SVEUČILIŠTE U ZAGREBU
PRIRODOSLOVNO - MATEMATIČKI FAKULTET
MATEMATIČKI ODSJEK

Uvod u složeno pretraživanje podataka

Prepoznavanje znakovnog jezika pomoću tenzora

Petra Sočo, Jelena Zaninović
Zagreb, 8.3.2021.

Sadržaj

1 Uvod **2**

2 Tenzorski model **2**
2.1 Osnovno o tenzorima 2
2.2 Prepoznavanje ruke 3
2.3 Reprezentacija podataka tenzorom 4
2.4 Prepoznavanje znakovnog jezika tenzorom 4

3 Eksperimentalni rezultati **5**
3.1 Priprema podataka 5
3.2 Rezultati 6

1 Uvod

Prepoznavanje gesti rukama lako nalazi primjenu u svakodnevnom životu. Računalno prepoznavanje gesti može zamijeniti direktni kontakt s ekranima u javnosti ili u prostorima koji se pokušavaju držati sterilnima (npr. operacijske sale). Osim toga, omogućila bi se intuitivnija i jednostavnija interakcija ljudi s računalima (smart uređaji, AR, VR). Jedna od zanimljivijih i korisnijih primjena je kod prepoznavanja znakovnog jezika; naime, računala bi mogla automatski transkribirati znakove u tekst ili govor, što bi moglo olakšati i ubrzati komunikaciju s osobama koje ga koriste.
Kako bi program što bolje funkcionirao u praksi, moramo razmisliti na koji će način primati informacije - u našem slučaju slike ruku. Osobi nije praktično gestikulirati iz jedne fiksne poze pa ima smisla prilagoditi program da prepoznaje znakove iz različitih kuteva. Budući da se ne može uvijek osigurati konzistentno, umjetno svjetlo, htjeli bismo izbjeći i osjetljivost na osvjetljenje.

2 Tenzorski model

2.1 Osnovno o tenzorima

Definicija 1 *Ako je polje $\mathcal{A}$ određeno s N indeksa, kažemo da je $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$* ***tenzor reda N*** *nad $\mathbb{R}$, $\mathcal{A} = (x_{i_1 i_2 \ldots i_{n-1} i_n i_{n+1} \ldots i_N})$, $i_1 = 1, \ldots I_1; \ldots;\ i_N = 1, \ldots I_N$.*

U daljnjem tekstu s a označavamo skalare, s $\mathbf{a}$ vektore, s $\mathbf{A}$ matrice, a s $\mathcal{A}$ tenzore. Većinski ćemo raditi s tenzorima reda 3.

Definicija 2 *Neka je $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ tenzor. I_n-dimenzionalni vektor dobiven tako da fiksiramo svaki indeks osim indeksa i_n zovemo* ***nit u modu n****.*

Definicija 3 *Neka je $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ i neka je $\mathbf{U} \in \mathbb{R}^{J_n \times I_n}$.* ***Produkt u modu n*** *tenzora $\mathcal{A}$ i matrice $\mathbf{U}$*, s oznakom $\mathcal{A} \times_n \mathbf{U}$, je $(I_1 \times I_2 \times \ldots I_{n-1} \times J_n \times I_{n+1} \times \cdots \times I_N)$-dimenzionalni tenzor zadan s

$$(\mathcal{A} \times_n \mathbf{U})_{i_1 i_2 \ldots i_{n-1} j_n i_{n+1} \ldots i_N} := \sum_{i_n} a_{i_1 i_2 \ldots i_{n-1} i_n i_{n+1} \ldots i_N} u_{j_n i_n}.$$

Produkt u modu n možemo izraziti i na sljedeći način

$$(\mathcal{A} \times_n \mathbf{U})_{(n)} = \mathbf{U} \cdot \mathbf{A}_{(n)},$$

gdje smo s $\mathbf{A}_{(n)}$ označili **matricizaciju** od $\mathcal{A}$ **u n-tom modu**.

Teorem 4 *Neka je $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$. Za $\mathbf{A} \in \mathbb{R}^{I_m \times J_m}$, $\mathbf{B} \in \mathbb{R}^{I_n \times J_n}$, $m \neq n$ vrijedi*

$$\mathcal{A} \times_m \mathbf{A} \times_n \mathbf{B} = (\mathcal{A} \times_m \mathbf{A}) \times_n \mathbf{B} = (\mathcal{A} \times_n \mathbf{B}) \times_m \mathbf{A}. \tag{1}$$

Teorem 5 (HOSVD) *Neka je* $\mathcal{A} \in \mathbb{R}^{l \times m \times n}$. *Tenzor* $\mathcal{A}$ *možemo zapisati kao*

$$\mathcal{A} = \mathcal{S} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)} \tag{2}$$

gdje su $\mathbf{U}_1 \in \mathbb{R}^{l \times l}, \mathbf{U}_2 \in \mathbb{R}^{m \times m}, \mathbf{U}_3 \in \mathbb{R}^{n \times n}$ *ortonormalne matrice, a* ***jezgreni tenzor*** $\mathcal{S}$ *je istih dimenzija kao* $\mathcal{A}$ *te ima svojstvo potpune ortogonalnosti. Vrijedi i da je*

$$\mathcal{S} = \mathcal{A} \times_1 (\mathbf{U}^{(1)})^T \times_2 (\mathbf{U}^{(2)})^T \times_3 (\mathbf{U}^{(3)})^T. \tag{3}$$

Definicija 6 (Moore-Penroseov pseudoinverz)
Neka je $\mathbf{A} \in \mathbb{R}^{m \times n}$. *Kažemo da je matrica* $\mathbf{A}^+ \in \mathbb{R}^{n \times m}$ ***pseudoinverz*** *matrice* $\mathbf{A}$ *ako zadovoljava sljedeće uvjete:*

(i) $\mathbf{A}\mathbf{A}^+\mathbf{A} = \mathbf{A}$

(ii) $\mathbf{A}^+\mathbf{A}\mathbf{A}^+ = \mathbf{A}^+$

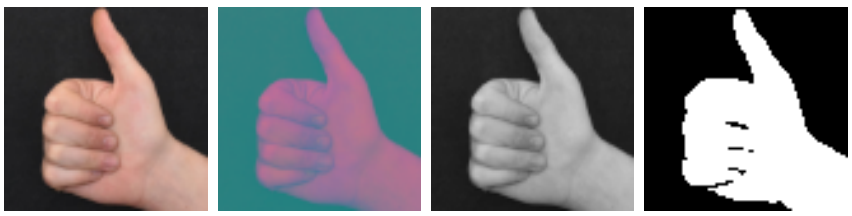
(iii) $(\mathbf{A}\mathbf{A}^+)^* = \mathbf{A}\mathbf{A}^+$

(iv) $(\mathbf{A}^+\mathbf{A})^* = \mathbf{A}^+\mathbf{A}$

2.2 Prepoznavanje ruke

Cilj nam je na slikama pomoću boja prepoznati ruku. Piksele dijelimo na 2 klastera: "boja kože" i "ostalo". Što se tiče modela boja, imamo mogućnost izbora. Premda je RGB praktičan za korištenje u većini slučajeva, nama on nije optimalni izbor zbog svoje osjetljivosti na osvjetljenje. Osim toga, RGB čuva više informacija o bojama nego što nam je potrebno.
YCbCr model je dosta učinkovitiji [1]. Model je osmišljen tako da Y prati svjetlinu slike, a Cb i Cr imaju informacije o bojama (konkretno, plavoj i crvenoj). Ako imamo prikaz u RGB-u, do YCbCr prikaza dođemo preko sljedeće formule

$$\begin{bmatrix} Y \\ C_b \\ C_r \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 65.481 & 128.553 & 24.966 \\ -37.797 & -74.203 & 112 \\ 112 & -93.786 & -18.214 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{4}$$



Slika 1: (slijeva nadesno) RGB, YCbCr, greyscale, binary

2.3 Reprezentacija podataka tenzorom

Neka su I_l, I_v i I_{pix} redom oznake broja slova, perspektiva i piksela. Tada je naš tenzor oblika $\mathcal{A} \in \mathbb{R}^{I_l \times I_v \times I_{pix}}$. Informacije o slovima i perspektivama faktoriziramo pomoću HOSVD-a i dobijemo

$$\mathcal{A} = \mathcal{S} \times_1 \mathbf{U}_l \times_2 \mathbf{U}_v \times_3 \mathbf{U}_{pix}, \tag{5}$$

gdje su $\mathbf{U}_l, \mathbf{U}_v$ i $\mathbf{U}_{pix}$ ortogonalne matrice reda I_l, I_v i I_{pix}, a $\mathcal{S} \in \mathbb{R}^{I_l \times I_v \times I_{pix}}$ jezgreni tenzor. Za sliku slova i u pogledu j ($i = 1, \ldots, I_l,\ j = 1, \ldots, I_v$) imamo

$$\mathcal{A}^{(i,j)} = \mathcal{S} \times_1 \mathbf{u}_i \times_2 \mathbf{u}_j \times_3 \mathbf{U}_{pix}$$

gdje su $\mathbf{u}_i$ i $\mathbf{u}_j$ redom retci matrica $\mathbf{U}_l$ i $\mathbf{U}_v$. Budući da smo fiksirali i i j, dobili smo nit $\mathcal{A}^{(i,j)} \in \mathbb{R}^{1 \times 1 \times I_{pix}}$. Iz rastava (5) i osnovnih svojstava množenja tenzora i matrice u modu 1 slijedi:

$$A = (\mathcal{S} \times_2 \mathbf{U}_v \times_3 \mathbf{U}_{pix}) \times_1 \mathbf{U}_l = \mathcal{B} \times_1 \mathbf{U}_l \tag{6}$$

2.4 Prepoznavanje znakovnog jezika tenzorom

Cilj je uklopiti neku testnu sliku u tenzor izgrađen na nekom konačnom skupu podataka, odnosno naći najbolju aproksimaciju među već dostupnim podacima. Testna slika $\mathcal{A}_{test} \in 1 \times 1 \times I_{pix}$ dana je nizom piksela i želimo da vrijedi rastav

$$\mathcal{A}_{test} = \mathcal{S} \times_1 \mathbf{u}_i \times_2 \mathbf{u}_j \times_3 \mathbf{U}_{pix}$$

za neke zasad nepoznate $\mathbf{u}_i$ i $\mathbf{u}_j$. Problem možemo preformulirati:

$$\underset{\mathbf{u}_i, \mathbf{u}_j}{\arg\min} \ \|\mathcal{A}_{test} - \mathcal{S} \times_1 \mathbf{u}_i \times_2 \mathbf{u}_j \times_3 \mathbf{U}_{pix}\|_2.$$

Prethodno zapisano bi za rezultat imalo aproksimaciju vektora $\mathbf{u}_i$ i $\mathbf{u}_j$. Budući da želimo aproksimirati $\mathbf{u}_i$, odnosno cilj nam je saznati "kojem slovu je bliska" naša testna slika, možemo računati tako da $\mathbf{u}_j$ budu generirani iz konačnog skupa. Konkretno, neka je $\mathbf{u}_j \in \{\mathbf{U}_v(k, 1 : I_v) : k = 1, \ldots, I_v\}$. Problem sada glasi:

$$\underset{\mathbf{u}_i}{\arg\min} \ \|\mathcal{A}_{test} - \mathcal{S} \times_1 \mathbf{u}_i \times_2 \mathbf{u}_j \times_3 \mathbf{U}_{pix}\|_2 \tag{7}$$

Korištenjem svojstava množenja tenzora i matrice u modu 1:

$$\underset{\mathbf{u}_i}{\arg\min} \ \|\mathcal{A}_{test} - \mathbf{u}_i \times (\mathcal{S} \times_2 \mathbf{u}_j \times_3 \mathbf{U}_{pix})_{(1)}\|_2 \tag{8}$$

pa imamo kandidata za aproksimaciju:

$$\mathbf{u}_i \ = \ \mathcal{A}_{test} \times (\mathcal{S} \times_2 \mathbf{u}_j \times_3 \mathbf{U}_{pix})^{+}_{(1)} \tag{9}$$

gdje s '+' označavamo Moore-Penroseov pseudoinverz. Ovim postupkom ćemo generirati I_v takvih aproksimacija $\{\mathbf{u}_i : i = 1, \ldots, I_v\}$ koje uspoređujemo s retcima matrice $\mathbf{U}_l$. Tražimo $k \in \{1, \ldots, I_l\}$ koji minimizira sljedeći izraz

$$\|\mathbf{u}_i - \mathbf{U}_l(k, 1 : I_l)\|_2.$$

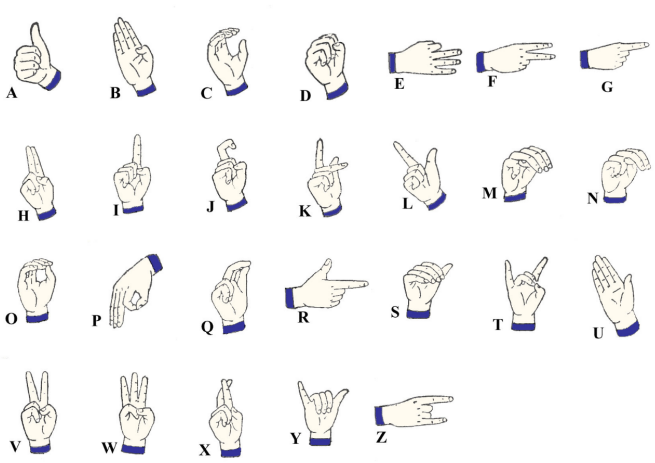
Pomoću kosinusa kuta 2 vektora $\mathbf{u}_i$ i $\mathbf{U}_l(k, 1 : I_l)$ možemo računati:

$$\arg\max_{i,k} \frac{\langle \mathbf{u}_i, \mathbf{U}_l(k, 1 : I_l) \rangle}{\|\mathbf{u}_i\| \|\mathbf{U}_l(k, 1 : I_l)\|} \tag{10}$$

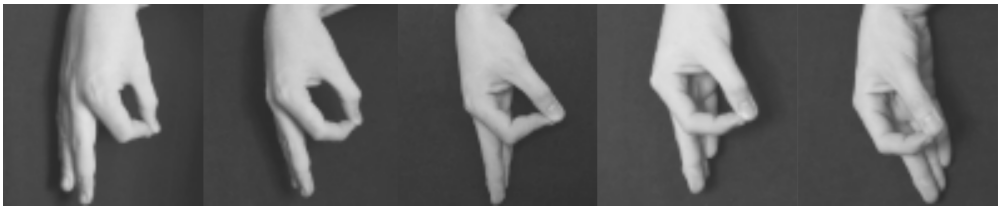
3 Eksperimentalni rezultati

3.1 Priprema podataka

Korištena je jednoručna abeceda engleskog znakovnog jezika (ASL, Slika 2). Fotografije su slikane iz 5 različitih kuteva (od 45° slijeva do 45° zdesna) u visini prsa, a pri tome fotoaparat nije bio pod nagibom u odnosu na pod. Od 26 znakova uspješno su obrađena 24 (nedostaju nam slova 'M' i 'N') te je u konačnici dobiveno 120 slika. Nakon rezanja viškova s fotografije, smanjena je rezolucija na 25×25 piksela. Za daljnju obradu korišten je Octave Forge 'Image' paket: RGB format slike konvertiran je u greyscale (*rgb2gray*) ili u YCbCr (*rgb2ycbcr*) te je potonje konvertirano u crno-bijele slike (*im2bw*).



Slika 2: Jednoručna abeceda ASL-a



Slika 3: Slovo P iz 5 različitih kuteva

3.2 Rezultati

Algoritam je testiran na način da se iz tenzora izolirala j-ta perspektiva ($\mathcal{A}(:, j, :)$) čija su slova ($\mathcal{A}(i, j, :)$) služila kao testni primjeri. Ostatak tenzora ($\mathcal{A}(:, k, :)$, $k \neq j$) je tretiran kao *training set*. Prethodni postupak je proveden za svaki $j = 1, \ldots, 5$. Za početak, program testiramo na 7 slova (A, L, H, P, R, U i Y) kojima se siluete međusobno značajno razlikuju. Rezultati su dani u tablicama 1 i 2. Na tom skupu podataka možemo zaključiti da siva skala slike daje nešto bolje rezultate. Dalje je eksperimentirano sa svim slovima (njih 24). Tada program uspješno identificira u prosjeku 13.4 slova od njih 24, što nije najoptimalniji rezultat. Ono što se svakako može u oba slučaja zaključiti je da Pogled 3 ima najveću uspješnost što i ima smisla jer su tada slova najviše diferencirana. Potencijalni razlog loših rezultata u drugom pokusu može biti obrada slika i skala koja se koristila. To je ključan korak koji doprinosi razlikama među slovima iz *training set*-a. Tako primjerice slova "B" i "U" izgledaju jako slično, kao i slova "C","D" i "O".

Perspektiva	Grey-scale	Binary
Kut 1	6	5
Kut 2	7	7
Kut 3	7	7
Kut 4	7	7
Kut 5	7	5
Prosjek	6.8	6.2

Tablica 1: Točnost po perspektivi (7 slova)

Perspektiva	Grey-scale	Binary
A, L, R, U	5	5
H	3	3
P	5	4
Y	5	4

Tablica 2: Broj pogodaka po slovu (7 slova)

Perspektiva	Grey-scale	Binary
Kut 1	10	11
Kut 2	14	13
Kut 3	17	16
Kut 4	13	11
Kut 5	13	14
Prosjek	13.4	13

Tablica 3: Točnost po perspektivi (sva slova)

Perspektiva	Grey-scale	Binary
A, J, P	5	5
R, W, Y	4	4
D, K, S, Z	3	3
C, E, O, T, U	2	2
B, Q	1	1
H	0	0
F, I	2	1
G	5	4
V	0	1
X	3	4

Tablica 4: Broj pogodaka po slovu (sva slova)

Literatura

[1] Su-Jing Wang, De-Cai Zhang, Cheng-Cheng Jia, Chung-Guang Zhou, Li-Biao Zhang, *A Sign Language Recognition Based on Tensor, 2010.*

[2] Zlatko Drmač, *Uvod u složeno pretraživanje podataka, predavanja 2020-2021*

[3] Dokumentacija Octave-forge 'Image' paketa, *octave.sourceforge.io/image/overview.html*