

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

SEMINAR

**Sazrijevanje računalnog vida: Automatsko
pronalaženje korespondencija**

Bojan Lukovac

Voditelj: Doc.dr.sc. Siniša Šegvić

Zagreb, travanj 2008.

Sadržaj

Sazrijevanje računalnog vida: Automatsko pronalaženje korespondencija	1
Sadržaj	2
Uvod u računalni vid.....	3
Uvod u SIFT.....	3
Algoritam i izvedba	4
Detekcija ekstrema u prostoru mjerila (Scale-space).....	4
Precizna lokalizacija interesnih točaka	4
Dodjela orijentacije	5
Lokalni opisnik interesne točke	6
Funkcionalnost i primjeri.....	9
Prepoznavanje objekata koristeći SIFT.....	9
Robotika	9
3D rekonstrukcija scene	9
Spajanje fotografija u panorame.....	9
Ostale primjene	10
Primjeri	10
Zaključak.....	18
Literatura	19

Uvod u računalni vid

Računalni vid je znanstvena i tehnološka disciplina koja se bavi teorijom izrade te samom izradom sustava koji služe pribavljanju informacija iz slike, bilo da je to iz jedne ili više fotografija, video sekvence ili medicinskih uređaja (primjerice CT skeneri). Neke od poddisciplina računalnog vida su rekonstrukcija scene, pronalaženje, detekcija i prepoznavanje objekata, praćenje, detekcija događaja, restauracija slike i slično.

Računalni vid je srodan mnogim drugim znanstvenim disciplinama – od optike i fotografije, preko automatike i robotike, obrade slika i raspoznavanja uzoraka, sve do umjetne inteligencije, s kojom je blisko vezan. Ujedno je i disciplina sa brojnim primjenama. Počnimo od medicinskih svrha, gdje računalni vid nalazimo u gotovo svim suvremenijim dijagnostičkim aparatima – mikroskopi, x-zrake, tomografija i ultrazvuk samo su neki od primjera. Još jedna važna primjena leži, naravno, i u industriji odnosno proizvodnji – za pozicioniranje robotskih ruku ili upravljanje kakvoćom, na primjer. Promet (samoupravljajuća vozila, prepoznavanje znakova), svemirska istraživanja (autonomna vozila, teleskopi), sigurnost (nadgledanje) te vojne svrhe (navođenje raketa, „battlefield awareness“) su sve dobri primjeri implementacije računalnog vida.

Uvod u SIFT

SIFT (Scale-Invariant Feature Transform) je jedan od algoritama koji se upotrebljava za neke od zadataka računalnog vida. SIFT, naime, služi za pronalazak i opisivanje lokalnih značajki neke slike. Ono što je pritom posebno važno jest njegova invarijantnost s obzirom na rotaciju i uvećanje odnosno smanjenje slike. Također je i relativno otporan na promjene u osvjetljenju na slici i promjene točke gledišta te zaklonjenost objekata.

Ovaj algoritam ima velike mogućnosti uporabe kod uparivanja odnosno prepoznavanja te kod 3-dimenzionalnog rekonstruiranja scene. Omogućava upoređivanje sa relativno opsežnim bazama podataka u približno realnom vremenu (ovisno o računalnom sustavu).

SIFT je razvijen i objavljen od strane Davida Lowea 1999. godine te je potom poboljšan i ponovno izdan 2004. Patentiran je u Američkom uredu za patente 2000. godine.

Algoritam i izvedba

Četiri su osnovna koraka na osnovu kojih SIFT obavlja svoj rad.

Detekcija ekstrema u prostoru mjerila (Scale-space)

U ovom koraku se detektiraju interesne točke. Algoritam prolazi kroz sve lokacije u svim veličinama slike da bi se detektirale potencijalne interesne točke. Funkcija koja se koristi jest razlika Gaussovih funkcija (1, 4) za svaku od veličina slike.

$$G(x, y, c) = \frac{1}{2\pi c^2} e^{-\frac{(x+y)^2}{2c^2}} \quad (1)$$

T. Lindeberg je 1994. godine dokazao da je Gaussova funkcija (1), jedina primjerena kao ishodište za prostor mjerila, kojim se Lowe koristi. Razlika Gaussiana (4) ima vrlo blisku vrijednost Laplasijanu Gaussiana koji je Lindeberg proučavao i koristio u svojim algoritmima.

Ulazna slika je dana slijedećom funkcijom:

$$I(x, y) \quad (2)$$

gdje su x i y koordinate točaka. Sliku zamućenu Gaussovom funkcijom prikazujemo sa:

$$L(x, y, c), \quad L(x, y, c) = I(x, y) * G(x, y, c) \quad (3)$$

gdje je c omjer odnosno uvećanje u odnosu na originalnu sliku, a „*” operator konvolucije (integral umnoška funkcija, gdje se druga od funkcija zrcali i posmakne). Konačno, razlika Gaussiana (DoG) je prikazana na slijedeći način:

$$D(x, y, c) = (G(x, y, c) - G(x, y, c * k)) * I(x, y) = \\ L(x, y, kc) - L(x, y, c) \quad (4)$$

Funkcija D (4) je zapravo nova slika proizašla iz gore navedenog procesa. Nakon što se dobiju DoG slike, svaki pixel iz slike jedne veličine se uspoređuje sa susjednih osam pixela na toj slici te odgovarajućih 9 pixela na slikama jednog stupnja veće ili manje veličine (ovaj izbor pixela nazivamo piramidom). Ako taj pixel iznosi minimum ili maksimum među svim uspoređenim pixelima, on postaje kandidat za interesnu točku odnosno keypoint.

Precizna lokalizacija interesnih točaka

Prvi stupanj algoritma je rezultirao kandidatnim točkama, ali problem je što ih daje previše, odnosno daje ih mnogo koje su nestabilne, podložne šumu na slici, niskog su kontrasta u odnosu na okolinu i slično. U ovom stupnju algoritma se, stoga, mjeri stabilnost kandidatnih lokacija i vrši izbor interesnih točaka.

Drugi stupanj SIFT-a se odvija u nekoliko točaka.

Prvo se interpolira lokacija maksimuma korištenjem razvoja Taylorovog reda funkcije D do drugog stupnja,

$$D(x) = D + \frac{\partial D^T}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 D}{\partial x^2} x \quad (5)$$

gdje je x odmak od kandidatne točke. Ako taj odmak iznosi više od 0.5, sigurno je da se taj ekstrem nalazi bliže nekoj drugoj kandidatnoj točki. Zbog niskog kontrasta, odbacuju se točke gdje funkcija ima vrijednost manju od 0.03.

Slijedeći problem jest taj što uz rubove objekata na slici dosadašnji dio algoritma daje mnogo kandidatnih točaka. Velik dio tih točaka je loše pozicioniran i unatoč otpornosti na šum, ostaje među odabranim točkama. Rješenje ovog dijela problema se zasniva na pretpostavci da je gradijent krivulje koja prolazi okomito na rub objekta na slici znatno manja nego gradijent krivulje uzduž ruba. Stoga valja riješiti slijedeću matricu drugog reda:

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{yx} & D_{yy} \end{bmatrix} \quad (6)$$

Možemo izbjeći traženje svojstvenih vrijednosti matrice time što nas jedino zanima omjer tih dviju vrijednosti. Veću od tih vrijednosti nazovimo α , a manju β .

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

$$\text{Det}(H) = \alpha\beta \quad (7)$$

U slučaju da se za determinantu (7) ispostavi da je negativna, ta se točka odbacuje. Preostaje usporediti kvadrat sume α i β podijeljen ovom determinantom sa određenim omjerom α i β , što rezultira omjerom svojstvenih vrijednosti za usporedbu s pragom.

Dodjela orijentacije

Ovo je ključni korak za postizanje invarijantnosti s obzirom ponajprije na rotaciju, ali i na uvećanje. Svakoj interesnoj točki se dodjeljuje orijentacija s obzirom na lokalne značajke tog dijela slike, odnosno smjera gradijenta (gradijenata) u tom dijelu. Stoga se opisnik interesne točke može prikazati u odnosu na tu prvotnu orijentaciju. Odabire se Gaussovom funkcijom zamućena slika L jednakog uvećanja interesne točke kako bi se postigla invarijantnost s obzirom na uvećanje. Računamo veličinu gradijenta, m i orijentaciju, θ slijedećim jednadžbama:

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2} \quad (8)$$

$$\theta(x, y) = \arctan \frac{L(x, y+1) - L(x, y-1)}{L(x+1, y) - L(x-1, y)} \quad (9)$$

Izračuni ove dvije jednadžbe se vrše za svaki pixel u okolici interesne točke. Formira se orijentacijski histogram sa 36 smjerova po 10° . Nalazi se točka najveće vrijednosti na histogramu, te se od nje i ostalih točaka do 80% vrijednosti najveće točke stvara interesna točka te orijentacije. Moguće je da se jednoj točki pridjeljuje više orijentacija (u 15% slučajeva), što se vrši tako da se stvori još jedna točka alternativnog smjera na istoj lokaciji.

U svojoj publikaciji (*Distinctive Image Features from Scale-Invariant Keypoints*), Lowe tvrdi da ovaj dio algoritma održava funkcionalnost i kod dodavanja većih količina šuma. Test je izvršen sa nasumičnim izborom rotacije i uvećanja slike, gdje je dodano 10% šuma. Varijacije u dobivenim rezultatima za originalnu sliku se kreću u rasponu od oko 2.5 stupnjeva, dok za dodani šum ta vrijednost raste na 3.9 stupnjeva, što je zavidan rezultat. Osnovni razlog za varijacije (odnosno greške), stoga, nije u dodjeli orijentacije već u samom izboru i detekciji lokacije i uvećanja.

Lokalni opisnik interesne točke

U ovom koraku preostaje dodijeliti opisni vektor svakoj interesnoj točki tako da se postigne invarijantnost na razlike u boji, osvjetljenju i točki gledišta. Ovaj dio algoritma je razvijen na osnovu modela koji su predstavili Edelman, Intrator i Poggio 1997. godine, a zasniva se na pokušaju oponašanja biološkog vida. Njihov pristup ima znatno bolje rezultate pri promjeni kuta gledanja (94% preciznosti, u odnosu na 35% alternativne metode).

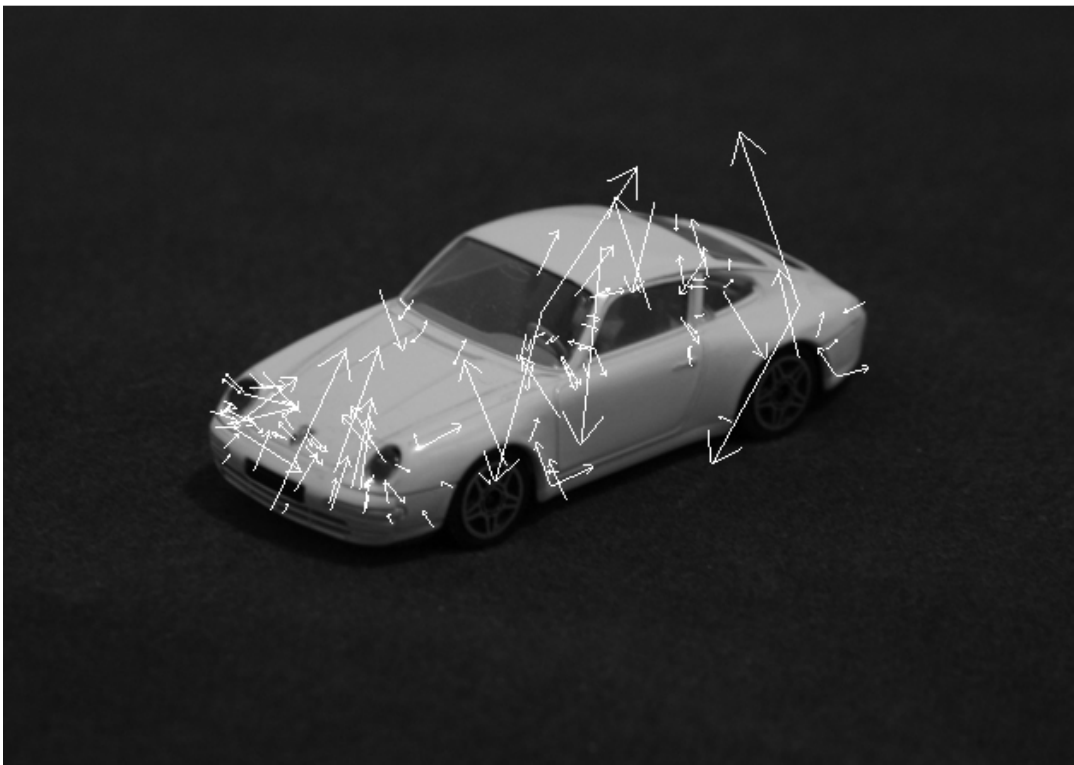
Uzimaju se područja od 4×4 pixela u okružju interesne točke te se predstavljaju kao skup orijentacijskih histograma. Od svakog od tih područja, stvara se vektor sa 8 potencijalnih smjerova, dakle ukupno je 128 podataka za svaku točku. Svaki smjer ima vrijednost proporcionalnu onoj predstavljenoj u histogramima u početnom 4×4 području. Osam je odabran kao broj smjerova (u odnosu na 4 ili 16) jer se pokazalo da se diljem raspona vrijednosti opisnog vektora zadržava najveću preciznost pravilne detekcije, čak i kod jako velikih zakreta u promjeni točke gledišta i dodacima šuma.

Vektor koji smo sad dobili, potrebno je dodatno normalizirati, dovesti na jediničnu duljinu. Ova normalizacija umanjuje potencijalne promjene u kontrastu. Promjene u svjetloći slike ne igraju ulogu u ovim izračunima, jer kod tih promjena konstantna vrijednost se dodaje svim pixelima, a mi uzimamo njihovu razliku, pa stoga ta promjena vrijednosti ne utječe na algoritam. S druge strane, moguć je utjecaj u slučajevima gdje svjetlost udara o površine koje drugačije reagiraju na nju u smislu transmisije, refleksije i apsorpcije te isto tako u općenitom 3-dimenzionalnom slučaju gdje svjetlost jednostavno udara na različite dijelove objekta drugačijim intenzitetom. Ove varijacije uglavnom utječu na veličinu vektora, a puno rjeđe na samu orijentaciju, pa se zbog toga u ovom koraku ograničavaju na određenu (eksperimentalno utvrđenu) vrijednost te ponovo renormaliziraju na jedinične vrijednosti, što znači da uparivanje veličina samih gradijenata nije toliko relevantno koliko su važni omjeri smjerova vektora. Rezultat ovoga jest dobiveni opisnik, čijim upoređivanjem pronalazimo korespondentne točke među slikama.

Ekstenzivnim testiranjem je dokazano da SIFT zadržava preciznost uparivanja od 70 do 85% za promjene u gledištu za 40° te iznad 50% za promjene od 50° . Skaliranje na manje mjerilo omogućuje precizno uparivanje za male i vrlo zaklonjene objekte dok su veća mjerila pogodna za slike sa mnogo šuma i zamućenosti. Konačno, možemo zaključiti da je SIFT invarijantan na uvećanje i rotaciju i gotovo invarijantan s obzirom na zaklonjenost, promjene u boji, osvjetljenosti, šum, zamućenje i razumne promjene u točki gledišta, a ti zavidni rezultati mu daju vrlo velik uporabni potencijal.



Slika 1 - Fotografija igračke autića



Slika 2 - Fotografija 1 sa označenim vektorima opisnicima

Slijedi primjer prikaza ispisa jedne ključne točke u XML formatu (konkretno iz programa koji koristi SIFT za spajanje slika u panoramu):

= <KeypointN>

<X>773.869729556549</X>

<Y>334.14281446069242</Y>

<Scale>3.3198600762446784</Scale>

<Orientation>3.0120025231173093</Orientation>

<Dim>128</Dim>

= <Descriptor>

<int>1</int>

<int>0</int>

<int>0</int>

<int>0</int>

<int>41</int>

<int>20</int>

<int>0</int>

<int>0</int>

<int>4</int>

<int>0</int>

<int>0</int> / niz je ovdje prekinut radi duljine, nepotpun prikaz isključivo radi primjera

</Descriptor>

</KeypointN>

Funkcionalnost i primjeri

SIFT je u načelu upotrebljiv u svim situacijama gdje je potrebno upariti lokacije ili objekte u dvije slike. Neki od primjera konkretne funkcionalnosti su slijedeći:

Prepoznavanje objekata koristeći SIFT

Robotika

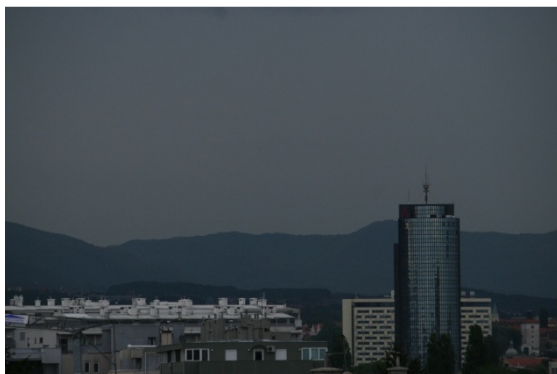
U robotici je najčešća uporaba SIFT-a pri određivanju lokacije ili iscrtavanju karte određenog prostora, odnosno konstrukciji 3-dimenzionalne karte okoline. U ovoj implementaciji, vrlo se često koristi stereo vid ili sustavi sa više kamera. Važno je, stoga, napomenuti Bayesovsko filtriranje koje se veže za kategoriju pozicioniranja robota u prostoru. Naime, u početku svog kretanja, robot se može smjestiti u točku (0,0) u koordinatnom sustavu, no kako se odmiče dalje od ishodišta koordinatnog sustava, vjerojatnost točnosti položaja u kojem robot misli da jest opada. Koristeći Bayesove rekursivne filtre, vjerojatnost o položaju se izračunava i konstantno ažurira novim podacima iz senzora.

3D rekonstrukcija scene

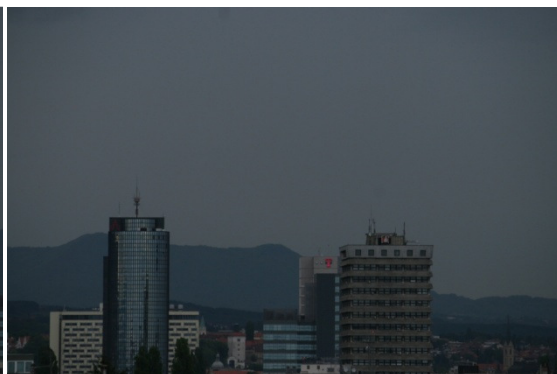
U ovoj primjeni, najprije se uzima nekoliko slika određenog prostora slikanih iz raznih položaja. SIFT traži korespondencije među svim pogledima na scenu te se gradi model određene scene u tri dimenzije. Kod već izrađenog modela okoline, moguće je koristiti SIFT za određivanje položaja kamera unutar te scene.

Spajanje fotografija u panorame

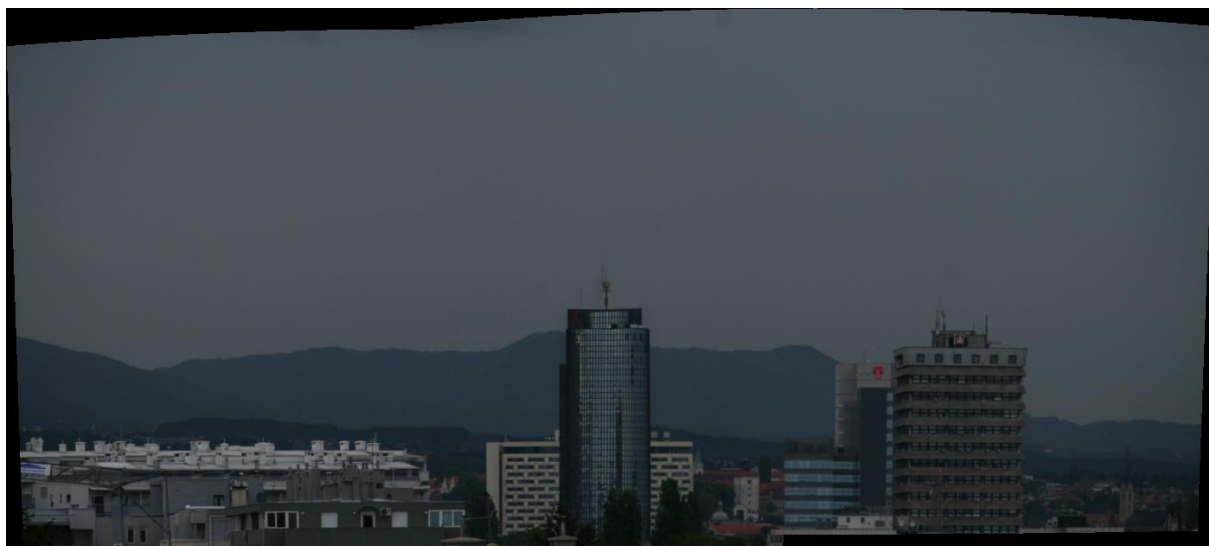
Upotreba je moguća i za spajanje fotografija u panorame. Traže se korespondencije između interesnih točaka svake od slika sa ostalim slikama te se na taj način slike pozicioniraju u panoramu. RANSAC algoritam se koristi za eliminaciju netočnih rezultata. RANSAC je jedan od algoritama koji izrađuje matematičke modele izolirajući iz skupine podataka one podatke numerički odvojene od ostatka podataka. Distantni podaci onda nemaju utjecaja na konačni rezultat algoritma. Primjer ove implementacije nalazimo kod autopano-sift programa (*Sebastian Nowozin*, <http://user.cs.tu-berlin.de/~nowozin/autopano-sift/>, 2005.).



Slika 3 - "Lijevi dio"



Slika 4 - "Desni dio"



Slika 5 - Kompletna panorama

Ostale primjene

Zbog svoje mogućnosti da vrlo brzo i precizno usporedi sliku sa velikim bazama podataka, SIFT ima veliku mogućnost za primjenu u nadgledanju, bilo za detekciju neželjenih objekata, prebrojavanje...

Prisjetimo se ranije navedene primjene u vojne svrhe, primjerice za navođenje raketa. Sve što bi bilo potrebno za 95% precizno navođenje rakete jest prethodna satelitska snimka objekta (ili približni model).

Primjeri

U ovom poglavlju biti će prikazani neki primjeri funkcionalnosti SIFT-a. Za ove primjere, korišten je prvenstveno Loweov algoritam, ali bit će prikazane i neke druge verzije.

Najprije je uzet primjer fotografije ključa (slika 6).



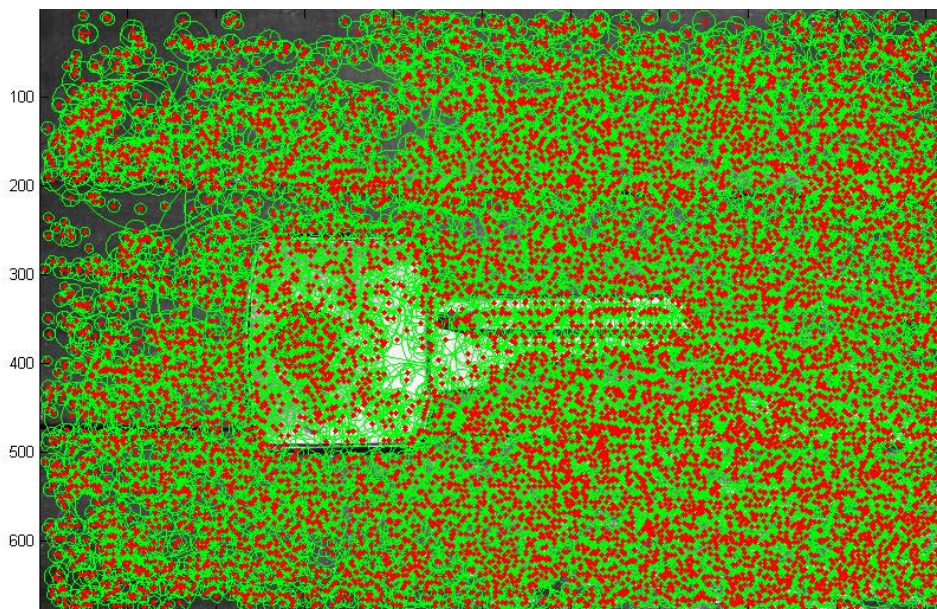
Slika 6 - Ključ

Ključ je fotografiran na kontrastno tamnoj površini te je pretpostavka da je ta fotografija dobra osnova za usporedbu sa drugim fotografijama. Slijedeća fotografija (7) će biti uspoređena s ovom.



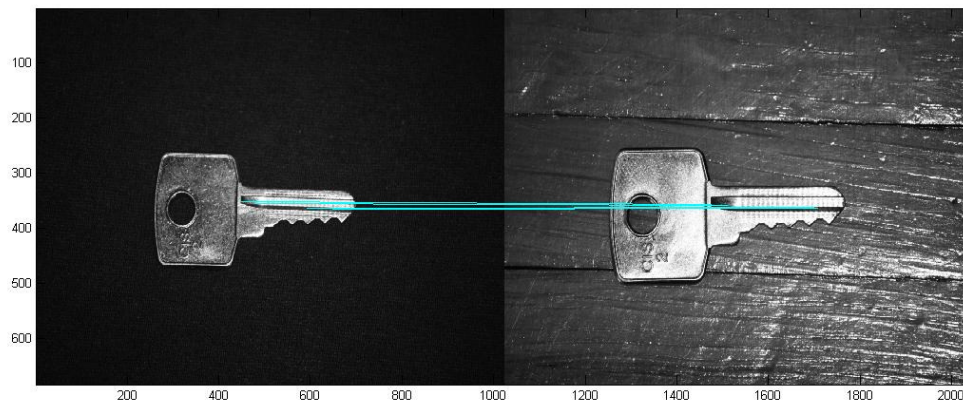
Slika 7 - Ključ (parket)

Na ovoj fotografiji (7), ključ leži na mnogo neprikladnijoj površini. Uočljivo je mnogo refleksije o parket i ključ.

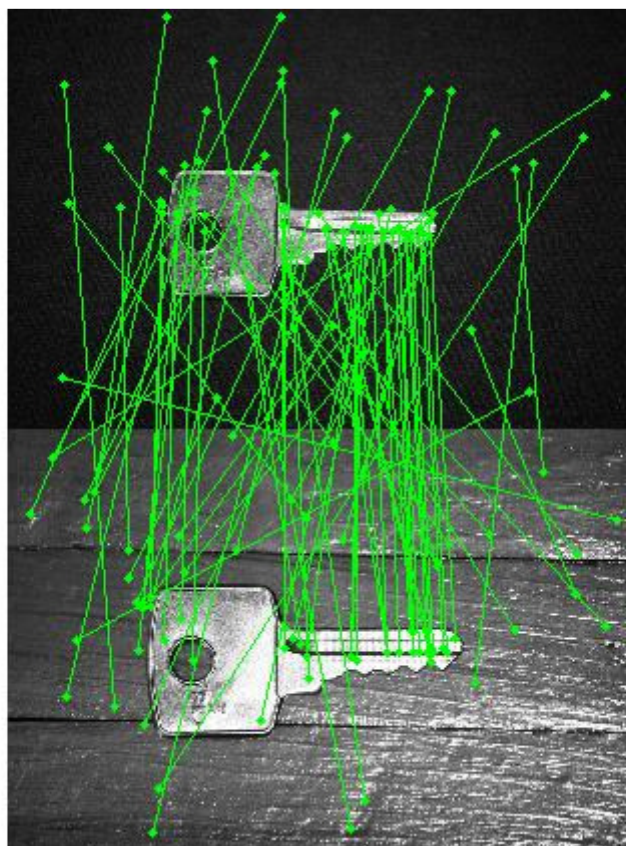


Slika 8 - Ključ - potencijalne interesne točke

Fotografija 8 je nastala iz fotografije 6 i prikazuje sve točke kandidate za interesne točke. Fotografije 9 i 10 su prikazi usporedbe fotografija 6 i 7 koristeći Loweov program (9) te alternativni program (A. Vedaldi, 10):



Slika 9 - Slike 6 i 7, Loweov program



Slika 10 - Slike 6 i 7, alternativni program

Loweova ivedba se ograničila na 5 parova točaka (neočekivano malo, ali dovoljno za uparivanje) dok je alternativni program pronašao mnogo više, no nažalost uz veliki broj netočno uparenih točaka. Vidljivo je da su se oba programa teško nosila sa testom sa ključem, vjerojatno zbog reflektivne površine metala i parketa te različitog osvjetljenja. Učinjeno je još nekoliko testova uz rotaciju i zaklonjenje ključa, ali oba programa su dala katastrofalne rezultate od niti jednog ili eventualno jednog netočno određenog para točaka. Slijedeći test će biti izveden koristeći subjekt čija je boja manje podložna promjenama osvjetljenja te sam materijal izrade više mat (slika 11).



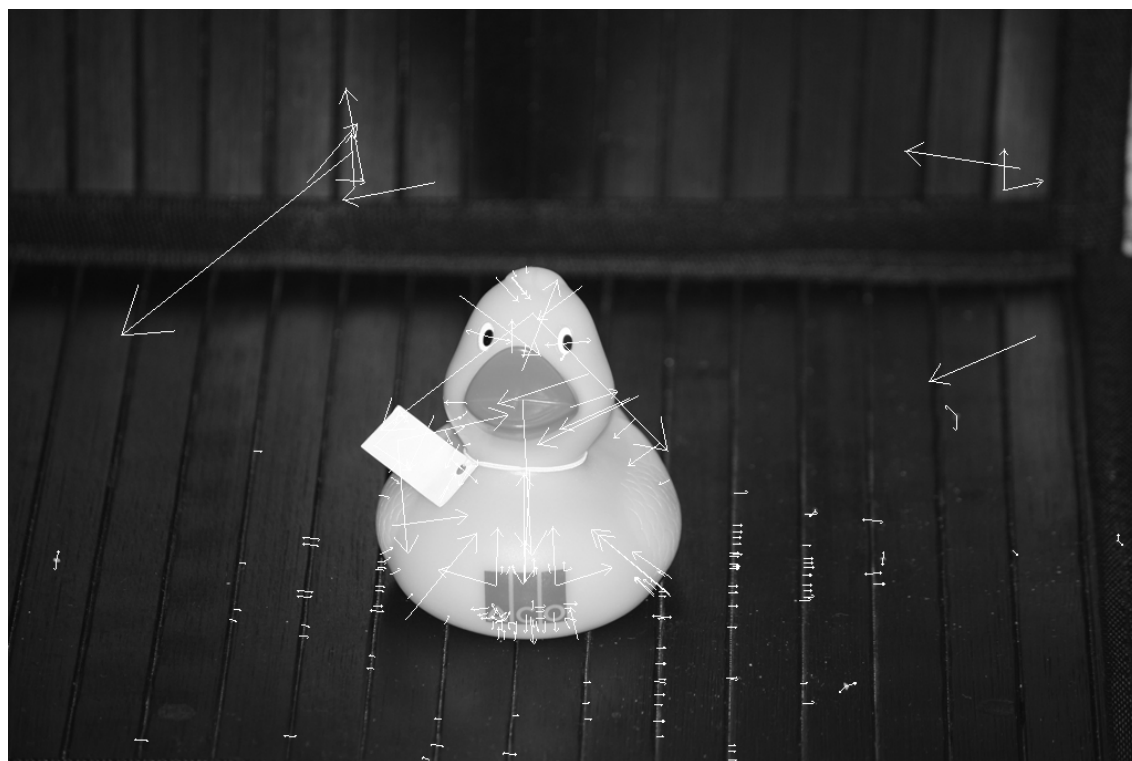
Slika 11 - Subjekt (gumena patkica)

Fotografiju gumene patkice (11), usporedit ćemo sa još četiri fotografije (12, 16, 17, 18) reprezentativne po rotaciji, promjeni veličine, te zaklonu na fotografiji.



Slika 12 - Subjekt sa vrlo malo promjena

Fotografija 12 je vrlo slična fotografiji 11, veličina, točka gledišta i kut rotacije su ostali jednaki, samo je stavljen dodatni poklopac na patkicu (kao minorna promjena na fotografiji s kojom se Loweov algoritam ne bi trebao mučiti). Fotografije 13 i 14 prikazuju detektirane interesne točke (opisnike) na fotografijama 11 i 12.

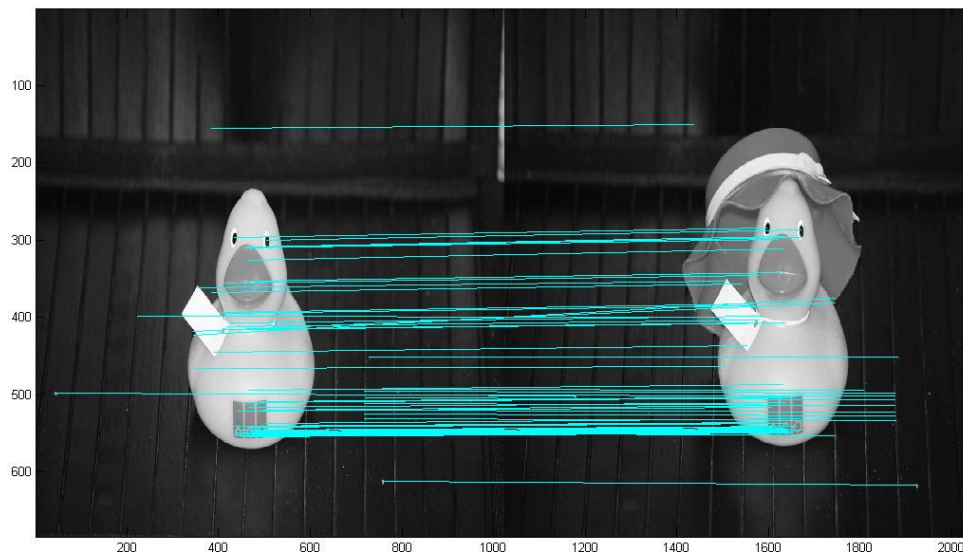


Slika 13 - Fotografija 11 sa prikazanim vektorima opisnicima



Slika 14 - Fotografija 12 sa prikazanim vektorima opisnicima

Konačno, usporedba tih dviju fotografija, daje nam slijedeću fotografiju, 15:



Slika 15 - Slike 11 i 12, parovi - Lowe

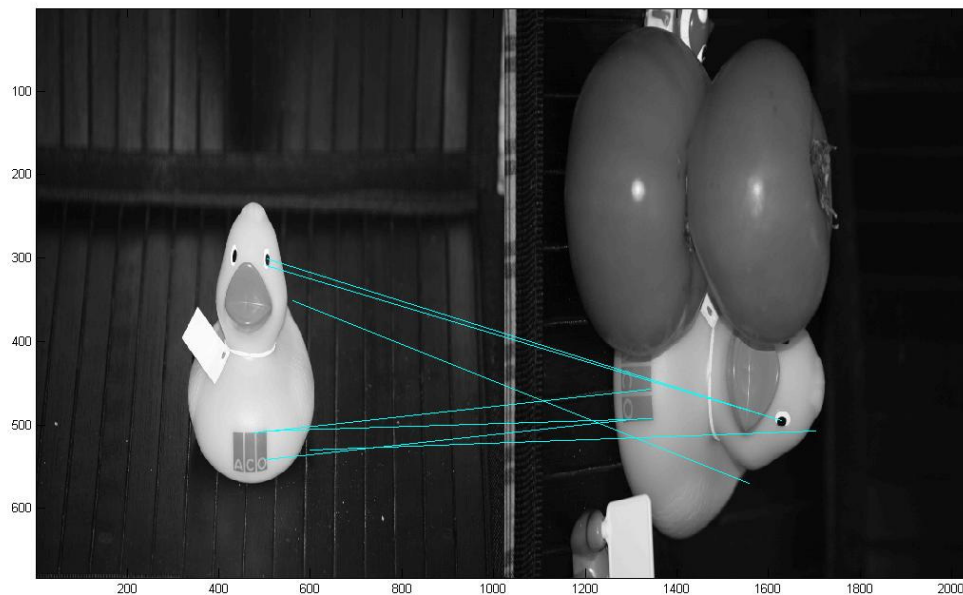
Paralelne crte korektno ukazuju na vrlo precizno uparivanje velikog broja interesnih točaka. Obje slike su imale oko 200 odabranih točaka, a upareno ih je oko pedeset. Algoritam se ovdje pokazao vrlo uspješnim. Fotografija 16 je prikaz usporedbe još jedne fotografije sa originalnom, 11. Ta je fotografija reprezentativna za pitanja zaklona i promjene veličine subjekta.



Slika 16 - Slika 11 upoređena sa zaklonjenom i umanjenom slikom

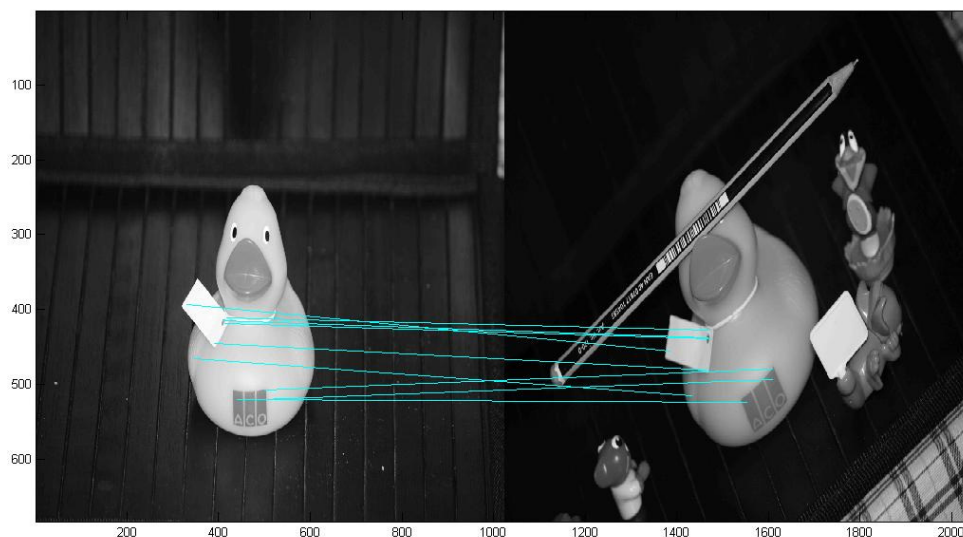
Algoritam se i ovdje relativno dobro snašao, pronašavši preko 10 odgovarajućih točaka od kojih su samo dvije netočno uparene. Načinjena su još dva testa sa patkicama, slika broj 17, gdje je cijela slika rotirana za 90° u smjeru kazaljke na satu uz zaklon i uvećanje. Pronađeno je sedam točaka od kojih su

sve točno određene. Na slikama 16 i 17 se vrlo dobro vidi kako se program oslanja na područja visokog kontrasta (primjerice natpis na „trbuhu“ patkice, oko i neki od rubova).



Slika 17 - Slika 11 upoređena sa zaklonjenom i zarotiranom te uvećanom fotografijom

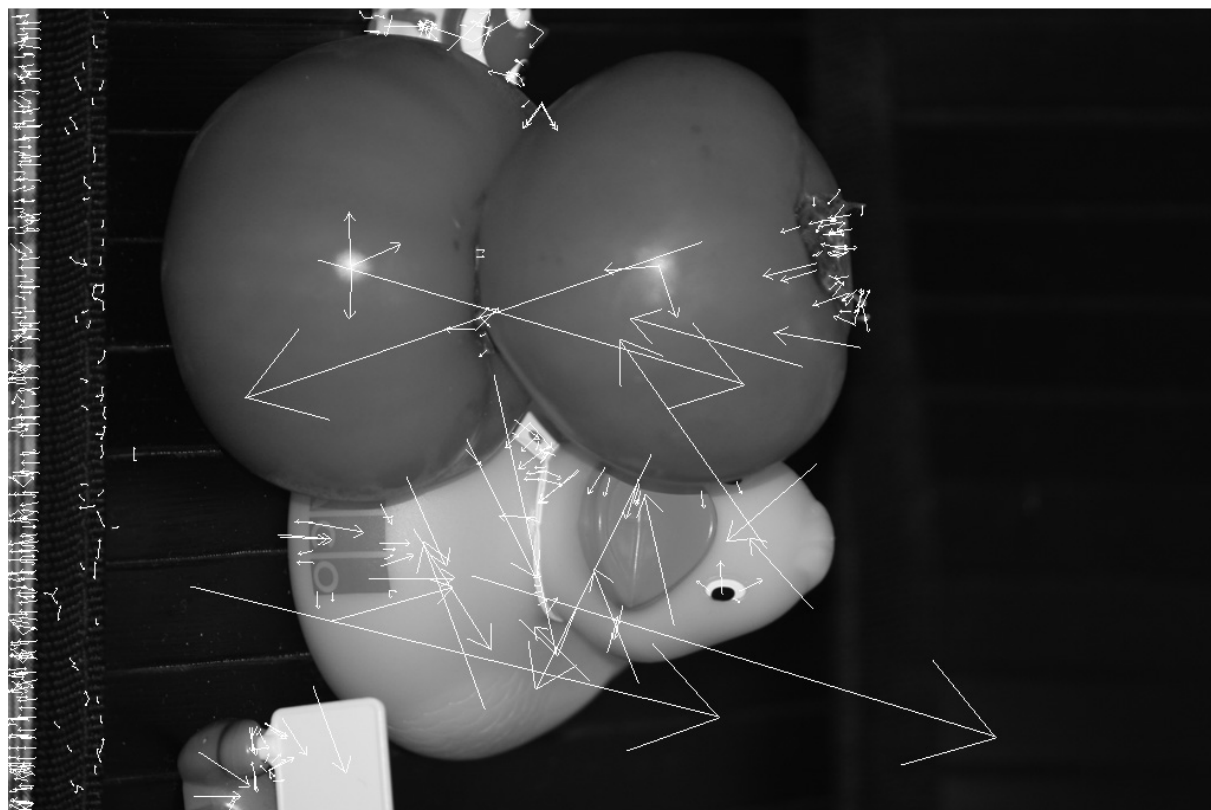
Posljednji testni rezultat prikazuje usporedbu slike 11 sa fotografijom (18) gdje je promjenjena točka gledišta s koje je fotografija snimljena, a samim time i osvjetljenje, fotografija je zarotirana za 30ak stupnjeva u smjeru suprotnom od onog kazaljke na satu te je položena olovka preko očiju patkice na koje se algoritam oslanjao do sad (fotografija je imala oko 600 odabranih točaka).



Slika 18 - Slika 11 upoređena sa fotografijom gdje je prisutna promjena točke gledišta

Broj pronađenih parova se čak i povećao na 9 te se algoritam opet pokazao preciznim.

Prvotni test s ključem se nije pokazao dobrim, no test na kojem se koristila gumena patkica je rezultirao vrlo uspješnim pokazateljima. Valja napomenuti da su oba programska rješenja izrađena kao aplikacije za Matlab. Provedba testiranja tim programima se pokazala relativno dugotrajnom i nepraktičnom jer je rad algoritma, koji je uključivao pronalazak točaka na dvije fotografije te uspoređivanje istih, trajao između jedne i tri minute za jedan par fotografija. Implementacije u programskim jezicima C, C++ ili C# se pokazuju daleko bržima, pogotovo kad ih se upoređuje sa unaprijed izgrađenom bazom podataka. Zanimljivo je još pogledati jednu fotografiju – 19, kopiju fotografije korištene za izradu slike 17. Na njoj se broj točaka približio jednoj tisući, jer se u kadru uz rub fotografije vidi dio stolnjaka sa područjem koje sadrži velik broj detalja i kontrastnih crta.



Slika 19 - Fotografija korištena za sliku 17 sa prikazanim vektorima opisnicima

Zaključak

U ovom radu je prikazana teorija i primjena jednog od najrelevantnijih pokušaja rješenja nekih od problema računalnog vida. Prikazan je algoritam koji rezultira doista zavidnim rezultatima, zadržavajući visoku razinu uspješnog rada unatoč mnogim potencijalnim problemima poput zakretanja, promjene veličine, šuma i ostalog ranije navedenog, što je i potkrijepljeno primjerima. U većini slučajeva, SIFT i ostali opisnici bazirani na SIFT-u (PCA-SIFT, ostali GLOH opisnici) daju mnogo bolje rezultate od konkurentih opisnika.

Računalni vid i njemu bliske znanstvene discipline tek čekaju neminovno razdoblje u kojem ih očekuje doista široka komercijalna primjena popraćena velikim ulaganjima. Za očekivati je, stoga, daljnji razvoj i sve češću primjenu ovog relativno novog algoritma.

Literatura

1. Lowe, D. G. Distinctive Image Features from Scale-Invariant Keypoints. Sječanj 2004. <http://www.cs.ubc.ca/~lowe/papers/ijcv04.pdf>, 2. travnja 2008.
2. Lowe, D. G. SIFT demo program. Srpanj 2005. <http://www.cs.ubc.ca/~lowe/keypoints/>, Demo Software: SIFT Keypoint Detector, 26. travnja 2008.
3. Lowe, D. G. Object Recognition from Local Scale-Invariant Features. Rujan 1999. International Conference on Computer Vision, Corfu
4. Scale-Invariant Feature Transform. Travanj 2008. http://en.wikipedia.org/wiki/Scale-invariant_feature_transform, 28. travnja 2008.
5. RANSAC. Travanj 2008. <http://en.wikipedia.org/wiki/RANSAC>, 28. travnja 2008.
6. Computer vision. Travanj 2008. http://en.wikipedia.org/wiki/Computer_vision, 26. travnja 2008.
7. Pinz, A. Object Categorization. 2006. Graz University of Technology, Austria