

Predefinisani projekat – soft kompjuting

Dejan Preradovic

Valentina Vlaški-Gvozdenov

Nataša Milanov

Univerzitet u Novom Sadu, Fakultet Tehnickih Nauka, Novi Sad

Uvod

Tematika ovog projekta obuhvata detekciju i prepoznavanje pokretnih cifara koje prelaze preko linija u datim video zapisima. Video zapis poseduje dve statičke linije, pri čemu je neophodno sabrati sve brojeve koji prelaze prvu liniju, a oduzeti one koji prelaze drugu. Potrebno je i ukloniti šumove koji ometaju detekciju regiona od interesa. U daljem izlaganju biće predstavljeni algoritmi i tehnologije pomoću kojih se postiže rešavanje datog problema.

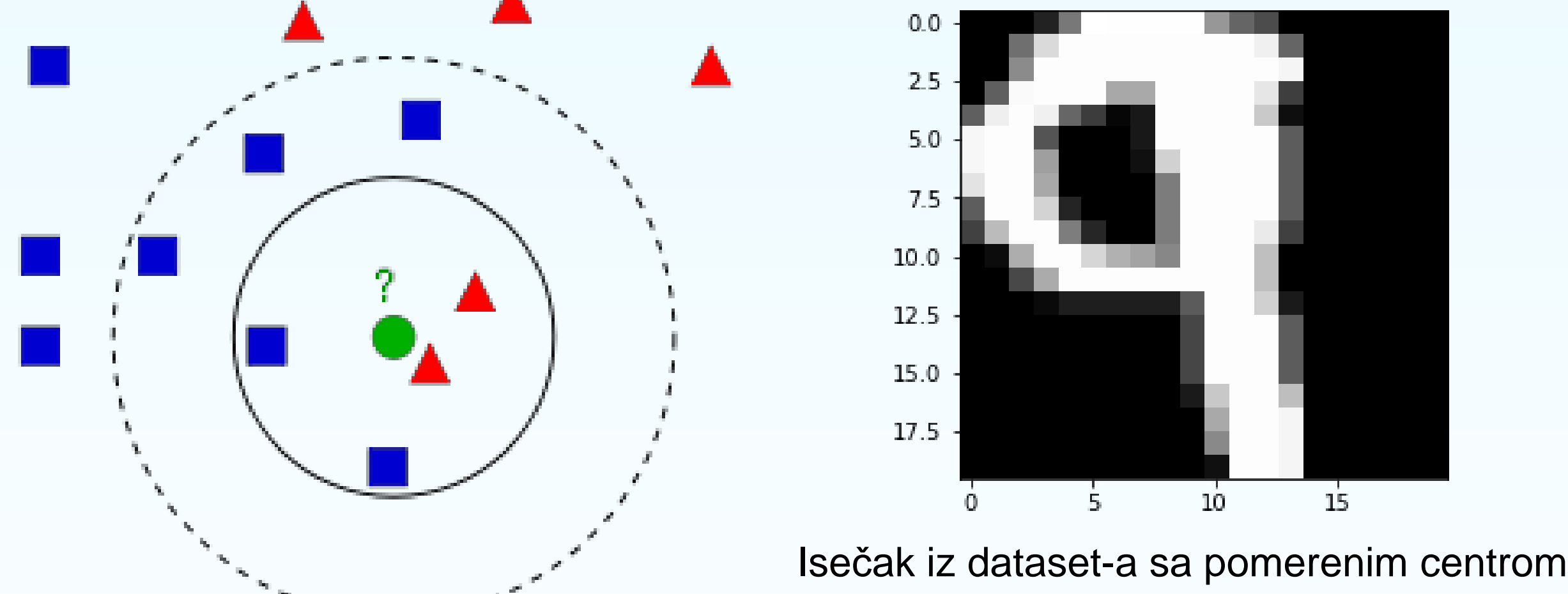
Obučavanje KNN algoritma

Za postizanje što boljih rezultata pri prepoznavanju cifara korišćen je k-Nearest Neighbour algoritam za klasifikaciju. Ovim algoritmom se vrši pretraga najboljeg poklapanja date cifre sa ostalima iz prostora cifara dataset-a.

Cifre iz dataset-a za obuku, kao i slike cifrara se isecaju sa svakog frejma pozicionirane su na sredini isečka, što čini algoritam nepreciznim ukoliko se desi da je centar cifre za prepoznavanje pomeren. Da bismo izbegli ovaj problem, cifre iz dataset-a postavljamo u gornji-levi ugao isečaka.

Potrebno je inicijalizovati k-Nearest Neighbour algoritam i proslediti mu dataset i odgovarajuće labela na obučavanje.

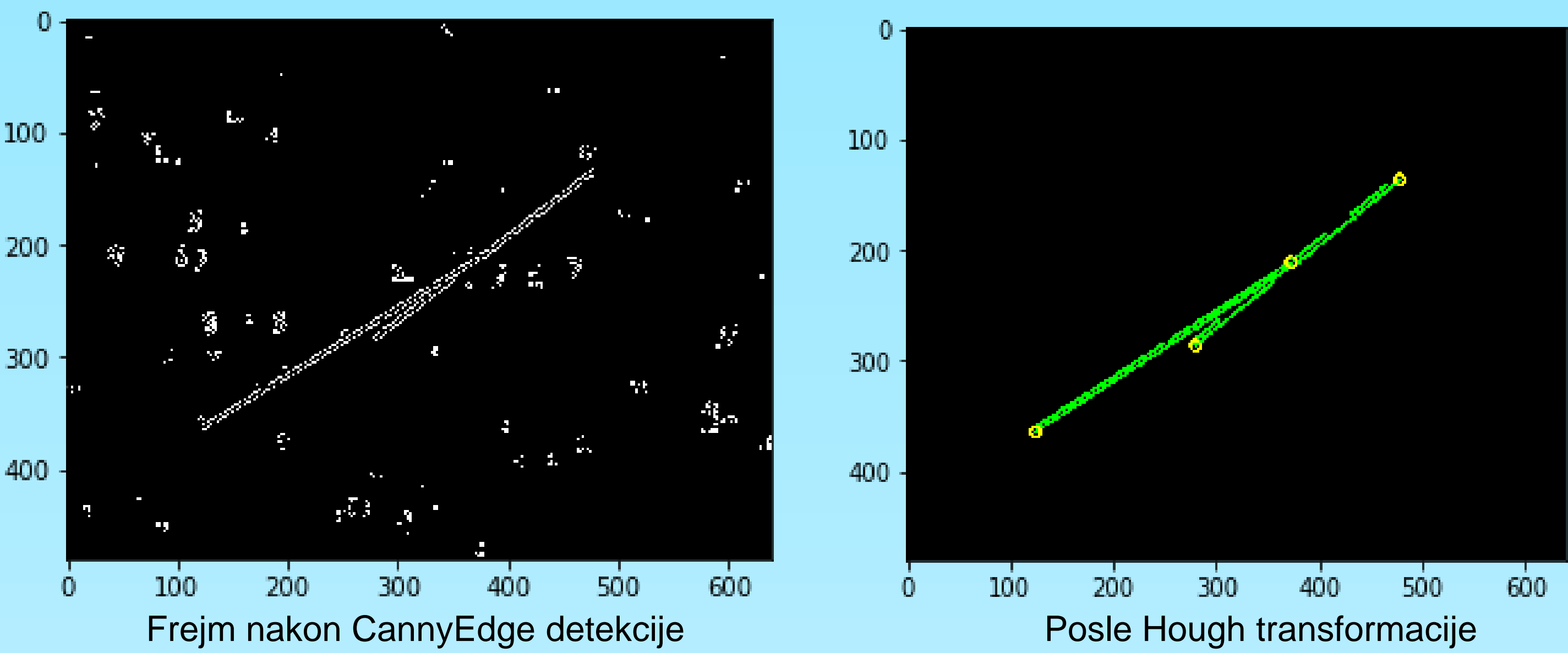
Prvobitno je upotrebljavana neuronska mreža, ali ista nije davala zadovoljavajuće rezultate.



Detekcija linija

Naredni korak obuhvata detekciju linija koja se obavlja na sledeći način:

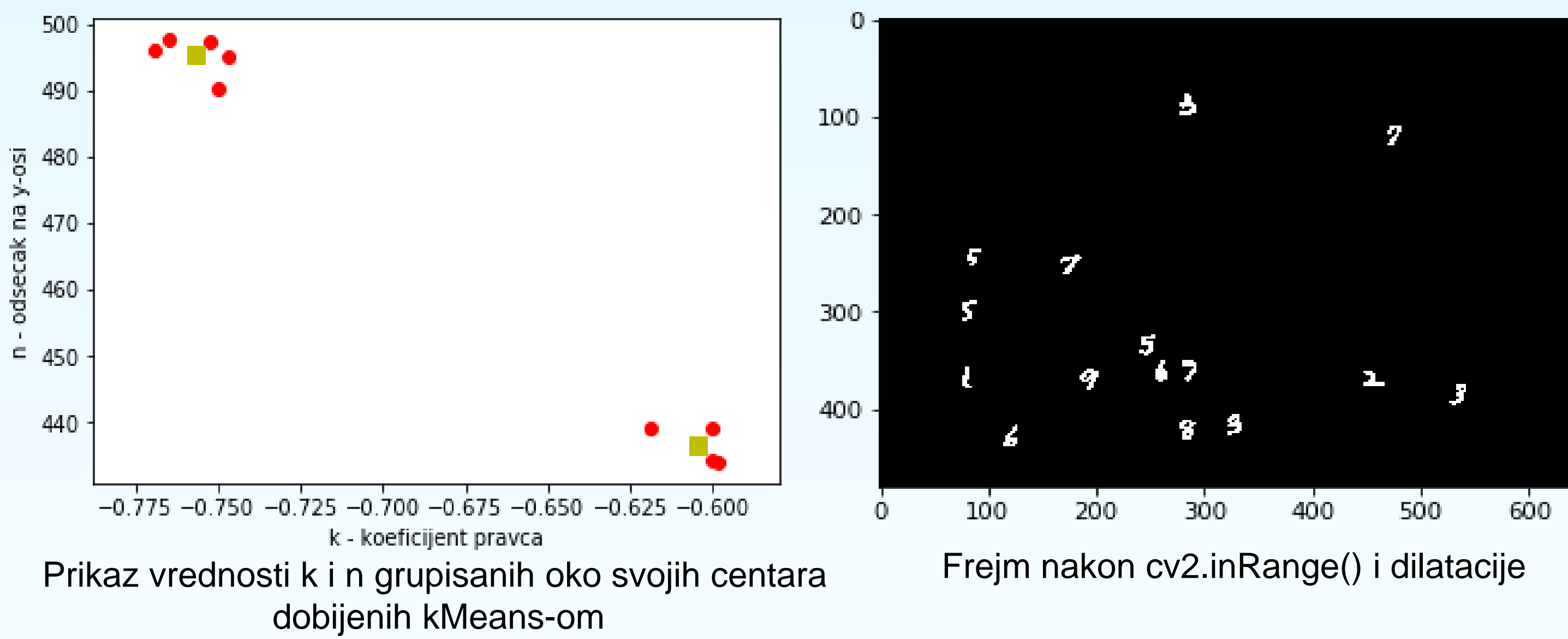
- Metodi za detekciju linija prosleđuje se slika prvog frejma i broj linija koji se očekuje na snimku.
- Vrš se konverzija slike u grayscale i prosleđuje se Canny Edge algoritmu za detekciju ivica. Rezultat ove operacije je binarna slika na kojoj vrednost 1 imaju pikseli koji predstavljaju ivice objekata sa slike.



• Slika dobijena Canny Edge algoritmom prosleđuje se funkciji HoughLinesP iz OpenCV biblioteke, koja proverava sve detektovane ivice i vraća niz početnih i krajnjih tačaka onih ivica koje ispunjavaju uslov prosleđenog praga, tj. predstavljaju pravu liniju.

• Niz dobijenih tačaka je dalje potrebno klasifikovati, odnosno za svaku od njih odrediti kojoj liniji na frejmu pripada. Da bi ovo bilo omogućeno, za svaki element niza koji predstavlja početnu i krajnju tačku računamo jednačinu prave. Dobijene koeficijente k i n prosleđujemo k-Means algoritmu za klasifikaciju, čiji je cilj formiranje kompaktnih grupa na osnovu sličnosti. Rezultat ovoga predstavlja niz labela koje određuju kojoj liniji sa frejma pripadaju linije dobijene Hough transformacijom.

• Za svaki klaster linija računaju se minimalne i maksimalne tačke po jednoj od osa i time se dobijaju konačne koordinate početnih i krajnjih tačaka linija sa frejma.



Obrada frejma

Za svaki frejm prethodno učitano video snimka:

- Vrš se uklanjanje šumova i linija na osnovu boja korišćenjem inRange() funkcije. Dobijena slika, na kojoj svi pikseli koji ne zadovoljavaju prosleđeni opseg dobijaju vrednost 0, a u suprotnom 1, koristi se kao maska i množi se sa originalnim frejmom. Na taj način se uklanjaju svi pikseli koji nisu od interesa (šumovi i linije).

• Na slici dobijenoj u prethodnom koraku radi se dilatacija kako bi se obezbedilo da cifra koja je eventualno razvojena u prethodnom koraku bude prepoznata kao jedan objekat od interesa.

• Korišćenjem funkcija label() i find_objects() iz Skipy biblioteke vrši se pronalaženje objekata od interesa i svakom od njih se dodeljuje jedinstvena labela. Rezultat ovoga je niz koji predstavlja početne i krajnje tačke objekata po x, odnosno y-osi.

Obrada detektovanog objekta

Za svaki objekat dobijen prethodnim korakom:

- Na osnovu početnih i krajnjih tačaka po obe ose izračunava se njegov centar i veličina.
- Objekte koji su veći od 11 piksela smatramo ciframa, pa je potrebno dalje proveriti da li se trenutni objekat prvi put pojavio na snimku. To se postiže pomoću inRange() metode koja u listi prethodno pronađenih objekata traži onaj koji se nalazi unutar prosleđene razdaljine od trenutnog objekta.
 - Ukoliko nije pronađen nijedan objekat u blizini, smatra se da se navedeni prvi put pojavio na ovom frejmu, pa mu se dodeljuje jedinstveni identifikator, prethodno izračunat centar, status prelaska preko linije i trenutni broj frejma. Obavlja se isecanje cifre sa frejma, pozivanje metode za postavljanje cifre u gornji-levi ugao isečka i njeno slanje algoritmu za prepoznavanje. Pored navedenih atributa pamti se i prepoznata vrednost cifre.
 - Ukoliko je trenutni objekat ranije identifikovan, dodeljuje mu se broj frejma, novi centar, pri čemu se isti dodaje u istoriju njegovih prethodnih centara.
- Za svaki objekat iz liste prethodno identifikovanih:
 - Izračunava se razdaljina do svake linije na frejmu:
 - Ukoliko je trenutni element iznad linije (ako je ispod, zanemarujemo ga) i ukoliko je razdaljina manja od 9 piksela, izvesno je da će element preći liniju, pa je njegov status prelaska preko linije potrebno postaviti na True za odgovarajuću liniju, i uvećati, odnosno umanjiti ukupnu sumu za vrednost njegovog atributa koja predstavlja prepoznatu cifru.
 - Na kraju se vrši prikaz rezultata i time se ovaj algoritam završava.

Zaključak

Kompleksnost projekta je naizgled prikrivena relativnom jednostavnošću izlaganja njegove problematike, te se stoga može smatrati da bi se isti mogao završitiu kratkom vremenskom periodu. Međutim, to ipak nije slučaj jer se sveobuhvatnost problema može sagledati tek nakon što se počne sa realizacijom projekta. Procenat prepoznavanja varira u zavisnosti od težine korišćenog video zapisa.