

O problemu:

Zadatak projekta je detekcija brojeva na snimku koji se kreću po ekranu sa gornje leve ka donjoj desnoj strani. Takođe snimak sadrži i dve nepokretne linije, plavu i zelenu. Cilj je da brojeve koji pređu preko plave linije dodamo na ukupan zbir (saberemo), a brojeve koji pređu preko zelene linije oduzmemo od ukupnog zbira. Realizaciju zadatka otežavaju bele tačkice koje se mogu okarakterisati kao šum. Postupak se ponavlja nezavisno za svaki zadati video snimak.

Prvi korak – DETEKCIJA LINIJE:

Na početku uzimamo prvi frejm snimka i na njemu detektujemo obe linije (jer su generalno linije fiksirane). Nakon što uzmemo frejm prebacujemo ga u HSV model boja kako bi detektovali zelenu i plavu boju (to radimo preko `inRange` metode). Nakon što uočimo boje izvlačimo

koordinate ove 2 linije pomoću `HoughLinesP` metode (biramo najduže linije). Kako bi znali da ih je program detektovao, radi samokontrole preko njih postavljamo dve crvene linije u vidu konture.

Drugi korak – DETEKCIJA BROJEVA:

Brojeve detektujemo posle prvog frejma kroz ceo snimak. Prvi korak je uklanjanje linija i belih tačkica (šuma). Ovde menjamo format boja u RGB, a preko `findContours` OpenCV metode uzimamo konture brojeva. Potom `boundingRect` funkcijom uzimamo koordinate gornjeg levog ugla broja, kao i visinu i širinu.



Treći korak – DETEKCIJA PRELASKA PREKO LINIJE:

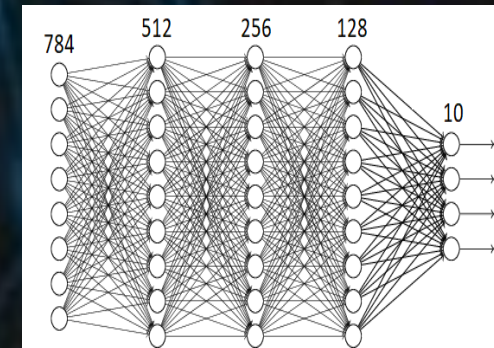
Suština ovog zadatka jeste kreiranje regiona od interesa,

dakle onaj deo frejma gde nas zanima da li je broj tu naišao ili nije. Konkretno mi imamo dva regiona od interesa, za plavu i zelenu liniju. Uočenu liniju posmatramo kao hipotenuzu pravouglog trougla. Ostale dve katete formiramo na osnovu dve kranje tacke te linije i koordinate levog gornjeg ugla broja. Ako je visina manja od nekog prosledjenog broja znaci da je taj broj na veoma malom rastojanju od linije i njega uzimamo, ostale odbacujemo. Pre ubacivanja koordinata broja u listu proveravamo razdaljinu svih koordinata u listi sa koordinatama konture koju ubacujemo. Ako se proverom ispostavi da je to zapravo isti broj samo se pomerio u drugom frejmu napustamo taj slučaj. U suprotnom slučaju koordinate tog broja ubacujemo u listu zajedno sa frejmom.

Četvrti korak – NEURONSKA MREŽA

Koristimo API Keras biblioteke za kreiranje neuronske mreže. Neuronska mreža sadrži tri nevidljiva sloja od po 512, 256 i 128 neurona. Prvi ulazni sloj

sadrži 784 neurona (velicina slike u pikselima 28x28), dok je izlazni sloj sadržan od 10 neurona (10 vrednosti cifara).



Prvo obrađujemo slike sa ciframa. Invertujemo boju pozadine slike i broja na njoj kako bi lakše obradili. Potom radimo tzv. otvaranje (to je postupak kada na sliku primenjujemo prvo eroziju – uklanjanje belih piksela, pa dilataciju – uklanjanje crnih piksela). Ovom operacijom dobijamo čistiju sliku jer se uklanja nepotrebni šum oko broja. Nakon ovoga radimo binarni threshold kako bi slika bila crno-bela. Potom sliku transformišemo u vektor od 784 bita (28x28) i to šaljemo na ulaz neuronske mreže.