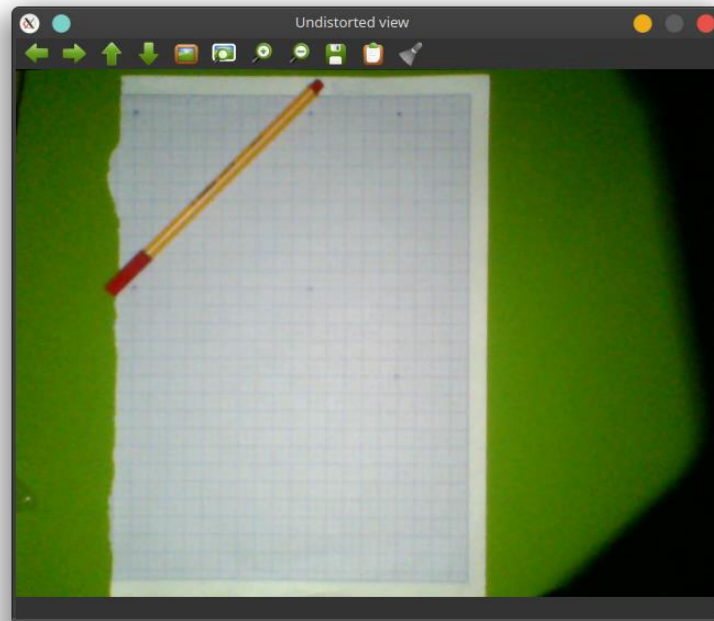


*Slika 3* Prikaz prepoznatih točaka na kalibracijskom panelu

Svaka kamera stvara određene distorzije na slici koje je potrebno ukloniti kako bi dobili vjerodostojne podatke o udaljenosti i rotaciji pravca u koordinatnom sustavu predmeta. Kako bi ispravili distorzije slike koristimo crno-bijelu šahovnicu (ovakav panel koristi se zbog visokog kontrasta i mogućnosti lakog prepoznavanja vrhova pojedinih kvadrata) i funkcije OpenCV biblioteke. Određivanjem broja kvadrata po visini i širini šahovnice te njihove veličine možemo dobiti potrebne parametre za ispravljanje distortirane slike. Kada provedemo algoritam vidljiv u priloženoj datoteci CameraCalib.cpp dobijemo podatke o intrinzičnim parametrima kamere i koeficijente distorzije koje spremamo u .xml datoteku kako bi ih kasnije mogli iskoristiti za ispravljanje slike.

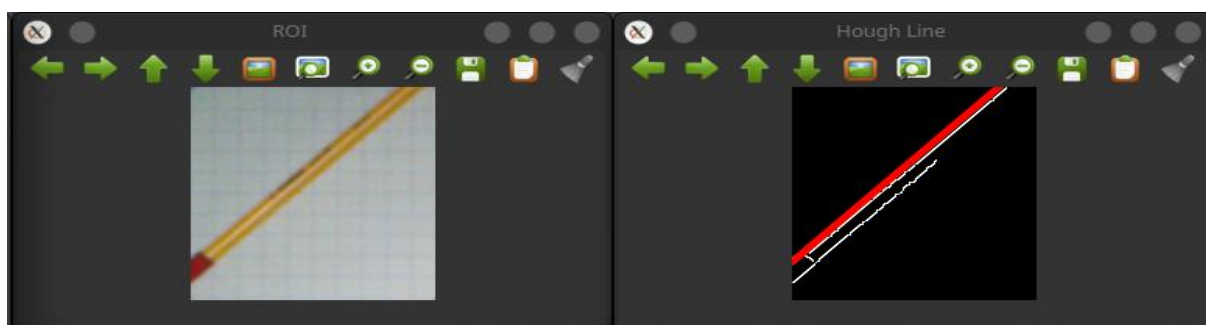
**2. Pronalazak pravca pomoću Houghove transformacije za pravce:**

Nakon što smo saznali parametre kamere možemo početi s računanjem Hough transformacije za pravce. Pokretanjem programa vidimo da se otvara prozor sa slikom kamere unutar kojeg možemo postaviti predmet čiji najdominantniji pravac ćemo kasnije pokušati prepoznati. U svrhu testiranja programa odabrana je olovka (zbog jasno definiranog dominantnog pravca) postavljena na milimetarski papir (zbog lakog postavljanja koordinatnog sustava predmeta i mjerenja kuta i udaljenosti s obzirom na ishodište).



Slika 4 Traženi predmet na milimetarskom papiru

Prvi korak ovog postupka svakako bi bio uklanjanje distorzija slike pomoću prethodno izračunatih parametara. Ovo možemo vrlo jednostavno postići koristeći funkciju *undistort* sadržanu u OpenCV biblioteci. Pritiskom na tipku 'c' korisniku omogućavamo odabir 4 točke za pravokutno područje interesa *ROI* (engl. *Region of interest*). Nakon odabira željenih točaka, odabrano područje izdvajamo u posebnu sliku te unutar tog područja provodimo daljnji postupak. Potrebno je prvo originalnu sliku u području interesa pretvoriti u sliku u sivim tonovima te tada nad njom provest prepoznavanje rubova. U ovu svrhu korištena je funkcija *Canny*. Nakon pronalaska svih rubova predmeta postavljenog u željeno područje nad istima možemo provesti traženje pravaca pomoću *HoughLines* metode koja vraća popis svih pronađenih pravaca kao popis njihovih parametara  $\theta$  i  $\rho$  ( $\theta, \rho$ ). Kako mi tražimo samo najdominantniji pravac na danom predmetu, potreban nam je samo prvi pravac iz dobivenog popisa parametara. Izdvojeni rubovi i pronađeni dominantni pravac vidljivi su na slici 5 u prozoru "Hough Line".



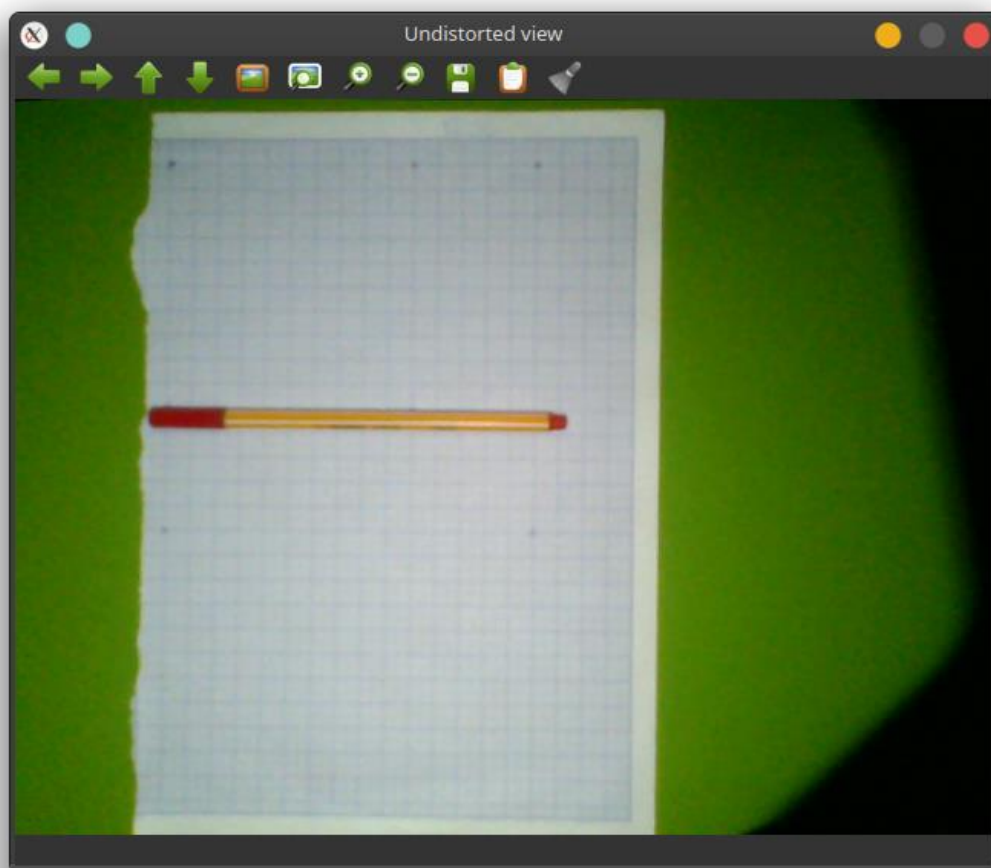
Slika 5 Odabrano područje interesa (lijevo) i detektirani rubovi s istaknutim najdominantnijim pravcem (desno)

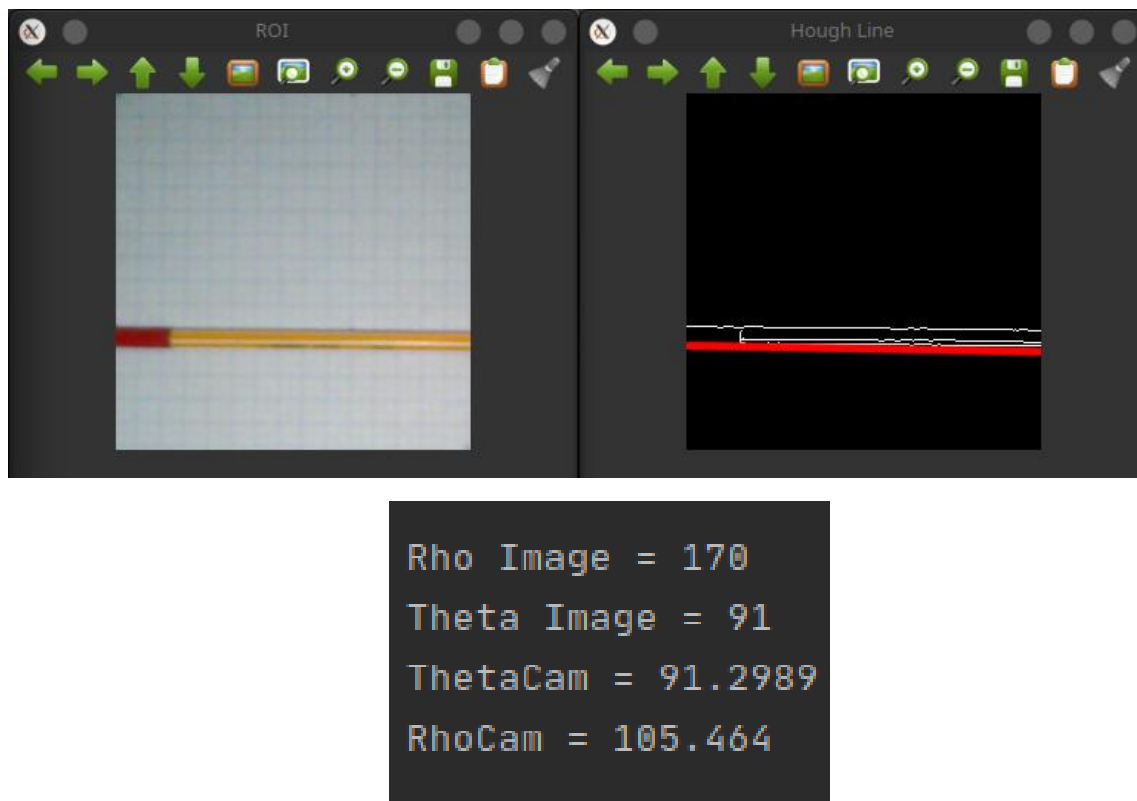
### 3. Prelazak iz sustava slike u sustav objekta:

Kako bi odredili stvarnu udaljenost pravca (predmeta) od ishodišta vlastitog koordinatnog sustava potrebno je iz 2D sustava slike pravac određen parametrima  $\theta$  i  $\rho$  preći u 3D sustav i pravac određen parametrima  $\theta'$  i  $\rho'$ . Pomoću jednadžbi zadanih u prilogu vježbe 3 možemo vrlo jednostavno preći iz sustava slike u stvarni sustav.

Provjerom parametara dobivenih zadanim jednadžbama i usporedbom istih sa stvarnom udaljenosti pravca od ishodišta možemo vidjeti da smo kalibraciju i pronalazak parametara pravca odradili kvalitetno te da nemamo velikih odstupanja.

Za dani primjer, stvarna udaljenost pravca od ishodišta bila je 58mm, a kut  $45^\circ$ . Parametri koje smo dobili računanjem pravca unutar područja interesa bili su 93 piksela udaljenosti pravca od ishodišta te kut od  $45^\circ$ , a nakon provođenja preračuna u stvarni koordinatni sustav objekta moglo se vidjeti kako je navedena udaljenost pravca od ishodišta 59.9296mm, a kut  $45.4358^\circ$ . Kako bi daljnje pokazali rad i preciznost algoritma iz datoteke HoughTransform.cpp u nastavku su ponuđene fotografije još jednog od mogućih slučajeva postavljanja predmeta.





*Slika 6* Predmet postavljen unutar okvira 150x150mm na udaljenost 100mm od ishodišta pod kutem od 90°

Stvarna udaljenost ovog predmeta od ishodišta bila je 100mm, a kut 90°. No kako je program kao dominantni pravac prepoznao donji rub olovke, u krajnji rezultat programa uračunata je debljina olovke te je tada udaljenost pravca od ishodišta bila 106mm.

## Izvori:

**Slika 1** – Predavanje profesora Roberta Cupeca, **ROBOTSKI VID, 4. Raspoznavanje 2D objekata**, Osijek, 07. 11. 2011.

**Slika 2** - [https://docs.opencv.org/3.4/d3/de6/tutorial\\_js\\_houghlines.html](https://docs.opencv.org/3.4/d3/de6/tutorial_js_houghlines.html) (Preuzeto 16. 01. 2021.)