EZ-Gimpy CAPTCHA prepoznavanje

Ivan Vrsajkov

RA155/2013

Uvod

CAPTCHA (Completely Automated Public Turing test to Tell Computers and Humans Apart) predstavlja program koji može da generiše test a potom i oceni navedeni rezultat koji služi za razlikovanje čoveka od kompjutera. Stoga, poenta testova je da ih čovek može sa relativnom lakoćom rešiti, dok kompjuter ne može! Sam koncept CAPTCHA-e je osmišljen kao odgovor na stvarne probleme sa kojima su se susretale razne internet kompanije koje su nudile mogućnost kreiranje besplatnog e-mail naloga za korišćenje od strane ljudi. Međutim, otkriveno je da su jako često bili korišćeni kompjuteri ("botovi") za masovno kreiranje naloga i slanja "junk" mejlova. Time što je traženo od korisnika da reši jednu CAPTCHA-u, bilo je moguće sprečiti ovakvo ponašanje jer "botovi" nisu uspevali da prođu test koju bi postavila CAPTCHA. Jedan od ranijih primera je bila EZ-Gimpy CAPTCHA koju je koristio Yahoo.

Primer EZ-Gimpy CAPTCHA-e

CAPTCHA je funkcionisala tako što se korisniku prikaže slika jedne reči nad kojom su izvršene razne transformacije. One su za cilj imale da reč i dalje ostane dovoljno čitljiva za čoveka, ali da istovremeno prosečan OCR softver ne može da je prepozna.

Cilj, algoritam i metode

Glavni cilj ovog projekta jeste da se proba napraviti softver koji će umeti na osnovu slike koju generiše EZ-Gimpy CAPTCHA prepozna sadržanu reč (sa zadovoljavajućom tačnošću).

Sam proces prepoznavanja je izdeljen na tri dela:

1. **Transformacija i obrada slike**
2. **Prepoznavanje pojedinačnih slova**
3. **Određivanje najverovatnijih reči**

Transformacija i obrada slike

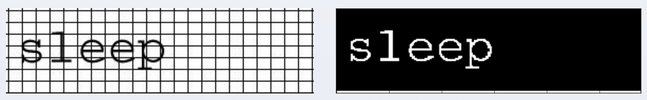
U ovom koraku se slika procesira sa ciljem da se utvrde njeni regioni koji najverovatnije sadrže slovo. Postoji više različitih vrsta slika koje se mogu pojaviti i svaka od njih zahteva poseban način obrade. Dakle, potrebno je prvo utvrditi o kojem tipu slike je reč, potom primeniti njoj specifičnu obradu i izolovati regione slike koji najverovatnije sadrže jedno slovo.

Slede vrste slika koje su uzete u obzir u ovom projektu:

1. Obična slova
   - U ovom slučaju ne postoji nikakav šum na slici. Pozadina ne mora biti bela a nad slovima je često izvršena distorzija. Potrebno je pomoću odgovarajućeg threshold-a zanemariti pozadinu i preostale regione proslediti na dalju obradu.

Primer obrade za tip "Obična slova"

2. Crni grid preko teksta
   - Preko reči je prebačen crni grid koji ne može da se ukloni običnim threshold-om. Stoga je potrebno pretvoriti sliku u binarnu sa nižim threshold-om nego obično, kako bi se što veći deo slova uhvatio. Potom se primenjuje morfološku operacija erozije radi uklanjanja grid-a i operacija zatvaranja, kako bi se slova spojila. Preostali regioni nakon transformacija se šalju na dalju obradu.

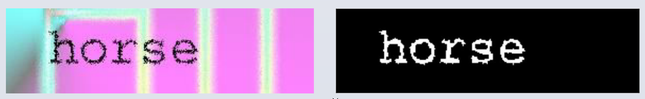


Primer obrade za tip "Crni grid preko teksta"

3. Slika sa šumom, slova su cela
   - Na slici se oko slova nalazi taman šum. Nakon što je slika pretvorena u binarnu, od svih regiona potrebno je izdvojiti dovoljno velike regione koji najverovatnije sadrže slovo a ostali se zanemaruju.

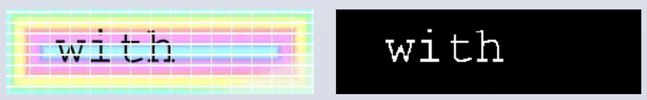
Primer obrade za tip "Slika sa šumom, slova su cela"

4. Šum među slovima
   - Šum se nalazi među samim slovima, te ona nisu cela. Nakon što se pomoću threshold-a eliminiše uticaj pozadine (jer nije uvek bela) i istovremeno se uhvati što više samih slova, primenjuje se zatvaranje kako bi se pokušali formirati regioni koji predstavljaju pojedinačna slova.



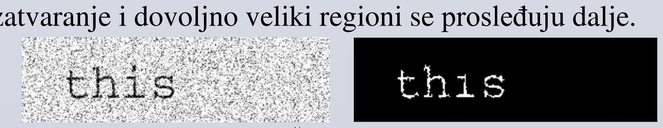
Primer obrade za tip "Šum među slovima"

5. Beli grid preko teksta
   - Preko slike je prebačen beli grid, nakon čega slova nisu cela. Slično kao i sa slučajem kada postoji šum među slovima, nakon što se pomoću threshold-a eliminiše uticaj pozadine, potrebno je primeniti zatvaranje kako bi se pokušala spojiti slova u regione koji predstavljaju pojedinačna slova.



Primer obrade za tip "Beli grid preko teksta"

6. Šum po celoj slici i među slovima
   - U ovom slučaju, taman šum manjih dimenzija prekriva celu sliku. Pomoću odgovarajućeg threshold-a se uzima što više od slova, a istovremeno što manje od šuma. Potom se primenjuje zatvaranje i dovoljno veliki regioni se prosleđuju dalje.



Primer obrade za tip "Šum po celoj slici i među slovima"

Prepoznavanje pojedinačnih slova

U ovom koraku se pravi predviđanje slova na osnovu prosleđenih regiona nakon obrade slike. Predviđanje se vrši pomoću K-nearest neighbour algoritma. Potom se utvrđuje redosled prepoznatih slova preko sortiranja regiona na osnovu njihove pozicije po X osi.

Određivanje najverovatnijih reči

Na osnovu prepoznatih slova i njihovog redosleda određuje se koje su najverovatnije reči u pitanju. Ovaj korak se radi kako bi se pokušale ispraviti moguće greške u prethodnim postupcima. Radi utvrđivanja najverovatnijih reči se koristi Levenshtein distance algoritam i skup prikupljenih reči na engleskom jeziku.

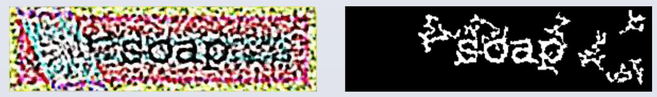
Rezultati

Radi utvrđivanja tačnosti, tri testa su korišćena:

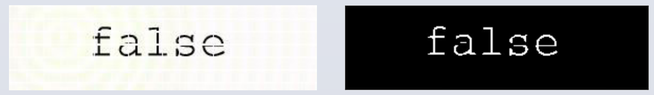
1. Test provere tačnih reči – provera da li je tačna reč sadržana u skupu vraćenih najverovatnijih reči. **Rezultat: 126/192**
2. Soft test računajući samo najbolje reči – procenat sličnosti najpribližnijih reči iz skupa najverovatnijih reči respektivnoj tačnoj reči pomoću Levenshtein distance. **Rezultat: 0.87%**
3. Soft test računajući sve vraćene reči – procenat sličnosti svake reči iz skupova najverovatnijih reči respektivnoj tačnoj reči pomoću Levenshtein distance. **Rezultat: 0.51%**

Problemi i moguća poboljšanja

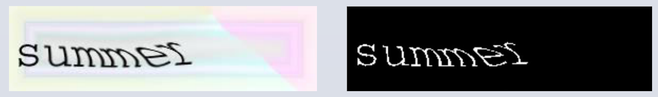
Bilo je prisutno više problema u pojedinim slučajevima prilikom obrade slika koji su bili suviše spicifični da se reše na odgovarajući način a da pri tome ne pokvare uspešnu obradu drugih primera.



Problem prevelikih regiona smetnji



Problem nemogućnosti spoja svih delova slova



Problem spojenih slova u originalu usled distorzije

Jedan od problema je bio i samo korišćenje K-nearest neighbour algoritma za prepoznavanje slova. Algoritam mora da koristi samo jedan najbliži sused zbog manjkavosti skupa sa označenim podacima. To znači da sama prepoznavanja često neće biti tačna, uprkos obradi slika označenog skupa i slika regiona koji najverovatnije sadrže po slovo.

Primer - Prepoznata slova:
"L", "E", "N", "O", "R", "L"

Takođe, problem je i broj reči koji će Levenshtein distance vratiti kao najverovatnije, jer se koristi veoma velik skup reči, koji sadrži i reči za koje su jako male šanse da će se ikada naći u CAPTCHA-i. Zato bi trebalo ili smanjiti skup reči za pretragu ili ubaciti faktor verovatnoće pojavljivanja reči u CAPTCHA-i kako bi se moglo lakše utvrditi manji skup najverovatnijih reči.