

# Predefinisani projekat za ocenu 7 i 8

Autor: Marko Kljajić RA105/2013

Fakultet Tehničkih Nauka Univerzitet u Novom Sadu

## O samom problemu

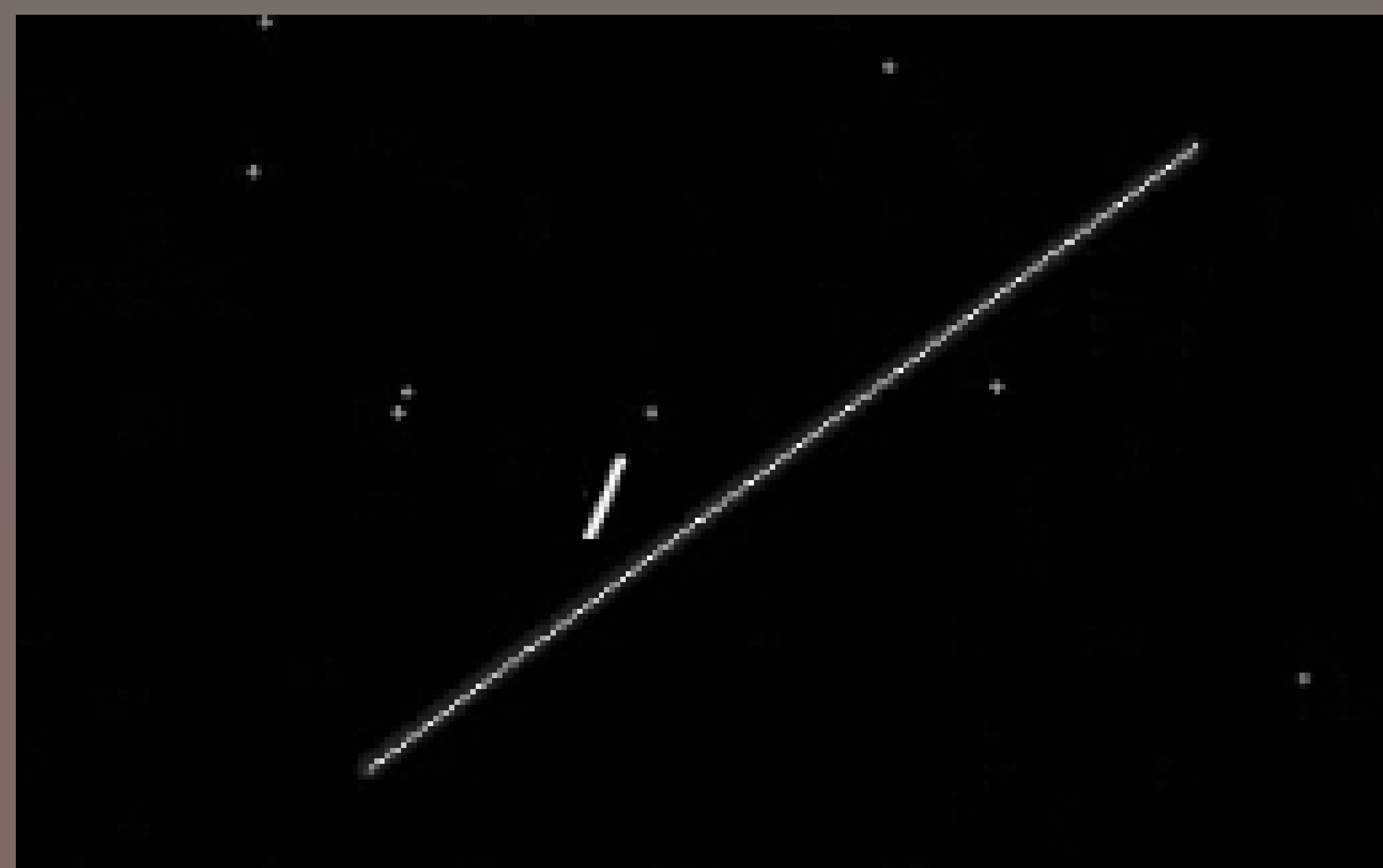
Snimak sadrži jednu pokretnu liniju koja je uvek iste boje. Potrebno je pomoću Hough-ove transformacije detektovati liniju. Na snimku se nalaze i cifre koje se kreću za slučajni broj koraka. Nakon detekcije linije potrebno je detektovati i sve cifre koje prolaze ispod linije. Prepoznate cifre potrebno je sumirati. Postići tačnost od bar 90%.

## Metod rešavanja problema

Za rešavanje navedenog problema korišćen je python programski jezik uz oslonac na NumPy, OpenCV, Scikit-image biblioteke kao i MNIST data set. Korišćenje OpenCV funkcija `inRange` i `HoughLinesP` uz prethodno prebacivanje slike u nijansu sive i primene trešholda su dovele do uspešne detekcije linije. Ponovno primenjivanje trešholda uz morfološke operacije dovodi do detekcije regiona brojeva.

## Rezultati

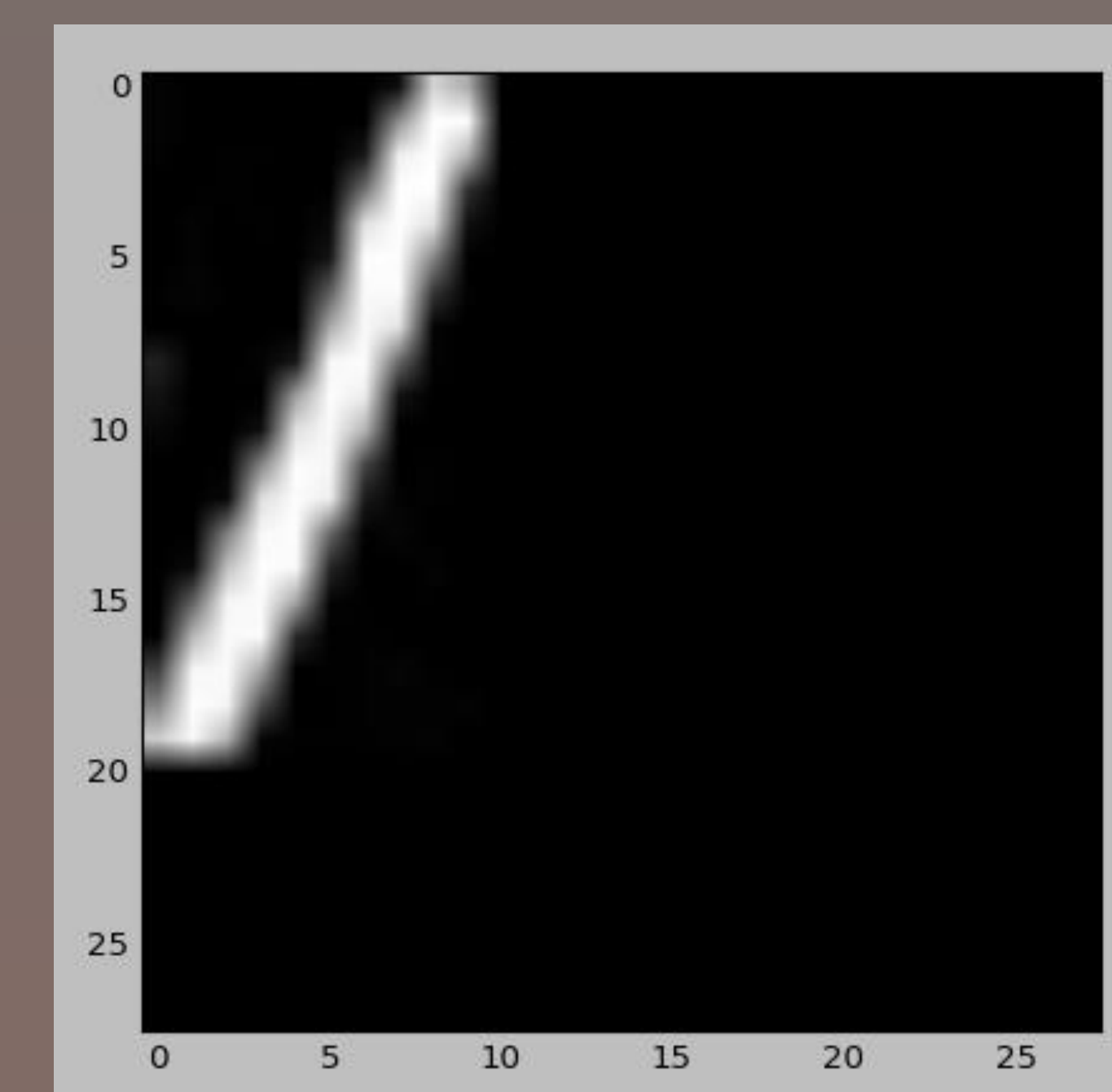
Kao što je već ranije navedeno za uspešnu detekciju linije potrebno je prvo sliku prebaciti u nijansu sive. Nakon toga primenjen je globalni trešhold da bi se izdvojile bitne informacije od pozadine. Nakon toga je urađena erozija koja je eliminisala informacije koje su uspele da prođu globalni trešhold a nisu nam od interesa. Na dobijenu sliku primenjivana je funkcija `skeletonize` iz `scikit-image` biblioteke da bi se dobio samo skelet linije koji nam je potreban za uspešno prepoznavanje linije. Dobijeni skelet se prosleđivao `HoughLinesP` funkciji OpenCV biblioteke uz parametre  $\rho$ ,  $\theta$ , definisani prag iznad čega će se nešto smatrati linijom, minimalnu dužinu linije i maksimalni dozvoljeni razmak između segmenata koji će se smatrati istom linijom. Nakon primene gore navedenog dobija se detektovana linija kao na sledećoj slici.



Slika 1 - Detektovana linija iz snimka

Radi uspešne detekcije potrebno je prethodno `mnist` data set pripremiti odnosno za svaku sliku odrediti region gde se nalazi cifra i premestiti cifru u gornji levi ugao slike.

Za uspešnu detekciju cifara potrebno je na sliku nijanse sive primeniti novi trešhold koji će izdvojiti bitne informacije o brojevima od pozadine. Na dobijenu sliku primeniti zatvaranje odnosno potrebno je eliminisati sitne otvore da jedan isti broj ne bi bio predstavljen u više regiona. Nakon zatvaranja sliku je potrebno predati Scikit-image funkciji `label` potom `regionprops` koja će odrediti regione gde se sve nalaze brojevi. Za dobijene brojeve potrebno je utvrditi da li se seku sa linijom. Ukoliko se seku potrebno je sa sive slike uz pomoć podataka o regionu izdvojiti sliku koja će sadržati broj i predati dobijenu sliku `knn` algoritmu radi prepoznavanja broja. Nakon toga se vrši provera da li je to novo detektovani broj ili je on ranije detektovan kao da seče liniju. Novo detektovani broj se zatim dodaje u listu koja će posle biti sumirana radi prikaza rezultata.



Slika 2 – Detektovan broj koji prolazi ispod linije

## Zaključak

Korišćenjem python programskog jezika i navedenih biblioteka vrlo jednostavno se dolazi do detekcije jedne ili više linija na slici što može biti korisno u sportu za detekciju pogodaka, auto industriji za održavanje automobila unutar trake, detekciju prekoračenja brzine itd. Detekcija brojeva takođe ima raznovrsnu primenu moguće ju je iskoristiti gde god je potrebna korisnička interakcija sa softverom. Radi smanjenja vremena potrebnog za izračunavanje sume detekcija brojeva u snimku se vrši na svaki drugi frejm.

## Literatura

[NumPy](#) – NumPy korisnička dokumentacija.  
[OpenCV](#) – OpenCV korisnička dokumentacija sa primerima.  
[Vežbe](#) – Github repozitorijum sa vežbi.  
[Scikit-image](#) – Scikit korisnička dokumentacija.  
[MNIST](#) – MNIST data set.