

# TENSOR IMAGE CLUSTERING

Karlo Grozdanić, Mislav Jelašić, Luka Karlič

14. veljače 2022.

## 1 Uvod

Količina elektroničkih podataka dostupnih za obradu raste velikom brzinom. Za cjelokupnu i uspješnu analizu potrebno je stoga pronaći efikasan način njihovog pohranjivanja. Digitalne slike prirodno je spremati u obliku matrice (odnosno tenzora drugog reda), gdje je svaki piksel određen odgovarajućim retkom i stupcem. Klasteriranje i kategorizacija predstavlja jedan dobar način interpretacije sadržaja na visokoj razini. Osnovna zamisao je grupirati mnoštvo slika u različite klase baze podataka prema zajedničkim vizualnim obilježjima. Takav koncept rezultira sažetim prikazom sadržaja, što omogućuje daljnje korištenje u raznim zadacima.

U ovom radu prezentiramo tenzorski pristup u analizi slika zasnovan na algebri tenzora i geometriji mnogostrukosti. Aktualni algoritmi učenja mogu se jedino primijeniti na vektorskim podacima. Budući da želimo očuvati lokalnu strukturu podataka, uvodimo novi način za reprezentaciju i klasteriranje slika u nižedimenzionalnom potprostoru tenzora drugog reda (matrica). Problem modeliramo pomoću grafa s ciljem pronalaženja optimalne particije minimizacijom zadane funkcije cilja. Iako naš algoritam tretira slike kao matrice, može se proširiti i na tenzore viših dimenzija.

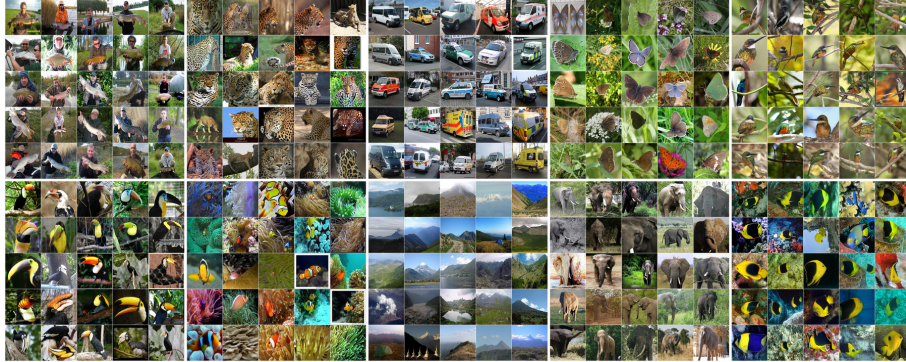


Figure 1: Grupiranje slika prema sadržaju

## 2 Algebra tenzora

Tensor reda  $k$  je realni multilinear funkcional nad  $k$  vektorskih prostora:

$$T : \mathbb{R}^{n_1} \times \dots \times \mathbb{R}^{n_k} \rightarrow \mathbb{R}$$

Broj  $k$  zove se red tenzora  $T$ . Multilinear funkcija linearna je kao funkcija svake varijable. Skup svih  $k$ -tenzora na  $\mathbb{R}^{n_i}$ ,  $i = 1, \dots, k$ , u oznaci  $\mathcal{T}^k$ , vektorski je prostor sa standardnim operacijama zbrajanja po točkama i skalarnog množenja:

$$\begin{aligned} (aT)(a_1, \dots, a_k) &= a(T(a_1, \dots, a_k)), \\ (T + T')(a_1, \dots, a_k) &= T(a_1, \dots, a_k) + T'(a_1, \dots, a_k), \end{aligned}$$

gdje su  $a_i \in \mathbb{R}^{n_i}$ . Za dane tenzore  $S \in \mathcal{T}^k$  i  $T \in \mathcal{T}^l$  definiran je njihov tenzorski produkt:

$$S \otimes T : \mathbb{R}^{n_1} \times \dots \times \mathbb{R}^{n_{k+l}} \rightarrow \mathbb{R},$$

$$S \otimes T(a_1, \dots, a_{k+l}) = S(a_1, \dots, a_k)T(a_{k+1}, \dots, a_{k+l}).$$

Iz multilinearnosti  $S$  i  $T$ , odmah je jasno da  $S \otimes T$  linearno ovisi o svakom argumentu pa je to tenzor reda  $k + l$ . Tenzori prvog reda nisu ništa drugo nego dualni vektori na  $\mathbb{R}^{n_1}$ . Drugim riječima,  $\mathcal{T}^1 = \mathcal{R}^{n_1}$ , gdje je  $\mathcal{R}^{n_1}$  dualni prostor od  $\mathbb{R}^{n_1}$ . Prostor tenzora drugog reda je produkt dva prostora tenzora prvog reda, tj.  $\mathcal{T}^2 = \mathcal{R}^{n_1} \otimes \mathcal{R}^{n_2}$ . Neka  $e_1, \dots, e_{n_1}$  čine standardnu kanonsku bazu od  $\mathbb{R}^{n_1}$ , a  $\varepsilon_1, \dots, \varepsilon_{n_1}$  bazu  $\mathcal{R}^{n_1}$ , dualnu standardnoj kanonskoj bazi  $\mathbb{R}^{n_1}$ . Analogno, neka su  $\tilde{e}_1, \dots, \tilde{e}_{n_2}$  i  $\tilde{\varepsilon}_1, \dots, \tilde{\varepsilon}_{n_2}$  baze prostora  $\mathbb{R}^{n_2}$ , odnosno  $\mathcal{R}^{n_2}$ . Dakle, imamo:

$$\begin{aligned}\varepsilon_i(e_j) &= \delta_{ij}, \\ \tilde{\varepsilon}_i(\tilde{e}_j) &= \delta_{ij},\end{aligned}$$

gdje je  $\delta_{ij}$  Kroneckerova delta funkcija. Sada lako vidimo da je  $\{\varepsilon_i \otimes \tilde{\varepsilon}_j, 1 \leq i \leq n_1, 1 \leq j \leq n_2\}$  baza za prostor  $\mathcal{R}^{n_1} \otimes \mathcal{R}^{n_2}$  pa proizvoljan 2-tenzor  $T$  možemo zapisati kao:

$$T = \sum_{\substack{1 \leq i \leq n_1 \\ 1 \leq j \leq n_2}} T_{ij} \varepsilon_i \otimes \tilde{\varepsilon}_j.$$

To pokazuje da svaki tenzor reda 2 iz  $\mathcal{R}^{n_1} \otimes \mathcal{R}^{n_2}$  odgovara točno jednoj  $n_1 \times n_2$  matrici. Za dva vektora  $\mathbf{a} = \sum_{k=1}^{n_1} a_k \mathbf{e}_k \in \mathbb{R}^{n_1}$  i  $\mathbf{b} = \sum_{l=1}^{n_2} b_l \tilde{\mathbf{e}}_l \in \mathbb{R}^{n_2}$  dobivamo

$$\begin{aligned}T(\mathbf{a}, \mathbf{b}) &= \sum_{i,j} T_{ij} \varepsilon_i \otimes \tilde{\varepsilon}_j \left( \sum_{k=1}^{n_1} a_k \mathbf{e}_k, \sum_{l=1}^{n_2} b_l \tilde{\mathbf{e}}_l \right) \\ &= \sum_{i,j} T_{ij} \varepsilon_i \left( \sum_{k=1}^{n_1} a_k \mathbf{e}_k \right) \tilde{\varepsilon}_j \left( \sum_{l=1}^{n_2} b_l \tilde{\mathbf{e}}_l \right) \\ &= \sum_{i,j} T_{i,j} a_i b_j \\ &= \mathbf{a}^T T \mathbf{b}\end{aligned}$$

Želimo transformirati standardan koordinatni sustav u prostor tenzora. Neka su  $\{\mathbf{u}_k, 1 \leq k \leq n_1\}$  i  $\{\mathbf{v}_l, 1 \leq l \leq n_2\}$  nove ortonormirane baze prostora  $\mathcal{R}^{n_1}$ , odnosno  $\mathcal{R}^{n_2}$ . Preciznije, imamo:

$$\begin{aligned}\mathbf{u}_k &= (u_{k1}, \dots, u_{kn_1})^T = \sum_{i=1}^{n_1} u_{ki} \varepsilon_i \\ \mathbf{v}_l &= (v_{l1}, \dots, v_{ln_2})^T = \sum_{j=1}^{n_2} v_{lj} \tilde{\varepsilon}_j\end{aligned}$$

pa se lako pokaže da vrijedi:

$$\begin{aligned}\varepsilon_i &= \sum_{k=1}^{n_1} u_{ki} \mathbf{u}_k \\ \tilde{\varepsilon}_j &= \sum_{l=1}^{n_2} v_{lj} \mathbf{v}_l\end{aligned}$$

Koristeći gornje tvrdnje, vidimo da se tenzor reda 2 može prikazati u novoj bazi na sljedeći način:

$$\begin{aligned}T &= \sum_{i,j} T_{ij} \varepsilon_i \otimes \tilde{\varepsilon}_j \\ &= \sum_{i,j} T_{ij} \left( \sum_{k=1}^{n_1} u_{ki} \mathbf{u}_k \right) \otimes \left( \sum_{l=1}^{n_2} v_{lj} \mathbf{v}_l \right) \\ &= \sum_{k,l} \left( \sum_{i,j} T_{ij} u_{ki} v_{lj} \right) \mathbf{u}_k \otimes \mathbf{v}_l \\ &= \sum_{k,l} (\mathbf{u}_k^T T \mathbf{v}_l) \mathbf{u}_k \otimes \mathbf{v}_l\end{aligned}$$

Gornja jednakost pokazuje da je  $\{\mathbf{u}_i \otimes \mathbf{v}_j, 1 \leq i \leq n_1, 1 \leq j \leq n_2\}$  nova baza za tenzorski produkt  $\mathcal{R}^{n_1} \otimes \mathcal{R}^{n_2}$ . Time tenzor  $T$  postaje  $U^T T V$  u transformiranoj bazi, gdje su  $U = (\mathbf{u}_1, \dots, \mathbf{u}_{n_1})$  i  $V = (\mathbf{v}_1, \dots, \mathbf{v}_{n_2})$ . Posebno, projekciju tenzora  $T$  na bazu  $\mathbf{u}_i \otimes \mathbf{v}_j$  možemo računati skalarnim produktom:

$$\langle T, \mathbf{u}_i \otimes \mathbf{v}_j \rangle = \langle T, \mathbf{u}_i \mathbf{v}_j^T \rangle = \mathbf{u}_i^T T \mathbf{v}_j.$$

Zaključujemo da se svaka slika može prikazati kao linearna kombinacija baznih tenzora  $\mathbf{u}_i \otimes \mathbf{v}_j$ , odnosno  $\mathbf{u}_i \mathbf{v}_j^T$  u matričnoj formi.

### 3 Klasteriranje slika pomoću tenzorske reprezentacije

#### 3.1 Određivanje funkcije cilja

Razmatramo problem spektralnog klasteriranja pomoću tenzorske reprezentacije, kojeg shvaćamo kao kombinaciju reduciranja dimenzije prostora i tradicionalnog algoritma klasteriranja podataka. Pretpostavljamo da su slike uzete iz neke nižedimenzionalne mnogostrukosti. S ciljem procjene geometrijskih i topoloških svojstava, tražimo odgovarajući linearni aproksimacijski potprostor za tenzore drugog reda.

Neka je dano  $m$  točaka  $\mathcal{X} = \{X_1, \dots, X_m\}$  iz potprostora tenzora drugog reda  $\mathcal{M} \in \mathcal{R}^{n_1} \otimes \mathcal{R}^{n_2}$ , koje predstavljaju učitane slikovne podatke. Formiramo graf susjedstva  $\mathcal{G}$  s ciljem modeliranja lokalne geometrijske strukture od  $\mathcal{M}$ . Označimo sa  $S$  pripadnu matricu težina, čiji su elementi:

$$S_{ij} = \begin{cases} 1, & \text{ako se } X_i \text{ nalazi među } p \text{ najbližih susjeda od } X_j \\ & \text{ili se } X_j \text{ nalazi među } p \text{ najbližih susjeda od } X_i, \\ 0, & \text{inače.} \end{cases}$$

Funkcija cilja određena je minimizacijom rezova particije grafa. Želimo optimalno podijeliti graf  $\mathcal{G}$  na dva disjunktna podgrafa  $\mathcal{G}_1, \mathcal{G}_2 \subseteq \mathcal{G}$ , tako da je  $\mathcal{G} = \mathcal{G}_1 \cup \mathcal{G}_2$  i  $\mathcal{G}_1 \cap \mathcal{G}_2 = \emptyset$ . Taj uvjet jednostavnije zapisujemo matematičkim izrazom:

$$\min_{\mathcal{G}_1, \mathcal{G}_2} \sum_{i \in \mathcal{G}_1} \sum_{j \in \mathcal{G}_2} S_{ij}.$$

Sada, svakoj točki  $X_i$  pridružujemo oznaku  $y_i \in \{-1, 1\}$  koja pokazuje pripadnost podgrafu  $\mathcal{G}_1$ , odnosno  $\mathcal{G}_2$ . Ako dvije točke  $X_i$  i  $X_j$  pripadaju istom podgrafu vrijedi  $y_i - y_j = 0$ , dok je u suprotnom  $|y_i - y_j| = 2$ . Gornji problem svodi se na traženje vektora  $\mathbf{y} = (y_1, \dots, y_m)$  koji minimizira sljedeću funkciju cilja:

$$\min_{\mathbf{y} \in \{-1, 1\}^m} \sum_{i, j} (y_i - y_j)^2 S_{ij}.$$

Koristimo spektralne metode aproksimacije optimalnog rješenja. Dopuštamo da je  $y_i$  bilo koji realni broj i da je pripadno preslikavanje s  $X_i$  u  $y_i$  linearno, tj. oblika  $y_i = \mathbf{u}^T X_i \mathbf{v}$ . Time konačno dolazimo do odgovarajuće funkcije cilja:

$$\min_{U, V} \sum_{i, j} \|U^T X_i V - U^T X_j V\|^2 S_{ij},$$

gdje  $U$  i  $V$  predstavljaju matrice transformacije. Minimiziranjem gornjeg izraza jamčimo da su za „bliske”  $X_i$  i  $X_j$ , tenzori  $U^T X_i V$  i  $U^T X_j V$  također „bliski”.

#### 3.2 Formiranje algoritma

Raspisujemo detaljnije našu funkciju cilja. Stavimo  $Y_i = U^T X_i V$  i uvedimo dijagonalnu matricu  $D$  s elementima  $D_{ii} = \sum_j S_{ij}$ . Budući da je  $\|A\|^2 = \text{tr}(AA^T)$  imamo:

$$\begin{aligned} & \frac{1}{2} \sum_{i, j} \|U^T X_i V - U^T X_j V\|^2 S_{ij} \\ &= \frac{1}{2} \sum_{i, j} \text{tr} \left( (Y_i - Y_j) (Y_i - Y_j)^T \right) S_{ij} \\ &= \frac{1}{2} \sum_{i, j} \text{tr} (Y_i Y_i^T + Y_j Y_j^T - Y_i Y_j^T - Y_j Y_i^T) S_{ij} \\ &= \text{tr} \left( \sum_i D_{ii} Y_i Y_i^T - \sum_{i, j} S_{ij} Y_i Y_j^T \right) \\ &= \text{tr} \left( \sum_i D_{ii} U^T X_i V V^T X_i^T U - \sum_{i, j} S_{ij} U^T X_i V V^T X_j^T U \right) \\ &= \text{tr} \left( U^T \left( \sum_i D_{ii} X_i V V^T X_i^T - \sum_{i, j} S_{ij} X_i V V^T X_j^T \right) U \right) \\ &= \text{tr} (U^T (D_V - S_V) U), \end{aligned}$$

gdje su  $D_V = \sum_i D_{ii} X_i V V^T X_i^T$  i  $S_V = \sum_{ij} S_{ij} X_i V V^T X_j^T$  novo označene matrice. Slično, iz  $\|A\|^2 = \text{tr}(A^T A)$  slijedi:

$$\begin{aligned}
& \frac{1}{2} \sum_{ij} \|U^T X_i V - U^T X_j V\|^2 S_{ij} \\
&= \frac{1}{2} \sum_{ij} \text{tr} \left( (Y_i - Y_j)^T (Y_i - Y_j) \right) S_{ij} \\
&= \frac{1}{2} \sum_{ij} \text{tr} (Y_i^T Y_i + Y_j^T Y_j - Y_i^T Y_j - Y_j^T Y_i) S_{ij} \\
&= \text{tr} \left( \sum_i D_{ii} Y_i^T Y_i - \sum_{ij} S_{ij} Y_i^T Y_j \right) \\
&= \text{tr} \left( \sum_i D_{ii} V^T X_i^T U U^T X_i V - \sum_{ij} S_{ij} V^T X_i^T U U^T X_j V \right) \\
&= \text{tr} \left( V^T \left( \sum_i D_{ii} X_i^T U U^T X_i - \sum_{ij} S_{ij} X_i^T U U^T X_j \right) V \right) \\
&= \text{tr} (V^T (D_U - S_U) V),
\end{aligned}$$

za matrice  $D_U = \sum_i D_{ii} X_i U U^T X_i^T$  i  $S_U = \sum_{ij} S_{ij} X_i U U^T X_j^T$ . Gornjim postupkom ustvrđujemo da treba istovremeno minimizirati izraze  $\text{tr}(U^T (D_V - S_V) U)$  i  $\text{tr}(V^T (D_U - S_U) V)$ . Dodatno, želimo maksimizirati globalnu varijancu mnogostrukosti prostora. Vjerojatnosnu mjeru određujemo preko dijagonalne matrice  $D$  i pretpostavljamo da podaci (točke) imaju srednju vrijednost nula. Na taj način, „težinsku” varijancu ocjenjujemo formulom:

$$\begin{aligned}
\text{var}(Y) &= \sum_i \|Y_i\|^2 D_{ii} \\
&= \sum_i \text{tr} (Y_i^T Y_i) D_{ii} \\
&= \sum_i \text{tr} (V^T X_i^T U U^T X_i V) D_{ii} \\
&= \text{tr} \left( V^T \left( \sum_i D_{ii} X_i^T U U^T X_i \right) V \right) \\
&= \text{tr} (V^T D_U V),
\end{aligned}$$

odnosno

$$\begin{aligned}
\text{var}(Y) &= \sum_i \text{tr} (Y_i Y_i^T) D_{ii} \\
&= \sum_i \text{tr} (U^T X_i^T V V^T X_i U) D_{ii} \\
&= \text{tr} \left( U^T \left( \sum_i D_{ii} X_i V V^T X_i^T \right) U \right) \\
&= \text{tr} (U^T D_V U),
\end{aligned}$$

gdje je sa  $Y = U^T X V$  označena nasumična varijabla tenzorskog prostora. Napokon, koristeći prethodne tvrdnje, dolazimo do sljedećeg glavnog optimizacijskog problema:

$$\begin{cases} \min_{U,V} \frac{\text{tr}(U^T (D_V - S_V) U)}{\text{tr}(U^T D_V U)} \\ \min_{U,V} \frac{\text{tr}(V^T (D_U - S_U) V)}{\text{tr}(V^T D_U V)} \end{cases}$$

Lako se vidi da rješenje treba tražiti generaliziranom metodom svojstvenih vektora  $(D_V - S_V, D_V)$ , odnosno  $(D_U - S_U, D_U)$ . Budući da matrice  $D_V$ ,  $S_V$ ,  $D_U$ ,  $S_U$  nisu konstantne, gornja dva izraza međusobno ovise jedno o drugome te se ne mogu riješiti nezavisno. Zato koristimo jednostavnu, ali efektivnu iterativnu metodu računanja. Prvo fiksiramo  $U$  i rješavamo generalizirani problem svojstvenih vrijednosti u varijabli  $V$ :

$$(D_U - S_U) \mathbf{v} = \lambda D_U \mathbf{v}.$$

Kada odredimo  $V$ ,  $U$  onda nalazimo rješavanjem drugog generaliziranog svojstvenog problema:

$$(D_V - S_V)\mathbf{u} = \lambda D_V \mathbf{u}.$$

U na početku inicijaliziramo kao jediničnu matricu. Optimalni  $U$  i  $V$  dobivamo konačnim brojem iteracija prethodna dva koraka.

## 4 Implementacija i testiranje

### 4.1 Odabir podataka

#### 4.1.1 Znamenke (MNIST) dataset

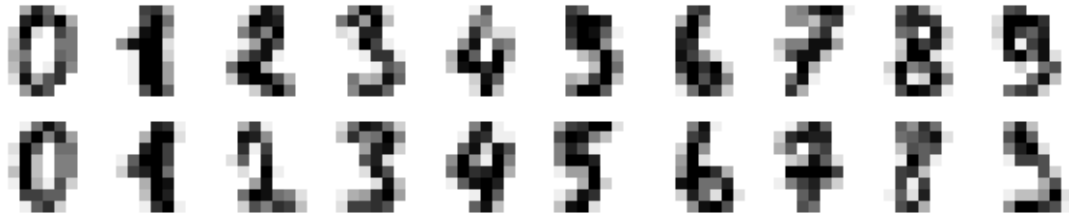


Figure 2: Slike rukom pisanih znamenaka

MNIST dataset nastao je 1998. kao kombinacija više skupova podataka, a sastoji se od slika rukom pisanih znamenki koje su pisali studenti te zaposlenici US Census Bureau-a. Sadrži 60000 slika za učenje i 10000 slika za testiranje. Jedan je od najpoznatijih skupova podataka u području znanosti o podacima.

#### 4.1.2 Lica



Figure 3: Originalne slike iz skupa podataka

Za lica osoba poslužili smo se javnim skupom podataka pod nazivom 'Yale face database'. Slike su preprocesuirane tako da im je dimenzija 32x32 piksela, poravnate su na način da su oči u istoj ravni, intenziteti piksela su skalirani na interval od 0 do 1. Podaci predstavljaju varijabilan broj pojedinaca, pri čemu za svaku osobu imamo 50 slika u različitim uvjetima (osvjetljenje, izraz lica i sl.). Broj klastera nam predstavlja koliko različitih pojedinaca smo uzeli iz skupa podataka.



Figure 4: Preprocesuirana lica iste osobe u različitim uvjetima (osvjetljenje, izraz lica..)

## 4.2 Metode

Uspoređujemo različite metode klasteriranja. Dok naš algoritam tretira slike kao matrice, odnosno tenzore drugog reda, ostali algoritmi spremaju sliku u obliku vektora. Takva reprezentacija nije lokalno optimalna jer ne poštuje odnose između piksela u prostoru. Implementirani ImageTensor pristup dijeli mnoga svojstva sa ostalim tradicionalnim algoritmima, ali je računski učinkovitiji. Nažalost, imamo ograničenost u traženju optimalnog rješenja samo na prostoru linearnih funkcija. Imamo sljedeće implementirane metode:

- k-means
  - klasterira podatke tako da iterativno mijenja klaster te njihove centroe
- PCA (Principal Component Analysis)
  - traži projekciju podataka na prostor u kojem se maksimizira varijanca
- LGE (Locality Graph Embedding)
  - traži projekciju podataka na prostor u kojemu se čuva lokalna povezanost
- Normalized cuts
  - spektralno klasteriranja preko normaliziranih rezova grafa
- TensorImage
  - traži projekciju prostora u kojoj su podaci prirodno razdvojeni

## 4.3 Prikaz rezultata

### 4.3.1 Znamenke

#### 1. PCA

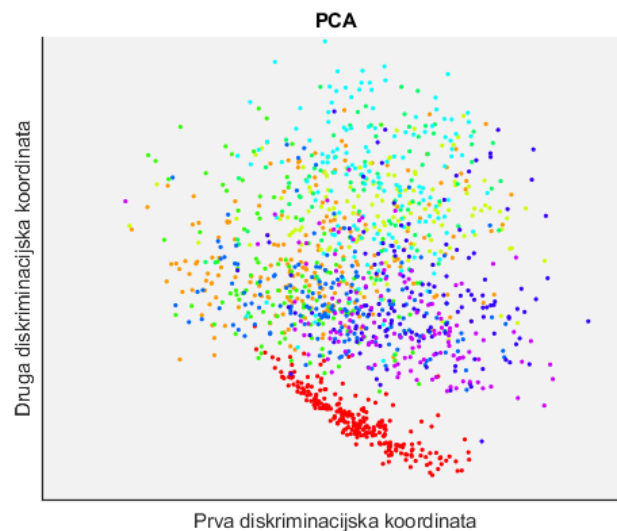


Figure 5: Prikaz PCA metode

PCA+k-means (prvih 10 disk. koord) daje točnost: **60.28%**

## 2. TensorImage

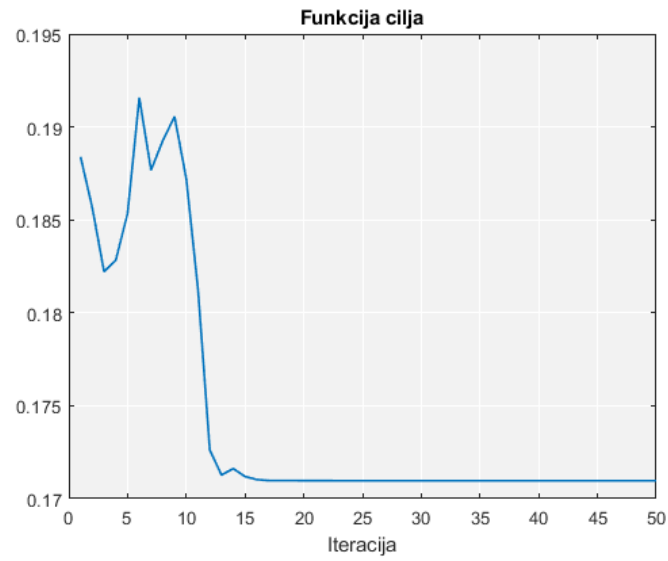


Figure 6: Pad funkcije cilja ovisno o iteracijama

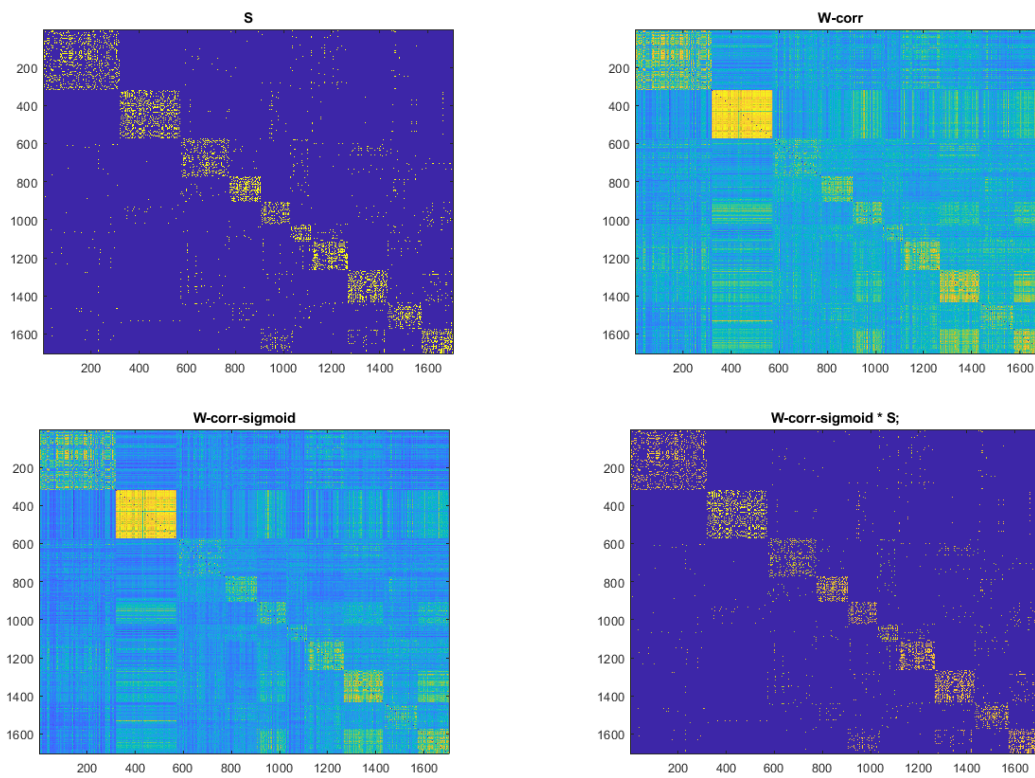


Figure 7: Razne matrice susjedstva

Table 1: Točnost TensorImage metode klasteriranja ovisno o odabranoj matrici susjedstva

Korištena matrica	Točnost (%)
$W_{corr}$	60.4
$\text{sigmoid}(W_{corr})$	<b>62.57</b>
$S$	55.01
$\text{sigmoid}(W_{corr}) \circ S$	51.56

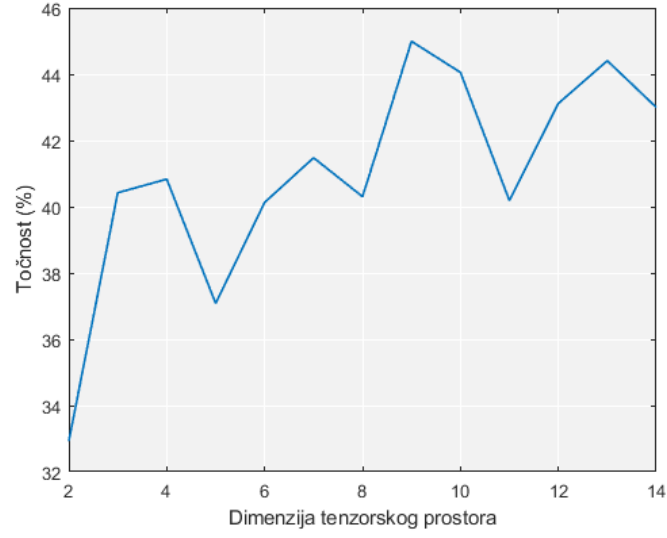


Figure 8: Točnost ovisno o odabranoj dimenziji prostora tenzora

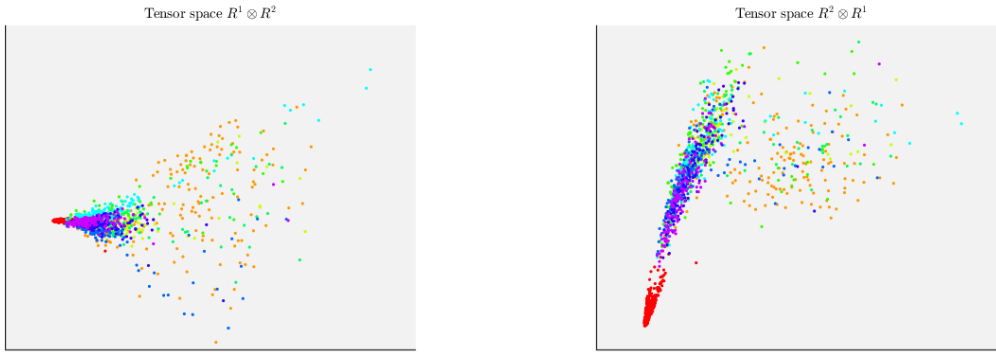


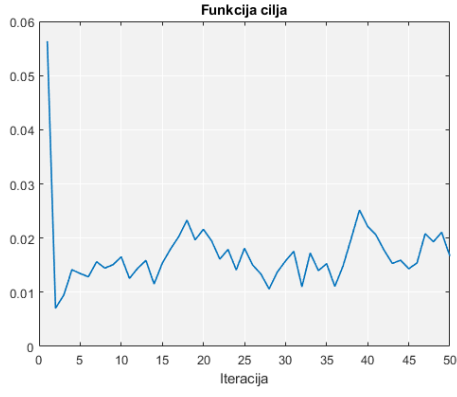
Figure 9: Projekcija podataka

#### 4.3.2 Lica

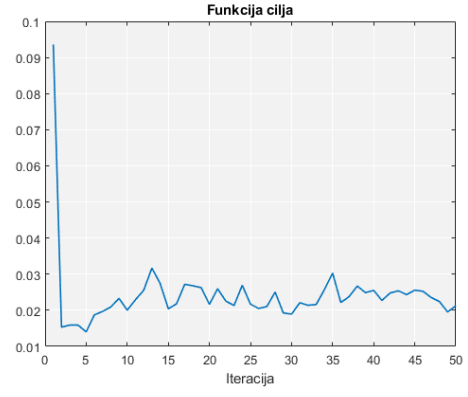
Table 2: Točnost TensorImage metode naspram NCut metodi u ovisnosti o parametrima matrice susjedstva

	max % TensorImage	max % Ncuts
$knn=20, cosine, k=5, d=2$	55%	77.6%
$knn=20, cosine, k=8, d=2$	50%	72.8%
$knn=20, euclid, k=5, d=2$	56%	81.25%
$knn=20, euclid, k=8, d=2$	50%	80.25%
$knn=20, euclid, k=8, d=5$	60%	75%
$knn=20, euclid, k=8, d=30$	55%	72%

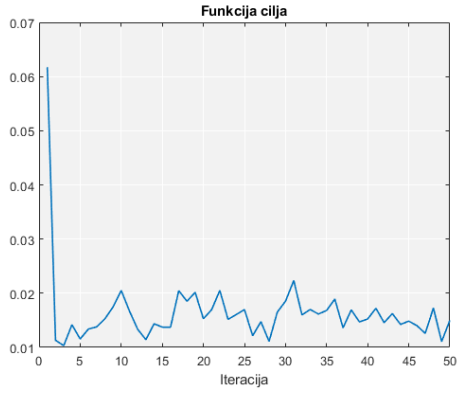




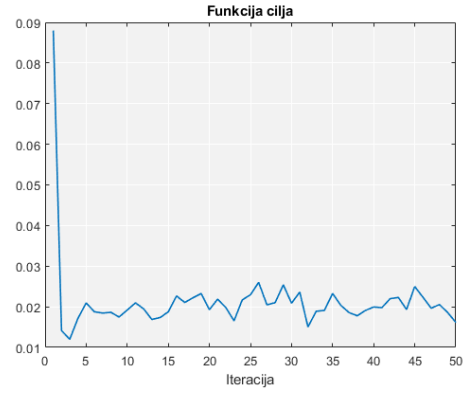
(a)  $knn = 20$ ,  $k = 5$ , metrika: cosine,  $d = 2$



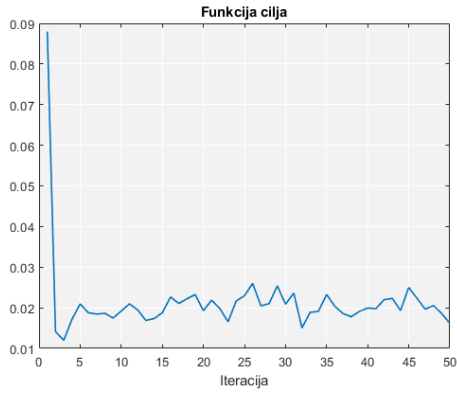
(b)  $knn = 20$ ,  $k = 8$ , metrika: cosine,  $d = 2$



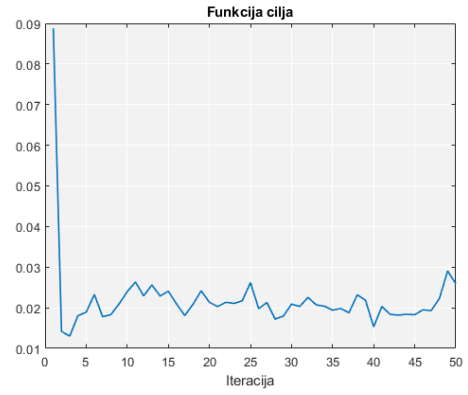
(c)  $knn = 20$ ,  $k = 5$ , metrika: euclid,  $d = 2$



(d)  $knn = 20$ ,  $k = 5$ , metrika: euclid,  $d = 5$

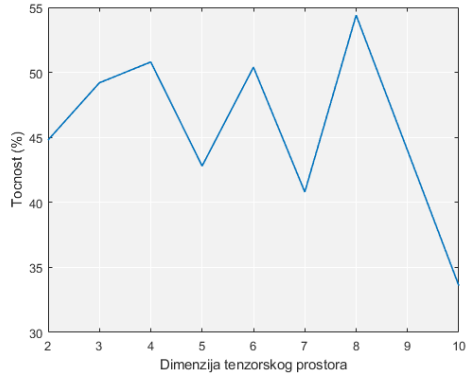


(e)  $knn = 20$ ,  $k = 8$ , metrika: euclid,  $d = 2$

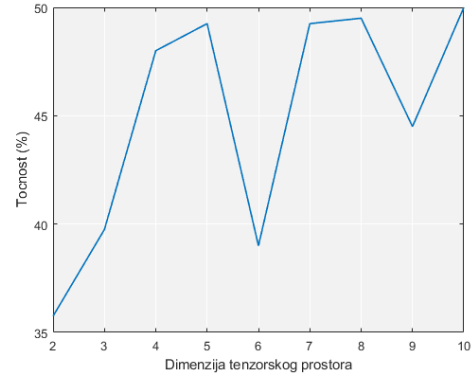


(f)  $knn = 20$ ,  $k = 8$ , metrika: euclid,  $d = 30$

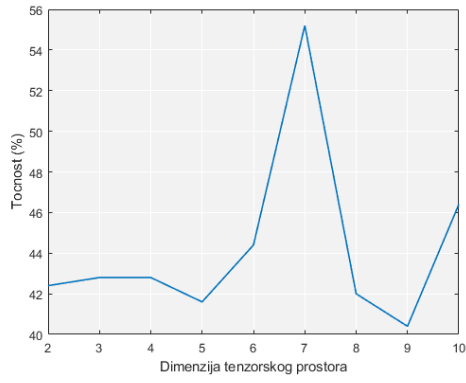
Figure 10: Funkcija cilja pri razlicitim parametrima konstrukcije matrice  $W$



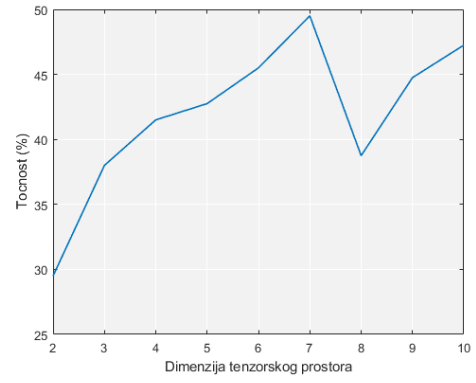
(a)  $knn = 20$ ,  $k = 5$ , metrika: cosine,  $d = 2$



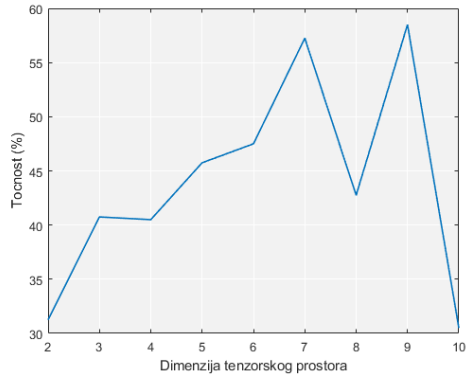
(b)  $knn = 20$ ,  $k = 8$ , metrika: cosine,  $d = 2$



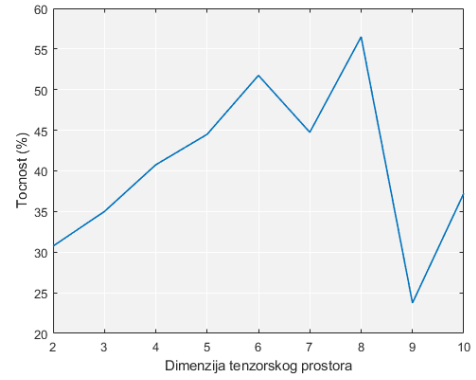
(c)  $knn = 20$ ,  $k = 5$ , metrika: euclid,  $d = 2$



(d)  $knn = 20$ ,  $k = 8$ , metrika: euclid,  $d = 2$

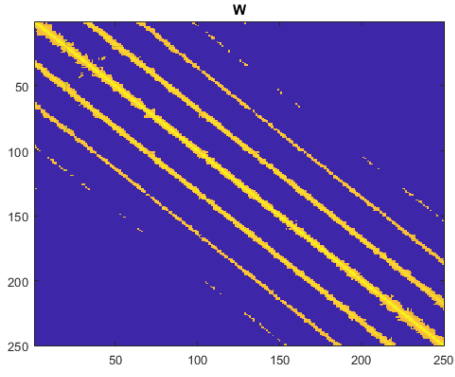


(e)  $knn = 20$ ,  $k = 8$ , metrika: euclid,  $d = 5$

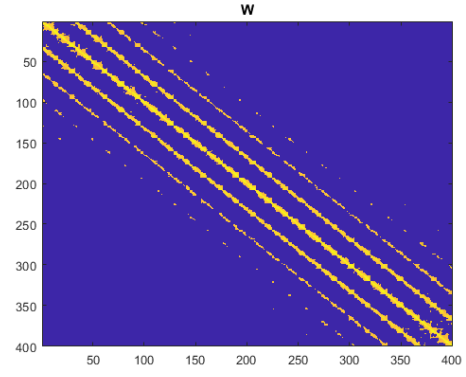


(f)  $knn = 20$ ,  $k = 8$ , metrika: euclid,  $d = 30$

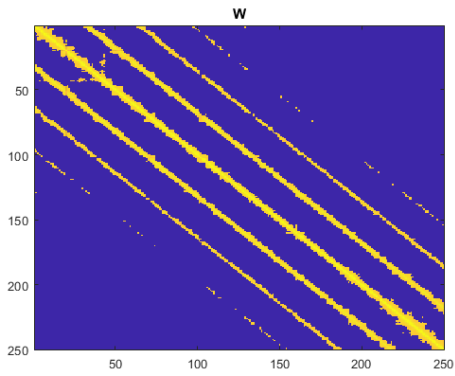
Figure 11: Postotak točnosti pogodjenih lica u ovisnosti o dimenziji tenzorskog prostora



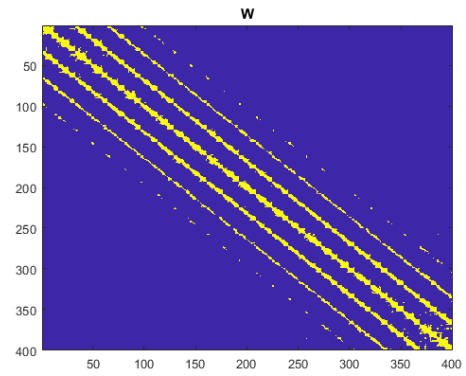
(a)  $knn = 20$ ,  $k = 5$ , metrika: cosine,  $d = 2$



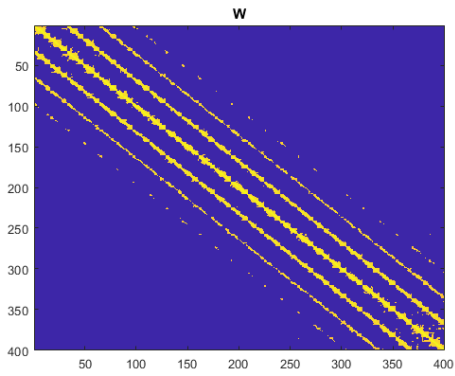
(b)  $knn = 20$ ,  $k = 8$ , metrika: cosine,  $d = 2$



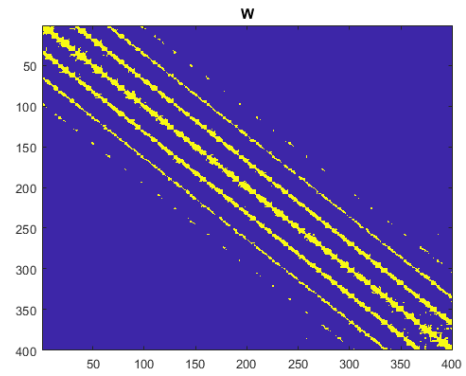
(c)  $knn = 20$ ,  $k = 5$ , metrika: euclid,  $d = 2$



(d)  $knn = 20$ ,  $k = 8$ , metrika: euclid,  $d = 5$



(e)  $knn = 20$ ,  $k = 8$ , metrika: euclid,  $d = 2$



(f)  $knn = 20$ ,  $k = 8$ , metrika: euclid,  $d = 30$

Figure 12: Graficki prikaz matrice  $W$  u ovisnosti o parametrima

## 5 Reference

1. S. X. Yu i J. Shi, *Multiclass Spectral Clustering*
2. X. Li, Z. Zhang, Y. Wang, W. Hu, *Multiclass Spectral Clustering Based on Discriminant Analysis*
3. J. Shi, J. Malik, *Normalized Cuts and Image Segmentation*
4. @INPROCEEDINGSCHHH07, AUTHOR = Deng Cai and Xiaofei He and Yuxiao Hu and Jiawei Han and Thomas Huang, TITLE = Learning a Spatially Smooth Subspace for Face Recognition, BOOKTITLE = Proc. IEEE Conf. Computer Vision and Pattern Recognition Machine Learning (CVPR'07), YEAR = 2007,
5. @INPROCEEDINGSCHH07b, AUTHOR = Deng Cai and Xiaofei He and Jiawei Han, TITLE = Spectral Regression for Efficient Regularized Subspace Learning, BOOKTITLE = Proc. Int. Conf. Computer Vision (ICCV'07), YEAR = 2007,
6. @ARTICLECHHZ06, AUTHOR = Deng Cai and Xiaofei He and Jiawei Han and Hong-Jiang Zhang, TITLE = Orthogonal Laplacianfaces for Face Recognition, JOURNAL = IEEE Transactions on Image Processing, YEAR = 2006, volume = 15, number = 11, pages = 3608-3614,
7. @ARTICLEHYHNZ05, AUTHOR = Xiaofei He and Shuicheng Yan and Yuxiao Hu and Partha Niyogi and Hong-Jiang Zhang, TITLE = Face Recognition Using Laplacianfaces, JOURNAL = IEEE Trans. Pattern Anal. Mach. Intelligence, YEAR = 2005, volume = 27, number = 3, pages = 328-340,
8. @ArticleGeBeKr01, author = "Georgiades, A.S. and Belhumeur, P.N. and Kriegman, D.J.", title = "From Few to Many: Illumination Cone Models for Face Recognition under Variable Lighting and Pose", journal = "IEEE Trans. Pattern Anal. Mach. Intelligence", year = 2001, volume = 23, number = 6, pages= "643-660"
9. @ArticleKCLee05, author = "K.C. Lee and J. Ho and D. Kriegman", title = "Acquiring Linear Subspaces for Face Recognition under Variable Lighting ", journal = "IEEE Trans. Pattern Anal. Mach. Intelligence", year = 2005, volume = 27, number = 5, pages= "684-698"
10. Deng Cai, Xiaofei He and Jiawei Han, "Document Clustering Using Locality Preserving Indexing" IEEE TKDE, Dec. 2005. Written by Deng Cai (dengcai2 AT cs.uiuc.edu), April/2004, Feb/2006, May/2007