SVEUČILIŠTE U ZAGREBU

FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

SEMINAR

Primjene opisnika značajki u računalnom vidu

Dražen Dostal

Voditelj: *Doc.dr.sc. Siniša Šegvić*

Zagreb, travanj 2010.

Sadržaj

Uvod

Različiti problemi kojima se bave područja računalnog vida i obrade slike uključuju razne postupke prepoznavanja, traženja ili praćenja objekata na slici ili nizu slika. Objekti se u računalnom vidu kao i u stvarnom životu jednoznačno mogu odrediti opisujući određene karakteristike (npr. geometrijski oblik, veličina, tekstura površine…).

Takve karakteristike u ovom području nazivamo značajkama, a kada komponente vektorskog prikaza značajke izlučimo iz predmetne slike, tada govorimo o opisniku značajke. Pojedini opisnici značajki mogu biti i izrazito kompleksni te se sastojati od velikog broja komponenti, pri čemu vrijedi pravilo da složeniji opisnik bolje opisuju objekt ali za to troši više računalnih resursa.

U ovom radu se razmatra nekoliko različitih opisnika značajki, njihova primjena te na kraju usporedba u obliku uspješnosti.

Opisnici značajki

Za bilo kakav objekt na slici se mogu naći točke interesa na tom objektu koje pružaju mogućnost za izlučivanje značajki objekta. Takav opis, izvađen iz slike za učenje, se može koristiti za pronalazak objekta na ispitnim slikama koje sadrže više drugih objekata. Kako bi se moglo obaviti pouzdano prepoznavanje objekta, važno je da je skup značajki izlučen iz slika za učenje otporan na promjene u veličini slike, šum, osvjetljenje i lokalna geometrijska iskrivljenja.

U ovom dijelu se opisuje nekoliko takvih opisnika značajki. To su redom: histogram orijentiranih gradijenata (HOG), transformacija značajki invarijantna na skaliranje (Scale-Invariant Feature Transform, SIFT), Haarove značajke i lokalna receptivna polja (LRF).

1.1 Histogram orijentiranih gradijenata

Ovaj pristup se temelji na prebrojavanju različitih orijentacija gradijenata u lokaliziranim dijelovima slike. Slična je metodama histograma orijentacije ruba, SIFT opisnicima te opisnicima temeljenim na kontekstu oblika, ali od njih se razlikuje po tome što koristi gustu mrežu uniformno raspoređenih ćelija i normalizaciju preklapanjem lokalnih kontrasta kako bi se poboljšale performanse postupka.

Osnovna ideja iza opisnika temeljenih na histogramu orijentiranih gradijenata (HOG) je ta da se lokalni izgled i oblik objekta u slici mogu opisati distribucijom intenziteta gradijenata ili orijentacija rubova. Implementacija ovakvih opisnika se može izvesti na način da ulaznu sliku podijelimo na manja spojena područja, koja nazivamo ćelijama, a za svaku ćeliju tada računamo histogram orijentacija gradijenta ili orijentacije rubova na temelju piksela unutar ćelije. Kombinacija ovakvih histograma tada predstavlja opisnik.

Kako bi se poboljšala uspješnost ovog postupka, lokalni histogrami se mogu podvrgnuti normalizaciji kontrasta računanjem mjere intenziteta veće regije slike, koju zovemo blok, te korištenja takve mjere za normalizaciju svih ćelija unutar tog bloka. Rezultat ove normalizacije je bolja invarijantnost na promjene u osvjetljenju ili zasjenjivanju objekata.

HOG opisnik sadrži nekoliko ključnih prednosti nad drugim metodama stvaranja opisnika. Budući da HOG opisnik djeluje na lokaliziranim ćelijama, metoda podupire invarijantnosti na geometrijske i fotometričke transformacije, takve promjene se pojavljuju tek u većim prostornim regijama. Štoviše, kao što je navedeno u [3], grubo prostorno uzorkovanje, fino uzorkovanje orijentacije i jaka normalizacija

lokalne fotometrije dozvoljavaju da se kretanja pojedinačnih tijela pješaka zanemare, tako dugo dok oni zauzimaju ugrubo određen uspravni položaju. HOG opisnik je stoga posebice pogodan za detekciju ljudi u slikama.

Implementaciju algoritma koji koristi HOG opisnike možemo podijeliti u nekoliko koraka:

1. Izračun gradijenta
2. Određivanje orijentacije
3. Grupiranje ćelija u blokove
4. Normalizacija blokova
5. Klasifikacija

1.1.1 Izračun gradijenta

Prvi korak izračuna u većini detektora značajki je pretprocesiranje slike čime se nastoji osigurati normalizirane vrijednosti boje i gama vrijednosti. Ovaj korak se može izostaviti kod HOG opisnika [3], jer se isti rezultat dobije normalizacijom po blokovima. Pretprocesiranje slike tako ima malo utjecaja na performanse. Umjesto toga, prvi korak je izračunavanje gradijenta.

Najčešći način je da se jednostavno primijeni 1-D centrirana, diskretna derivacijska maska u horizontalnom i/ili vertikalnom smjeru. Ova metoda dakle zahtijeva filtriranje intenziteta piksela slike sa sljedećim filtrima:

$$[-1\ 0\ 1]\ i\ [-1\ 0\ 1]^T$$

Dalal i Triggs [3] su testirali i druge, kompleksnije maske kao npr. 3x3 Sobelovu masku (Sobelov operator) ili dijagonalne maske, ali su pružile slabije performanse pri detekciji ljudi. Također su eksperimentirali s Gaussovim zaglađivanjem prije primjene maske, ali su se isto tako pokazali bolji rezultati ako se zaglađivanje izostavi.

1.1.2 Određivanje orijentacije

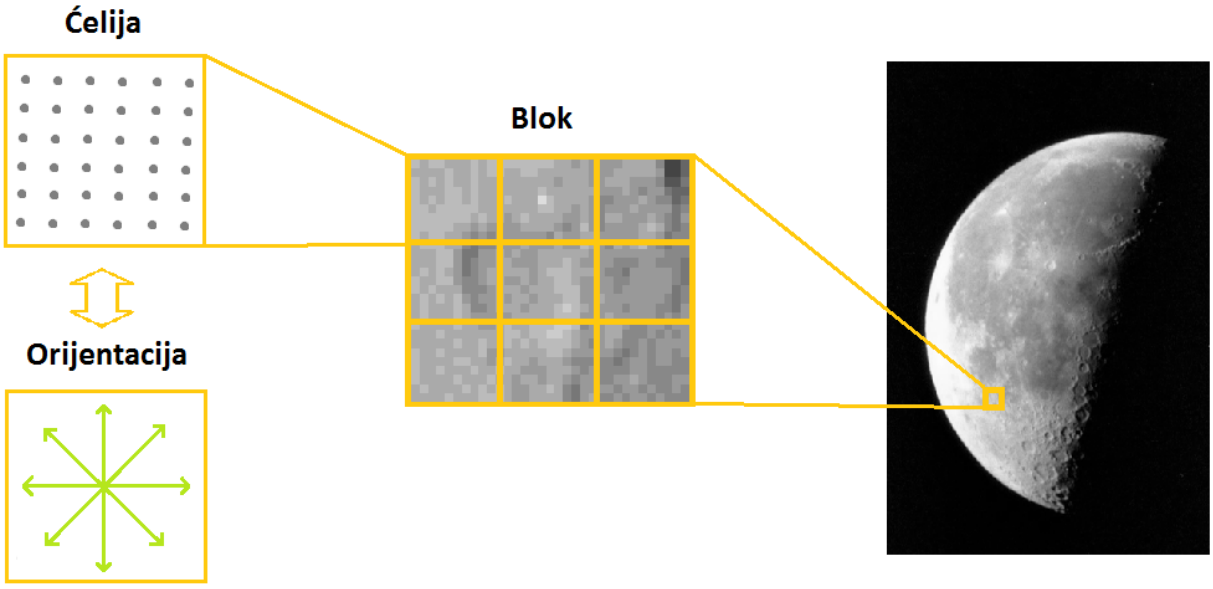
Drugi korak izračuna uključuje stvaranje histograma ćelija. Svaki piksel unutar ćelije sudjeluje u glasanju za orijentaciju na temelju izračunatog gradijenta u njemu. Svaki glas pri ovom postupku može imati svoju težinu koja ovisi o položaju unutar ćelije. Ćelije mogu biti pravokutnog ili kružnog oblika, dok kutovi u histogramu mogu biti jednoliko raspoređeni na interval od 0 do 180 stupnjeva ili od 0 do 360 stupnjeva, ovisno o tome da li se u izračun uzima apsolutni iznos gradijenta ili postoje i negativne vrijednosti.

Dalal i Triggs su u svom radu utvrdili da korištenje apsolutnih gradijenata uz 9 kutova histograma daje najbolje rezultate pri detekciji ljudskih oblika na slici. Što se tiče težinskog glasovanja, doprinos svakog piksela može biti sam iznos gradijenta ili nekih funkcija te veličine (npr. kvadratni korijen ili kvadrat iznosa gradijenta), a pri testiranju je utvrđeno da se dobivaju najbolji rezultati ako se upravo koristi iznos gradijenta.

1.1.3 Grupiranje ćelija u blokove

Da bi se u obzir uzele promijene osvjetljenja i kontrasta, iznosi gradijenata moraju biti lokalno normalizirani, što zahtijeva grupiranje ćelija zajedno u veće, prostorno povezane blokove. HOG opisnik je tada vektor komponenti normaliziranih histograma ćelija iz svih elemenata bloka. Ovi blokovi obično se preklapaju, što znači da svaka ćelija više puta pridonosi izračunu konačnog opisnika. Postoje dva glavna oblika blokova: pravokutni blokovi (*eng. Rectangular HOG*, R-HOG) i kružni blokovi (*eng. Circular HOG*, C-HOG).

R-HOG blokovi su uglavnom kvadratne rešetke, predstavljene s tri parametra: broj ćelija po bloku, broj piksela po ćeliji, te broj različitih kutova po histogramu ćelije. U eksperimentu kojeg su proveli Dalal i Triggs, optimalni parametri su se pokazali kao blokovi s 3x3 ćelije koje obuhvaćaju 6x6 piksela te 9 mogućih kutova (0°-360°) u koje se ćelije mogu grupirati (slika 1). Štoviše, pronašli su da se manja poboljšanja u performansama mogu se dobiti primjenom Gaussovog prostornog prozora unutar svakog bloka prije tabeliranje glasova histograma kako bi se težina piksela oko ruba blokova umanjila.



Slika 1 – Prikaz R-HOG bloka

R-HOG blokovi se mogu činiti vrlo slični SIFT opisnicima, no iako su im formacije slične, R-HOG blokovi se računaju na gustim mrežama u jednom poravnanju bez orijentacije, dok se SIFT opisnici računaju iz rijetkih, ključnih točaka invarijantnih na skaliranje pri čemu se onda opisnik rotira kako bi se poklopio sa orijentacijom. Osim toga, R-HOG blokovi se koriste zajedno za izvođenje prostorne informacije, dok se SIFT opisnici koriste pojedinačno.

C-HOG blokovi imaju dvije varijante: one s jednom središnjom ćelijom i one sa središnjom ćelijom podijeljenom na kutove. Ovi blokovi se mogu opisati s četiri parametra: broj kutnih i radijalnih smjerova, radijus centralnog smjera, i faktor širenja radijusa dodatnih radijalnih smjerova. Dalal i Triggs su otkrili da obije varijante daju jednake performanse, te da su najbolji parametri: dva radijalna smjera s četiri kutna, centralni radijusu od 4 piksela i faktor širenja 2. Usrednjavanje Gaussovim filtrom nije pružilo bolje performanse. C-HOG blokovi se doimaju slični opisnicima temeljenim na kontekstu oblika, ali se u velikoj mjeri razlikuju po tome što C-HOG blokovi sadrže ćelije sa više mogućih orijentacija, dok opisnici temeljeni na kontekstu oblika koriste jedino prebrojavanje prisutnosti pojedinih rubova.

1.1.4 Normalizacija blokova

Dalal i Triggs su ispitali četiri različite metode normalizacije blokova. Svaki blok (ćelije unutar tog bloka) se zasebno normalizira na temelju svih ćelija bloka. Ovim postupkom se postiže lokalna normalizacija blokova te cijeli postupa postaje robusniji na promjene u osvjetljenju i zasjenjivanje objekta.

Neka je v nenormalizirani vektor koji sadrži sve histograme ulaznog bloka, neka su $\|v\|_1$ i $\|v\|_2$1-norma i 2-norma te neka je e mala konstanta (prava vrijednost nije bitna), tada faktor normalizacije možemo računati na ove načine:

L2-norm: $$f = \frac{v}{\sqrt{\|v\|_2^2 + e^2}}$$

L1-norm: $$f = \frac{v}{\|v\|_1 + e}$$

L1-sqrt: $$f = \sqrt{\frac{v}{\|v\|_1 + e}}$$

$$\|v\|_k = \left(\sum_{i=1}^{n} |v_i|^k\right)^{\frac{1}{k}}, \quad k = 1,2$$

Dalal i Triggs su utvrdili da L2-norm i L1-sqrt normalizacije daju slične performanse, dok L1-norm normalizacija pruža nešto manje pouzdane performanse, ali sve ove normalizacije su pokazale značajno poboljšanje u usporedbi sa nenormaliziranim podacima.

1.1.5 Klasifikacija

Konačni korak prepoznavanja objekata koristeći HOG opisnike je predavanje opisnika nekakvom sustavu za prepoznavanje temeljenom na nadziranom učenju. Jedan od takvih sustava je i metoda potpornih vektora (Support Vector Machine, SVM). To je binarni klasifikator koji traži optimalnu hiperravninu koja pak predstavlja decizijsku ravninu. Jednom kada se SVM nauči sa slikama koje sadrže određene objekte, tada može donositi odluke o prisutnosti tih objekata na ulaznim slikama koje naravno prije nikad nije vidio. Dalal i Triggs su koristili upravo ovakav sustav za detekciju ljudi na slikama.

1.2 SIFT

SIFT (Scale-Invariant Feature Trasform) je metoda koji se koristi za pronalaženje i opisivanje značajki objekata na slikama. Razvio ju je David Lowe 1999. godine [2] i patentiran je u SAD-u.

Loweova patentirana metoda je u stanju pouzdano prepoznavati objekte čak i među mnogo drugih objekata te ako je objekt dijelom sakriven. Ovo je moguće jer je njegov SIFT opisnik značajki invarijantan na skaliranje i orijentaciju te je djelomice invarijantan na promjene u osvjetljenju.

Osnovni postupak se može opisati u četiri koraka. Prvo se na ulaznoj slici pronalaze potencijalne interesne točke (kandidati za interesne točke). Pošto ima mnogo takvih točaka, u slijedećem koraku se pokušava ukloniti lošije kandidate koji su nestabilni u pogledu osjetljivosti na šum ili niskog kontrasta u odnosu na okolinu. Jednom kada se odredi skup pravih kandidata, tj. prave interesne točke, onda im se dodjeljuje orijentacija. Orijentacije svake točke se računa na temelju lokalnih značajki okoline promatrane točke. Konačni korak služi dodjeljivanju opisnog vektora svakoj od interesnih točaka.

Ova četiri koraka također formiraju i korake SIFT algoritma, čiji opis slijedi u nastavku.

1.2.1 Detekcija ekstrema u prostoru mjerila

Ovaj korak opisuje pronalazak potencijalnih interesnih točaka, tj. kandidata. Ulazna slika se konvoluira sa Gaussovim filtrima različitih veličina te se primjenjuje na različitim veličinama slika. Zatim se računa razlika tako zamućenih slika. Kandidati za interesne točke su zapravo maksimumi ili minimumi te razlike (Difference of Gaussians, DoG). Razlika takvih slika je dana sa:

$$D(x, y, \sigma) = L(x, y, k_i\sigma) - L(x, y, k_j\,\sigma$$

gdje su $L(x, y, k_i\sigma)$ i $L(x, y, k_j\,\sigma)$ zapravo ulazna slika $I(x, y)$ konvoluirana sa Gaussovim filtrima veličina $k_i\sigma$ i $k_j\,\sigma$:

$$L(x, y, k\sigma) = G(x, y, k\sigma) * I(x, y)$$

Rezultat razlike tih dvaju slika je ponovno slika (DoG slika) iste veličine na kojoj se sada traže kandidati. Svaki piksel DoG slike se uspoređuje sa svojih 8 susjeda te sa još 9 susjeda prve veće (ili manje) slike obrađene istim postupkom. Ako je promatrani piksel najmanji ili najveći u tom svom susjedstvu, onda se proglašava kandidatom.

1.2.2 Precizna lokalizacija interesnih točaka

Izlaz prošle faze je mnogo mogućih kandidata od kojih nisu svi prave interesne točke. Ovaj korak upravo odabire najbolje interesne točke i odvija se u nekoliko podkoraka.

Prvo se provodi interpolacija ekstrema (kandidata), tj. funkcije $D(x, y, \sigma)$ Taylorovim redom drugog stupnja:

$$D(x) = D + \frac{\delta D^T}{\delta x}x + \frac{1}{2}x^t\,\frac{\delta^2 D}{\delta^2 x}x$$

gdje je x odmak od kandidata. Ako je taj odmak veći od 0.5 onda postoji neki drugi kandidat kojemu je promatrani ekstrem bliži, a ako je manji od 0.03 onda se ekstrem radi niskog kontrasta odbacuje.

Funkcija $D(x, y, \sigma)$ ima veliki broj kandidata blizu rubova koji imaju loše određene položaje. Za tako loše definirane ekstreme, gradijent okomit na rub je mnogo veći nego onaj uz rub. Kako bismo uklonili takve kandidate potrebno je izračunati svojstvene vrijednosti matrice H momenata drugog reda funkcije D:

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix}$$

Štoviše, računanje egzaktnih svojstvenih vrijednosti se može i izbjeći računanjem njihovog omjera. Ako te svojstvene vrijednosti označimo sa α i β dobivamo slijedeće jednadžbe (α je veća od njih):

$$D_{xx} + D_{yy} = \alpha + \beta$$

$$Det(H) = D_{xx} D_{yy} - D_{xy}^2 = \alpha \beta$$

$$\frac{D_{xx} + D_{yy}}{Det(H)} = \frac{\alpha + \beta}{\alpha \beta}$$

Ako je determinanta matrice H negativna onda se točka odbacuje, u suprotnom se koristi usporedba omjera $\frac{\alpha+\beta}{\alpha \beta}$ s određenim pragom. Ako je izračunata vrijednost veća od praga, takva točka se također odbacuje.

1.2.3 Dodjela orijentacije

U ovom koraku, svim interesnim točkama koje su prošle selekciju se dodjeljuje orijentacije na temelju značajki iz njihove okoline, tj. smjerova gradijenata tog dijela. Ovdje se postiže invarijantnost na rotaciju.

Prvo se uzima slika $L(x, y, \sigma)$ zamućena Gaussovim filtrom veličine σ koji odgovara promatranoj interesnoj točki te se računaju amplituda i kut gradijenta:

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2}$$

$$\theta(x, y) = \tan^{-1}\left(\frac{L(x, y+1) - L(x, y-1)}{L(x+1, y) - L(x-1, y)}\right)$$

Amplituda i kut gradijenta se računaju za svaki piksel u susjedstvu (npr. 4x4 susjedstvo) interesne točke. Formira se histogram orijentacija sa 36 smjerova, pri čemu svaki smjer pokriva 10 stupnjeva. Svaki uzorak uzet iz susjedstva se pridodaje odgovarajućem smjeru u histogramu sa iznosom jednakim iznosu gradijenta u njemu. Smjerovi sa najvećim vrijednostima odgovaraju dominantnim orijentacijama. Kada se histogram popuni, interesnoj točki se dodjeljuje orijentacija najveće vrijednosti u histogramu i orijentacije koje odgovaraju vrijednostima unutar 80% najveće vrijednosti.

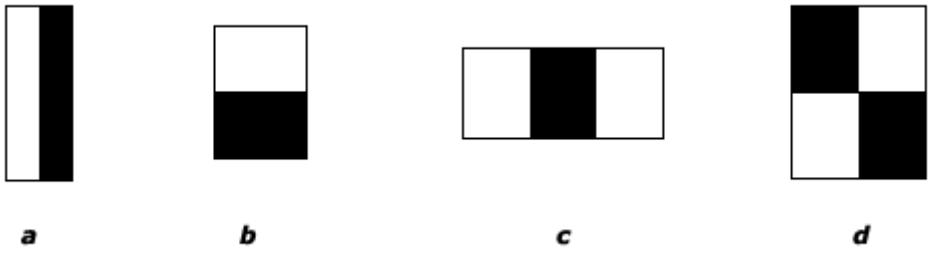
1.2.4 Opisnik interesne točke

Prijašnji koraci su osigurali invarijantnost na položaj, skaliranje i rotaciju. Sada se pokušava dobiti vektor opisnik za svaku interesnu točku koji je invarijantan na ostale varijacije kao što je npr. osvjetljenje.

Prvo se uzima polje veličine 4x4 u okruženju interesne točke te se za njih uzimaju odgovarajući histogrami. Za svako od tih područja, na temelju spomenutih histograma, se formira vektor sa 8 mogućih smjerova, što konačno određuje veličinu takvog opisnika na 128 dimenzija. Ovakav vektor je potrebno normalizirati kako bi se povećala invarijantnost na promjene u osvjetljenju.

1.3 Haarove značajke

Pristup koji su koristili Viola i Jones [4] umjesto korištenja usporedbe vrijednosti piksela na slici sa skupom za učenje koristi jednostavne značajke. Najveća prednost leži u vremenu izvođenja. Korištene Haarove značajke prikazane su na slici 2.



Slika 2 – Osnovne Haarove značajke; Sve ostale značajke se dobivaju skaliranjem i rotacijom (45°)

Značajke na slici 2.a se dobivaju razlikom zbroja intenziteta pisela koji se nalaze u bijelom pravokutniku i sume intenziteta koje se nalaze u crnom pravokutniku. Sve ostale (2.b,2.c,2.d) se dobivaju istim postupkom. Prolazak ovih značajki po slici je vremenski zahtjevno za računanje kao i svaka 2-D konvolucija, no za ubrzavanje algoritma se koristi integralna slika.

Jednadžba za dobivanje integralne slike glasi:

$$IntImg(x,y) = \sum_{x' \leq x;\, y' \leq y} Img(x', y')$$

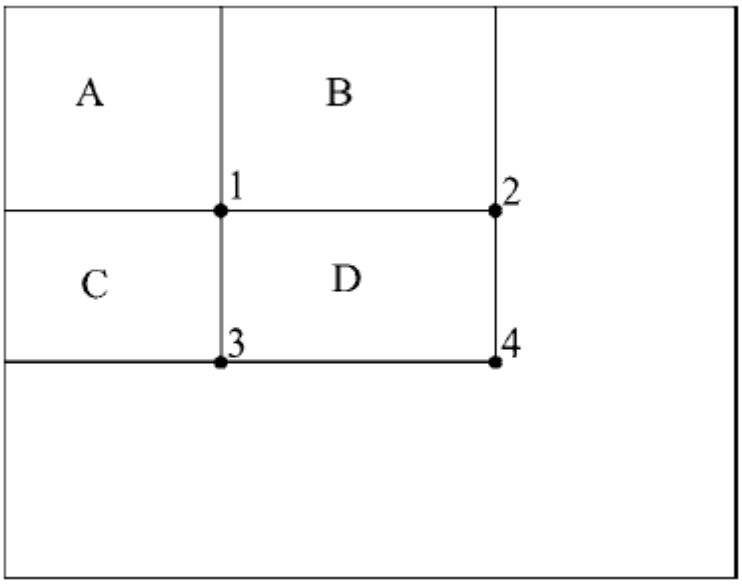
Koristeći rekurzivne relacije:

$$s(x, y) = s(x, y - 1) + Img(x, y)$$

$$IntImg(x, y) = IntImg\ (x - 1, y) + s(x, y)$$

integralna slika se može izračunati u jednom prolazu ($s(x, y)$ je kumulativna suma reda, $s(x, -1) = 0$, $IntImg(-1, y) = 0$). Dakle, vrijednost intenziteta $IntImg(x, y)$ jednaka je sumi svih intenziteta u originalnoj slici koji se nalaze iznad i lijevo od x,y uključujući i intenzitet u vrijednosti x,y.

Za računanje vrijednosti integralne slike u točki D potrebna su nam 4 područja (vidi sliku 3). Vrijednost u području 1 je suma intenziteta u pravokutniku A. Vrijednost u lokaciji 2 je A+B, lokacija 3 A+C i lokacije 4 je A+B+C+D. Ovaj princip računanja se primjenjuje u računanju značajki. Budući da se područja potrebna za računanje Haarovih značajki preklapaju potreban je manji broj pristupanja integralnoj slici od n*broj pravokutnika. Za prvu i drugu Haarovu značajku (Slika 2 – a,b) potrebno je pristupiti integralnoj slici 6 puta, za treću 8 puta, za četvrtu 9 puta. Npr. za značajku na slici 1 – A potrebno je koristiti 4 pravokutnika za računanje bijele površine i 4 pravokutnika za računanje crne, ali se 2 pravokutnika preklapaju pa je broj različitih korištenih pravokutnika zapravo 4.



Slika 3 - Prikaz računanja integralne slike

1.3.1 Pojačani (boostani) Haarovi klasifikatori

Značajnije performanse u polju detekcije objekata moguće je postići uvođenjem tzv. *kaskade slabih (boostanih) klasifikatora*. Osnovna ideja temelji se na konceptu

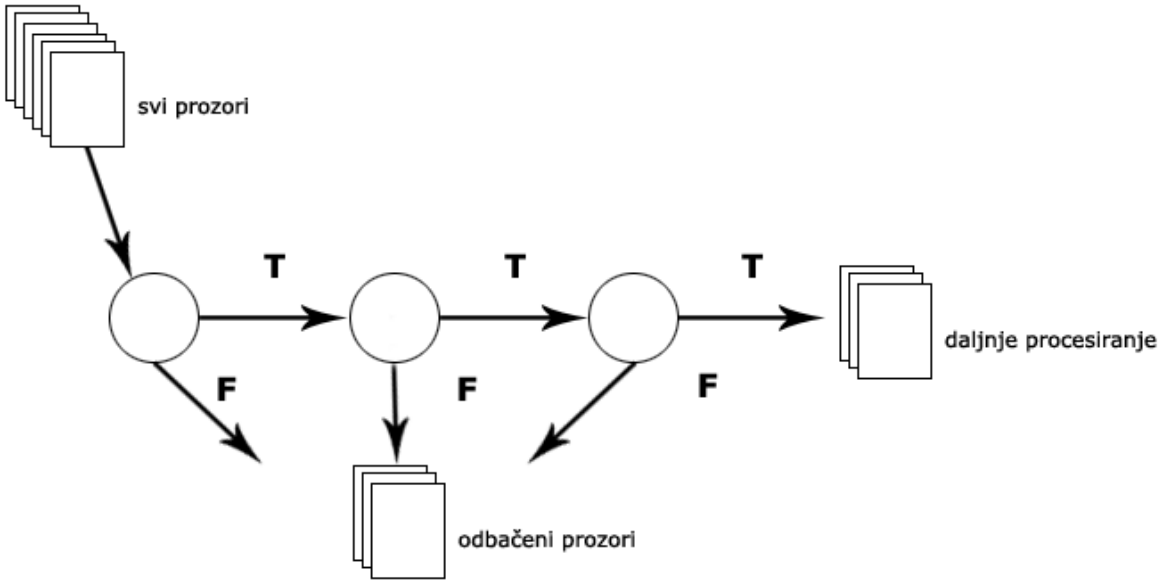
eliminacije negativnih prozora i detekciji svih pozitivnih instanci u kaskadnom spoju *boostanih* klasifikatora (slika 4). Pritom se za eliminaciju glavnine prozora koriste jednostavni klasifikatori, dok se kompleksniji pozivaju u svrhu smanjenja *false-positive* odnosa.

Ukoliko je *boostani* klasifikator u *i*-tom segmentu kaskade zadovoljio prag, postoji mogućnost da se traženi objekt nalazi u prozoru, te se algoritam nastavlja u *i*+1-om segmentu. U protivnom, zaključuje se da traženi objekt nije u prozoru, te se odvija pomak na novu poziciju.

Algoritam se može opisati sa svega nekoliko operacija:

1. evaluiraj Haarove značajke (zahtjeva između 6 i 9 referenci integralne slike po značajki)
2. evaluiraj slabi klasifikator za svaku Haarovu značajku (zahtjeva jednu usporedbu s pragom po značajki)
3. kombiniraj slabe klasifikatore (zahtjeva jedno množenje po značajki, zbroj, te usporedbu s pragom u konačnici)

Forma ovakvoj procesa jednaka je degenerativnom decizijskom stablu, zbog čega se služimo pojmom «kaskada».



Slika 4: Algoritam

Optimizacija performansi može se postići ako prolaz kroz kaskadu uključuje manji broj lokacija slike, zbog čeka se snažni klasifikatori slažu po postotku uspješnosti eliminacije prozora u kojima nema traženog objekta. Prvim klasifikatorom postaje onaj koji je najuspješniji u tom zadatku.

Dublja kaskada implicira veći broj klasifikatora i manje pogrešnih detekcija. Možemo pisati:

$$F = \prod_{i=1}^{K} f_i$$

gdje *F* predstavlja procjenu pogrešnih detekcija, *K* broj klasifikatora, a f_i procjenu pogrešnih detekcija za pojedini klasifikator.

Analogno, s porastom dubine kaskade raste i broj pogrešnih detekcija:

$$D = \prod_{i=1}^{K} d_i$$

D označava procjenu ispravno detektiranih objekata cijele kaskade, a d_i procjenu detektiranih objekata za pojedini klasifikator.

1.3.2 Normalizacija prozora za detekciju

U postupku učenja provodi se normalizacija varijancom, te je iz tog razloga nužno istu normalizaciju provesti i u postupku detekcije. Za izračun varijance, pogodne su integralne slike. Koristi se obična integralna slika i integralna slika kvadriranih slikovnih elemenata.

Izraz za varijancu jest:

$$\sigma^2 = m^2 - \frac{1}{N}\prod_{i,j} p_{ij}^2$$

gdje *m* predstavlja srednju vrijednost, p_{ij} vrijednost slikovnog elementa unutar prozora, a *N* ukupan broj elemenata prozora.

Uzevši u obzir širinu *w* i visinu *h* prozora, srednju vrijednost računamo preko integralnih slika na sljedeći način:

$$m = \frac{S_i(x,y) - S_i(x+w,y) - S_i(x,y+h) + S_i(x,y)}{(x+w)(y+h)}$$

dok sumu kvadriranih elemenata prozora aproksimira sljedeća relacija:

$$\sum_{i,j} p_{ij}^2 = S_q(x,y) - S_q(x+w,y) - S_q(x,y+h) + S_q(x,y)$$

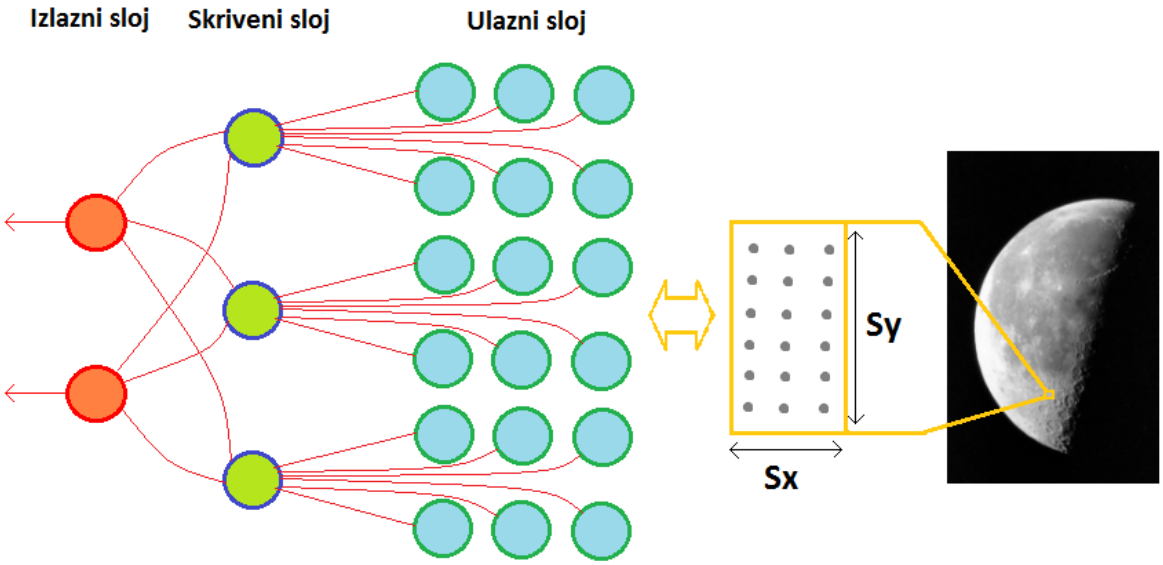
$$S_q(x,y) = \sum_{x' \leq x, y' \leq y} S(x',y')^2$$

U postupku detekcije se normalizacija izvodi dijeljenjem vrijednosti značajke sa devijacijom, uzevši u obzir gorenavedene izraze.

1.4 Lokalna receptivna polja

Pojam lokalnog receptivnog polja (Local Receptive Field, LRF) ima svoje korijene u receptivnim poljima neurona vidnog sustava živih bića. Ti neuroni pale prilikom svake promijene u području vidnog polja kojeg pokrivaju i tako se stvara osjet vida. Ako cijelo vidno polje zamislimo kao ulaznu sliku, dakle 2-D matricu, pojedini neuroni pokrivaju samo malene dijelove te matrice. Kada mozak primi impulse od velikog broja takvih neurona možemo vidjeti sliku određenog objekta. Ovu ideja se krije i iza algoritama računalnog vida za traženje i prepoznavanje objekata temeljenim na lokalnim receptivnim poljima. Pošto se vidi da su neuronske mreže vrlo bliske ovoj problematici, u ovom radu se opisuje upravo jedna takva struktura.

Osnovnu neuronsku mrežu za potrebe ovog rada možemo opisati sa tri sloja. Ulazni sloj odgovara elementima ulazne slike, a veličina tog sloja je zadana kao $S_x * S_y$ pri čemu su S_x i S_y broj horizontalnih i broj vertikalnih elemenata slike. Vrijednost koja se pridodaje ulazu u svaki ulazni neuron je zapravo iznos sive razine odgovarajućeg piksela (slika se prvotno pretvara u sivu sliku). Skriveni sloj (drugi sloj) sadrži manji broj neurona koji su povezani sa ulaznim slojem. Okidanje tih neurona se računa na temelju zbroja svih ulaza od kojih svaki ima svoju težinu. Ti neuroni nisu spojeni sa svim neuronima ulaznog sloja već samo sa određenim dijelom tih neurona koji predstavljaju područje na slici (npr. jedan neuron skrivenog sloja je povezan sa 3x3 susjedna neurona ulaznog sloja). Upravo takva područja na slici nazivamo lokalnim receptivnim poljima. Izlazni, tj. treći sloj se također sastoji od više neurona, ali ovaj put je svaki od njih povezan sa svim neuronima skrivenog sloja. Ovaj zadnji sloj zapravo donosi odluku da li je traženi objekt prisutan na slici ili daje opis cijele scene sažet u više parametara čiji broj odgovara broju izlaznih neurona.



Slika 5 – Primjer opisane neuronske mreže

Ovu neuronsku mrežu možemo naučiti da raspoznaje različite objekte, predstavljanjem većeg broja primjera koji odgovaraju ciljnom objektu (potrebni su i lažni primjeri) i traženju da za takve slike dobivamo određene izlazne vrijednosti u izlaznom sloju. Ovaj postupak bi se moga provoditi korekcijom težina gradijentnim spustom.

Jednom kada takvu mrežu naučimo da raspoznaje jednu ili više klasa objekata, takve objekte možemo tražiti na neviđenim slikama, odnosno možemo ostvariti prepoznavanje i detekciju takvih objekata na slikama. Postupak pretraživanja bi se mogao odvijati na način da sliku prolazimo sa prozorom određene veličine. Za svaki položaj tog prozora, slikovne elemente unutar njega predajemo naučenoj neuronskoj mreži, koja nam odgovara da li dio slike u tom prozoru odgovara traženom objektu, tj. da li se objekt nalazi u određenom prozoru. Već i ovakvim jednostavnim postupkom možemo ostvariti invarijantnost na skaliranje jer prozor možemo slobodno skalirati po volji a neuronskoj mreži predajemo konstantnu veličinu ulazne slike koja se računa na temelju prozora (stapanje piksela).

Neuronsku mrežu možemo naučiti za više objekata pri čemu onda očekujemo i drugačije izlaze za svaki objekt ili možemo za svaki objekt imati jednu neuronsku mrežu (što se čini nepraktičnim).

Primjene opisnika značajki

Ovdje opisani opisnici značajki se mogu koristiti nebrojeno mnogo primjena kako na područjima računalnog vida i obrade slike tako posredno ili neposredno i u drugim područjima. Najočitije primjene se odnose na prepoznavanje, pronalaženje, pretraživanje ili praćenje objekata ili čak cijelih scena na slikama.

Neke primjene:

- Raspoznavanje
- Određivanje korespondencija
- Dohvaćanje slika na temelju sadržaja

2.1 Raspoznavanje

Ovaj pojam pokriva raspoznavanje objekata ili čitavih scena na slikama ili slijedu slika. Za čovjeka je ovo trivijalan zadatak, već se sa malo truda mogu prepoznati objekti u svim mogućim rotacijama, translacijama, skaliranjima, promjenama u osvjetljenju te čak i kad su objekti djelomično sakriveni. Računalima je to još uvijek izazovan zadatak.

Sve spomenute metode mogu izvršavati ovakve zadatke, jer sve opisuju objekte na svoj način te je s njima moguće pronaći tražene objekte na slikama.

2.2 Određivanje korespondencija

Problem određivanja korespondencija na dvjema slikama ili na slijedu slika se svodi na traženje istih (sličnih) interesnih točaka na slikama. Ovo je nativno područje primjene SIFT algoritma koji koristi upravo strukture interesnih točaka kako bi opisao objekte na slikama.

Ako se tražene korespondencije predstave kao cijeli objekti, ponovno se sve opisane metode mogu primijeniti.

2.3 Dohvaćanje slika na temelju sadržaja

Konačni cilj dohvaćanja slika na temelju sadržaja (Content Based Image Retrival, CBIR) je tražilica u koju bi se upisalo ime nekog objekta a ona bi nam vratila sve slike na kojima se pojavljuje takav objekt.

Pod traženjem se ovdje podrazumijeva stvarno pretraživanje sadržaja slike na temelju elemenata same slike, a ne na tekstu opisa ili dodijeljenih oznaka slike. Kao primjer se može uzeti tražilica operacijskog sustava. Njoj bi se moglo zadati da traži slike na kojima se nalazi šalica. Takva tražilica bi morala biti upogonjena s algoritmima računalnog vida koji bi na ulaz primali sve slike koje tražilica pronađe na računalu a na izlazu bi pružali informaciju da li ta slika sadrži šalicu ili ne.

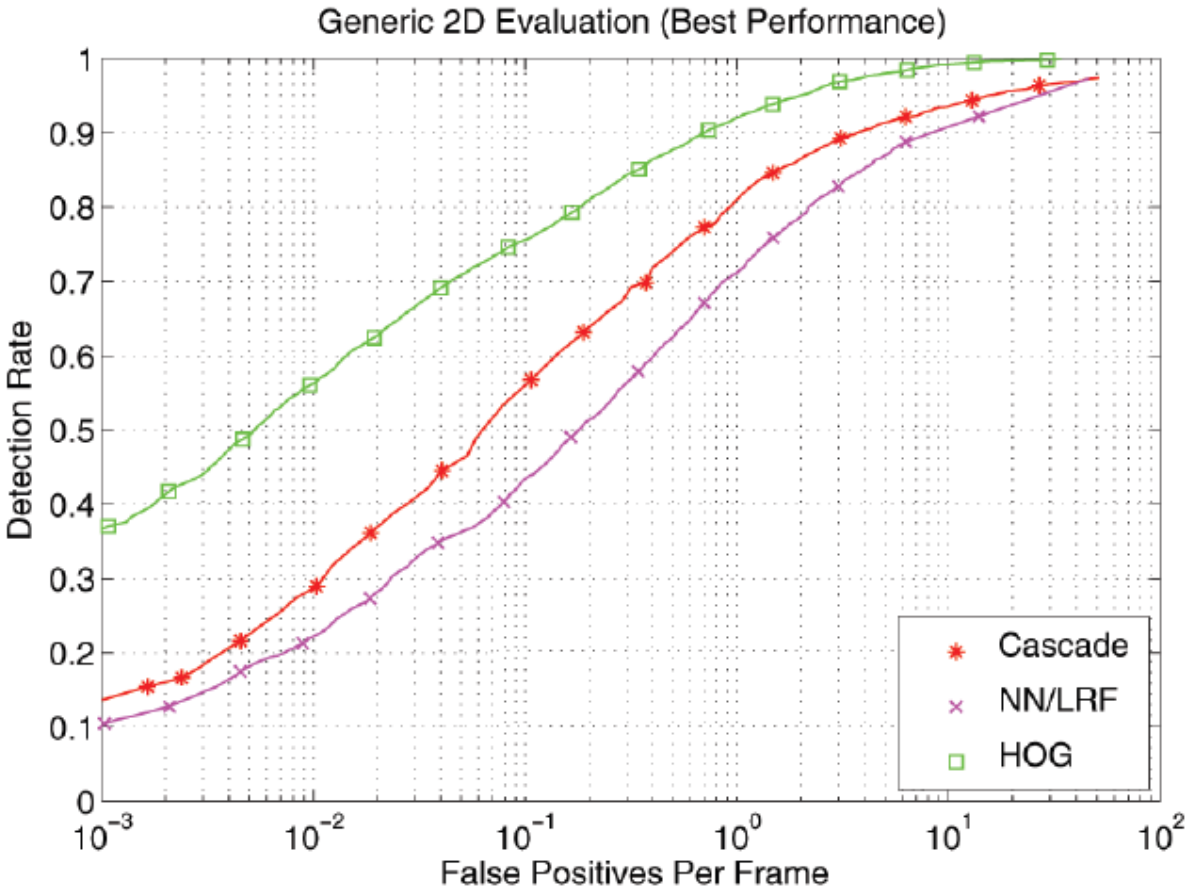
Ovakvim razmišljanjem se može otići i malo dalje. Opisana tražilica bi mogla pružati dodatne informacije u obliku jednostavnijih (npr. broj pronađenih šalica na slici, veličine šalica, posebne oblike) ili složenih opisnih struktura koje bi obuhvaćale raspoznavanje sadržaja cijele slike (npr. šalica se nalazi na stolu i to u kuhinji) što se još uvijek čini kao daleka budućnost.

Usporedba opisnika značajki

U ovom dijelu se navode rezultati koje su pružili različiti opisnici pošto su očite razlike već objašnjene u prošlim poglavljima.

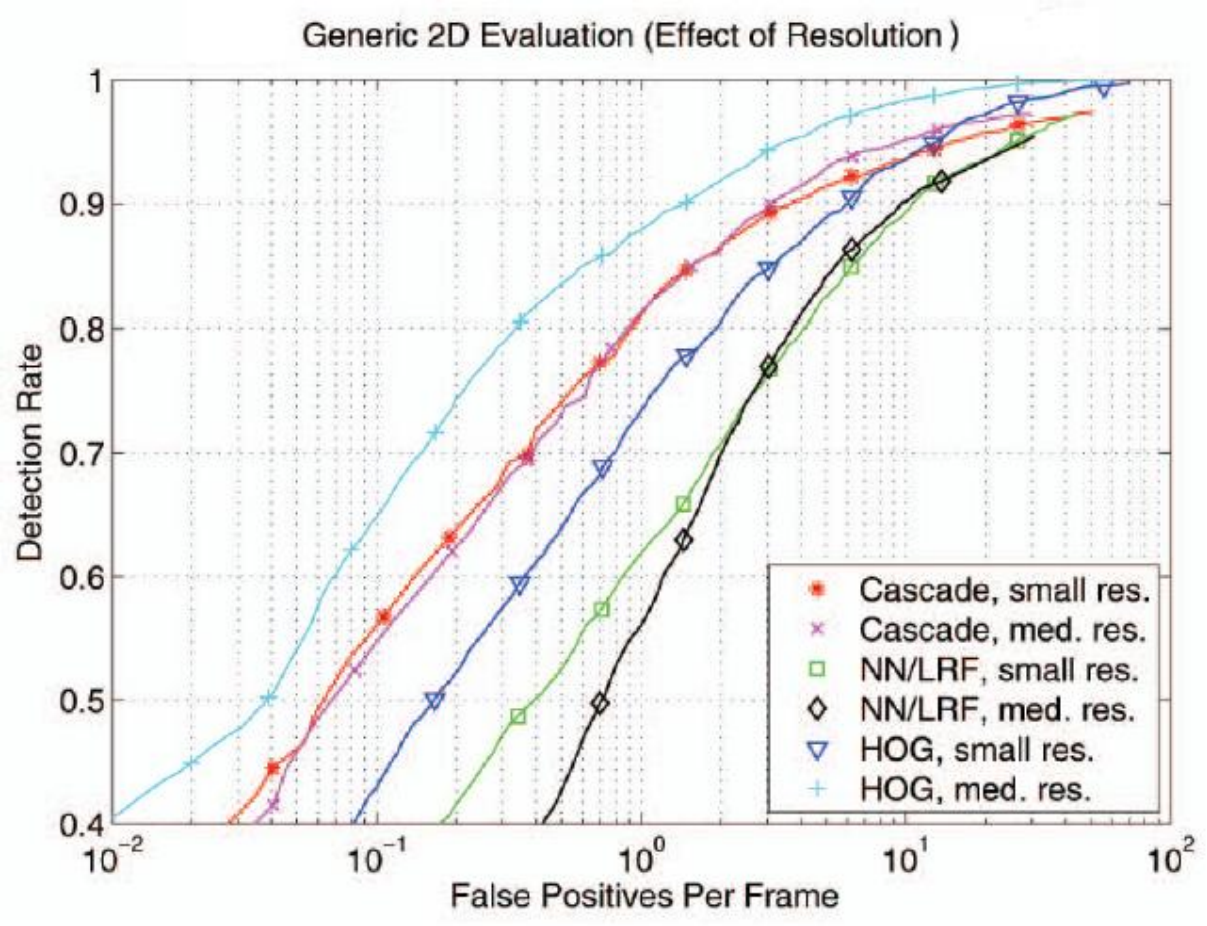
Rezultati testiranja za Haarove značajke, lokalna receptivna polja te HOG su navedeni u [5]. Koriste se kaskada Harrovih značajki (nadalje kaskada), neuronska mreža upogonjena lokalnim receptivnim poljima (NN/LRF) te SVM upogonjen HOG opisnicima (HOG/SVM). Za treniranje klasifikatora je korišteno 15660 pozitivnih i 6744 negativnih primjera, dok se testira na 21790 slika veličine 640x480 piksela i sa ukupno 56492 labela ljudi na njima. Za pronalaženje pješaka na slikama su korišteni klizeći prozori.

Na sljedećoj slici se može primijetiti kako je pristup temeljen na HOG opisnicima bolji od NN/LRF i kaskade ako se gleda omjer uspješno detektiranih pješaka i lažno pozitivno detektiranih pješaka, dok je kaskada nešto bolja od NN/LRF pristupa:



Slika 6 – Konačni rezultati usporedbe triju opisnika

Prilikom promjene rezolucije pješaka, kaskada Haarovih značajki i NN/LRF pokazuju slične rezultate pri smanjenim rezolucijama pješaka, dok je HOG/SVM lošiji na manjim rezolucijama:



Slika 7 – Usporedni rezultati za promijenjenu rezoluciju pješaka (small/medium res.)

Usporedba eksperimentalnih rezultata korištenja triju opisnika je prikazana u tablici 1. Svi opisnici pokazuju bliske rezultate u pogledu odziva sustava. Uz ovako slične odzive valja zamijetit preciznost koja je najbolja u slučaju korištenja HOG/SVM metode. Treći redak u tablici označava broj lažno pozitivno detektiranih pješaka na 1000 slika i po minuti računanja.

	Kaskada	**NN/LRF**	**HOG/SVM**
Odziv	65.4%	65.3%	64.1%
Preciznost	56.1%	33.5%	90.2%
FP / 10^3, min	156	307	16
Prosječno vrijeme računanja za 10^3 detekcijskih prozora	20ms	660ms	430ms

Tablica 1 – Brojčani rezultati triju opisnika

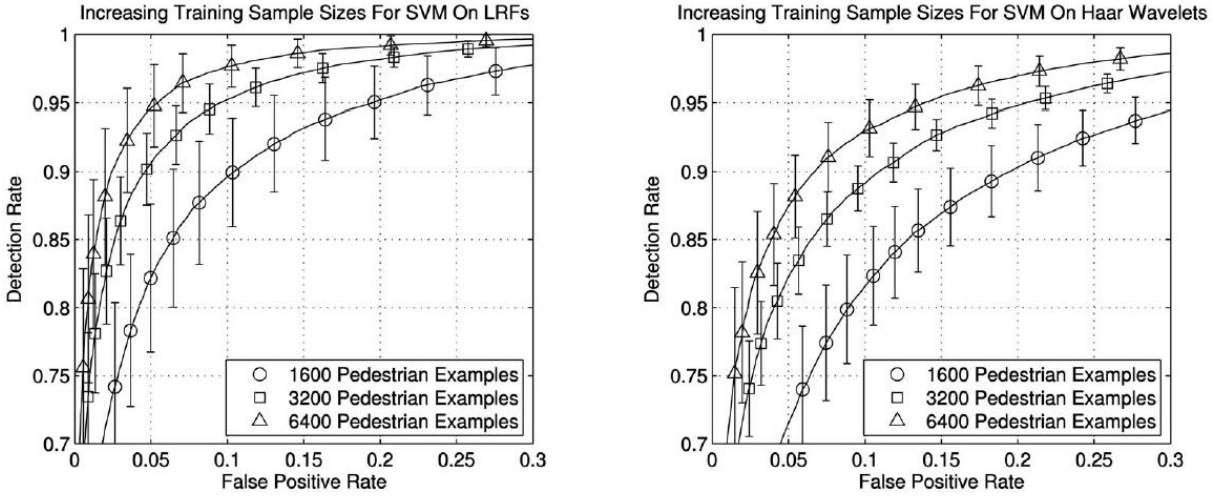
Ako se vrijeme računanja ograniči, postižu se i drugačiji rezultati. Kaskada Haarovih značajki je općenito mnogo brža od ostala dva pristupa što se i vidi u

rezultatima za vremensko ograničenje od 250ms računanja po slici (tablica 2). Pri ograničenju od 2.5s po slici se ne vide neke veće razlike od prošlog postupka, osim povećanja preciznosti kaskade. Kada se ograničenje smanji na 250ms, kaskada pokazuje svoje prave prednosti. U ovom slučaju su se preciznosti HOG/SVM i NN/LRF pristupa osjetno pogoršale dok je preciznost kaskade ostala očuvana.

	Kaskada	**NN/LRF**	**HOG/SVM**
Odziv (2.5s)	64.9%	65.5%	64.3%
Preciznost (2.5s)	77.2%	53.4%	88.7%
FP / 10^3, min (2.5s)	32	102	11.7
Odziv (250ms)	64.9%	67.0%	67.4%
Preciznost (250ms)	77.2%	43.4%	47.6%
FP / 10^3, min (250ms)	32	171	143
Prosječno vrijeme računanja za 10^3 detekcijskih prozora	20ms	440ms	430ms

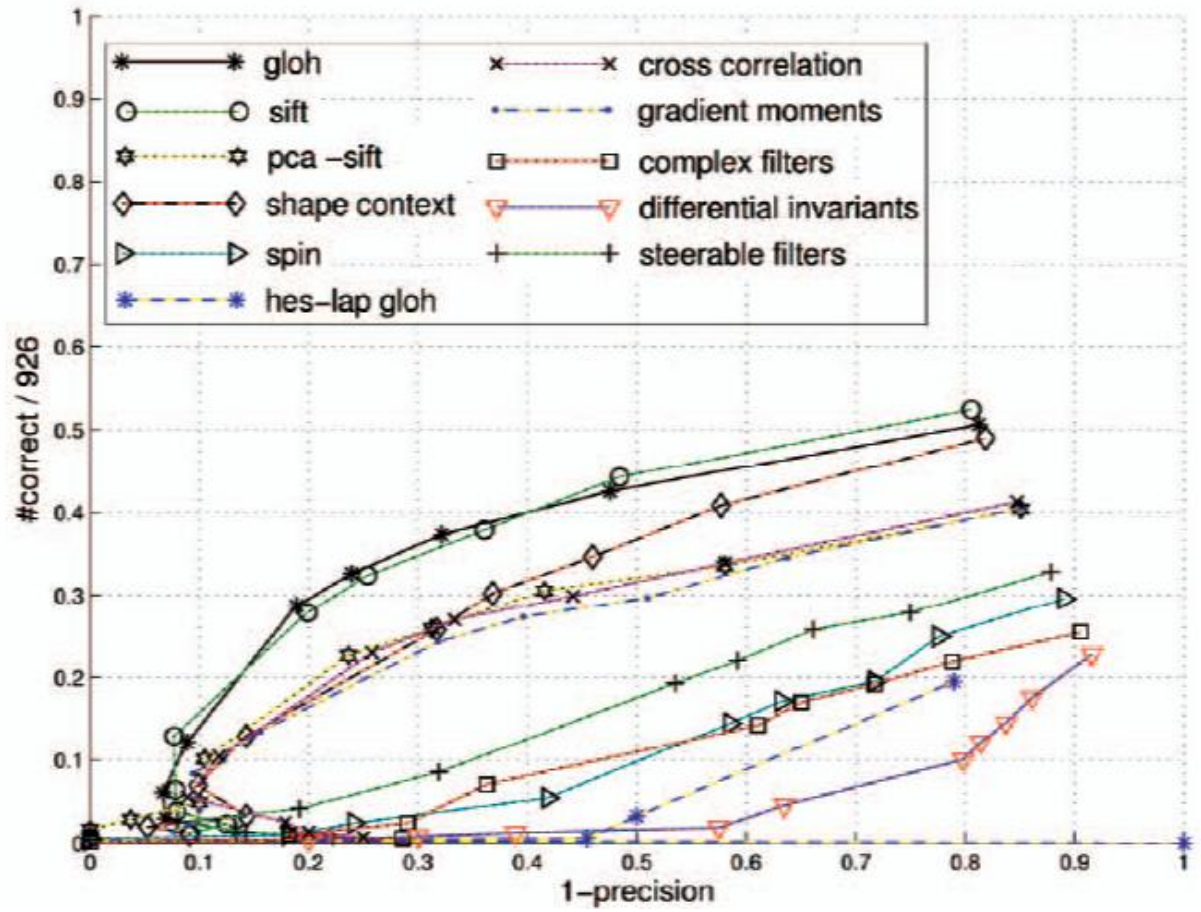
Tablica 2 – Rezultati opsinika uz ograničeno vrijeme računanja

Na rezultate primjene pojedinih opisnika utječe i veličina skupa slika za učenje na kojima su ti opisnici trenirani [6]. Ako se taj skup (pozitivnih i negativnih primjera) poveća, poboljšavaju se i rezultati postupaka (bolja detekcija te manje lažno pozitivnih rezultata). Na slici 8 se vidi usporedba za dva opisnika za različite veličine skupova za učenje. To su SVM upogonjen sa LRF opisnicima te SVM zajedno sa Haarovim značajkama. U ovom primjeru se pokazalo da udvostručavanje broja pozitivnih i negativnih primjera vodi do gotovo dva puta manje pogrešaka klasifikacije.



Slika 8 - Prikaz ovisnosti rezultat opisnika o veličini skupa za učenje tih opisnika

Nešto bolja usporedba SIFT opisnika sa više ostalih opisnika značajki je dana u [7]. U ovom radu se okvirno pokazalo da je opisnik SIFT bolji od ostalih opisnika na slikama tekstura i strukturiranim slikama. Na slici 9 se vide upravo rezultati iz toga rada. Prikazan je omjer točno klasificiranih pješaka za različite iznose preciznosti na skupu od 926 ispitnih slika. SIFT opisnik je u tom radu pokazao bolje rezultate za faktere skaliranja 2-2.5 i faktor rotacije 30-45 stupnjeva. Lošijim se pokazao na zamućenim slikama gdje su opisnici temeljeni na gradijentnim operatorima ipak bolji.



Slika 9 – Točnost klasifikacije za različite preciznosti

Zaključak

Različiti opisnici značajki opisani u ovom radu se mogu ravnopravno nositi sa većinom problema raspoznavanja, pronalaženja, pretraživanja ili praćenja objekata na području računalnog vida. Razlike između njih postoje kako u samim idejama iza metoda tako i u rezultatima koje takve metode pružaju, stoga pojedini opisnici mogu biti bolji za neke primjene od drugih.

Osnova iza SIFT opisnika su interesne točke čijom se usporedbom mogu pronaći korespondencije, tj. slične točke na raznim slikama koje jednoznačno mogu odrediti položaj traženog objekta, teksture ili čak čitavih prizora na neviđenim slikama. Takve korespondencije se mogu predstaviti kao objekti čime dobivamo funkcionalnost prepoznavanja i traženja objekata na slikama. SIFT opisnici su se pokazali za malu mjeru uspješnijim od ostalih opisnika, pogotovo ako se radi o specifičnim transformacijama koje su opisane u prošlom poglavlju.

HOG opisnici se temelje na amplitudi i kutu gradijenta u svakom slikovnom elementu, pri čemu se mogu raspoznavati objekti na temelju njihovog oblika, tj. položaja rubova i njihove orijentacije. Ovaj opisnik se pokazao nešto boljim od LRF opisnika i kaskade Haarovih značajki, pri čemu je pokazao lošije rezultate na manjim rezolucijama traženog objekta i detekcijske rešetke.

Metoda prepoznavanja objekata kaskadom Haarovih značajki se pokazala najbržom metodom ali je isto tako pružila i nešto lošije rezultate od HOG i LRF opisnika ako vrijeme računanja nije ograničeno, u protivnom je pokazala bolje rezultate te je pogodna za razmatranje u primjenama gdje je potrebna brza obrada.

LRF opisnik se temelji na dijeljenju slike na manja područja koja se predaju neuronskoj mreži. Neuronska mreža se zatim ući na mnogo primjera određenog objekta, a na kraju joj se može predočiti neviđeni objekt i ona je u stanju klasificirati taj objekt kao traženi ili ga odbaciti. LRF opisnik se pokazao sličnim HOG opisniku, ali je isto tako pokazao otpornost na smanjenje rezolucije traženog objekta i detekcijske rešetke.

Kod ovih opisnika su se pokazali bolji rezultati povećavanjem skupa primjera za učenje tako da ostaje prostora za poboljšavanje ukupnih rezultata u konačnim primjenama, ali za pouzdano praćenje objekata u stvarnom vremenu potrebno je dodatno unaprijediti ove metode.

Literatura

[1] Wöhler, C., Anlauf, J.K.: *An adaptable time-delay neural-network algorithm for image sequence analysis*, IEEE Computational Intelligence Society, 1999., pp 1531 - 1536

[2] David G. Lowe: *Object Recognition from Local Scale-Invariant Features*, *Proceedings of the International Conference on Computer Vision*. 2., 1999., pp. 1150–1157

[3] Navneet Dalal and Bill Triggs: *Histograms of Oriented Gradients for Human Detection*, IEEE Computer Society, 2005.,pp 886-893

[4] Viola and Jones: *Rapid object detection using boosted cascade of simple features*, Computer Vision and Pattern Recognition, 2001

[5] Markus Enzweiler, Student Member, IEEE, and Dariu M. Gavrila: *Monocular Pedestrian Detection: Survey and Experiments*, IEEE Pattern Analisys And Machine Inteligence, vol. 31, no. 12, 2009.

[6] S. Munder and D.M. Gavrila: *An Experimental Study on Pedestrian Classification*, IEEE Pattern Analisys And Machine Inteligence, vol. 28, no. 11, 2006.

[7] Krystian Mikolajczyk and Cordelia Schmid: *A Performance Evaluation of Local Descriptors*, IEEE Pattern Analisys And Machine Inteligence, vol. 27, no. 10, 2005.

Sažetak

U ovom radu je napravljen općeniti opis različitih opisnika značajki korištenih na području računalnog vida. Među obrađene opisnike spadaju: histogram orijentiranih gradijenata (HOG), transformacija značajki invarijantna na skaliranje (Scale-Invariant Feature Transform, SIFT), Haarove značajke i lokalna receptivna polja (LRF).

Za svaki opisnik je prvo opisana temeljna ideja, zatim slijedi opis implementacije takve ideje, odnosno koraci algoritma stvaranja opisnika. Nakon spomenutih opisa slijedi opis nekoliko različitih područja primjene. To su područja raspoznavanja objekata, određivanja korespondencija, dohvaćanja slika na temelju njihova sadržaja te ostale primjene. Na kraju slijedi usporedba uspješnosti obrađenih opisnika na različitim problemima gdje se prikazuju njihove prednosti i mane kao i konačni eksperimentalni rezultati primjene.