

Preliminarno ispitivanje sadržaja slike pomoću histograma slike korišćenjem SVM algoritma i neuronske mreže

Student
Igor Valjević

Mentor
prof. dr. Vladimir Filipović

Matematički fakultet
Univerziteta u Beogradu
april 2016.

Sadržaj

| | |
|--|----|
| 1. Rezime..... | 4 |
| 2. Uvod..... | 5 |
| 2.1. Pregled i struktura rada | 6 |
| 3. Modeli za predstavljanje boja u računarstvu: RGB, HSI (HSV, HSL), intenzitet..... | 7 |
| 3.1. RGB model boja | 7 |
| 3.2. HSI (HSV, HSL) model boja | 7 |
| 4. Histogram slike | 9 |
| 4.1. Histogram baziran na intenzitetu | 9 |
| 4.2. Histogram baziran na bojama..... | 9 |
| 5. Veštačke neuronske mreže..... | 11 |
| 5.1. Perceptron..... | 11 |
| 5.2. Algoritam propagacije unatrag (eng. Back propagation)..... | 13 |
| 5.3. Bolcmanova mreža sa ograničenjem (RBM) | 15 |
| 6. Metoda podržavajućih vektora (eng. Support Vector Machine - SVM)..... | 17 |
| 6.1. Primena na nelinearne slučajeve..... | 18 |
| 6.2. Pregled popularnih jezgra koji se koriste u SVM | 19 |
| Linearno jezgro..... | 19 |
| Polinomijalno jezgro | 19 |
| Gausovo jezgro | 19 |
| Laplasijanovo jezgro | 20 |
| Jezgro hiperboličke tangente (Sigmoidno) | 20 |
| 7. Računarska i softverska platforma..... | 21 |
| 7.1. Paint 2.0 | 21 |
| Struktura programa..... | 22 |
| 7.2. Prepoznavanje slike na osnovu histograma | 23 |
| Struktura modula za kreiranje RBM..... | 24 |
| Struktura modula za kreiranje SVM..... | 24 |
| 8. Izbor optimalnih parametara | 25 |
| 8.1. Izbor slika | 25 |
| 8.2. Izbor parametara za RBM..... | 27 |
| 32 vrednosti nijansi po boji | 27 |
| 64 vrednosti nijansi po boji | 27 |
| 256 vrednosti nijansi po boji | 27 |

| | |
|---|----|
| 8.3. Izbor parametara za SVM..... | 28 |
| 32 vrednosti nijansi po boji | 28 |
| 64 vrednosti nijansi po boji | 28 |
| 256 vrednosti nijansi po boji | 28 |
| 9. Rezultati – veštačke neuronske mreže | 29 |
| 32 vrednosti nijansi po boji | 29 |
| 64 vrednosti nijansi po boji | 30 |
| 256 vrednosti nijansi po boji | 30 |
| 10. Rezultati – SVM..... | 31 |
| 32 vrednosti nijansi po boji | 31 |
| 64 vrednosti nijansi po boji | 32 |
| 256 vrednosti nijansi po boji | 32 |
| 11. Zaključak | 33 |
| 12. Reference..... | 34 |

1. Rezime

U radu se razmatra problem prepoznavanja sadržaja slike. Testira se u kojoj se meri, u procesu prepoznavanja, korisnik može osloniti na podatke koji se odnose isključivo na paletu boja slike. U tu svrhu iz slika su izdvojeni histogrami osnovnih boja (RGB) crvena, zelena i plava i time kompletno zanemareni specijalni podaci slike.

Korišćene su metoda podržavajućih vektora (eng. Support Vector Machine – SVM) i Veštačka neuronska mreža, preciznije Bolcmanova mreža sa ograničenjem (eng. Restricted Boltzmann Machine - RBM).

Uspešno je pokazano da je moguće uopštiti algoritam koristeći 64 nijanse po boji, umesto svih 256 kao i da algoritam sa 256 nijansi daje lošije rezultate zbog preprilagođavanja (eng. overfitting).

Ustanovljeno je da veliki deo vremena izvršenja algoritma, neočekivano, otpada na pretprocesiranje i da algoritam za kreiranje histograma, iako veoma jednostavan, treba dodatno usavršiti.

Primećena je i velika dihotomija prilikom potrage za skupom slika na kojima bi bilo moguće učiti algoritme: ili je baza slika bila veoma usko specijalizovana, ili nije bila unapred klasifikovana. Ako se ovo uzme u obzir i činjenica da Bolcmanove mreže sa ograničenjem, pored učenja sa nadzorom, omogućavaju i učenje bez nadzora (eng. unsupervised learning), može se zaključiti da se, pri rešavanju proučavanog problema, u prvom planu nalazi RBM algoritam [1], [2].

2. Uvod

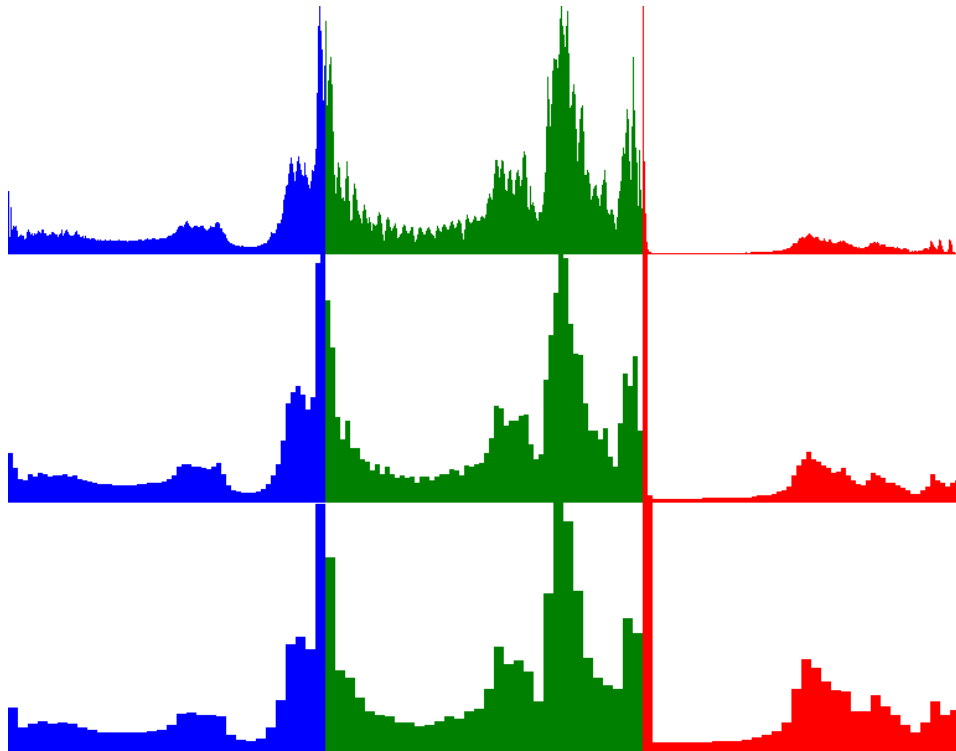
Pored prepoznavanje lica, obrazaca i oblika [3], [4], jedan od prvih utisaka koji se stiže prilikom pregledanja slika je uočavanje palate boja. Ljudski mozak je uslovljen da povezuje boje, npr. zelenu sa travom, lišćem, prirodom; plavu sa nebom, vodom; boju kože sa licem, ljudskim telom, itd [5].



Slika 1. Primer slika pogodnih za klasifikaciju putem histograma boja

Svaka slika ima definisanu klasu kojoj pripada. Klase su definisane na osnovu subjekta slike, na primer: šuma, lice, kuća, more... Svakoj slici odgovara jedna klasa, što je unapred, ručno, određeno. Cilj svih klasifikacionih algoritama veštačke inteligencije je, da na osnovu početnog skupa podataka sa poznatim klasama, da odredi klasu novog podatka sa nepoznatom klasom. Tako i u ovom slučaju, cilj je da se na osnovu početnog (klasifikovanog) skupa histograma slika odredi klasa nove slike sa nepoznatom klasom.

U ovom istraživanju napravljen je pokušaj da se proveriti da li gornja pretpostavka važi i u domenu računarskog prepoznavanja sadržaja slike. Za svaku sliku izračunat je njen histogram po svakoj od komponenata osnovnih boja (crvena, plava i zelena). Tako dobijene vrednosti su ulazni parametri za dva najpopularnija pristupa klasifikaciji podataka koji su danas u upotrebi. Prva, zasnovana na veštačkim neuronskim mrežama, preciznije Bolcmanovim mrežama sa ograničenjem (RBM) i druga, Metoda podržavajućih vektora (eng. Support Vector Machine – SVM). Obe metode kao ulazne podatke očekuju niz vrednosti, dobijenih izračunavanjem histograma za svaku sliku, kao i kojoj klasi data slika pripada.



Slika 2. Primer histograma sa različitim brojem nijansa po boji. Prvi red ima 256, drugi 64 i treći 32 nijanse.

U cilju da se uporedi koliko ima uticaja preciznost histograma na izbor klase za svaku sliku su izračunati histogrami sa 32 nijanse boja, 64 nijanse i sa 256 nijansi (Slika 3.). Dalje, obe metode su nezavisno obučavane sa sva tri tipa histograma i dobijeni rezultati upoređeni.

2.1. Pregled i struktura rada

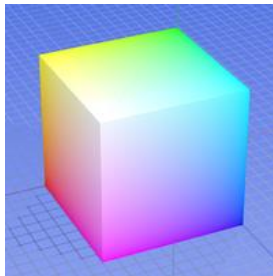
U četiri odeljka, koja slede, (od trećeg do šestog) ukratko su opisani korišćeni algoritmi, njihove varijante i parametri. Potom je u sedmom odeljku predstavljena softverska platforma, kao i korišćene biblioteke. Dalje, u osmom odeljku, predstavljena je metodologija za izbor slika i parametara algoritama. U devetom i desetom odeljku su predstavljeni dobijeni rezultati i izvršeno je razmatranje rezultata. Sledeći, jedanaesti odeljak sadrži zaključna razmatranja kao i moguće pravce daljih istraživanja. Na kraju je dat spisak korišćene literature.

3. Modeli za predstavljanje boja u računarstvu: RGB, HSI (HSV, HSL), intenzitet...

Za predstavljanje boja u računarskim sistemima se koriste apstraktni matematički modeli koji se nazivaju modeli boja. Modeli boja opisuju kako se svaka pojedinačna boja može opisati kao n-torka brojeva. U različitim modelima intenzitet boja se računa na različite načine, pa vrednost intenziteta u jednom modelu ne odgovara direktno vrednosti u drugom [6].

3.1. RGB model boja

RGB je aditivni model sa tri osnovne boje: crvena, zelena i plava koje dodavanjem u različitim proporcijama predstavljaju ostale boje [7]. Baziran je na ljudskoj percepciji boja sa tri tipa receptora koji postoje u ljudskom oku. Kao takav je najlakši za razumevanje. RGB model se najčešće predstavlja kao kocka (Slika 4.)



Slika 4. RGB model boja predstavljen kao kocka

Postoje različite formule za izračunavanje vrednosti intenziteta boja u RGB modelu. Osnovni je jednostavna srednja vrednost tri komponente boja.

$$Y = (R + G + B)/3 \quad (1)$$

Ova formula je korektna, jer sliku u boji pretvara u sliku sa nijansama sive. Međutim, primećeno je da neke nijanse sive, dobijene na ovaj način, ne odgovaraju našoj uobičajenoj percepciji. Zato su napravljene druge formule za izračunavanje vrednosti intenziteta.

$$Y = 0,2126 * R + 0,7152 * G + 0,0722 * B \quad (1)$$

$$Y = 0,299 * R + 0,587 * G + 0,114 * B \quad (2)$$

$$Y = (R + R + B + G + G + G)/6 \quad (3)$$

3.2. HSI (HSV, HSL) model boja

Dok se RGB model obično predstavlja ortogonalnim koordinatama, modeli HSI (HSV, HSL) se predstavljaju u cilindričnim koordinatama. Ovi modeli transformišu RGB model u modele koji bliže odgovaraju ljudskoj vizualnoj percepciji. Boja, zasićenost i

vrednost (odnosno osvetljenost za HSL) se predstavljaju trojkama i izomorfne su sa RGB modelom, što znači da se jedna boja u RGB modelu preslikava u tačno jednu boju u HSV(L) modelu, i obrnuto [8].

Modelima HSI (HSV, HSL) je zajedničko da je boja definisana ortogonalnim uglom, gde crvena počinje na 0° , zelena na 120° i plava na 240° . Udaljenost od centralne ose predstavlja zasićenost sa najzasićenijim bojama na rubu, a centralna osa je rezervisana za nijanse sive sa belom i crnom bojom na vrhu i dnu (Slika 5.). Razlika između ova dva modela je da su osnovne boje kod HSL modela pri vrednosti polovine vrednosti osvetljenosti, dok su kod HSV modela osnovne boje kod vrednosti 1. Uočava se da je HSL model bliži onome što se intuitivno očekuje, da je bela boja kod vrednosti osvetljenja 1.



Slika 5. Cilindrična reprezentacija HSL (a) i HSV (b) modela boja

Preciznije bi se HSL model mogao prikazati kao bi-prizma, a HSV model kao prizma.



Slika 6 Reprezentacija HSL modela kao bi-prizma (a) I HSV (b) modela kao prizma

4. Histogram slike

Histogram slike se definiše kao graf koji na horizontalnoj osi ima sve nijanse boje (intenziteta osvetljenosti), a na vertikalnoj broj piksela koji su te nijanse. Popularizacijom digitalne fotografije histogram je dostigao široku upotrebu pošto može veoma očigledno da pokaže da li je slika ispravno eksponirana.



Slika 7. Slika i njen histogram

Kao što se može videti, Slika 7b nema previše piksela u blizini vrednosti 0 ili 255 što dovodi do zaključka da su podešavanja bila ispravna prilikom slikanja i da detalji nisu izgubljeni zbog preekspozicioniranosti ili podekspozicioniranosti.

U polju obrade slika histogrami se često koriste kao vodilje prilikom algoritama izbora graničnih vrednosti, detekcije ivice, segmentacije slike, itd. Histogrami se mogu koristiti i u nekim tehnikama za obradu slike gde nude veliku dozu kontrole nad procesom, npr. slaganje histograma i izjednačavanje histograma.

4.1. Histogram baziran na intenzitetu

Histogrami bazirani na intenzitetu (svetlosti) celokupnu sliku transformišu u histogram tako što za svaki piksel izračunavaju intenzitet svetlosti po jednoj od formula objašnjenim u 2.1. Za potrebe testiranja u ovom radu takav pristup nije bio prihvatljiv pošto se podaci o komponentama boja gube prilikom sumiranja.

Formula kojom se izračunava intenzitet je linearna kombinacija vrednosti po bojama (Formula 2). Iz tog razloga je pretpostavljeno da ne postoji razlog za uključanje histograma baziranog na intenzitetu u konačan algoritam.

4.2. Histogram baziran na bojama

Prilikom kreiranja histograma slike po bojama, kreiraju se tri histograma u RGB modelu po jedan za svaku komponentu boje (crvena, zelena i plava). Svaki histogram se kreira tako što se prebrojavaju pikseli koji imaju iste vrednosti za svaku od komponenta boja.

Pošto je u RGB modelu za svaku boju obezbeđeno osam bitova, jasno je da postoji ograničenje na maksimalno 256 nijansi po komponenti boje.

Za potrebe ovog istraživanja, a u interesu smanjivanja kompleksnosti domena [4], upoređivani su rezultati dobijeni ako se koristi manji broj nijansi po boji. Broj nijansi je smanjivan tako što su agregirane vrednosti piksela bliskih po nijansi boje. Na primer, da bismo dobili histogram sa samo 32 nijanse agregirano je po osam bliskih nijansi u jednu. Od 0x00 do 0x08 je prva nijansa, od 0x08 do 0x10 je druga, 0x10 do 0x18 je treća, itd.

U ovom istraživanju poređeni su histogrami bazirani na bojama sa 32, 64 i 256 nijansi po komponenti boje.

5. Veštačke neuronske mreže

Izučavanje veštačkih neuronskih mreža započeto je 1943. kada su neurolog McCulloch i logičar Pitts uveli ideju o modelu sa pojednostavljenim neuronima [9]. U ovom radu autori su pokušali da objasne kako mozak radi i kako se jednostavne ćelije mogu organizovati kako bi se oformili kompleksni modeli razmišljanja. Predstavljen je veoma pojednostavljen model neurona kao osnovni gradivni blok. Međusobnim povezivanjem ovakvih neurona formiraju se kompleksne mreže. Neuronu su predstavljeni kao modeli zasnovani na biološkim neuronima i kao osnovne komponente logičkih kola koje mogu da izvrše različite vrste izračunavanja [10].

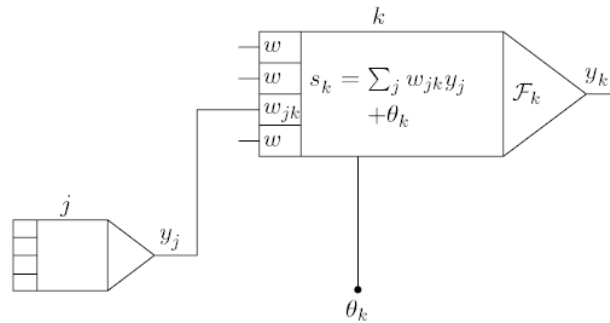
Velikoj popularnosti veštačkih neuronskih mreža doprineo je veliki uspeh i nagla ekspanzija knjiga sa naučno fantastičnim temama. Autori kao što su Isak Asimov i Artur Klark su samo neki od najpoznatijih autora koji su očekivali velike stvari od veštačkih neuronskih mreža i veštačke inteligencije uopšte. Međutim, kada su Minski i Papert objavili knjigu "Perceptrons" 1969. [11], u kojoj su predstavili nedostatke modela perceptrona, ti rezultati su direktno doveli do toga da veliki broj istraživača napusti ovo polje istraživanja. Samo je nekoliko naučnika ostalo da se bavi istraživanjima u ovoj oblasti, od kojih su najpoznatiji: Teuvo Kohonen, Stephen Grossberg, James Anderson i Kunihiko Fukushima.

Tek je početkom osamdesetih godina ponovo zaživelo interesovanje za veštačke neuronske mreže kada su unapređeni algoritmi i dobijeni značajniji teorijski osnovi na kojima je bilo moguće pravilno, i sa realnim očekivanjima, zasnovati ovu granu istraživanja.

5.1. Perceptron

Perceptron je osnovna gradivna jedinica veštačkih neuronskih mreža [12]. U svom osnovnom obliku to je binarni klasifikator koji slika $R^D \rightarrow \{+1, -1\}$, gde je R skup realnih brojeva, a D pozitivan ceo broj. U svojim složenijim oblicima može da bude alat regresivne analize koji, umesto binarnog, za izlaz ima neprekidnu realnu funkciju obično ograničenu na $[0, 1]$.

Svaki perceptron ima relativno jednostavan osnovni zadatak: da prihvati ulazne impulse od susednih perceptrona ili spoljašnjih izvora i izračuna izlaz. Svaki ulaz u perceptron ima pridružen odgovarajući težinski koeficijent koji određuje koliko ulogu ima prilikom izračunavanja izlazne vrednosti. Drugi zadatak perceptrona je da, prilikom treniranja prilagodi izlaznu vrednost očekivanoj u što je većoj meri. To se postiže menjanjem težinskih koeficijenata za neku vrednost (koeficijent učenja). Slika 8 predstavlja shematski prikaz perceptrona i na njoj se mogu uočiti dva perceptrona j i k , te izlaz y_j prvog perceptrona j povezan na ulaz w_{jk} drugog perceptrona k .

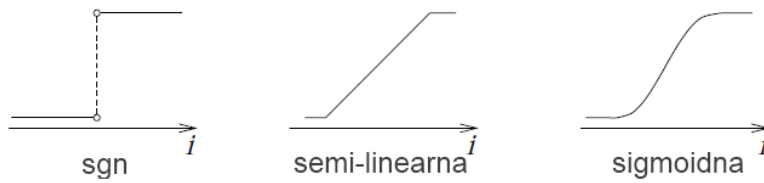


Slika 8. Shematski prikaz perceptrona

Izlaz se često računa kao težišna suma ulaza. Najjednostavniji način računanja izlaza prikazan je sledećom formulom:

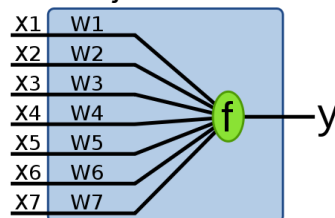
$$s_k(t) = \sum_j w_{jk}(t) y_j(t) + \theta_k(t) \quad (3)$$

Ako je suma veća od neke unapred zadate vrednosti, smatra se da je perceptron aktiviran i učestvovaće u izračunavanjima sledećih perceptrona, u suprotnom je neaktivan. Aktivaciona funkcija perceptrona je obično funkcija limita (tzv. threshold) koja može biti stepenasta, semi-linearna ili sigmoidna (Slika 9) [13].



Slika 9. Prikaz različitih aktivacionih funkcija

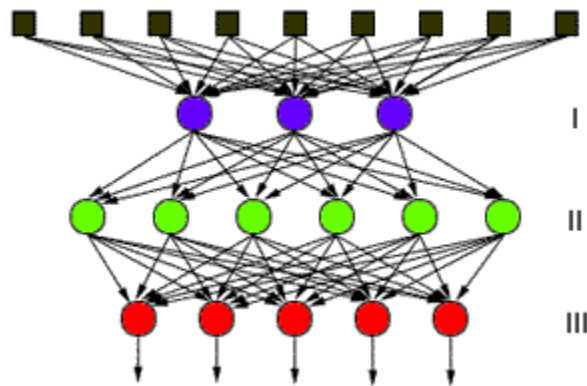
Moguće je trenirati jedan perceptron da obavlja jednostavnu klasifikaciju. Proces se zove treniranje perceptrona i u suštini se isti postupak koristi i kada imamo veliki broj perceptrona povezanih u veštačku neuronsku mrežu. Proces treniranja perceptrona može pomoći da se razume način na koji rade veštačke neuronske mreže.



Slika 10. Perceptron sa ulazima (x_1-x_7), težiškim faktorima (w_1-w_7), aktivacionom funkcijom f i izlazom y

5.2. Algoritam propagacije unatrag (eng. Back propagation)

Algoritam su prvi put opisali 1969. godine Bryson i Ho [14] kao pokušaj da se uopšti upotreba perceptrona na višeslojne mreže, ali je, zbog ograničenosti računarskih resursa i nemogućnosti nalaženja globalnih optimuma, bio uglavnom ignorisan do sredine 1980-tih godina [15]. Tada je, uporedo sa unapređenjima u tehnologiji, brzini i raspoloživoj memoriji računarskih sistema algoritam našao mesto za uspešno učenje veštačkih neuronskih mreža.



Slika 11. Primer veštačke neuronske mreže sa jednim skrivenim slojem

Algoritam se izvršava u dve faze:

- Propagacija
 - Unapred: ulazni podaci se „propuštaju“ kroz mrežu da bi se izračunale aktivacione funkcije
 - Unazad: računaju se razlike dobijenih i očekivanih koeficijenata
- Popravka težinskih faktora perceptrona
 - Izračunavanje stepena razlike težinskih faktora
 - Korekcija težina unapred zadatim faktorom (koeficijent učenja)

Ove dve faze se ponavljaju dok se ne nađe rešenje koje zadovoljava uslove tačnosti [16].

Algoritam, takođe, može biti prikazan pseudo kodom [17]:

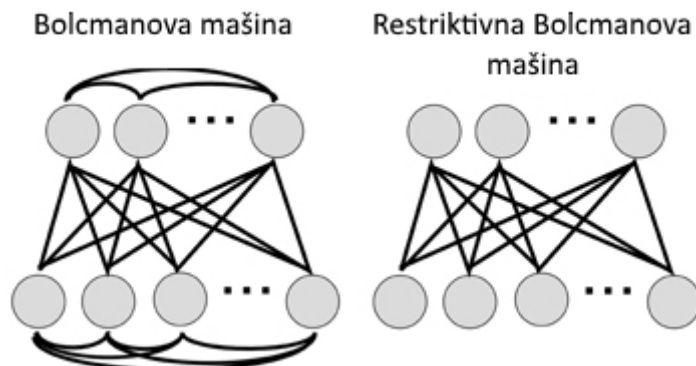
```

PropagacijaUnapred():
forEach  $u$  in  $inputNodes$  do
     $h_u \leftarrow$  odgovarajuću vrednost ulaznog vektora  $x$ 
end
forEach  $v$  in  $calculatedNodes$  do //čvorovi kojima je roditelj,  $par(v)$ , izračunat
     $a_v \leftarrow \sum_{u \in par(v)} w_{(u,v)} h_u$ 
     $h_v \leftarrow \tanh(a_v)$  //izračunavanje funkcije aktivacije čvora
end
return  $a_y$  //vrednost izlaznog čvora
PropagacijaUnatrag()
run PropagacijaUnapred(x) //izračunavanje aktivacija
 $e_y \leftarrow y - a_y$  //izračunavanje greške
forEach  $v$  in  $cal(e_v)$  do //sve vrednosti koje imaju izračunatu grešku
    forEach  $u \in par(v)$  do
         $g_{u,v} \leftarrow -e_v h_u$  //izračunavanje gradijenata
         $e_u \leftarrow e_u + e_v w_{u,v} (1 - \tanh^2(a_u))$  //izračunavanje greške
    end
end
return  $g_e$  //svi gradijenti

```

5.3. Bolcmanova mreža sa ograničenjem (RBM)

Bolcmanove mreže su tip veštačkih neuronskih mreža koje su interesantne zbog svoje stohastičke i generativne prirode. Lako se mogu paralelizovati i mogu se relativno lako naučiti da prepoznaju interne karakteristike skupa podataka na kojem se uče. Ove osobine ih čine odličnim teorijskim modelom, ali zbog svoje interne složenosti nisu pogodne za računarsku implementaciju.



Slika 12. Bolcmanova i Bolcmanova mreža sa ograničenjem

Postavljanjem ograničenja da se neuroni mogu povezivati samo između različitih slojeva (ne postoje veze unutar istog sloja) od Bolcmanove mreže se dobija Bolcmanova mreža sa ograničenjem. Ovo ograničenje omogućava implementaciju značajno efikasnijeg algoritma učenja zasnovanog na gradijentnom spustu.

Pojednostavljen model sa samo dva sloja: ulazni (vidljiv) i izlazni (skriven) se može vektorski izraziti:

$$E(v, h) = -h^T W v - c^T v - b^T h \quad (4)$$

Pri tome je \mathbf{v} vektor ulaznih parametara, \mathbf{h} vektor izlaznih parametara, \mathbf{c} vektor težinskih faktora za ulazni (vidljivi) sloj i \mathbf{b} vektor težinskih faktora za izlazni sloj. Gornja formula daje funkciju energije datog sistema, dok je funkcija verovatnoće da je sistem u određenoj konfiguraciji data sa:

$$p(v, h) = \frac{e^{-E(v, h)}}{Z} \quad (5)$$

Ovde E predstavlja vrednost energije sistema za date vrednosti ulaznih i izlaznih parametara, a Z je parametar normalizacije.

Ono što je svakako velika prednost RBM algoritma je mogućnost učenja slojeva „jedan po jedan“. Moguće je naučiti sloj h_1 na osnovu vrednosti vektora \mathbf{v} , pa onda na osnovu h_1 vektora naučiti sloj h_2 i tako dalje:

$$Z = \sum_{v, h} e^{-E(v, h)} \quad (6)$$

Iako je moguće izračunati vrednost Z , izračunavanje je eksponencijalne kompleksnosti za svaku moguću vrednost \mathbf{v} i \mathbf{h} . Iz tog razloga se koristi funkcija nazvana kontrastna divergencija. RBM se trenira tako da se nađe maksimum proizvoda

verovatnoća $p(v)$ za trening skup V . Algoritam koji se koristi je modifikacija gradijentnog spusta [17], [2].

$$\begin{aligned}W &\leftarrow W + \alpha(h(x^{(t)})x^{(t)T} - h(\tilde{x})\tilde{x}^T) \\b &\leftarrow b + \alpha(h(x^{(t)}) - h(\tilde{x})) \\c &\leftarrow c + \alpha(x^{(t)} - \tilde{x})\end{aligned}\tag{7}$$

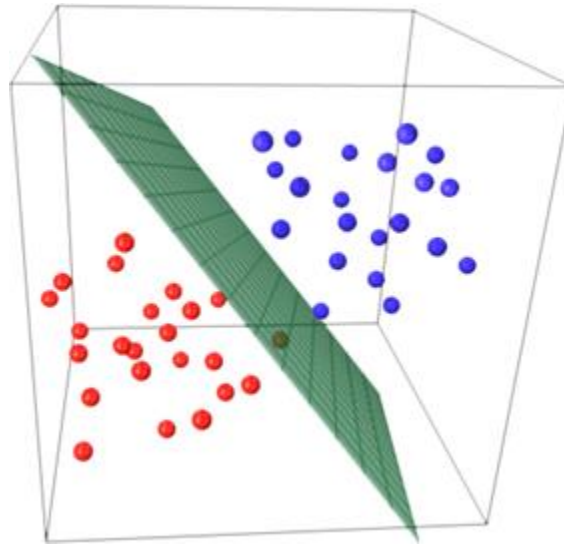
6. Metoda podržavajućih vektora (eng. Support Vector Machine - SVM)

SVM je algoritam mašinskog učenja sa nadgledanjem. Osnovni algoritam je primenjiv samo na linearno razdvojivim tačkama.

U osnovnom obliku imamo dva skupa tačaka (Slika 13.) koje imaju svoje koordinate x_i i vrednost y_i . Vrednost y_i još nazivamo i klasom tačke x_i .

$$x_i \in R^p; y_i \in \{-1, 1\}; i = 1..n \quad (8)$$

Funkcija algoritma je nalaženje optimalne hiper-ravani koja razdvaja ova dva skupa tačaka.



Slika 14. Dva skupa linearno odvojena uz pomoć hiperravni

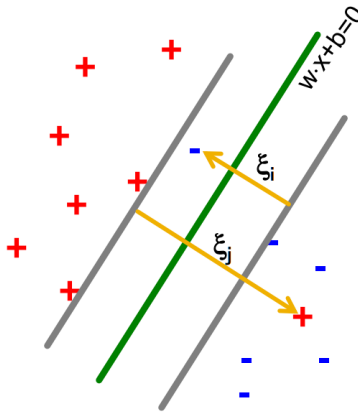
Hiperravan je definisana jednačinom $\langle w, x \rangle - b = 0$, gde je $\langle w, x \rangle$ vektorski proizvod normalnog vektora hiper-ravni i tačke x .

Margina je definisana formulama $\langle w, x \rangle - b = \frac{b}{2}$ i $\langle w, x \rangle - b = -\frac{b}{2}$. Geometrijski, $\frac{b}{\|w\|}$ je širina margine definisana ovom hiper-ravni. Funkcija koja svakoj tački određuje klasu može se zapisati:

$$f(x, w, b) = \text{sgn}(\langle w, x \rangle + b) \quad (9)$$

$$\forall_{i=1}^n: y_i * (\langle w, x_i \rangle + b) \geq 1 - \zeta_i$$

Gde je ζ_i empirijska greška (rizik) u modelu koja omogućava da nađemo hiperravan koja nije apsolutno tačna. Takva hiperravan pogrešno klasifikuje određeni (mali) skup tačaka, ali dovodi do značajno bolje klasifikacije. Parametar ζ_i je uveden da bi se ublažio uticaj potencijalnih autlajera (eng. outlier). Ovakvu marginu nazivamo mekom (eng. soft margin)[18].



Slika 15. Primer meke margine

Optimalna hiperravan je ona koja deli dva skupa maksimizujući marginu. Da bismo maksimizovali marginu $\frac{b}{\|w\|}$ potrebno je minimizovati vrednost $\|w\|$:

$$\min_{w,b} \frac{1}{2} \|w\|^2 + C \sum_i \zeta_i \quad (10)$$

Gde C predstavlja koeficijent kojim se utiče na visinu „kazne“ koju primenjujemo na tačke koje su pogrešno klasifikovane. Ako je C preveliko dolazimo u opasnost da veoma dobro klasifikujemo tačke iz trening skupa, ali da značajno lošije klasifikujemo tačke na kojima nismo trenirali. Ovakvu pojavu nazivamo preprilagođavanjem skupu na kome učimo. Uvođenjem Lagranžovih koeficijenata problem možemo predstaviti kao:

$$\min L(w, b, a) = \frac{1}{2} \|w\|^2 + C \sum_i \alpha_i (y_i (\langle w, x_i \rangle + b) - 1 + \zeta_i) \quad (11)$$

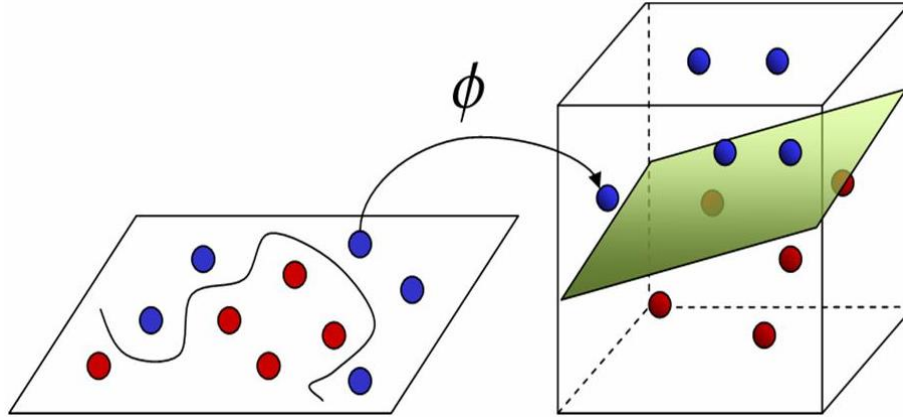
α_i su Lagranžovi koeficijenti koji daju težinu tačkama. Pošto većina α_i mora biti jednaka nuli, one tačke za koje je $\alpha_i > 0$ nazivaju se podržavajući vektori. Ista funkcija se može zapisati u dualnom obliku:

$$\max \sum_i \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j \langle w, x_i \rangle; 0 \leq \alpha_i \leq C \quad (12)$$

Velika prednost SVM algoritma zapisanog u ovom obliku je što se izvršava veoma brzo, u polinomskom vremenu $O(n^3m)$ gde je n broj tačaka, a m dimenzija vektora.

6.1. Primena na nelinearne slučajeve

Dosadašnje razmatranje se odnosilo samo na linearne probleme, ali često se susrećemo sa modelima koji nisu linearno razdvojivi. Na sreću, u SVM-u je moguće definisati funkciju koja svaku tačku preslikava u višedimenzionalni prostor u kojem je moguće formirati razdvajajuću hiperravan [19].



Slika 16. Preslikavanje iz nelinearnog prostora u višedimenzionalni gde je moguće linearno rastavljanje

Takva funkcija se naziva jezgro (eng. kernel). Jedno od popularnijih jezgara je Gausovo:

$$K(x, y) = e^{-\frac{\|x-y\|^2}{2\sigma^2}} \quad (13)$$

Ovo jezgro predstavlja funkciju sličnosti između dve tačke. Ako su tačke slične vrednost funkcije je bliska 1. Sa druge strane, ako je razlika velika, vrednost funkcije jezgra je bliska 0.

6.2. Pregled popularnih jezgra koji se koriste u SVM

Linearno jezgro

Najjednostavnije jezgro. Definiše se skalarnim proizvodom $\langle x, y \rangle$ plus konstanta c .

$$K(x, y) = x^T z + c \quad (14)$$

Polinomijalno jezgro

Jezgro pogodno za situacije kada su ulazni podaci dobro normalizovani. Parametri su koeficijent nagiba α , konstanta c i stepen polinomijala d .

$$K(x, y) = (\alpha x^T + c)^d \quad (15)$$

Gausovo jezgro

Primer jezgra sa radijalnom funkcijom osnove.

$$K(x, y) = e^{-\frac{\|x-y\|^2}{2\sigma^2}} \quad (16)$$

Parametar σ velikim delom utiče na ponašanje jezgra, Ako mu se dodeli prevelika vrednost jezgro će se ponašati skoro u potpunosti linearno. Sa druge strane, ako mu se dodeli premala vrednost, funkcija će biti pod prevelikim uticajem šuma u podacima.

Laplasijanovo jezgro

Je takođe jezgro za radijalnom funkcijom osnove i veoma je slično Gausovom jezgro osim što je manje osetljiv na promene vrednosti σ parametra.

$$K(x, y) = e^{-\frac{|x-y|}{2\sigma}} \quad (17)$$

Jezgro hiperboličke tangente (Sigmoidno)

Sigmoidno jezgro je našlo veliku primenu pri izučavanju veštačkih neuronskih mreža, pošto SVM koristeći ovo jezgro postaje ekvivalentno dvoslojnoj veštačkoj neuronskoj mreži.

$$K(x, y) = \tanh(\alpha x^T y + c) \quad (18)$$

Ima dva parametra: nagib α i konstantu c . Uobičajena vrednost za α je $\frac{1}{N}$ gde je N dimenzija ulaznih podataka.

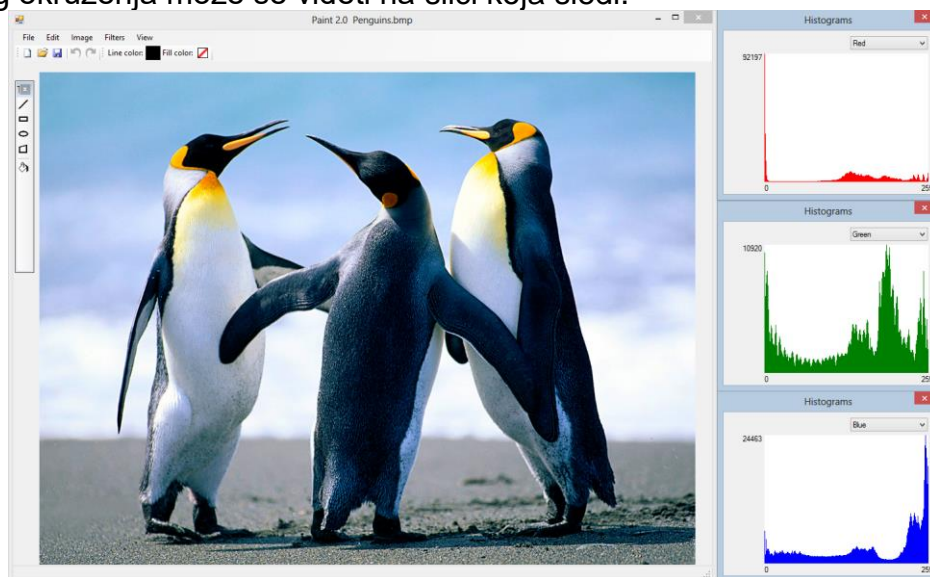
7. Računarska i softverska platforma

Za potrebe ovog istraživanja napravljena je aplikacija koja omogućava da se pretpostavke empirijski provere, kao i da se uporede rezultati dobijeni korišćenjem veštačkih neuronskih mreža i SVM algoritma.

Aplikacija se sastoji iz dva dela. Prvi deo je modifikacija već postojećeg programa Paint 2.0 [20] izrađenog u .NET WinForms tehnologiji, a drugi deo je implementacija veštačkih neuronskih mreža i SVM algoritama.

7.1. Paint 2.0

Aplikacija je inicijalno bila napravljena za potrebe predmeta Računarska Grafika 2 i u njoj je implementirano efikasno crtanje svih geometrijskih primitiva, osnovne operacije za modifikaciju slika (kontrast, boje, konvertovanje u nijanse sive, itd.) kao i osnovni algoritmi računarske vizije (eng. Computer vision) kao što su detekcija ivica, Laplasova detekcija ivica i algoritmi uklapanja histograma (engl. Histogram matching). Izgled korisničkog okruženja može se videti na slici koja sledi.



Slika 17. Izgled Paint 2.0 korisničkog okruženja sa histogramima za datu sliku

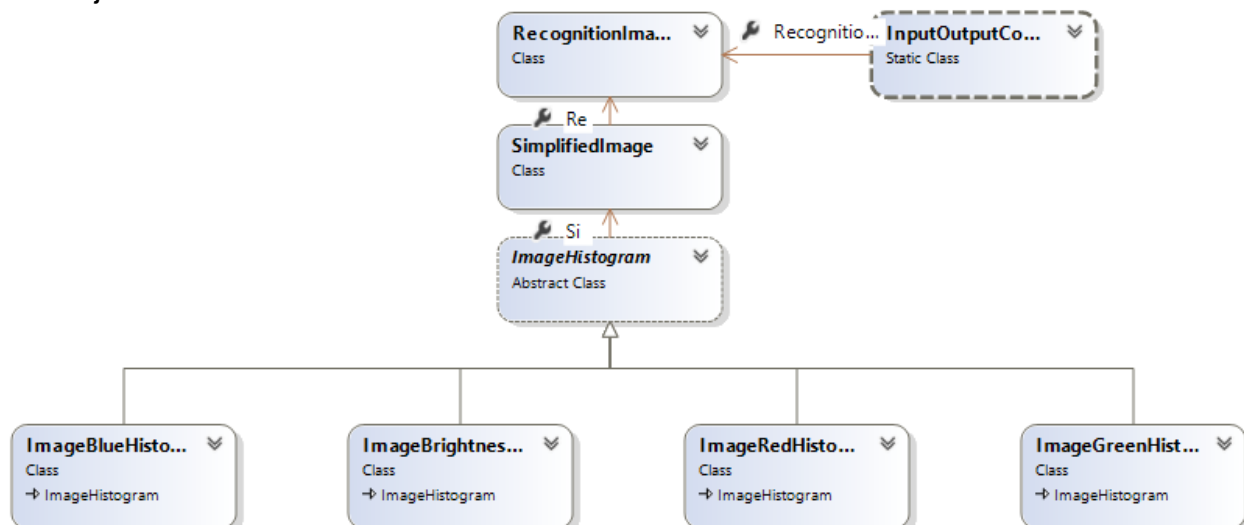
Sastavni deo programa je prikaz histograma u realnom vremenu. Akcenat prilikom pravljenja prikaza histograma je bio na efikasnosti, da bi se mogao koristiti u realnom vremenu. Testiranjem algoritama, razvijenog za potrebe aplikacije Paint 2.0 došlo se do zaključka da je dovoljno efikasan za potrebe izračunavanja histograma za veliki broj slika. Naročito kada se uzme u obzir da se ne moraju obrađivati u realnom vremenu.

Struktura programa

Svaka slika (klasa *SimplifiedImage* u programu) se čuva sa atributima:

1. Putanja do slike,
2. Klasa slike,
3. Broj nijansa histograma,
4. Tri objekta koji predstavljaju histogram slike (*ImageHistogram*) po nijansama crvene, plave i zelene boje,
5. Histogram baziran na intenzitetu.

Radi jednostavnijeg rada sa programom, za klasu slike se uzima ime datoteke roditelja slike.



Slika 18. Dijagram klasa zaduženih za formiranje histograma iz slika

ImageHistogram je apstraktna klasa koja implementira obrazac dizajna pod nazivom Factory. Ova klasa zadužena je za kreiranje odgovarajuće potklase u zavisnosti od izabrane boje kao i za serijalizaciju i deserijalizaciju histograma. Prilikom inicijalizacije kreiraju se dva histograma: jedan sa apsolutnim brojem piksela odgovarajuće nijanse i drugi sa procentualnim brojem piksela po nijansi u odnosu na ukupan broj piksela. Izračunavanje nijanse histograma boja se realizuje jednostavnim prebrojavanjem piksela koji imaju određenu nijansu kao komponentu u RGB modelu. Prilikom izračunavanja histograma intenziteta koristi se Formula 2.2 definisana u poglavlju 3.

Lista svih obrađenih slika čuva se u klasi *RecognitionImageSet* koja prilikom inicijalizacije očekuje putanju do početnog foldera u kome se nalaze sve slike. Aplikacija skenira sve podfoldere počevši od zadatog foldera i kreira *SimplifiedImage* objekte za svaku pronađenu sliku. Podržani su standardni formati slika kao što su JPEG, GIF, PNG i BMP.

Posle inicijalizacije svih slika i izračunavanja vrednosti za sve histograme *RecognitionImageSet* klasa serijalizuje sve histograme u datoteku u csv formatu koji se može sačuvati i dalje obrađivati. Primer formata izlazne datoteke može se videti na Slici 16.

Ovaj deo programa je prilagođen za efikasno konvertovanje slika u histograme kao i izvoz u format koji se kasnije može koristiti kao ulazna vrednost za veštačke neuronske mreže i SVM algoritam. Da bi se moglo proveriti u kolikoj meri rezultat zavisi od

preciznosti histograma, moguće je da se histogram generiše sa 32, 64 i 256 nijansa po boji.

| Klasa | Putanja do slike | Vrednosti po nijansama... |
|-------|------------------|---------------------------|
|-------|------------------|---------------------------|

Slika 19. Obrazac izlazne .csv datoteke

7.2. Prepoznavanje slike na osnovu histograma

Drugi deo aplikacije se sastoji od implementacije SVM algoritama i veštačke neuronske mreže. Pri realizaciji tog dela aplikacije korišćena je Accord.NET biblioteka u kojoj su implementirana oba algoritma.

Accord.NET je „open source“ biblioteka za naučna izračunavanja u .NET-u. Nastala je proširivanjem AForge.NET biblioteke i uključuje veliki izbor algoritama za naučna izračunavanja, statističku obradu podataka, mašinsko učenje, prepoznavanje obrazaca, računarsku viziju i računarski sluh (eng. Computer audition). Biblioteka nudi veliki izbor odgovarajućih distribucija verovatnoća, algoritama učenja, testova i jezgara [21].

Za potrebe klasifikacije slika uz pomoć metode podržavajućih vektora izabran je SVM sa Gausovim jezgrom. Pošto je SVM po svojoj prirodi binarni klasifikator, korišćena je njegova ekstenzija po principu jedan na jedan [22]. Tako je kompleksan problem koji se odnosio na više klasa razbijen na $n*(n-1)/2$ jednostavnijih, binarnih problema.

Za potrebe klasifikacije slika uz pomoć veštačkih neuronskih mreža izabrana je Bolcmanova mreža sa ograničenjem.

Ulazni podatak za oba modula je csv (eng. Comma Separated Value) dokument koji je generisan na kraju poglavlja Paint 2.0 (Slika 20.). Svaki red odgovara po jednoj slici predstavljenoj svojom klasom, putanjom do slike i nizom vrednosti histograma. Na osnovu ovih podataka kreiraju se dve matrice: prva koja sadrži samo jednu kolonu sa klasama slika i drugu koja sadrži vrednosti histograma svih slika. Druga matrica je dimenzija $n * 3m$ gde je n ukupan broj slika a m broj nijansi histograma po boji.

Struktura modula za kreiranje RBM

Biblioteka Accord.NET već u sebi sadrži sve potrebne klase za uspešno kreiranje i korišćenje Bolcmanove mreže sa ograničenjem. Upotrebljena je Gausova aktivaciona funkcija koja je implementirana u *Accord.Neuro.ActivationFunctions.GaussianFunction* klasi.

Na osnovu ovako kreirane funkcije kreirana je RBM koja se nalazi u *Accord.Neuro.Networks.RestrictedBoltzmannMachine*.

Za potrebe učenja ovako dobijene mreže korišćena je metoda gradijentnog spusta koju implementira klasa *Accord.Neuro.Learning.ContrastiveDivergenceLearning*. Metoda gradijentnog spusta kao parametre uzima ubrzanje, koeficijent učenja i koeficijent opadanja težinskih faktora čije su optimalne vrednosti nađene empirijskim putem [2].

Struktura modula za kreiranje SVM

Na sličan način kao što je implementiran modul za RBM kreira se i modul za metode podržavajućih vektora. Na osnovu ulaznih vektora izračunat je parametar Sigma (σ u Formula 19.) korišćenjem ugrađene funkcije *Estimate()* Gausovog jezgra koja se može naći u klasi *Accord.Statistics.Kernels.Gaussian*.

Na osnovu dobijene vrednosti koeficijenta Sigma kreira se novo Gausovo jezgro. Dalje, na osnovu ovog jezgra kreira se SVM koji se može naći u klasi *Accord.MachineLearning.VectorMachines.KernelSupportVectorMachine*.

Učenje se odvija regresijom implementiranom u klasi *Accord.MachineLearning.VectorMachines.Learning.SequentialMinimalOptimizationRegression*, sa parametrom C (Formula 20.) čija je vrednost izračunata na osnovu ulaznih vektora metodom minimalne sekvencijalne optimizacije implementirane u klasi *Accord.MachineLearning.VectorMachines.Learning.SequentialMinimalOptimization*.

8. Izbor optimalnih parametara

Jedan od najvećih izazova prilikom implementacije bilo kog algoritma mašinskog učenja je izbor optimalnih parametara (Gausov koeficijent i prihvatljiva empirijska greška za SVM; a za RBM broj slojeva, koeficijent učenja, itd.)

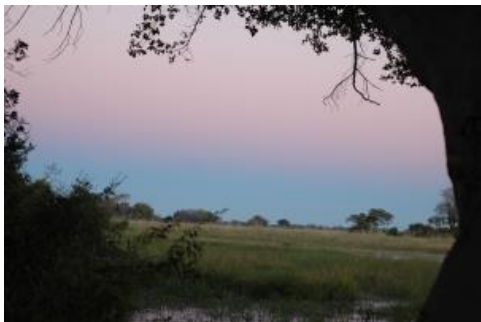
8.1. Izbor slika

Pošto se većina istraživanja bazirana na mašinskom učenju bavi veoma uskim temama (prepoznavanje lica, segmentacijom, promena na koži, tkivu, itd.), bio je pravi izazov naći slike generalne namene (priroda, životinje, portreti, itd.) koje sadrže neophodne klase. Pronađena je baza slika¹ pod nazivom „Natural images from the birthplace of the human eye“ koja je već korišćena u radu pod istim imenom [23].

Korišćene slike su izuzetno visoke rezolucije (3008x2000), klasifikovane su po predmetu (Slika 21.): putevi, divlje životinje, drveće, nebo, reke, noć, itd. Sve ukupno je korišćeno 348 slika raspoređenih u 9 kategorija.

Prednost korišćenog algoritma je u tome što ne zavisi od rezolucije slike jer sam proces pretvaranja slike u njen histogram taj podatak se gubi i on ne utiče na dalju obradu. Naravno, ovo važi ako se ne koriste apsolutne vrednosti broja piksela određene nijanse boja, nego procenat učešća date nijanse u celom histogramu za datu boju.

¹ <http://tofu.psych.upenn.edu/~upennidb/gallery2/Zipper/index.php?file=Zipper/lists/XjaEDHgOcnSOxusQJBg3p3upTedKXZu0>



Slika 22. Primer slika u izabranoj bazi

8.2. Izbor parametara za RBM

Poštujući preporuke izložene u radovima G. Hinton [2] i testiranjem za razne vrednosti parametara, zaključeno je da se najbolje vrednosti dobijaju za sledeće vrednosti parametara:

32 vrednosti nijansi po boji

| | |
|--|--|
| Ukupan broj ulaznih parametara | 96 |
| Tip neurona | Gausov |
| Broj skrivenih slojeva | 2 |
| Broj neurona u skrivenim slojevima | 30, 9 |
| Koeficijent učenja | 0,01 |
| Početne vrednosti vidljivih težina | Gausove random vrednosti sa Standardnom devijacijom 0,01 |
| Ubrzanje | 0,9 |
| Koeficijent opadanja težinskih faktora | 0,001 |

Tabela 1 Izabrane vrednosti parametara prilikom treniranja RBM sa 32 nijanse

64 vrednosti nijansi po boji

| | |
|--|--|
| Ukupan broj ulaznih parametara | 192 |
| Tip neurona | Gausov |
| Broj skrivenih slojeva | 2 |
| Broj neurona u skrivenim slojevima | 50, 9 |
| Koeficijent učenja | 0,01 |
| Početne vrednosti vidljivih težina | Gausove random vrednosti sa Standardnom devijacijom 0,01 |
| Ubrzanje | 0,9 |
| Koeficijent opadanja težinskih faktora | 0,001 |

Tabela 2 Izabrane vrednosti parametara prilikom treniranja RBM sa 64 nijansi

256 vrednosti nijansi po boji

| | |
|--|--|
| Ukupan broj ulaznih parametara | 768 |
| Tip neurona | Gausov |
| Broj skrivenih slojeva | 2 |
| Broj neurona u skrivenim slojevima | 50, 9 |
| Koeficijent učenja | 0,01 |
| Početne vrednosti vidljivih težina | Gausove random vrednosti sa Standardnom devijacijom 0,01 |
| Ubrzanje | 0,9 |
| Koeficijent opadanja težinskih faktora | 0,001 |

Tabela 3 Izabrane vrednosti parametara prilikom treniranja RBM sa 256 nijansi

8.3. Izbor parametara za SVM

Poštujući preporuke izložene u radovima Chih-Wei Hsu, Shih-Shung Chang, i sar. [24] i testiranjem za razne vrednosti parametara, zaključeno je da se najbolje vrednosti dobijaju za sledeće vrednosti parametara:

32 vrednosti nijansi po boji

| | |
|--------------------------------|---------|
| Ukupan broj ulaznih parametara | 96 |
| Jezgro | Gausovo |
| Sigma | 14,37 |
| C | 1,07 |

Tabela 4 Izabrane vrednosti parametara prilikom treniranja SVM sa 32 nijanse

64 vrednosti nijansi po boji

| | |
|--------------------------------|---------|
| Ukupan broj ulaznih parametara | 192 |
| Jezgro | Gausovo |
| Sigma | 5,09 |
| C | 1,03 |

Tabela 5 Izabrane vrednosti parametara prilikom treniranja SVM sa 64 nijansi

256 vrednosti nijansi po boji

| | |
|--------------------------------|---------|
| Ukupan broj ulaznih parametara | 768 |
| Jezgro | Gausovo |
| Sigma | 2,62 |
| C | 1,05 |

Tabela 6 Izabrane vrednosti parametara prilikom treniranja SVM sa 256 nijansi

Vrednost C predstavljaju koeficijent prilagođavanja skupu na kome se vrši učenje (opisano detaljnije u odeljku 6) i dobijene su sekvencijalnom minimalnom optimizacijom. Dobijene vrednosti (približno 1) su snažan indikator da bi i linearno jezgro bilo sasvim dovoljno. Međutim, pošto je Gausovo jezgro u najmanju ruku dobro koliko i linearno [25], nije utvrđena dalja potreba za ispitivanjem linearnog jezgra.

Vrednosti Sigma su dobijene heuristikom za dobijanje sigma funkcije u jezgru sa radijalnom osnovom [26].

9. Rezultati – veštačke neuronske mreže

Dobijeni rezultati za RBM su se u većoj ili manjoj meri poklopili sa očekivanim. Aproksimacija na 32 vrednosti nijansi po boji je bilo preveliko uopštenje i klasifikacija nije dala najbolje rezultate. Sa druge strane, primer sa svim nijansama je kao prvo bio daleko teže izračunati (28 sekundi u odnosu na 253 za 50 ciklusa po sloju), a kao drugo dao i lošije rezultate od primera sa 64 nijanse boja.

Pretpostavka je da je primer sa 256 nijansi dao i suviše detalja na kojima se algoritam „zadržao“ i dao lošije rezultate. Pokušano je i sa povećanjem broja slojeva u veštačkoj neuronskoj mreži i sa povećanjem broja neurona po sloju, ali nisu dobijene vrednosti sa manjom greškom.

Slede tablice konfuzije u kojima su tabelarno prikazani rezultati. Tri odvojene tabele odgovaraju vrednostima preciznosti histograma koji su obrađivani: 32, 64 i 256 nijansa po boji.

U kolonama se nalaze „stvarne“ klase slike, a u redovima se nalazi klasa slike koju je RBM dodelio datoj slici. Vrednosti na glavnoj dijagonali broj tačno pogodjenih klasa (stvarna klasa i pogodena klasa je ista), dok vrednosti van glavne dijagonale predstavljaju greške prilikom klasifikacije.

U donjem desnom uglu se nalazi ukupan procenat tačnosti koji prestavlja odnos sume vrednosti na glavnoj dijagonali i ukupnog broja slika.

32 vrednosti nijansi po boji

| | | Stvarna klasa | | | | | | | | | |
|----------------|--------|---------------|-----|-----|-----|-----|-----|------|-----|-------|--------|
| | | 01A | 17A | 23A | 26B | 27A | 32A | 33B1 | 41A | 61A | Ukupno |
| Pogođena klasa | cd01A | 1 | 4 | 5 | 1 | 0 | 2 | 1 | 4 | 0 | 18 |
| | cd17A | 0 | 35 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 36 |
| | cd23A | 1 | 0 | 42 | 2 | 0 | 5 | 0 | 0 | 0 | 50 |
| | cd26B | 0 | 4 | 3 | 11 | 0 | 0 | 1 | 5 | 0 | 24 |
| | cd27A | 0 | 0 | 4 | 0 | 14 | 13 | 0 | 0 | 0 | 31 |
| | cd32A | 0 | 1 | 0 | 0 | 5 | 55 | 0 | 0 | 0 | 61 |
| | cd33B1 | 0 | 0 | 0 | 2 | 0 | 0 | 22 | 0 | 4 | 28 |
| | cd40A | 2 | 0 | 1 | 0 | 0 | 1 | 0 | 42 | 0 | 46 |
| | cd61A | 0 | 0 | 0 | 2 | 0 | 0 | 7 | 0 | 45 | 54 |
| | Ukupno | 4 | 44 | 56 | 18 | 19 | 76 | 31 | 51 | 49 | 348 |
| | | | | | | | | | | 76,7% | |

Tabela 7 Matrica konfuzije dobijenih rezultata za RBM sa 32 nijanse

64 vrednosti nijansi po boji

| Pogođena klasa | Stvarna klasa | | | | | | | | | Ukupno |
|----------------|---------------|-----|-----|-----|-----|-----|------|-----|-----|--------|
| | 01A | 17A | 23A | 26B | 27A | 32A | 33B1 | 41A | 61A | |
| cd01A | 2 | 1 | 2 | 1 | 0 | 0 | 0 | 0 | 0 | 6 |
| cd17A | 0 | 42 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 42 |
| cd23A | 1 | 0 | 51 | 1 | 0 | 3 | 0 | 0 | 0 | 56 |
| cd26B | 0 | 1 | 2 | 14 | 0 | 0 | 1 | 5 | 0 | 23 |
| cd27A | 0 | 0 | 1 | 0 | 16 | 15 | 0 | 0 | 0 | 32 |
| cd32A | 0 | 0 | 0 | 0 | 3 | 58 | 0 | 0 | 0 | 61 |
| cd33B1 | 0 | 0 | 0 | 0 | 0 | 0 | 28 | 0 | 2 | 30 |
| cd40A | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 46 | 0 | 47 |
| cd61A | 0 | 0 | 0 | 2 | 0 | 0 | 2 | 0 | 47 | 51 |
| Ukupno | 4 | 44 | 56 | 18 | 19 | 76 | 31 | 51 | 49 | 348 |
| | | | | | | | | | | 87,3% |

Tabela 8 Matrica konfuzije dobijenih rezultata za RBM sa 64 nijansi

256 vrednosti nijansi po boji

| Pogođena klasa | Stvarna klasa | | | | | | | | | Ukupno |
|----------------|---------------|-----|-----|-----|-----|-----|------|-----|-----|--------|
| | 01A | 17A | 23A | 26B | 27A | 32A | 33B1 | 41A | 61A | |
| cd01A | 1 | 4 | 5 | 3 | 1 | 2 | 1 | 4 | 0 | 21 |
| cd17A | 0 | 34 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 35 |
| cd23A | 1 | 0 | 41 | 2 | 0 | 3 | 0 | 0 | 0 | 47 |
| cd26B | 0 | 4 | 3 | 9 | 0 | 0 | 1 | 5 | 0 | 22 |
| cd27A | 0 | 0 | 4 | 0 | 12 | 18 | 0 | 0 | 0 | 34 |
| cd32A | 2 | 2 | 0 | 0 | 6 | 50 | 0 | 0 | 0 | 60 |
| cd33B1 | 0 | 0 | 0 | 2 | 0 | 0 | 22 | 0 | 4 | 28 |
| cd40A | 0 | 0 | 2 | 0 | 0 | 3 | 0 | 42 | 0 | 47 |
| cd61A | 0 | 0 | 0 | 2 | 0 | 0 | 7 | 0 | 45 | 54 |
| Ukupno | 4 | 44 | 56 | 18 | 19 | 76 | 31 | 51 | 49 | 348 |
| | | | | | | | | | | 73,6% |

Tabela 9 Matrica konfuzije dobijenih rezultata za RBM sa 256 nijansi

10. Rezultati – SVM

SVM je algoritam primarno namenjen za binarnu klasifikaciju. Ipak, korišćena biblioteka već ima implementiran algoritam koji pretvara više klasni problem sa 9 klasa u 36 binarnih problema ($9 * (9 - 1) / 2$).

Slično kao i za RBM slede tablice konfuzije u kojima su tabelarno prikazani rezultati. Tri odvojene tabele odgovaraju vrednostima preciznosti histograma koji su obrađivani: 32, 64 i 256 nijansa po boji.

U kolonama se nalaze „stvarne“ klase slika, a u redovima se nalazi klasa slike koju je SVM dodelio datoj slici. Vrednosti na glavnoj dijagonali broj tačno pogođenih klasa (stvarna klasa i pogođena klasa je ista), dok vrednosti van glavne dijagonale predstavljaju greške prilikom klasifikacije.

U donjem desnom uglu se nalazi ukupan procenat tačnosti koji prestavlja odnos sume vrednosti na glavnoj dijagonali i ukupnog broja slika.

32 vrednosti nijansi po boji

| | Stvarna klasa | | | | | | | | | Ukupno |
|-------------------------|---------------|-----|-----|-----|-----|-----|------|-----|-----|--------|
| | 01A | 17A | 23A | 26B | 27A | 32A | 33B1 | 41A | 61A | |
| Pogođena klasa cd01A | 1 | 0 | 3 | 1 | 0 | 4 | 4 | 6 | 1 | 18 |
| cd17A | 0 | 36 | 5 | 0 | 0 | 0 | 0 | 1 | 0 | 36 |
| cd23A | 1 | 0 | 39 | 3 | 0 | 3 | 0 | 0 | 10 | 50 |
| cd26B | 0 | 7 | 4 | 8 | 0 | 0 | 2 | 8 | 0 | 24 |
| cd27A | 0 | 0 | 2 | 0 | 13 | 17 | 0 | 0 | 1 | 31 |
| cd32A | 0 | 1 | 0 | 0 | 3 | 48 | 0 | 4 | 0 | 61 |
| cd33B1 | 1 | 0 | 0 | 1 | 0 | 0 | 20 | 0 | 5 | 28 |
| cd40A | 1 | 0 | 3 | 0 | 0 | 2 | 0 | 32 | 0 | 46 |
| cd61A | 0 | 0 | 0 | 5 | 3 | 2 | 5 | 0 | 32 | 54 |
| Ukupno | 4 | 44 | 56 | 18 | 19 | 76 | 31 | 51 | 49 | 348 |
| | | | | | | | | | | 65,8% |

Tabela 10 Matrica konfuzije dobijenih rezultata za SVM sa 32 nijanse

64 vrednosti nijansi po boji

| | | Stvarna klasa | | | | | | | | Ukupno | |
|----------------|--------|---------------|-----|-----|-----|-----|-----|------|-----|--------|-----|
| | | 01A | 17A | 23A | 26B | 27A | 32A | 33B1 | 41A | | 61A |
| Pogođena klasa | cd01A | 2 | 3 | 2 | 1 | 0 | 1 | 0 | 0 | 0 | 6 |
| | cd17A | 0 | 35 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 42 |
| | cd23A | 0 | 0 | 45 | 3 | 0 | 1 | 1 | 0 | 3 | 56 |
| | cd26B | 2 | 2 | 9 | 12 | 0 | 0 | 1 | 3 | 0 | 23 |
| | cd27A | 0 | 0 | 0 | 0 | 11 | 9 | 0 | 0 | 0 | 32 |
| | cd32A | 0 | 0 | 0 | 0 | 3 | 63 | 0 | 0 | 0 | 61 |
| | cd33B1 | 0 | 4 | 0 | 0 | 4 | 0 | 22 | 0 | 1 | 30 |
| | cd40A | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 48 | 0 | 47 |
| | cd61A | 0 | 0 | 0 | 2 | 1 | 2 | 7 | 0 | 45 | 51 |
| | Ukupno | 4 | 44 | 56 | 18 | 19 | 76 | 31 | 51 | 49 | 348 |
| | | | | | | | | | | 81,3% | |

Tabela 11 Matrica konfuzije dobijenih rezultata za SVM sa 64 nijansi

256 vrednosti nijansi po boji

| | | Stvarna klasa | | | | | | | | Ukupno | |
|----------------|--------|---------------|-----|-----|-----|-----|-----|------|-----|--------|-----|
| | | 01A | 17A | 23A | 26B | 27A | 32A | 33B1 | 41A | | 61A |
| Pogođena klasa | cd01A | 1 | 4 | 3 | 2 | 1 | 0 | 0 | 4 | 0 | 21 |
| | cd17A | 0 | 33 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 35 |
| | cd23A | 1 | 0 | 42 | 0 | 1 | 7 | 0 | 0 | 0 | 47 |
| | cd26B | 0 | 2 | 5 | 11 | 0 | 0 | 0 | 5 | 0 | 22 |
| | cd27A | 0 | 0 | 2 | 0 | 14 | 11 | 0 | 0 | 0 | 34 |
| | cd32A | 2 | 5 | 0 | 0 | 3 | 55 | 0 | 0 | 0 | 60 |
| | cd33B1 | 0 | 0 | 0 | 2 | 0 | 0 | 27 | 0 | 2 | 28 |
| | cd40A | 0 | 0 | 3 | 0 | 0 | 3 | 0 | 42 | 0 | 47 |
| | cd61A | 0 | 0 | 0 | 2 | 0 | 0 | 4 | 0 | 47 | 54 |
| | Ukupno | 4 | 44 | 56 | 18 | 19 | 76 | 31 | 51 | 49 | 348 |
| | | | | | | | | | | 78,2% | |

Tabela 12 Matrica konfuzije dobijenih rezultata za SVM sa 256 nijansi

11. Zaključak

Budući da RBM predstavlja jedan od najpopularnijih algoritama mašinskog učenja moglo se pretpostaviti da će i u ovom slučaju dati najbolje rezultate, što se potvrdilo i u praksi. Dobijena je maksimalna tačnost od 87,3% za veštačke neuronske mreže sa 64 nijanse boje u odnosu na 81,3% za SVM sa istim brojem nijansi.

Smanjenje broja nijansi boja kao ulaznih parametara se pokazalo kao korisno uopštenje pošto su se najbolji rezultati dobili korišćenjem 64 nijansi. Sa druge strane, 32 nijanse su se pokazale kao nedovoljne – pojavio se prevelik gubitak informacija.

U slučaju 256 boja, SVM algoritam se pokazao značajno boljim od veštačkih neuronskih mreža, time opravdavajući svoju reputaciju robusnosti kada su ulazni podaci visoke dimenzije.

Iako se u ovom istraživanju pokazalo da prepoznavanje slika može da se koristi za grubu klasifikaciju, može se uočiti da u opštem slučaju gubitak prostornog rasporeda detalja na slici predstavlja preveliki gubitak informacija da bi bilo moguće osloniti se samo na histograme boja radi klasifikacije slika.

Velika moguća prednost RBM algoritma je da može da radi i ako skup ulaznih podataka nema unapred određene klase, odnosno može da se koristi kao alat za učenje bez nadgledanja i praktično sam da uradi klasterizaciju ulaznih slika. Ovakva mogućnost nije trivijalna pošto postoji veliki izbor slika na internetu pogodnih za testiranje algoritama mašinskog učenja koje nisu unapred klasifikovane.

Potpunijom analizom vremena izvršavanja uočava se da su pretprocesiranje slika u histograme i normalizacija ulaznih vrednosti vektora uzeli neočekivano puno vremena. U tu svrhu bi u daljim istraživanjima bilo poželjno proveriti da li je moguće brže izračunavanje vrednosti histograma slika na procesorima grafičkih kartica (GPU) uz pomoć, na primer, NVIDIA CUDA [27].

Pretpostavka da će SVM algoritam biti značajno efikasniji se u ovom slučaju nije pokazala ispravnom. Razlog tome je relativno mali broj slika, veliki broj rezultujućih klasa kao i pretprocesiranje koje je uzelo više vremena nego što je bilo prvobitno pretpostavljeno. Zbog toga je i procentualni udeo vremena, koje je utrošeno na SVM i RBM algoritme, bio srazmerno manji, pa je imao i manji uticaj na brzinu izvršavanja celokupnog programa. Povećanje broja slika bi sasvim sigurno uticalo na brzinu izvršavanja u korist SVM algoritma.

Dalje povećanje efikasnosti oba algoritama bi se moglo postići smanjivanjem broja klasa. Jedan od mogućih načina je ograničiti broj klasa: na osnovne boje, tada bi postojale samo tri klase ili, ako se proširi podskup boja i na izvedene, 6 klasa. U tom slučaju bi se potencijalno izgubili finiji detalji kao što su grupisanje boja u slici. Na primer, uz zelenu (lišće, trava) često ide plava (nebo, voda), uz boju kože često ide crna (portret) i sl.

Budući da izbor slika i klasa u ovom istraživanju nije u potpunosti slučajna, dobijeni rezultati verovatno predstavljaju jedan od najboljih slučajeva u korist prepoznavanja sadržaja. Međutim, i pored toga, pokazana je dobit korišćenjem ovog algoritma koja ga kvalifikuje da bude komponenta u nekom složenijem algoritmu prepoznavanja slika.

12. Reference

- [1] T. Tieleman, “Training Restricted Boltzmann Machines using Approximations to the Likelihood Gradient,” *Proc. 25th Int. Conf. Mach. Learn.*, vol. 307, 2008.
- [2] G. Hinton, “A Practical Guide to Training Restricted Boltzmann Machines,” *Computer (Long. Beach. Calif.)*, vol. 9, no. 3, p. 1, 2010.
- [3] K. Smith, “Face Recognition,” *NSTC Subcommittee on Biometrics*, no. January 2001, pp. 1–10, 2006.
- [4] I. Guyon and A. Elisseeff, “An Introduction to Variable and Feature Selection,” *J. Mach. Learn. Res.* 3, vol. 3, pp. 1157–1182, 2003.
- [5] M. S. Castelhana and J. M. Henderson, “The influence of color on the perception of scene gist,” *J. Exp. Psychol. Hum. Percept. Perform.*, vol. 34, no. 3, pp. 660–675, 2008.
- [6] R. C. Gonzalez, R. E. Woods, and P. Hall, *Digital Image Processing*, Second Edi. New Jersey: Prentice-Hall, Inc., 2002.
- [7] J. Ž. Tasić, “Procesiranje slikovnih analogija neuronskim mrežama,” UNIVERSITY OF BELGRADE, 2012.
- [8] J. C. Russ, *The Image Processing Handbook, Third Edition*, 3rd ed. CRC Press LLC, 1998.
- [9] W. S. McCulloch and W. Pitts, “A logical calculus of the ideas immanent in nervous activity,” *Bull. Math. Biophys.*, vol. 5, pp. 115–133, 1943.
- [10] B. Krose and P. Smagt, *An introduction to Neural Networks*. The University of Amsterdam, 1996.
- [11] M. Minsky and S. Papert, “Perceptrons: An Introduction to Computational Geometry,” *MIT Press*, p. 268, 1969.
- [12] A. Đurišić, “Implementacija i evaluacija tehnika klasifikacije teksta zasnovanih na Bajesovoj teoremi,” 2014.

- [13] M. Petrović, “Osnovi veštačkih neuronskih mreža i značaj njihove primene,” *Zb. Rad.* 20, 2011.
- [14] A. E. Bryson and H. Yu-Chi, “Applied optimal control: optimization, estimation, and control,” *Blaisdell Publ. Co. or Xerox Coll. Publ.*, 1969.
- [15] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning representations by back-propagating errors,” *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.
- [16] S. J. Russell and P. Norvig, “Artificial Intelligence A Modern Approach.,” p. 578, 1969.
- [17] I. Sutskever and T. Tieleman, “On the Convergence Properties of Contrastive Divergence,” *Proc. 13th Int. Conf. Artif. Intell. Stat.*, pp. 789–795, 2010.
- [18] A. Kartelj, N. Mitić, V. Filipović, and D. Tošić, “Electromagnetism-like algorithm for support vector machine parameter tuning,” *Soft Comput. Springer-Verlag*, vol. 18, no. 10, pp. 1985–1998, 2014.
- [19] W. S. Noble, “What is a support vector machine?,” *Nat. Biotechnol.*, vol. 24, no. 12, pp. 1565–1567, 2006.
- [20] I. Valjevic and A. Tuba, “Histogram manipulations as a pre-step to image recognition,” in *First international Students Conference on Informatics - ICDD*, 2011.
- [21] C. R. de Souza, “The Accord.NET Framework.” [Online]. Available: <http://accord.googlecode.com>.
- [22] C.-W. Hsu and C.-J. Lin, “A comparison of methods for multiclass support vector machines,” *IEEE Trans. Neural Networks*, vol. 13, no. 2, pp. 415–425, 2002.
- [23] G. Tkačik, P. Garrigan, C. Ratliff, G. Milčinski, J. M. Klein, L. H. Seyfarth, P. Sterling, D. H. Brainard, and V. Balasubramanian, “Natural Images from the Birthplace of the Human Eye,” *PLoS One*, vol. 6, no. 6, 2011.
- [24] H. Chih-Wei, C. Chih-Chung, and L. Chih-Jen, “A Practical Guide to Support Vector Classification,” 2010.
- [25] S. S. Keerthi and C.-J. Lin, “Asymptotic behaviors of support vector machines with

Gaussian kernel,” *Neural Comput.*, vol. 15, no. 7, pp. 1667–89, 2003.

- [26] B. Caputo, K. Sim, F. Furesjo, and A. Smola, “Hyperparameter estimation for the Gaussian Radial Basis kernel,” *Whistler*, 2002.
- [27] R. Shams and R. a Kennedy, “Efficient Histogram Algorithms for NVIDIA CUDA Compatible Devices,” *Memory*, pp. 1–5, 2007.