

Predefinisani projekat za 60 bodova(ocene 9 i 10)

- Video zapis poseduje 2 pokretne linije koje je potrebno detektovati Hough transformacijom
- Cifre prelaze preko pokretnih linija
- Cifre koje pređu ispod prve linije treba sabrati, a cifre koje pređu ispod druge linije treba oduzeti od konačnog rezultata
- Pokušati rešiti slučaj preklapajućih cifara koje istovremeno prelaze preko linije, a nalaze se jedna preko druge u određenoj meri
- Za najvišu ocenu je potrebno postići tačnost prepoznavanja od bar 95%

Metod rešavanja problema

- 1** Programski jezik Python i odgovarajuće biblioteka:
- NumPy,
 - Matplotlib,
 - OpenCV,
 - Tensorflow(Keras)

- 2** Kreiranje neuronske mreže pomoću Tensorflow-a. Upotrebljen Keras Sequential model sa sledećom layer strukturom:
- Flatten - input_shape = (28,28),
 - Dense
 - Dropout
 - Dense - output space = 10

- 3** Linije detektovane upotrebom HoughLinesP metodom uz specifikaciju parametara minLineLength(100) i maxLineGap (10).
- Dobijene krajnje tačne linija

Rezultati

Detekcija linija:

Linije su detektovane pomoću **HoughLinesP** metode(OpenCV biblioteka). Svaka linija je pronađena zasebno, u korist je uzeta različita boja dve linije. Pre upotrebe metode korišćen je **Binarni Threshold**. Parametri su podešeni na:

- Threshold vrednost piksela - 20
- maxValue postavljen na 255

Kao rezultat dobijene su **krajnje tačne linija**

Izdvajanje cifara sa snimka, predikcija broja i računanje sume:

Cifre su detektovane u svakom frejmu. Nakon prebacivanja slike u gray scale primenjen je binarni threshold sa istim parametrima kao i za detekciju linija. Iz frejma su zatim detektovane sve konture. Odrađena je selekcija na osnovu veličine pravougaonika minimalne površine.

Za svaki odgovarajući pravougaonik nad konturom broja računa se distanca od linije. Na osnovu pozicioniranja pravougaonika predviđa se da li će broj doći u kontakt sa bližom linijom. Uzimaju se u obzir jedino pravougaonici koji se nalaze iznad najbliže linije.

Ukoliko su uslovi pozicioniranja pozitivni(pravougaonik je ocenjen da je dovoljno blizu linije da se može predvideti kontakt sa linijom). Sa frejma se izdvaja deo koji odgovara pravougaoniku sa brojem unutar njega i priprema kao ulaz u neuronsku mrežu.

Neuronska mreža je obučena na MNIST dataset-om. Rezultat predikcije neuronske mreže se prosleđuje dalje programu kako bi se ustanovilo da li je možda isti broj 2 puta detektovan iznad linije kojoj je najbliži.

Problemi:

Sledeći problemi su izdvojeni kao najuticajniji tokom implementacije:

- Detekcijom brojeva unutar frejma nije rešen problem preklapanja cifara. Tokom implementacije korišćeni su i Tracker-i, međutim, rezulat njihove implementacije nije davao bolje rezultate
- Implementacija detekcije brojeva i izračunavanje njihove udaljenosti od najbliže linije donela je komplikacije i potrebu za dodatnim proverama da li je isti broj detektovan više puta. Poređenje brojeva vršeno je na osnovu rezultata predikcije neuronske mreže za deo slike na kojoj se nalazi broj i veličine pravougaonika kojem broj pripada. Javlja se mogućnost da u različitim frejmovima isti broj pripada pravougaonicima različitih veličina.
- Predikcija neuronske mreže za region u kojem se nalazi broj pokazuje veoma nisku tačnost. U velikom broju slučajeva konture koje pripadaju cifri 0(nula) su detektovane kao cifra 7(sedam)

Link do projekta: <https://github.com/BobanPoznanovic/softPredef>