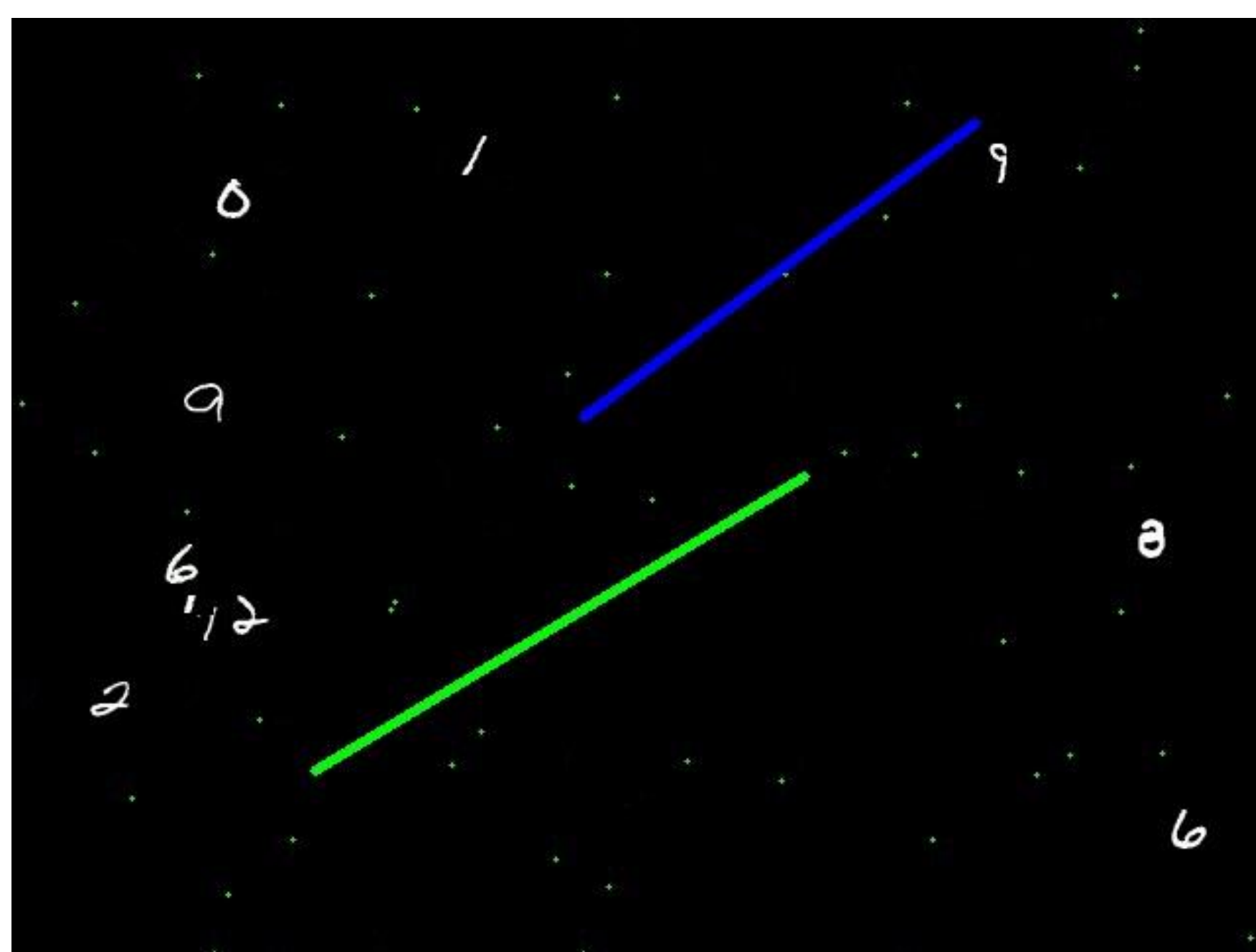


# SOFT KOMPJUTING 2017 - PREDEFINISANI PROJEKAT ZA OCENU 9/10

Ivan Jančić, ra157-2014

## UVOD

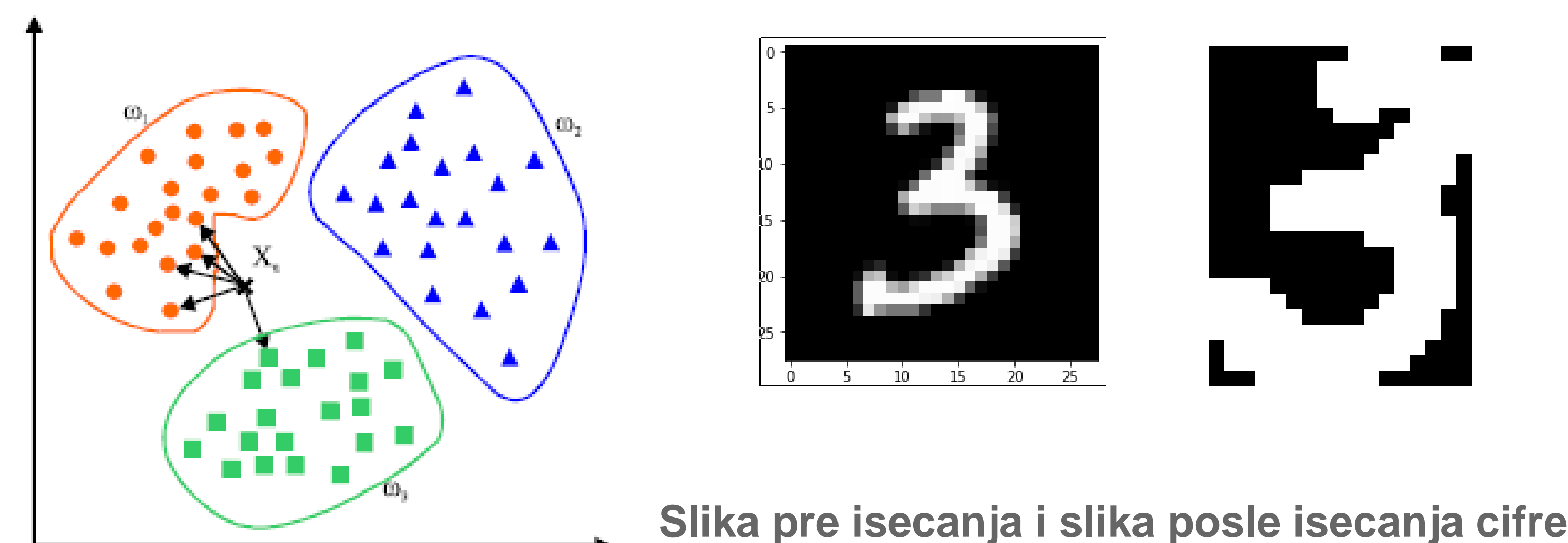
U okviru projekta se nalazi 10 video snimaka. Na svakom snimku se nalaze dve statične linije – jedna plava i jedna zelena, kao i brojevi koji se kreću u toku snimka. U pozadini se takođe nalaze šumovi koje je potrebno ukloniti radi preciznijeg izdvajanja regiona od interesa. Cilj projekta je prepoznati i sabrati brojeve koji pređu preko plave linije, a oduzeti one koji pređu preko zelene. Za izradu projekta korišćen je Python programski jezik zajedno sa numpy, OpenCV, scikit bibliotekama i MNIST datasetom.



## OBUČAVANJE KNN KLASIFIKACIONOG ALGORITMA NA MNIST DATASET-U

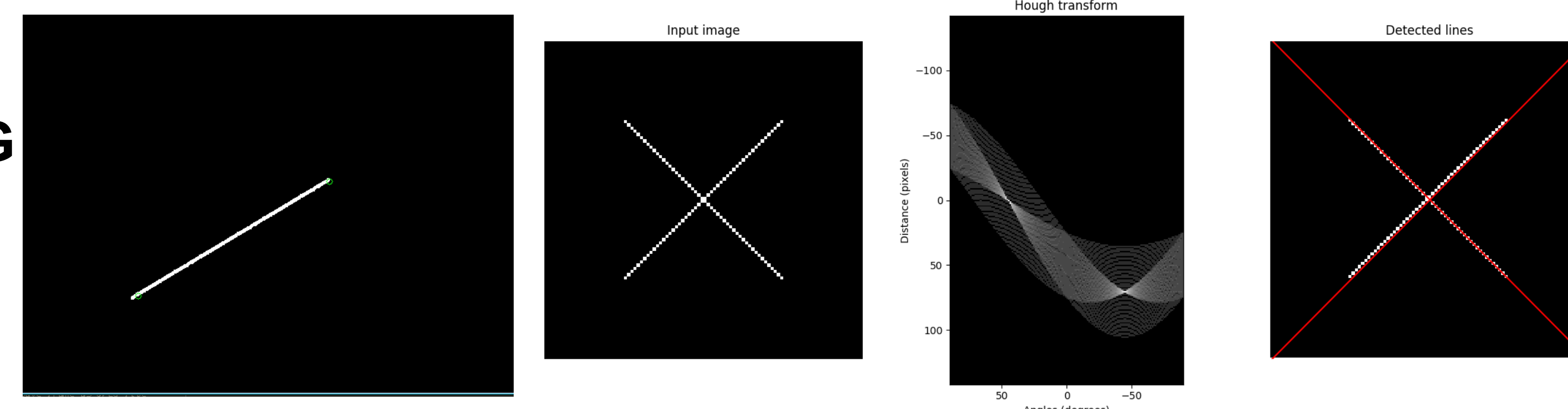
K nearest neighbor je algoritam koji ćemo koristiti za klasifikaciju cifara. Za obuku se koristi 70.000 cifara iz MNIST dataset-a. Svaku cifru iz dataseta obrađujemo tako što primenimo dilataciju i eroziju kako bismo izdvojili broj što bolje za funkciju iz biblioteke cv2 findContours(), zatim nad nađenom konturom pozovemo metodu boundingRect() koja će ga uokviriti što ćemo iskoristiti da isečemo broj iz njegovog okruženja. Isečenom broju promenimo dimenzije na 28x28 i vraćamo ga u niz za obuku.

Proveravamo kakve rezultate daje k(broj komsija) za vrednosti k u rasponu od 1-30, i uzimamo najbolji rezultat za obuku (uglavnom to bude k=1).



## DETEKCIJA LINIJA HOUGH-OVOM TRANSFORMACIJOM

Na prvom frejmu se vide samo 2 linije, pa se metodi za detekciju linija salje slika prvog frejma nakon izvršenog tresholda za zelenu, pa zatim za plavu boju. Nad takvom slikom pozivamo metodu Canny koja algoritmom Canny Edge pronalazi ivice objekata sa slike i predstavlja ih u obliku binarne slike. Zatim se takva slika prosleđuje funkciji HoughLinesP koja vraća niz početnih i krajnjih tačaka onih ivica koje čine pravu liniju. U datom nizu pronadjemo minimalnu i maksimalnu krajnu i početnu tačku na osnovu kojih ćemo odrediti datu pravu. Koristimo funkciju pnt2line koja nam daje informaciju o tome koliko je tačka koja se kreće udaljena od najbliže tačke na liniji.



Slika nadjene zelene linije

Primer pronalaženja linija Hough transformacijom

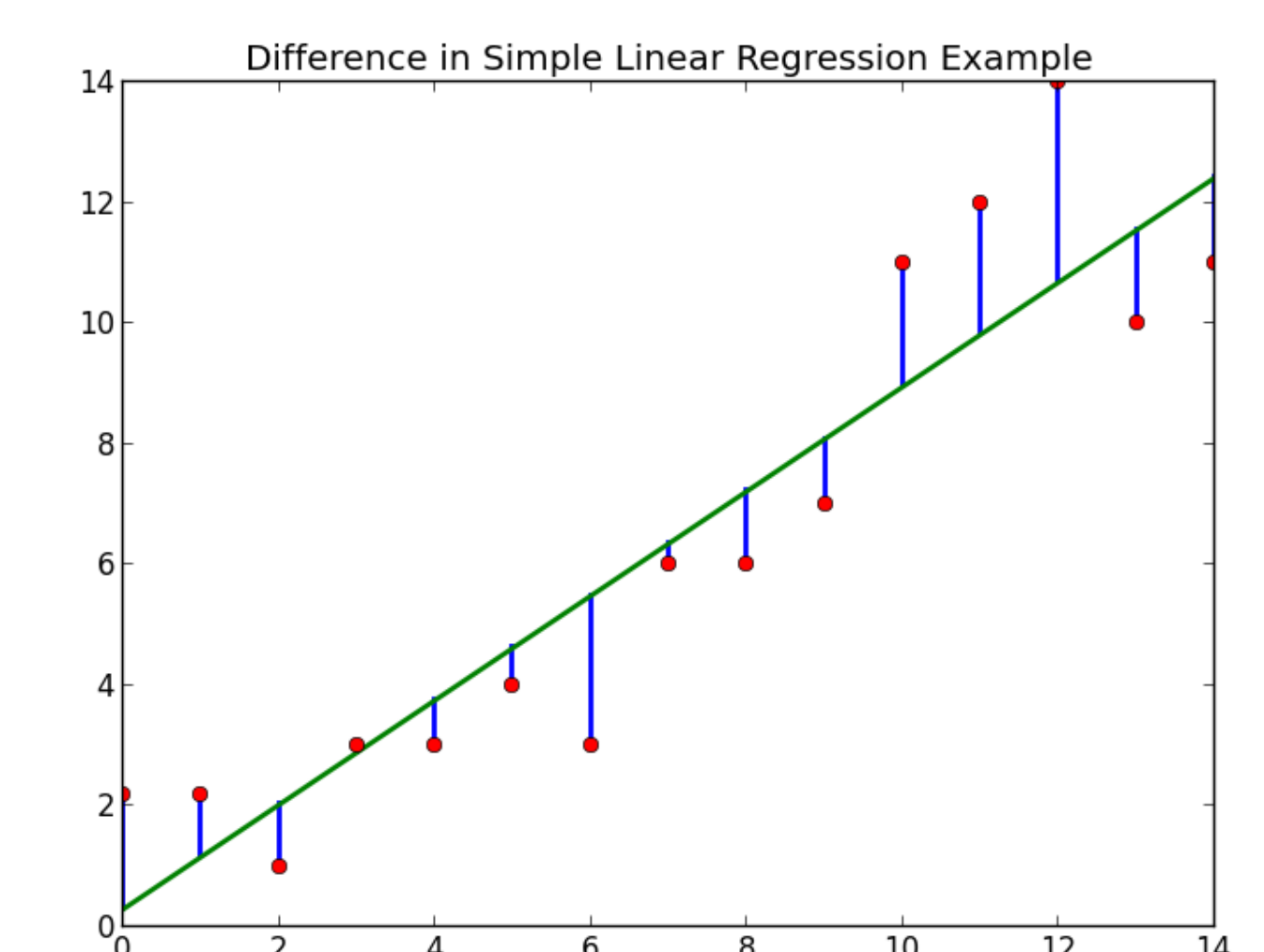
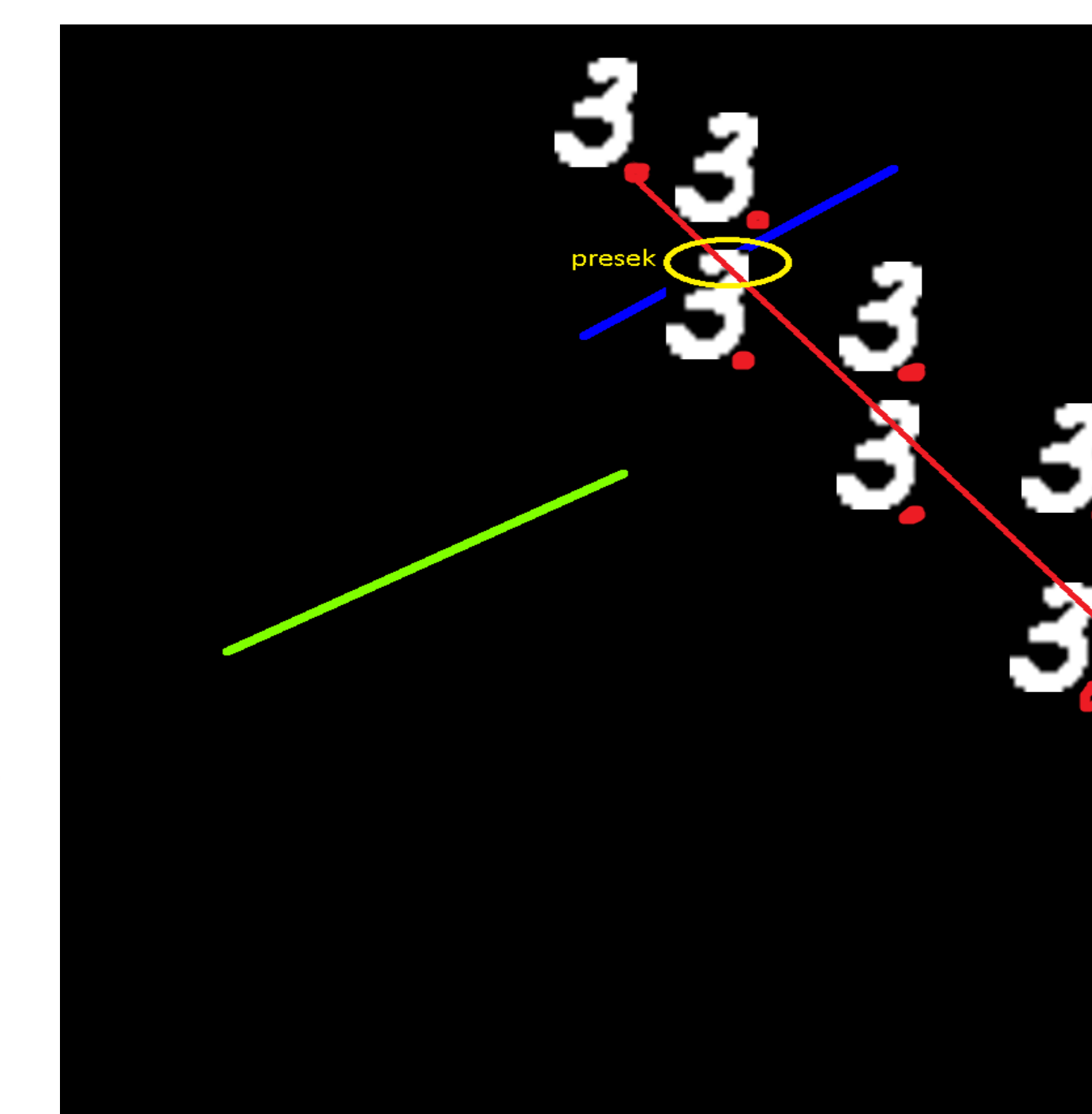
## DETEKCIJA CIFARA I OBRADA FREJMA

Iz svakog videa redom učitalamo frejmove. Za svaki frejm se vrši otklanjanje šumova i linija korišćenjem funkcije cv2.inRange(). Zatim se izvrši dilatacija da bi se bele površine brojeva povećale i kasnije bili pogodniji za detekciju. Funkcija ndimage.label() obeležava regione od interesa na slici, i zatim koristimo funkciju ndimage.find\_Objects() kako bismo izdvojili sve nadjene regione(brojeve) u jedan niz. Nakon toga idemo redom kroz niz svih brojeva i isecamo brojeve sa početne slike na osnovu koordinata nadjenih regiona. Isečenom broju promenimo dimenzije na 28x28 i transformišemo ga u oblik jednak brojevima iz dataseta. Izvršimo predikciju broja i povratnu vrednost saberemo ili oduzmemo od ukupne sume.

## DETEKCIJA PRELASKA CIFRE PREKO LINIJE

Radjeno je na dva načina:

- 1) Prvi način podrazumeva računanje distance centra broja od njemu najbliže tačke na jednoj od 2 linije. Ako je distanca dovoljno mala to znači da je broj prešao preko linije i sabira se/oduzima se od ukupne sume. Takođe postoji flag koji govori da li je broj prešao već jednom datu liniju koji sprečava da se broj više puta računa za istu liniju.
- 2) Koristimo linearnu regresiju. Za broj računamo razliku između trenutnog frejma i frejma kada se on poslednji put pojavio. Ako je razlika veća od nekog praga onda to znači da je broj izašao sa videa. U tom trenutku za taj broj korišćenjem linearne regresije nadjemo pravu koja predstavlja njegov put kroz video. Ako ta prava seče liniju to znači da je broj prošao kroz datu liniju. Postoji zaštita kojom proveravamo da li je y koordinata prvog pojavljivanja broja manja od y koordinate tačke preseka broja sa plavom/zelenom linijom. Tako proveravamo da li je broj inicijalno bio ispod linije ili ne.



Linearna regresija primer

## ZAKLJUČAK

Rezultati dobijeni na prvi način pokazuju prepoznavanje cifara 33,73% dok se za drugi način dobija rezultat 48,19% što je i očekivano iz razloga što prvi način ne rešava problem preklapajućih cifara na liniji. S obzirom da se u drugom načinu cifre sabiraju/oduzimaju tek kada izađu iz frejma i to tako što se proveruje da li su sekle jednu od dve linije, problem preklapajućih cifara nema nikakav efekat na krajnji rezultat.