PREPOZNAVANJE LICA

KORIŠTENJEM T-SVD METODE

Travanj 2020.

Bratulić Petar
Stanišić Matea

Uvod i notacija

Tenzori višeg reda

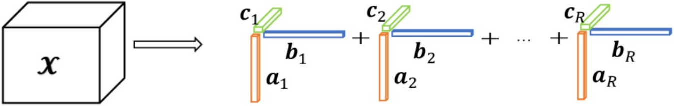
- Tenzor je višedimenzionalno polje. U mnogo slučajeva, primjerenije je pohraniti podatke u višedimenzionalnim poljima višeg reda, nego u ekvivalentnoj matričnoj formi.
- Tenzori višeg reda pojavljuju se u primjeni mnogih znanstvenih disciplina poput psihometrije ili obradi signala i slika.
- Rezultat česte primjene su razni algoritmi i dekompozicije koji su razvijeni u tu svrhu poput Higher-Order SVD-a (HOSVD), TUCKER-a ili CANDECOMP/PARAFAC-a (CP).

Dekompozicija tenzora I

- CP dekompozicija

$$\mathcal{A} = \sum_{l=1}^{R} u^{(l)} \circ v^{(l)} \circ w^{(l)} \Longrightarrow (\mathcal{A})_{i,j,k} = \sum_{l=1}^{R} u_i^{(l)} v_j^{(l)} w_k^{(l)},$$

gdje je $\mathcal{A}$ tenzor dimenzija $I \times J \times K$, $u^{(l)} \in \mathbb{R}^I$, $v^{(l)} \in \mathbb{R}^J$, $w^{(l)} \in \mathbb{R}^K$, za $l = 1, 2, \ldots, R$



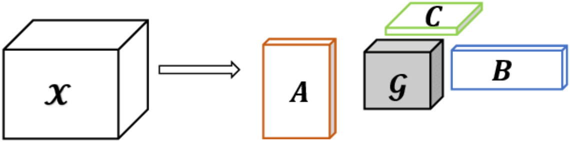
Slika 1: CP dekompozicija

Dekompozicija tenzora II

- Tucker dekompozicija

$$\mathcal{A} = \mathcal{G} \times_1 \mathbf{U} \times_2 \mathbf{V} \times_3 \mathbf{W} = \sum_{i=1}^{R_1} \sum_{j=1}^{R_2} \sum_{k=1}^{R_3} \sigma_{ijk} (u^{(i)} \circ v^{(j)} \circ w^{(k)})$$

gdje je $R_1 \leq I, R_2 \leq J, R_3 \leq K$ i za $i, j, k, u^{(i)} \in \mathbb{R}^I, v^{(j)} \in \mathbb{R}^J, w^{(k)} \in \mathbb{R}^K, \mathcal{G} = \sigma_{ijk} \in \mathbb{R}^{R_1 \times R_2 \times R_3}, \mathbf{U} = [u^1, u^2, \dots, u^{R_1}], \mathbf{V} = [v^1, v^2, \dots, v^{R_2}], \mathbf{W} = [w^1, w^2, \dots, w^{R_3}]$



Slika 2: Tucker dekompozicija

Notacija I

- Red tenzora - modovi

$$\mathcal{A} = \left(a_{i_1 i_2 \dots i_p}\right) \in \mathbb{R}^{n_1 \times n_2 \times \dots \times n_p}.$$

- Nit tenzora

  $A(:, j, k)$
- Odsječak tenzora

  $A^{(k)} = A(:, :, k), A(:, k, :), A(k, :, :)$ redom označavaju frontalni, laterlani i horizontalni odsječak

Notacija II

- Blok cirkularna matrica veličine $ln \times mn$ (nastala iz $A \in \mathbb{R}^{l \times m \times n}$)

$$\mathtt{bcirc}(\mathcal{A}) = \begin{bmatrix} \mathbf{A}^{(1)} & \mathbf{A}^{(n)} & \mathbf{A}^{(n-1)} & \dots & \mathbf{A}^{(2)} \\ \mathbf{A}^{(2)} & \mathbf{A}^{(1)} & \mathbf{A}^{(n)} & \dots & \mathbf{A}^{(3)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}^{(n)} & \mathbf{A}^{(n-1)} & \mathbf{A}^{(n-2)} & \dots & \mathbf{A}^{(1)} \end{bmatrix}$$

- Unfold

$$\mathtt{unfold}(\mathcal{A}) = \begin{bmatrix} \mathbf{A}^{(1)} \\ \mathbf{A}^{(2)} \\ \vdots \\ \mathbf{A}^{(n)} \end{bmatrix}$$

Notacija III

- Fold

$$\mathtt{fold}(\mathtt{unfold}(\mathcal{A})) = \mathcal{A}.$$

- Frobeniusova norma tenzora $A \in \mathbb{R}^{l \times m \times n}$

$$\|\mathcal{A}\|_F = \sqrt{\sum_{i=1}^{l} \sum_{j=1}^{m} \sum_{k=1}^{n} a_{ijk}^2},$$

PCA

PCA - *eigenfaces*

- PCA je jedna od klasičnih tehnika u prepoznavanju slika i kompresiji podataka.
- *glavne komponente* - transformirane nekorelirane varijable dobivene iz velikog broja orginalnih varijabli koje su moguće korelirane
- Prvih nekoliko glavnih komponenti objašnjava većinu varijance u podacima, dok ostale imaju mali doprinos.

PCA algoritam

Input: Slike za treniranje: $\mathbf{I}_i$; Test slika $\mathbf{J}$, indeks kraćenja k
Output: Uspredba test slike sa slikom iz skupa za treniranje
for $i = 1$ **to** N **do**
  $\mathbf{L}(:, i) \leftarrow$ vektorizirani $\mathbf{I}_i$
end for
$\mathbf{A} \leftarrow$ srednja devijacijska forma od $\mathbf{L}$
$\mathbf{U} \leftarrow$ lijevi singularni vektori od $\mathbf{A}$
$\mathbf{G} \leftarrow \mathbf{U}(:, 1:k)^T * \mathbf{A}$
$\mathbf{T} \leftarrow (\mathbf{J} - \mathbf{M})$; $\mathbf{M}$ srednja slika
$\mathbf{t} \leftarrow$ vektorizirani forma od $\mathbf{T}$
$\mathbf{c} \leftarrow \mathbf{U}(:, 1:k)^T * \mathbf{t}$
for $j = 1$ **to** N **do**
  Izračunaj $\|c - \mathbf{G}(:, j)\|_F$
end for
Slika za treniranje čiji su koeficijenti najbliži test slici je rezultat.

Algorithm 1: Tradicionalna matrična PCA metoda

Tensorfaces

TensorFaces

- TensorFaces je bio prvi algoritam koji je koristio tenzore u rješavanju problema prepoznavanja lica.
- Podaci su reprezentirani kao tenzor s različitim modovima za različite faktore. Kao i u PCA, slike su vektorizirane. Ostali modovi se koriste kako bi se reprezentirali ostala svojstva slike.
- Aproksimacija tenzora se postiže multidimenzionalnom metodom *higher-order* SVD (HOSVD).

TensorFaces algoritam I

Input: Slike za treniranje:
$\mathbf{I}_{p,v,t,e}, \; p = 1, \ldots, P, \; v = 1, \ldots, V, \; i = 1, \ldots, I, \; e = 1, \ldots, E;$
Test slika $\mathbf{J}$

Output: Koeficijenti testne slike s nekim tenzorskim bazama i usporedbama

for $p = 1 \ldots, P, \; v = 1 \ldots, V, \; i = 1 \ldots, I, \; e = 1 \ldots, E$ **do**
  $\mathcal{L}(p, v, i, e, :) \leftarrow$ vektorizirani $\mathbf{I}_{p,v,i,e}$
end for

Napravi HOSVD da se dobije jezgreni tenzor $\mathcal{Z}$ i ortogonalne faktorske matrice $\mathbf{U}_i$:

$\mathcal{L} \leftarrow \mathcal{Z} \times_1 \mathbf{U}_p \times_2 \mathbf{U}_v \times_3 \mathbf{U}_i \times_4 \mathbf{U}_e \times_5 \mathbf{U}_pix$

$\mathcal{B} \leftarrow \mathcal{Z} \times_2 \mathbf{U}_v \times_3 \mathbf{U}_i \times_4 \mathbf{U}_e \times_5 \mathbf{U}_pix$

$\mathcal{T}(1, 1, 1, 1 :) \leftarrow$ vektorizirana slika za testiranje

TensorFaces algoritam II

for $v = 1, \ldots, V,\ i = 1, \ldots, I,\ e = 1, \ldots, E$ **do**

$c_{v,i,e} \leftarrow (\mathcal{B}^{\dagger}_{v,i,e})^T \, \mathcal{T}(1, 1, 1, 1, :)$

for $p = 1, \ldots, P$ **do**

Izračunaj $\|c_{v,i,e} - \mathbf{U}_p(p, :)^T\|_F$

end for

end for

Slika za treniranje čiji su koeficijenti najbliži test slici je rezultat.

Algorithm 2: TensorFaces metoda

Tensor framework

Tenzor - tenzor množenje I

Definicija 4.1

Neka je $\mathcal{A}$ $l \times p \times n$ i $\mathcal{B}$ $p \times m \times n$, tada definiramo produkt tenzora $\mathcal{A} * \mathcal{B}$ kao $l \times m \times n$ tenzor

$$\mathcal{A} * \mathcal{B} = \texttt{fold}(\texttt{bcirc}(\mathcal{A}) \cdot \texttt{unfold}(\mathcal{B})) \tag{1}$$

U praksi tenzor - tenzor množenje računamo malo drukčije.
Pretpostavimo da je $\mathcal{A} \in \mathbb{R}^{l \times m \times n}$ i $\mathbf{F}_n$ je $n \times n$ DFT matrica.
Dobivamo $\texttt{bcirc}(\mathcal{A})$ koja je $n \times n$ blok matrica s $l \times m$ blokova.
Nakon kratke manipulacije dobivamo:

Tenzor - tenzor množenje II

$$(\mathbf{F}_n \otimes \mathbf{I}_l)\ \texttt{bcirc}(\mathcal{A})\ (\mathbf{F}_n^* \otimes \mathbf{I}_m) = \mathbf{P}_1^T \begin{bmatrix} \mathbf{D}_{11} & \mathbf{D}_{12} & \cdots & \mathbf{D}_{1m} \\ \mathbf{D}_{21} & \mathbf{D}_{22} & \cdots & \mathbf{D}_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{D}_{l1} & \mathbf{D}_{l3} & \cdots & \mathbf{D}_{lm} \end{bmatrix} \mathbf{P}_2^T$$

$$= \begin{bmatrix} \hat{\mathbf{A}}^{(1)} & & & \\ & \hat{\mathbf{A}}^{(2)} & & \\ & & \ddots & \\ & & & \hat{\mathbf{A}}^{(n)} \end{bmatrix}.$$

Tenzor - tenzor množenje III

Primijenjujući brzu Fourierovu transformaciju duž trećeg moda $\mathcal{A}$ daje nam alternativni način računanju $\hat{\mathbf{A}}^{(i)}$.

Input: $\mathcal{A} \in \mathbb{R}^{l \times p \times n}, \mathcal{B} \in \mathbb{R}^{p \times m \times n}$
Output: $\mathcal{C} := \mathcal{A} * \mathcal{B}$
$\hat{\mathcal{A}} \leftarrow \texttt{fft}(\mathcal{A}, [], 3)$; $\hat{\mathcal{B}} \leftarrow \texttt{fft}(\mathcal{B}, [], 3)$
for $i = 1$ **to** n **do**
$\hat{\mathcal{C}}(:,:,i) \leftarrow \hat{\mathcal{A}}(:,:,i)\,\hat{\mathcal{B}}(:,:,i)$
end for
$\mathcal{C} \leftarrow \texttt{ifft}(\hat{\mathcal{C}}, [], 3)$

Algorithm 3: Tenzor - tenzor množenje, koristeći računanje u Fourierovoj domeni

Pojmovi i svojstva potrebni za T-SVD I

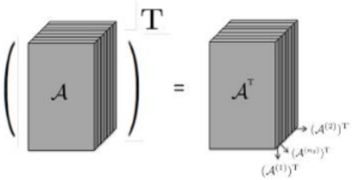
Lema 4.1

$\mathcal{A} * (\mathcal{B} * \mathcal{C}) = (\mathcal{A} * \mathcal{B}) * \mathcal{C}$

Definicija 4.2

Ako je $\mathcal{A}$ tenzor dimenzija $l \times m \times n$, onda je $\mathcal{A}^T$ tenzor dimenzija $m \times l \times n$ dobiven transponiranjem svakog frontalnog odsječka te mijenjanjem poretka transponiranih frontalnih odsječaka od 2 do n.

Pojmovi i svojstva potrebni za T-SVD II



Slika 3: Transponiranje

Definicija 4.3

Kažemo da je tenzor f-dijagonalan ako je svaki frontalni odsječak dijagonalna matrica.

Pojmovi i svojstva potrebni za T-SVD III

Definicija 4.4

Tenzor dimenzija $l \times l \times n$ $\mathcal{I}_{lln}$ je identiteta čiji je prvi frontalni odsječak $l \times l$ matrica identiteta, a svi ostali frontalni odsječci su nulmatrice.

Definicija 4.5

Tenzor realnih vrijednosti $\mathcal{Q}$ dimenzija $l \times l \times n$ je ortogonalan ako

$$\mathcal{Q}^T * \mathcal{Q} = \mathcal{Q} * \mathcal{Q}^T = \mathcal{I}. \tag{2}$$

T-SVD

T-SVD dekompozicija

Teorem 5.1

Neka je $\mathcal{A}$ tenzor realnih vrijednosti dimenzija $l \times m \times n$. Tada se $\mathcal{A}$ može faktorizirati u obliku

$$\mathcal{A} = \mathcal{U} * \mathcal{S} * \mathcal{V}^T$$

pri čemu su $\mathcal{U}$ ortogonalni tenzor dimenzija $l \times l \times n$, $\mathcal{V}$ ortogonalni tenzor dimenzija $m \times m \times n$ te $\mathcal{S}$ f-dijagonalni tenzor dimenzija $l \times m \times n$.

Dokaz teorema 5.1 I

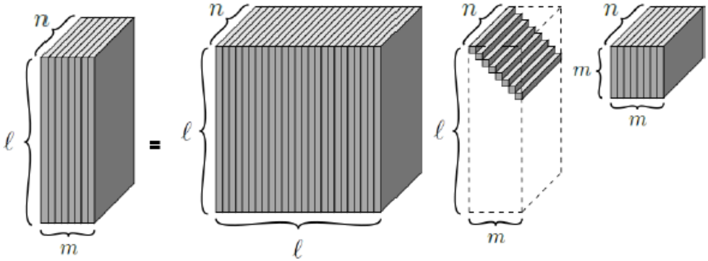
Prvo, transformirajmo $\mathtt{bcirc}(\mathcal{A})$ u Fourierovu domenu:

$$(\mathbf{F}_n \otimes \mathbf{I}_l)\,\mathtt{bcirc}(\mathcal{A})\,(\mathbf{F}_n^* \otimes \mathbf{I}_m) = \begin{bmatrix} \hat{\mathbf{A}}^{(1)} & & & \\ & \hat{\mathbf{A}}^{(2)} & & \\ & & \ddots & \\ & & & \hat{\mathbf{A}}^{(n)} \end{bmatrix}$$

Izračunajmo matrični SVD za svaki $\hat{\mathbf{A}}^{(i)}$ kao $\hat{\mathbf{A}}^{(i)} = \hat{\mathbf{U}}^{(i)}\,\hat{\mathbf{S}}^{(i)}\,(\hat{\mathbf{V}}^{(i)})^T$

Sada nakon što izaberemo odgovarajuće dimenzije za $\mathbf{I}$, primijenimo $\mathbf{F}_n^* \otimes \mathbf{I}$ slijeva i $\mathbf{F}_n \otimes \mathbf{I}$ s desna svake blok dijagonalne matrice da dobijemo tri cirkularne matrice čiji su prvi blok stupci $\mathtt{unfold}(\mathcal{U})$, $\mathtt{unfold}(\mathcal{S})$, $\mathtt{unfold}(\mathcal{V}^T)$. Primijenjujući fold dobivamo $\mathcal{U}, \mathcal{S}, \mathcal{V}^T$.

Dokaz teorema 5.1 II



Slika 4: T-SVD

Korištenje T-SVD-a u dekompoziciji slika

- Predložena metoda razlikuje se od PCA za matrice na način da se podaci čuvaju u tenzorima trećeg reda te se koriste tenzor-teznor faktorizacije bazirane na operatoru $*$.
- razlikuje se od TensorFaces algoritma jer slike nisu pikselizirane, nema dodatnih dimenzija koje predstavljaju osvjetljenje, poze i slično te koristi T-SVD umjesto HOSVD-a.

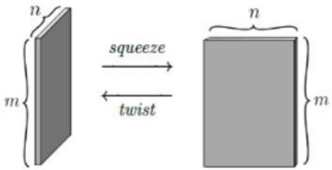
Squeeze i *Twist* operacije I

- U postupku će nam koristiti sljedeća notacija: `squeeze` operacija koja uzima tenzor $\mathcal{X}$ iz $\mathbb{R}^{l\times 1\times n}$ i daje matricu

$$\mathbf{X} := \texttt{squeeze}(\mathcal{A}) \Rightarrow \mathbf{X}(i,j) := \mathcal{X}(i,1,j)$$

dok je `twist` inverzna operacija operaciji `squeeze`

$$\texttt{twist}\left(\texttt{squeeze}\left(\mathcal{X}\right)\right) = \mathcal{X}.$$



Slika 5: Prikaz operacija `squeeze` i `twist`

Algoritam T-SVD 1

Input: Slike za treniranje: $\mathbf{I}_{i},$; Test slika $\mathbf{J}$, indeks kraćenja k
Output: Odgovarajuća slika za test sliku iz skupa za treniranje

for $i = 1$ **to** N **do**
  $\mathcal{L}(:, i, :) \leftarrow \mathbf{I}_i$
end for
$\mathbf{M} \leftarrow$ srednja slika
$\mathcal{A} \leftarrow$ srednja devijacijska forma od $\mathcal{L}$
$\mathcal{U} \leftarrow$ lijevi singularni vektori tenzora $\mathcal{A}$
$\mathcal{C} \leftarrow \mathcal{U}(:, 1:k, :)^T * \mathcal{A}$
$\mathcal{T}(:, 1, :) \leftarrow \mathtt{twist}(\mathbf{J} - \mathbf{M})$
$\mathcal{B} \leftarrow \mathcal{U}(:, 1:k, :)^T * \mathcal{T}$
for $j = 1$ **to** N **do**
  Izračunaj $\|\mathcal{B} - \mathcal{C}(:, j, :)\|_F$
end for
Slika za treniranje čiji su koeficijenti najbliži test slici je rezultat.

Poboljšani T-SVD I

Ako označimo

- $\mathbf{Y}_j := \texttt{squeeze}(\mathbf{A}(:, j, :))$ j-tu slika za treniranje
- $\texttt{circ}(v)$ cirkularnu matricu čiji je prvi stupac v

onda se može pokazati da je

$$\mathbf{Y}_j \approx \sum_{t=1}^{k} \mathbf{W}_t \, \texttt{circ}(\mathbf{c}_{t,j}) \tag{3}$$

gdje su

- $\mathbf{W}_t := \texttt{squeeze}(\mathcal{U}(:, t, :)) \in \mathbb{R}^{l \times n}$ matrica
- $\mathbf{c}_{t,j}^T := \texttt{squeeze}(\mathbf{C}(t, j, :)^T)$ vektor redak duljine n

Poboljšani T-SVD II

Teorem 5.2

Neka $\mathbf{F}$ *predstavlja* $n \times n$ *DFT matricu. Neka su* $\widehat{\mathbf{Y}_j} := \mathbf{Y}_j \mathbf{F}, \widehat{\mathbf{W}_j} := \mathbf{W}_j \mathbf{F}$ *i* $\widehat{\mathbf{c}_{t,j}}$ *Fourierovi koeficijenti vektora stupca* $\mathbf{c}_{t,j}$ *duljine* n*,*

$$\widehat{\mathbf{Y}_j} \approx \left[\sum_{t=1}^{k} \widehat{\mathbf{c}_{t,j}}^{(1)} \hat{\mathbf{U}}^{(1)}(:,t), \sum_{t=1}^{k} \widehat{\mathbf{c}_{t,j}}^{(2)} \hat{\mathbf{U}}^{(2)}(:,t), \ldots, \sum_{t=1}^{k} \widehat{\mathbf{c}_{t,j}}^{(n)} \hat{\mathbf{U}}^{(n)}(:,t) \right] \tag{4}$$

gdje $\widehat{\mathbf{c}_{t,j}}^{(i)}$ *predstavlja i-tu komponentu tog vektora i* $\widehat{\mathbf{W}}_t(:,i) = \hat{\mathbf{U}}^{(i)}(:,t)$*, gdje je* $\hat{\mathbf{U}}^{(i)}$ *unitarna matrica iz dokaza teorema T-SVD-a. Slijedi da je algoritam 2 ekvivalentan simultanoj PCA izvršenoj na svakom stupcu transformirane slike u Fourireovom prostoru, koristeći iste indekse kraćenja k na svakom stupcu.*

Poboljšani T-SVD III

Rezultat:

$$\widehat{\mathbf{Y}_j} \approx \left[\sum_{t=1}^{k_1} \widehat{\mathbf{c_{t,j}}}^{(1)} \widehat{\mathbf{U}}^{(1)}(:,t), \sum_{t=1}^{k_2} \widehat{\mathbf{c_{t,j}}}^{(2)} \widehat{\mathbf{U}}^{(2)}(:,t), \ldots, \sum_{t=1}^{k_n} \widehat{\mathbf{c_{t,j}}}^{(n)} \widehat{\mathbf{U}}^{(n)}(:,t) \right], \tag{5}$$

gdje, da bi sačuvali konjugatnu simetriju, indeks kraćenja za stupce $j,\ 2 \leq j \leq \frac{n}{2}$ je isti kao i za stupce $n - j + 2$.

Tada je suma svih indeksa kraćenja jednaka

$$K = k_1 + 2k_2 + \ldots + 2k_{\frac{n}{2}} + k_{\frac{n}{2}+1} \tag{6}$$

Računanje parametara kraćenja $k_i, i = 1, \ldots, \frac{n}{2} + 1$

- Vrijedi:
  $\|\widehat{\mathbf{A}}(:,:,1:\frac{n}{2}+1)\|_F^2 = \sum_{i=1}^{\frac{n}{2}+1} \|\hat{\mathbf{A}}(:,:,i)\|_F^2 = \sum_{i=1}^{\frac{n}{2}+1} \sum_{j=1}^{\min(l,m)} (\hat{\mathbf{s}}_j^{(i)})^2 (7)$
- Mjera relativne energije:

$$\frac{\sum_{i=1}^{\frac{n}{2}+1} \sum_{j=1}^{k_i} \left(\hat{\mathbf{s}}_j^{(i)}\right)^2}{\|\hat{\mathbf{A}}(:,:,1:\frac{n}{2}+1)\|_F^2} \tag{8}$$

- Parametre kraćenja računamo traženjem najmanjeg indeksa t za koji vrijedi, uz oznaku *treshold* za relativnu energijsku vrijednost, i $\mathbf{q}$ za vektor koji sadrži $\hat{\mathbf{s}}_j^{(i)}$ od najveće do najmanje:
  $\sum_{i=1}^{\frac{n}{2}+1} \sum_{j=1}^{k_i} \{(\hat{\mathbf{s}}_j^{(i)})^2 : (\hat{\mathbf{s}}_j^{(i)})^2 > \mathbf{q}(t)^2\} > \|\widehat{\mathbf{A}}(:,:,1:\frac{n}{2}+1)\|_F^2 \times \mathit{treshold} (9)$
- Kako vrijedi $\hat{\mathbf{s}}_1^{(i)} \geq \hat{\mathbf{s}}_2^{(i)} \geq \ldots \hat{\mathbf{s}}_{\min\{l,m\}}^{(i)}$, k_i se definiraju tako da je $\hat{\mathbf{s}}_{k_i}^{(i)} > \mathbf{q}(t)$, ali $\hat{\mathbf{s}}_{k_i+1}^{(i)} \leq \mathbf{q}(t)$

Algoritam T-SVD 2 I

Input: Slike za treniranje: $\mathbf{I}_i,\ i = 1, 2, \ldots, N$; Test slika $\mathbf{J}$, indeks kraćenja k

Output: Usporedba test slike s reprezentacijom skupa za treniranje u Fourierovoj domeni

for $i = 1$ **to** N **do**
  $\mathcal{L}(:, :, i) \leftarrow \mathbf{I}_i$
end for
$\mathcal{M} \leftarrow$ srednja slika
$\mathcal{A} \leftarrow$ srednja devijacijska forma od $\mathcal{L}$
Frontalni odsječci SVD-a od $\hat{\mathbf{A}}^{(i)} \leftarrow \hat{\mathcal{A}}(:, :, i),\ i = 1, \ldots, \frac{n}{2} + 1$
Drop strategija; sadrži jedino $\hat{\mathbf{U}}^{(i)}(:, 1 : k_i)$
$\mathcal{T}(:, 1, :) \leftarrow \mathtt{twist}(\mathbf{J} - \mathcal{M})$
for $i = 1 : \frac{n}{2} + 1,\ t = 1 : k_t,\ j = 1 : N$ **do**
  $\hat{\mathbf{c}}_{t,j}^{(i)} \leftarrow (\hat{\mathbf{U}}^{(i)}(:, t))^H \hat{\mathbf{A}}^{(i)}(:, j)$
end for

Algoritam T-SVD 2 II

for $i = 1 : \frac{n}{2} + 1,\ t = 1 : k_t$ **do**

$\quad \hat{\mathbf{b}_t}^{(i)} \leftarrow (\hat{\mathbf{U}}^{(i)}(:, t))^H \, \hat{\mathbf{T}}^{(i)}(:, 1)$

end for

for $j = 1 : N$ **do**

$\quad$ Izračunaj $\|\hat{\mathbf{b}_t}^{(i)} - \hat{\mathbf{c}}_{t,j}^{(i)}\|_F$

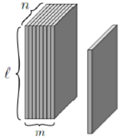
end for

Slika za treniranje čiji su koeficijenti najbliži test slici je rezultat.

T-PQ

Alternativni pristup pomoću tenzorske QR faktorizacije I

Alternativna metoda tenzorske faktorizacije s potencijalnom prednošću u slučaju ažuriranja baze lica, ali s nedostatkom da faktorizacija neće biti optimalna.



Slika 6: Ažuriranje baze podataka

Alternativni pristup pomoću tenzorske QR faktorizacije II

Definicija 6.1

Permutacijski tenzor $\mathcal{B} \in \mathbb{R}^{m \times m \times n}$ je tenzor čiji su elementi nule i jedinice te vrijedi:

$$\mathcal{P}^T * \mathcal{P} = \mathcal{P} * \mathcal{P}^T = \mathcal{I}. \tag{10}$$

Teorem 6.1

Neka je $\mathcal{A} \in \mathbb{R}^{l \times m \times n}$ tenzor realnih vrijednosti. Tada se $\mathcal{A}$ može faktorizirati u obliku

$$\mathcal{A} * \mathcal{P} = \mathcal{Q} * \mathcal{R} \tag{11}$$

gdje je $\mathcal{Q} \in \mathbb{R}^{l \times l \times n}$ ortogonalni tenzor, $\mathcal{R} \in \mathbb{R}^{l \times m \times n}$ f-gornje trokutasti tenzor i $\mathcal{P}$ permutacijski tenzor.

Dokaz Teorema 6.1

- Prebacimo se u Fourierovu domenu
- Napravimo *QR*-faktorizacije pivotirane po stupcima:

$$\hat{\mathbf{A}}^{(1)}\mathbf{P}^{(1)} = \hat{\mathbf{Q}}^{(1)}\hat{\mathbf{R}}^{(1)} \tag{12}$$

$$\hat{\mathbf{A}}^{(i)}\mathbf{P}^{(1)} = \hat{\mathbf{Q}}^{(i)}\hat{\mathbf{R}}^{(i)} \tag{13}$$

- Definiramo

$$\mathbf{P} := (\mathbf{F}^* \otimes \mathbf{I})(\text{blok dijagonalna matrica sa } \mathbf{P}^{(1)} \text{na glavnim dijagonalama })(\mathbf{F} \otimes \mathbf{I}) \tag{14}$$

$$\hat{\mathbf{Q}} = (\mathbf{F}^* \otimes \mathbf{I})(\text{blok dijagonalna matrica sa } \hat{\mathbf{Q}}^{(i)} \text{na glavnim dijagonalama })(\mathbf{F} \otimes \mathbf{I}) \tag{15}$$

$$\hat{\mathbf{R}} = (\mathbf{F}^* \otimes \mathbf{I})(\text{blok dijagonalna matrica sa } \hat{\mathbf{R}}^{(i)} \text{na glavnim dijagonalama })(\mathbf{F} \otimes \mathbf{I}) \tag{16}$$

- $\mathcal{P}, \mathcal{Q}, \mathcal{R}$ dobivamo primijenjujući `fold` na prvi blok stupac blok cirkularnih matrica 14, 15 i 16, respektivno.

Algoritam za T-QR

Input: Slike za treniranje: $\mathbf{I}_i$, $i = 1, 2, \ldots, N$; Test slika $\mathbf{J}$, indeks kraćenja k

Output: Koeficijenti testnih slika s bazom smanjene dimenzije

for $i = 1$ **to** N **do**
  $\mathcal{L}(:,:,i) \leftarrow \mathbf{I}_i$
end for
$\mathcal{M} \leftarrow$ srednja slika
$\mathcal{A} \leftarrow$ srednja devijacijska forma od $\mathcal{L}$
$[\mathcal{Q}, \mathcal{R}] \leftarrow$ tenzor (pivotirana) QR dekompozicija od $\mathcal{A}$
$\mathcal{C} \leftarrow \mathcal{Q}(:, 1:k, :)^T * \mathcal{A}$
$\mathcal{T}(:, 1, :) \leftarrow \texttt{twist}(\mathbf{J} - \mathcal{M})$
$\mathcal{B} \leftarrow \mathcal{Q}(:, 1:k, :)^T * \mathcal{T}$
for $j = 1$ **to** N **do**
  Izračunaj $\|\mathcal{B} - \mathcal{C}(:, j, :)\|_F$
end for
Slika za treniranje čiji su koeficijenti najbliži test slici je rezultat.

Rezultati

Podaci

- Slike preuzete iz [7] korišteni su po uzoru na [2]

Slika 7: Primjer fotografija iz preuzete baze podataka [7]

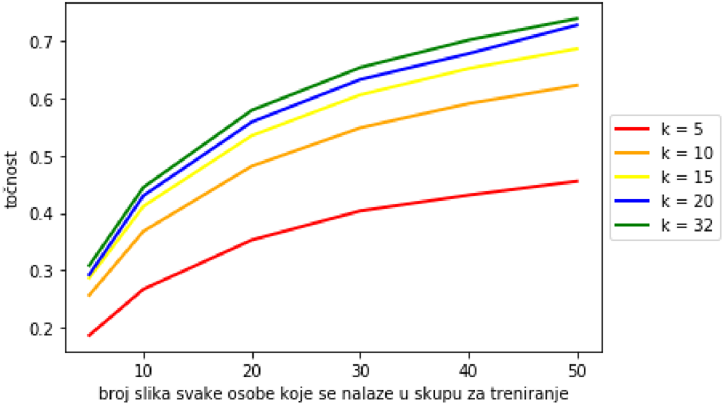
- Preuzeta baza podataka sadrži 16128 slika 28 osoba uslikanih u 9 različitih pozicija i u 64 različita osvjetljenja

Testiranje

- `Matlab` okruženje
- Faza *treniranja* i *testiranja*
- $p = 5, 10, 20, 30, 40, 50$ predstavlja broj različitih slika svake osobe koje se nalaze u skupu za treniranje te je za svaki p na nasumičan način generirano 50 mogućnosti skupa za treniranje, odnosno skupa za testiranje algoritma
- Za svaki p pokrenuli smo algoritme na svih 50 mogućnosti i usporedili rezultate
- Točnost algoritma određivali smo formulom:

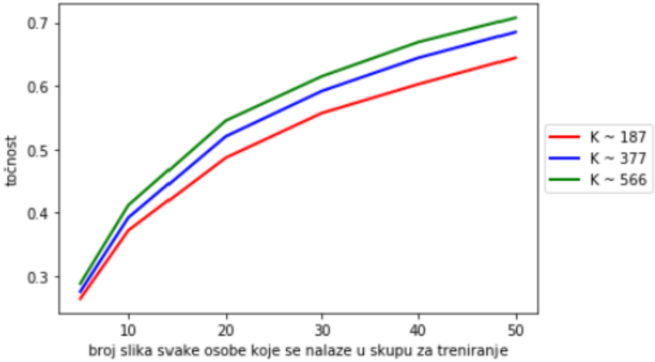
$$\frac{\textit{broj dobro prepoznatih lica}}{\textit{broj testnih fotografija}} \tag{17}$$

Rezultati - TSVD I



Slika 8: Rezultati pokretanja algoritma koji koristi TSVD metodu dekompozicije tenzora s fiksnim indeksom suženja k na podacima iz [7].

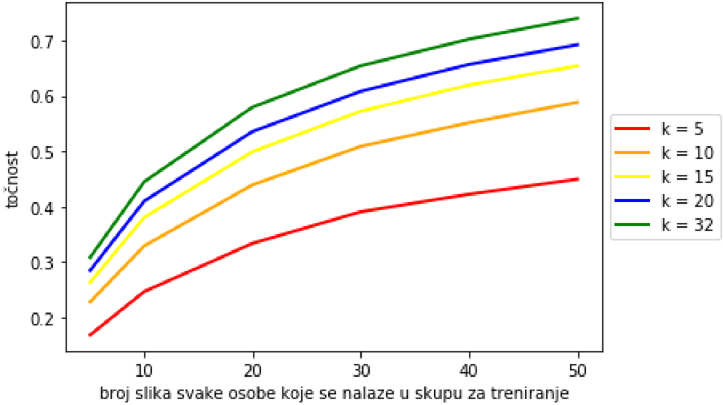
Rezultati - TSVD II



Slika 9: Rezultati pokretanja algoritma koji koristi TSVD metodu dekompozicije tenzora s različitim indeksima kraćenja k_i na podacima iz [7].

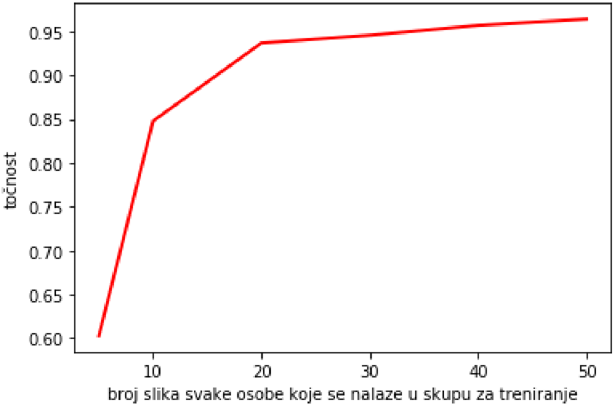
Rezultati - TSVD III

Rezultati - TQR



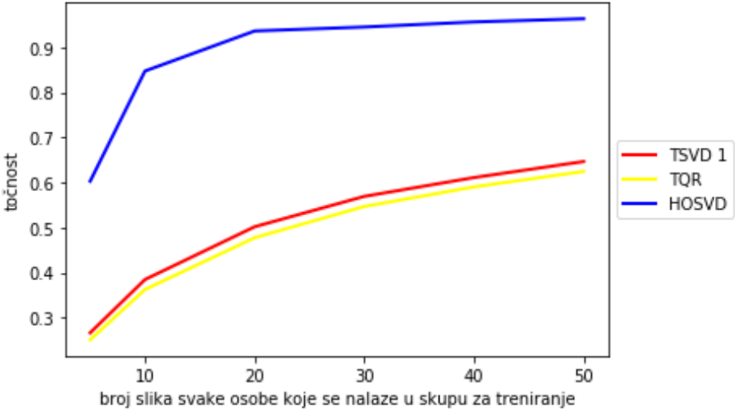
Slika 10: Rezultati pokretanja algoritma koji koristi TQR metodu dekompozicije tenzora na podacima iz [7].

Rezultati - HOSVD



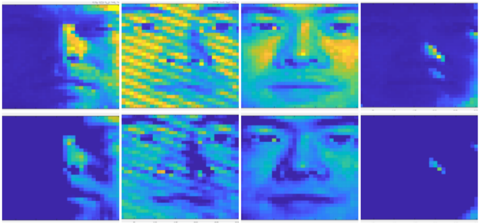
Slika 11: Rezultati pokretanja algoritma koji koristi HOSVD metodu dekompozicije tenzora na podacima iz [7].

Zaključak

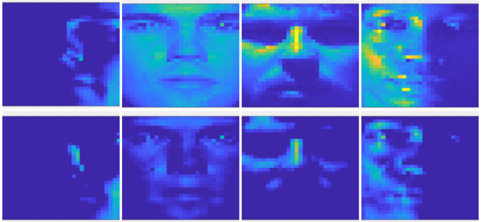


Slika 12: Usporedba svih rezultata pokretanja prethodno navedenih algoritma na podacima iz [7].

Fotografije prije i poslije primjena TSVD-a



Slika 13: Primjer dobro prepoznatih lica



Slika 14: Primjer loše prepoznatih lica

Primjena TSVD metode na videu

▸ Link

Slika 15: Video prije i nakon primjene TSVD-a na njegov tenzor podataka

Literatura

Literatura I

- A.S. Georghiades, P.N. Belhumeur i D.J. Kriegman. „From Few to Many: Illumination Cone Models for Face Recognition under Variable Lighting and Pose". *IEEE Trans. Pattern Anal. Mach. Intelligence* 23.6 (2001), str. 643–660.
- Ning Hao i dr. „Facial Recognition Using Tensor-Tensor Decompositions". *SIAM Journal on Imaging Sciences [electronic only]* 6 (veljača 2013). doi: 10.1137/110842570.
- M.E. Kilmer, C.D. Martin i L. Perrone. „A third-order generalization of the matrix svd as a product of third-order tensors". *Tufts Computer Science Technical Report* (2008).
- Liang Liao i Stephen Maybank. „Generalized Visual Information Analysis Via Tensorial Algebra". *Journal of Mathematical Imaging and Vision* (veljača 2020). doi: 10.1007/s10851-020-00946-9.

Literatura II

Neil Muller, Lourenço Magaia i Ben Herbst. „Singular Value Decomposition, Eigenfaces, and 3D Reconstructions". *Society for Industrial and Applied Mathematics* 46 (rujan 2004), str. 518–545.

Berkant Savas i Lars Eldén. „Handwritten digit classification using higher order singular value decomposition". *Pattern Recognit.* 40 (2007), str. 993–1003.

The Extended Yale Face Database B. `http://vision.ucsd.edu/~leekc/ExtYaleDatabase/ExtYaleB.html.`. Dohvaćeno: 30.3.2020.