

# Optičko praćenje prstiju i prepoznavanje karakterističnih putanja

---

*Realizovan je sistem za prepoznavanje cifara gestikulacijom prstiju, u vazduhu, i prepoznavanje istih. Sistem je implementiran u programskom jeziku Matlab. Za prepoznavanje korišćen je Matlabov alat Neural network toolbox. Sistem je testiran na arapskim cifarama koje se zapisuju jednopotezno. Problem predstavlja zapisivanje odradenih karaktera koji zahtevaju više od jednog manevra pri zapisivanju.*

---

## Princip rada

Sistem se sastoji od kamere i indikatora pričvršćenih na vrhove palca i kažiprsta. Pri spajanju indikatora kamera ne beleži nikakvo kretanje, u slučaju da se indikatori razdvoje kamera beleži putanju kažiprsta sve dok opet ne dođe do kontakta između indikatora. Sa slike 1 vidimo da su obezbeđeni idealni uslovi radi preciznijeg izdvajanja indikatora na video snimku.

Video zapis se analizira u HSV formatu. Za izdvajanje definišu se pragovi za svaku komponentu (H, S i V). Pragovi predstavljaju interval vrednosti za svaku komponentu koja definiše boju indikatora. Po izdvajanju boje iscrtavaju se centri mase (slika 2) svakog indikatora.

Vrednosti koordinata centra mase izračunavaju se predstavljenim formulama (Davies 2005):

$$T_x = \frac{1}{p} \cdot \sum_{i=1}^m \sum_{j=1}^n A(i, j) \cdot j$$

$$T_y = \frac{1}{p} \cdot \sum_{j=1}^n \sum_{i=1}^m A(i, j) \cdot i$$

gde je  $p$  – ukupan broj piksela koji odgovaraju indikatoru,  $A$  – slika,  $i, j$  – koordinate svakog piksela, a  $m$  i  $n$  – dužina i širina slike.

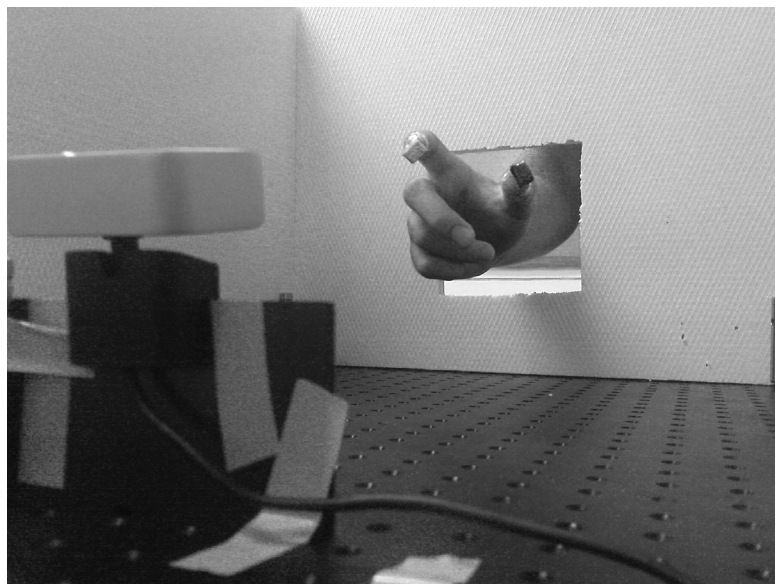
---

*Nemanja Janković  
(1992), Loznica, Vojvode  
Putnika L2/16, učenik 4.  
razreda Tehničke škole u  
Loznici*

### MENTORI:

*Filip Panjević, Beograd,  
Microsoft Development  
Center Serbia*

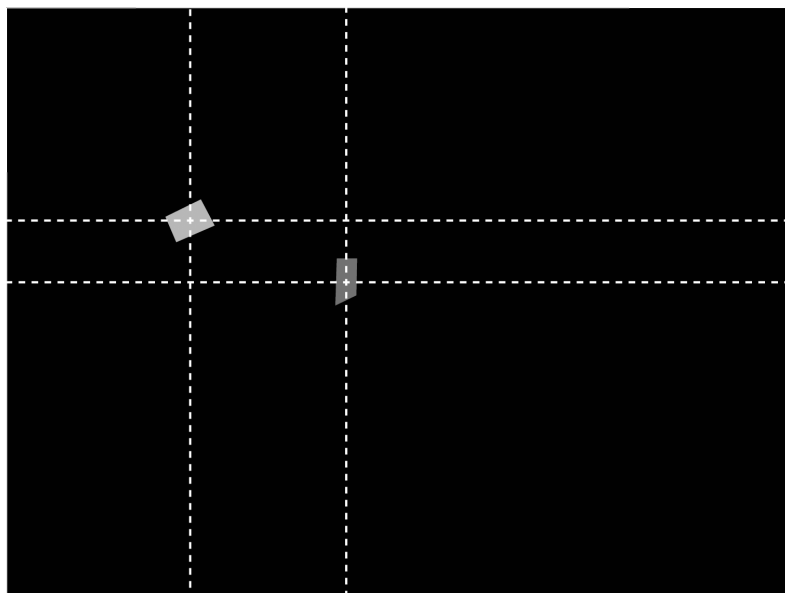
*Ivan Razumenić, Vršac,  
student Elektrotehničkog  
fakulteta Univerziteta u  
Beogradu*



Slika 1.  
Okruženje sistema

---

Figure 1.  
System environment

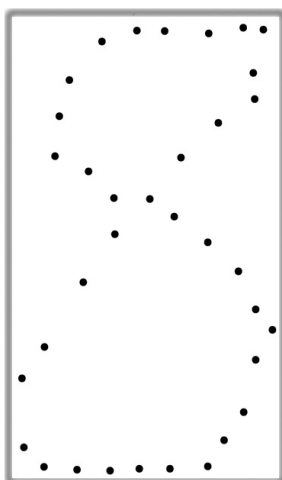


Slika 2.  
Izdvojeni indikatori  
sa iscrtanim  
težištima

---

Figure 2.  
Recognized finger  
indicators and  
centres of their mass

Prilikom traženja težišta indikatora na prvom frejmu, obrađuje se cela slika. U analizi narednih frejmova obuhvaćena je predefinisana okolina prethodnog težišta. U slučaju da u tom opsegu ne nađe težište opseg se proširuje. U slučaju kada se traži težište u celom opsegu vreme koje je potrebno za nalaženje je približno 7 ms, dok vreme sa optimizovanim kodom iznosi približno 0.8 ms, na ovaj način se povećava broj obrađenih frejmova i povećava rezolucija putanje prsta. Na slici 3 prikazan je primer određene putanje prsta.



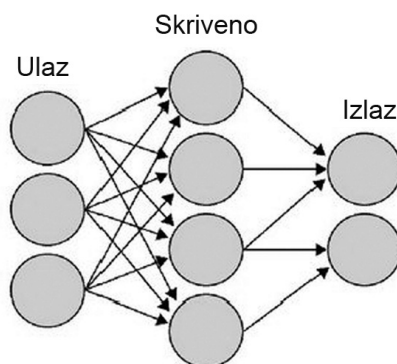
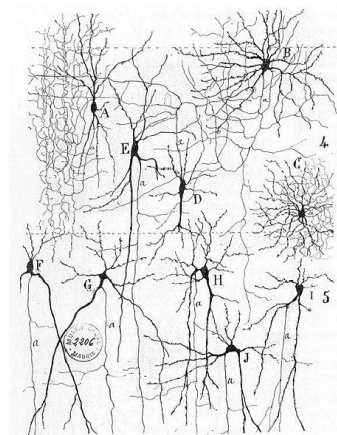
Slika 3.  
Isčrtana težišta za  
svaki frejm

Figure 3.  
Traced focuses for  
every frame

## Prepoznavanje

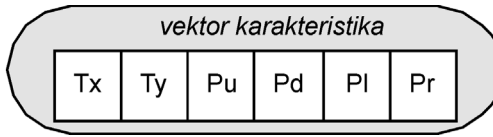
Neuronske mreže (slika 4) se koriste za modelovanje kompleksnih odnosa između ulaza i izlaza sistema koje je teško ili nemoguće opisati analitički (Hagan *et al.* 1996). Zato se često koriste u problemima prepoznavanja oblika. Korišćena je troslojna neuronska mreža.

Vektor karakterističnih osobina, ili vektor karakteristika (eng. feature vector) predstavlja skup osobina koje opisuju zapisan karakter. U ovom radu definisano je šest osobina za vektor karakteristika. Nakon zapisivanja putanja indikatora, rezultat se prosleđuje algoritmu koji izračunava osobine putanje i smešta ih u vektor karakteristika (slika 5), koji se prosleđuje neuronskoj mreži.



Slika 4.  
Biološka (levo) i  
veštačka (desno)  
neuronska mreža

Figure 4.  
Biological (left) and  
artificial (right)  
neural network



Slika 5.  
Vektor karakteristika

Figure 5  
Feature vector

Prve dve osobine predstavljaju centre masa svih izdvojenih težišta na video snimku. Centri mase se računaju u okviru u kome se karakter nalazi (slika 3). Npr. centar mase broja 9 očekujemo u gornjoj polovini slike, broja 6 u donjoj. Sledeće osobine su broj koliko prosečno puta horizontalna linija preseca zapisani karakter. Algoritam je napisan tako da za svaki pixel na Y osi povlači horizontalnu liniju i broji koliko puta je presekle zapisan karakter. Računa se prosečan broj preseka u gornjoj i donjoj polovini slike.

Formule za računanje broja preseka su:

$$P_d = \frac{\sum_{i=1}^{h/2} S_i}{\frac{h}{2}}$$

$$P_u = \frac{\sum_{i=h/2}^h S_i}{\frac{h}{2}}$$

gde je  $P_d$  – prosečan broj preseka u donjoj polovini,  $P_u$  – prosečan broj preseka u gornjoj polovini,  $h$  – visina slike, a  $S$  – broj preseka po jednoj horizontali. Na primer, očekivani broj preseka broja jedan u donjoj polovini je jedan. Očekivani broj preseka broja osam za obe polovine je 2.

Poslednje dve osobine vektora karakteristika predstavljaju broj koliko prosečno puta vertikalna linija preseca zapisani karakter. Princip računanja je isti kao i kod prethodne dve osobine s tim što se u ovom slučaju posmatraju leva i desna polovina slike.

Formule za računanje broja preseka su:

$$P_L = \frac{\sum_{i=1}^{w/2} S_i}{\frac{w}{2}}$$

$$P_R = \frac{\sum_{i=w/2}^w S_i}{\frac{w}{2}}$$

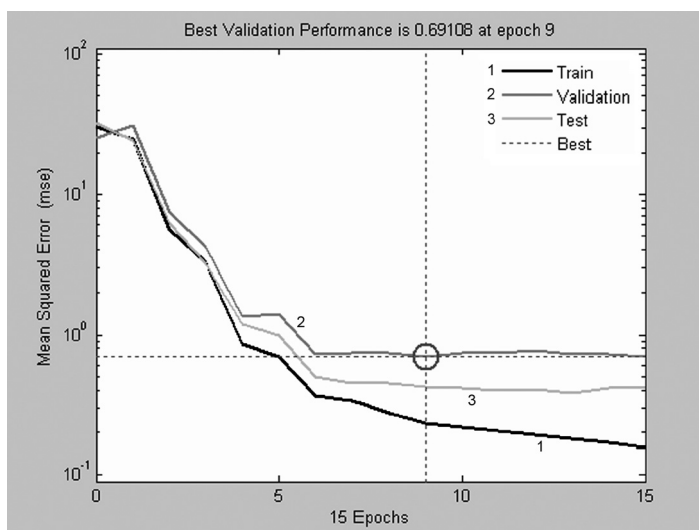
gde je  $P_L$  – prosečan broj preseka na levoj polovini,  $P_R$  – prosečan broj preseka na desnoj polovini,  $w$  – širina slike, a  $S$  – prosečan broj preseka po jednoj vertikali. Na primer, očekivani prosečan broj preseka, na levoj strani, za broj jedan je 1. Za broj osam na obe strane je 4.

Za implementaciju neuronske mreže, čiji zadatak je prepoznavanje karaktera, korišćen je Matlabov alat Neural Network Toolbox (www.mathworks.com). Levenberg-Marquardt algoritam (Lourakis 2005) je korišćen za obučavanje mreže. Potrebno mu je proslediti set podataka kojim se obučava da uradi određeni zadatak. Pri generisanju neuronske mreže potrebno je podesiti koliki deo ulaznih podataka se koristi za trening, verifikaciju i test. Podešeno je 60% uzoraka za trening, 20% za verifikaciju i 20% za test. Takođe je bilo potrebno podesiti arhitekturu mreže. Namješteno je da mreža operiše sa 12 skrivenih neurona. Broj neurona u skrivenom sloju je duplo veći od broja osobina. Nije vršeno istraživanje u smeru kako broj neurona u srednjem sloju utiče na performanse mreže.

## Rezultati i diskusija

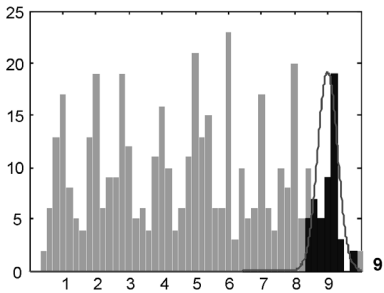
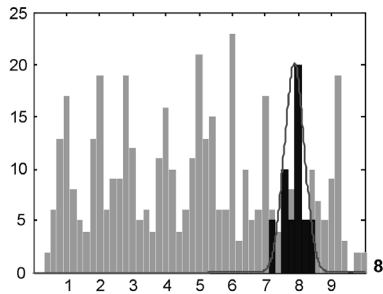
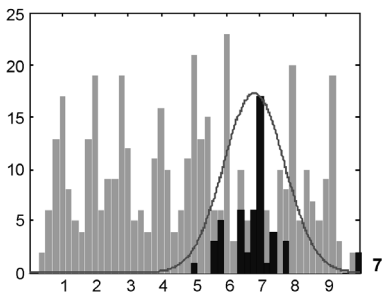
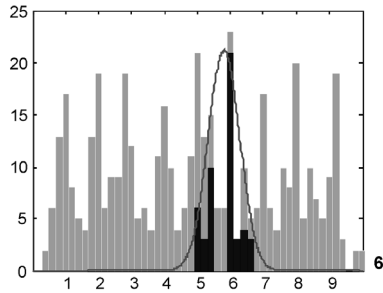
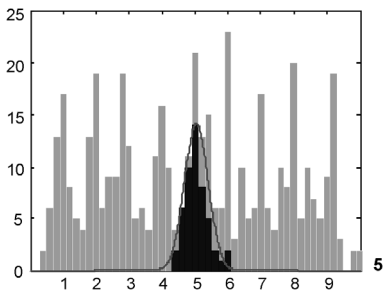
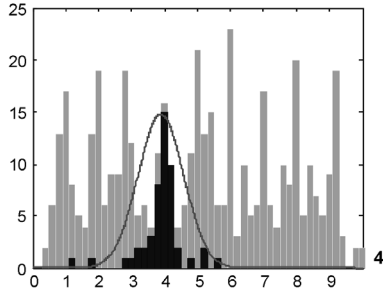
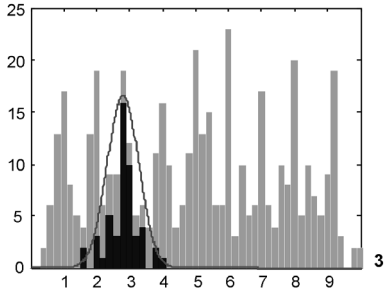
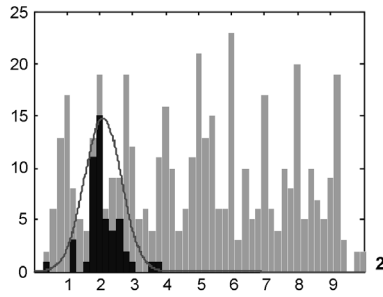
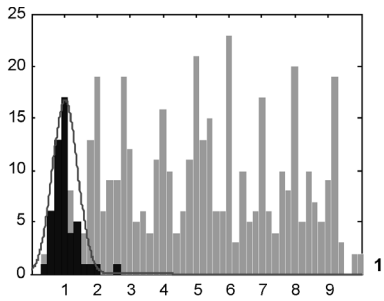
Korišćeno je 450 uzoraka prikupljenih od različitih osoba. Svaka osoba napravila je po 10 uzoraka za svaku cifru, ukupno 5 osoba. Mreža na izlazu vraća niz realnih brojeva, koji predstavljaju vrednosti koje je mreža izračunala posredstvom skrivenih neurona. Za svaki uzorak se određenom funkcijom, koristeći osobine vektora karakteristika, računa kojoj klasi taj uzorak pripada. Sve dobijene vrednosti klasifikuju se u 9 klasa. Klase predstavlja prepoznate brojeve od 1 do 9

Na grafiku (slika 6) je predstavljena zavisnost greške od epoha. Epoha je proces u kom mreža jednom obradi sve uzorke. Najmanja greška koju je mreža uspeła da svede prikazana je na grafiku. Crnom linjom (1) predstavljeni su uzorci korišćeni za trening, tamnosivom (2) uzorci za verifikaciju a svetlosivom (3) test uzorci.



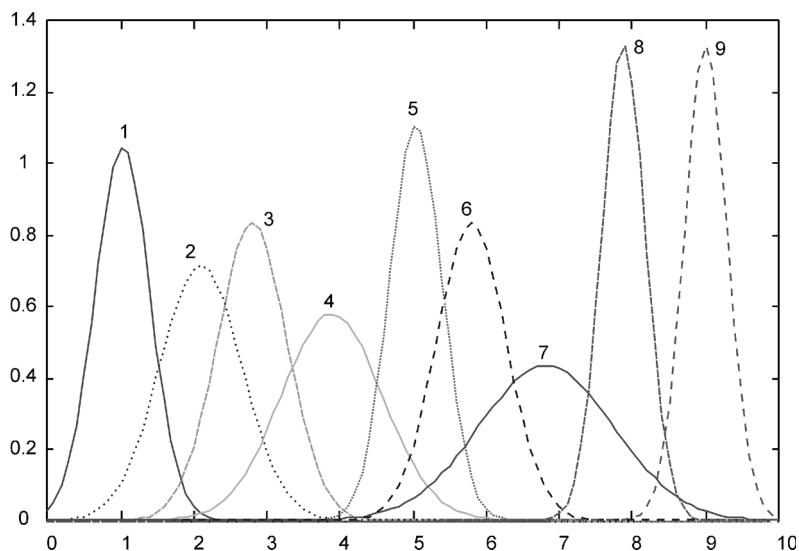
Slika 6.  
Performanse  
neuronske mreže

Figure 6.  
Neural network  
performance



Slika 7.  
Histogrami  
prepoznavanja za  
svaku cifru

Figure 7.  
Recognition rate  
histograms for each  
number



Slika 8.  
Prepoznavanje pojedinačnih cifara – aproksimacija Gausijanom (na slici grafici nisu normalizovani)

Figure 8.  
Recognition rate for each number – Gaussian approximation (not normalized)

Procenat prepoznavanja cifara dobija se brojanjem koliko se uzoraka našlo u određenom intervalu. Za svaku cifru uzet je interval  $\pm 0.4$  od tražene vrednosti (na primer interval koji odgovara cifri jedan je  $[0.6, 1.4]$ ).

U tabeli 1 prikazan je procenat slučajeva kada je cifra prepoznata.

Tabela 1. Postotak prepoznavanja cifara

Cifra	1	2	3	4	5	6	7	8	9
Procenat	76	72	66	78	86	62	62	90	76

Na grafikonima (slika 7) predstavljeni su histogrami za svaku cifru pojedinačno, a na slici 8 odgovarajuće aproksimacije normalnom raspodelom. Parametri ovih raspodela dati su u tabeli 2.

Tabela 2. Parametri normalnih raspodela

Cifra	1	2	3	4	5	6	7	8	9
Sr. vrednost	1.03	2.1	2.8	3.89	5.04	5.86	6.82	7.88	9.00
St. devijacija	0.38	0.56	0.48	0.69	0.36	0.48	0.92	0.30	0.30

## Zaključak

Ostvareno je prepoznavanje karaktera zapisanih gestikulacijom prstiju. Uspešno je optimizovan kod za rad na željenom nivou. Kao nadogradnja projekta može se realizovati zapisivanje više od jednog karaktera, kao i zapisivanje jednog karaktera iz više manevara. Prosečno je prepoznato 74% uzoraka.

## Literatura

- Davies E. R. 2005. *Machine Vision*. Morgan Kaufman
- Hagan M. T., Demuth H. B., Beale M. H. 1996. *Neural network design*. Stillwater (Oklahoma): Martin Hagan
- Lourakis M. 2005. A Brief Description of the Levenberg-Marquardt Algorithm Implemented by levmar. Dostupno na <http://www.ics.forth.gr/~lourakis/levmar/levmar.pdf>
- [http://www.igi.tugraz.at/lehre/EW/tutorials/nnt\\_intro/nnt\\_intro.pdf](http://www.igi.tugraz.at/lehre/EW/tutorials/nnt_intro/nnt_intro.pdf)
- [http://www.gcseguide.co.uk/standard\\_deviation.htm](http://www.gcseguide.co.uk/standard_deviation.htm)

---

*Nemanja Janković*

### Optical Tracking and Recognition of Fingertips' Characteristic Path

This paper presents a system for the recognition of the characteristic path made by fingertip gesticulations. Recognition is achieved by using neural networks. The system is implemented in MATLAB and tested for recognition of numbers drawn with fingertip.

