

Prepoznavanje lica pomoću tenzorske dekompozicije singularnih vrijednosti

Zirdum, Ivan

Master's thesis / Diplomski rad

2017

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Science / Sveučilište u Zagrebu, Prirodoslovno-matematički fakultet**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:217:788907>

Rights / Prava: [In copyright](#)/Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-10-11**



Repository / Repozitorij:

[Repository of the Faculty of Science - University of Zagreb](#)



SVEUČILIŠTE U ZAGREBU
PRIRODOSLOVNO–MATEMATIČKI FAKULTET
MATEMATIČKI ODSJEK

Ivan Zirdum

**PREPOZNAVANJE LICA POMOĆU
TENZORSKE DEKOMPOZICIJE
SINGULARNIH VRIJEDNOSTI**

Diplomski rad

Voditelj rada:
doc. dr. sc. Nela Bosner

Zagreb, studeni 2017.

Ovaj diplomski rad obranjen je dana _____ pred ispitnim povjerenstvom u sastavu:

1. _____, predsjednik
2. _____, član
3. _____, član

Povjerenstvo je rad ocijenilo ocjenom _____.

Potpisi članova povjerenstva:

1. _____
2. _____
3. _____

Sadržaj

Uvod	1
1 Vektorske i matrice norme	3
1.1 Vektorske norme	3
1.2 Matrice norme	4
2 Dekompozicija matrice na singularne vrijednosti (SVD)	7
2.1 SVD	7
2.2 Neke primjene dekompozicije matrice na singularne vrijednosti	13
3 Tenzori	16
3.1 Uvod	16
3.2 Osnovni koncepti tenzora	18
3.3 Tenzorski SVD (HOSVD)	26
3.4 Aproksimacija tenzora HOSVD-om	30
4 Prepoznavanje Lica Korištenjem HOSVD-a	35
4.1 Tenzorski prikaz fotografija	35
4.2 Prepoznavanje lica	39
5 Primjeri dobiveni MATLAB rutinama	41
5.1 Klasifikacija	44
Bibliografija	51

Uvod

Moderna tehnologija i stalan porast broja stanovnika na Zemlji razlog su velike produkcije podataka. Svaki dan se proizvede ogromna količina novih podataka koje je potrebno spremati, obraditi, a često i slati. Zbog toga je bilo potrebno osmisliti načine za efikasno spremanje podataka kako bi oni zauzimali što je manje moguće prostora, a da glavna informacija bude očuvana. Lijep primjer toga je kompresija digitalne slike. Uzmimo za primjer sliku dimenzije 3000×2000 piksela. Kako bismo takvu sliku spremili u memoriju potrebno nam je $3000 \cdot 2000 \cdot 24 = 144000000$ bitova memorije (za pamćenje jednog piksela koristimo 24 bita) što iznosi 18 megabajta.



Slika 1: Izvorna slika - Komprimirana slika.

U ovom radu bavit ćemo se matričnom(tenzorskom) dekompozicijom koja nam omogućava da iz nekog skupa podataka, prikazanog pomoću matrice(tenzora), izdvojimo "najvažniji" dio, a ostatak izostavimo. Koristeći tu tehniku možemo komprimirati slike, razviti algoritme koji: klasificiraju rukom pisane znamenke, prepoznaju ljudska lica i još mnogo toga. Zamislite sada da ste zaposleni u pošti i da je vaša zadaća sortirati pristigla pisma prema poštanskom broju kako bi ona stigla na pravu adresu. Možda biste radije bili prometni policajac na autocesti koji je zadužen za praćenje brzine automobila koje ulaze u tunel. Vaš posao je snimiti tablice automobila koji ne poštuju ograničenje brzine, a zatim u bazi podataka registracija vozila

pronaći tu tablicu te na priloženu adresu vlasnika poslati kaznu za prebrzu vožnju. Biste li svaki dan isti posao radili ručno ili biste radije pokušali taj postupak automatizirati? Srećom, ljudi su se već susreli s ovim problemom i smislili mnogo načina kako taj posao ubrzati i uštediti dragocjeno vrijeme. Svi navedeni problemi mogu se riješiti upotrebom jednog moćnog alata numeričke matematike koji ćemo vam pokušati približiti u ostatku ovog rada.

Poglavlje 1

Vektorske i matrice norme

1.1 Vektorske norme

Vektorske i matrice norme osnovno su sredstvo koje koristimo kod ocjene grešaka vezanih uz numeričke metode, posebno u numeričkoj linearnoj algebri.

Definicija 1.1.1. (Vektorska norma) Vektorska norma je svaka funkcija $\|\cdot\| : \mathbb{C}^n \rightarrow \mathbb{R}$ koja zadovoljava sljedeća svojstva:

1. $\|x\| \geq 0, \forall x \in \mathbb{C}^n$, a jednakost vrijedi ako i samo ako je $x = 0$,
2. $\|\alpha x\| = |\alpha| \|x\|, \forall \alpha \in \mathbb{C}, \forall x \in \mathbb{C}^n$,
3. $\|x + y\| \leq \|x\| + \|y\|, \forall x, y \in \mathbb{C}^n$. Ova je nejednakost poznatija pod imenom nejednakost trokuta (zbroj duljina bilo koje dvije stranice trokuta veći je od duljine treće stranice).

Analogno se definira vektorska norma na bilo kojem vektorskom prostoru V nad poljem $F = \mathbb{R}$ ili \mathbb{C} .

Neka je x vektor iz \mathbb{C}^n s komponentama $x_i, i = 1, \dots, n$, u oznaci $x = (x_1, \dots, x_n)^T$, ili, skraćeno $x = [x_i]$. U numeričkoj linearnoj algebri najčešće se koriste sljedeće tri norme:

1. 1-norma ili l_1 norma, u engleskom govornom području poznatija kao “Manhattan” ili “taxi-cab” norma

$$\|x\|_1 = \sum_{i=1}^n |x_i|,$$

2. 2-norma ili l_2 norma ili euklidska norma

$$\|x\|_2 = (x^*x)^{1/2} = \left(\sum_{i=1}^n |x_i|^2 \right)^{1/2},$$

3. ∞ -norma ili l_∞ norma

$$\|x\|_\infty = \max_{i=1, \dots, n} |x_i|.$$

Primijetimo da je samo 2-norma izvedena iz skalarnog produkta, dok ostale dvije to nisu.

2-norma ima bitno svojstvo koje je čini posebno korisnom. Ona je invarijantna na unitarne transformacije vektora x , tj. ako je Q unitarna matrica ($Q^*Q = QQ^* = I$), onda je

$$\|Qx\|_2 = (x^*Q^*Qx)^{1/2} = (x^*x)^{1/2} = \|x\|_2.$$

Sve ove tri norme specijalni su slučaj Hölderove p -norme (l_p norme) definirane s:

$$\|x\|_p = \left(\sum_{i=1}^n |x_i|^p \right)^{1/p}, \quad p \geq 1.$$

Za Hölderove p -norme vrijedi i poznata Hölderova nejednakost

$$|x^*y| \leq \|x\|_p \|y\|_q, \quad \frac{1}{q} + \frac{1}{p} = 1.$$

Posebni slučaj Hölderove nejednakosti za $p = q = 2$ je Cauchy-Schwarzova nejednakost

$$|x^*y| \leq \|x\|_2 \|y\|_2.$$

1.2 Matrične norme

Zamjenimo li u definiciji (1.1.1) vektor $x \in \mathbb{C}^n$ matricom $A \in \mathbb{C}^{m \times n}$, dobivamo matričnu normu.

Definicija 1.2.1. (Matrična norma) Matrična norma je svaka funkcija $\|\cdot\| : \mathbb{C}^{m \times n} \rightarrow \mathbb{R}$ koja zadovoljava sljedeća svojstva:

1. $\|A\| \geq 0, \forall A \in \mathbb{C}^{m \times n}$, a jednakost vrijedi ako i samo ako je $A = 0$,
2. $\|\alpha A\| = |\alpha| \|A\|, \forall \alpha \in \mathbb{C}, \forall A \in \mathbb{C}^{m \times n}$,
3. $\|A + B\| \leq \|A\| + \|B\|, \forall A, B \in \mathbb{C}^{m \times n}$.
4. Za matričnu normu ćemo reći da je konzistentna ako vrijedi $\|AB\| \leq \|A\| \|B\|$

kad god je matrični produkt AB definiran. Oprez, norme od A, B i AB ne moraju biti definirane na istom prostoru (dimenzije)!

Neki autori smatraju da je i ovo posljednje svojstvo sastavni dio definicije matrične norme (tada to svojstvo obično zovu submultiplikativnost). Ako su ispunjena samo prva tri svojstva, onda to zovu generalizirana matrična norma.

Matrične norme mogu nastati na dva različita načina. Ako matricu A promatramo kao vektor s $m \times n$ elemenata, onda, direktna primjena vektorskih normi (uz oznaku a_{ij} matričnog elementa u i -tom retku i j -tom stupcu) daje sljedeće definicije:

1. l_1 norma

$$\|A\|_1 := \sum_{i=1}^m \sum_{j=1}^n |a_{ij}|,$$

2. l_2 norma (euklidska, Frobeniusova)

$$\|A\|_2 := \|A\|_F = (\text{tr}(A^*A))^{1/2} = \left(\sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2 \right)^{1/2},$$

3. l_∞ norma

$$\|A\|_\infty := \max_{\substack{i=1, \dots, m \\ j=1, \dots, n}} |a_{ij}|.$$

tr je oznaka za trag matrice – zbroj dijagonalnih elemenata matrice.

S druge strane, matrične norme možemo dobiti kao operatorske norme iz odgovarajućih vektorskih korištenjem definicije

$$\|A\| = \max_{x \neq 0} \frac{\|Ax\|}{\|x\|} \text{ (ili } = \max_{\|x\|=1} \|Ax\|). \quad (1.1)$$

Kad se uvrste odgovarajuće vektorske norme u (1.1), dobivamo

1. matrična 1-norma, "maksimalna stupčana norma"

$$\|A\|_1 = \max_{j=1, \dots, n} \sum_{i=1}^m |a_{ij}|,$$

2. matrična 2-norma, spektralna norma

$$\|A\|_2 = (\rho(A^*A))^{(1/2)} = \sigma_{\max}(A),$$

3. matrična ∞ -norma, "maksimalna retčana norma"

$$\|A\|_\infty = \max_{i=1, \dots, m} \sum_{j=1}^n |a_{ij}|,$$

pri čemu je ρ oznaka za spektralni radijus kvadratne matrice (maksimalna po apsolutnoj vrijednosti svojstvena vrijednost)

$$\rho(B) = \max\{|\lambda| \mid \det(B - \lambda I) = 0\}, \quad (B \text{ kvadratna!}),$$

a σ je standardna oznaka za tzv. singularnu vrijednost matrice. Detaljnu definiciju što je to singularna vrijednost, dobit ćete u poglavlju koje će se baviti dekompozicijom singularnih vrijednosti.

Poglavlje 2

Dekompozicija matrice na singularne vrijednosti (SVD)

2.1 SVD

Jedna od najkorisnijih dekompozicija i s teorijske strane (za dokazivanje) i s praktične strane, je dekompozicija singularnih vrijednosti (engl. singular value decomposition) ili, skraćeno, SVD.

Sljedeći teorem pokazuje da za svaku matricu postoji njezina dekompozicija singularnih vrijednosti.

Teorem 2.1.1. (SVD)

Neka su m i n prirodni brojevi te A proizvoljna $n \times m$ realna matrica. Tada postoji dekompozicija

$$A = U\Sigma V^T,$$

gdje su $U \in \mathbb{R}^{n \times n}$ i $V \in \mathbb{R}^{m \times m}$ ortogonalne matrice, a $\Sigma \in \mathbb{R}^{n \times m}$ dijagonalna matrica,

$$\Sigma = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_m),$$

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_m \geq 0.$$

Dokaz: Pretpostavka $n \geq m$ nije restrikcija: u suprotnom primijenimo teorem na A^T . Dokaz se provodi indukcijom po n i m . Pretpostavljamo da postoji dekompozicija singularnih vrijednosti od matrice dimenzije $(n-1) \times (m-1)$ i dokazujemo da tada postoji i za $n \times m$ matricu. Pretpostavljamo da je $A \neq 0$, jer u protivnom je

$\Sigma = 0$, a U i V su proizvoljne unitarne matrice.

Baza indukcije je za $m = 1$, jer je $n \geq m$.
Definirajmo jedinični vektor

$$u = \frac{A}{\|A\|_2}$$

te ga dopunimo matricom \widehat{U} tako da $U = (u, \widehat{U})$ bude ortogonalna $n \times n$ matrica. Sada za

$$U = (u, \widehat{U}), \quad \Sigma = \begin{pmatrix} \|A\|_2 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \quad V = 1, \quad \text{gdje je } \Sigma \in \mathbb{R}^{n \times 1}$$

slijedi $A = U\Sigma V^T$.

Za korak indukcije, izaberemo vektor v , takav da je $\|v\|_2 = 1$ i na njemu se baš dostiže maksimum 2-norme za A , tj. vrijedi

$$\|A\|_2 = \max_{\|x\|_2=1} \|Ax\|_2 = \|Av\|_2.$$

Definiramo jedinični vektor

$$u = \frac{Av}{\|Av\|_2}.$$

Vektore u i v dopunimo matricama \widetilde{U} , odnosno \widetilde{V} , tako da

$$U_0 = (u, \widetilde{U}), \quad V_0 = (v, \widetilde{V})$$

budu redom $n \times n$ i $m \times m$ unitarne matrice. Sad možemo pisati

$$U_0^T A V_0 = \begin{pmatrix} u^T \\ \widetilde{U}^T \end{pmatrix} A(v, \widetilde{V}) = \begin{pmatrix} u^T A v & u^T A \widetilde{V} \\ \widetilde{U}^T A v & \widetilde{U}^T A \widetilde{V} \end{pmatrix}. \quad (2.1)$$

Po definiciji vektora u i v je

$$u^T A v = \frac{v^T A^T}{\|Av\|_2} A v = \frac{\|Av\|_2^2}{\|Av\|_2} = \|Av\|_2 = \|A\|_2 := \sigma.$$

Nadalje, zbog ortogonalnosti stupaca unitarne matrice U_0 , svi stupci matrice \widetilde{U} su okomiti na vektor u , pa je $\widetilde{U}^T u = 0$. Onda je i

$$\widetilde{U}^T A v = \widetilde{U}^T u \|Av\|_2 = 0.$$

Tvrdimo i da je $u^T A \tilde{V} = 0$. Označimo s

$$A_1 = U_0^T A V_0, \quad w^T = u^T A \tilde{V}, \quad B = \tilde{U}^T A \tilde{V}.$$

Relacija (2.1) tada glasi

$$A_1 = \begin{pmatrix} \sigma & w^T \\ 0 & B \end{pmatrix}.$$

Zbog unitarne invarijantosti 2-norme je

$$\sigma = \|A\|_2 = \|U_0^T A V_0\|_2 = \|A_1\|_2.$$

S druge strane, za proizvoljni vektor $z \neq 0$ vrijedi

$$\|A_1\|_2 = \max_{x \neq 0} \frac{\|A_1 x\|_2}{\|x\|_2} \geq \frac{\|A_1 z\|_2}{\|z\|_2},$$

odnosno

$$\|A_1\|_2 \|z\|_2 \geq \|A_1 z\|_2.$$

Izaberimo

$$z = \begin{pmatrix} \sigma \\ w \end{pmatrix}.$$

Onda je

$$\begin{aligned} \|A_1\|_2^2 \|z\|_2^2 &= \|A_1\|_2^2 (\sigma^2 + \|w\|_2^2) \geq \|A_1 z\|_2^2 = \left\| A_1 \begin{pmatrix} \sigma \\ w \end{pmatrix} \right\|_2^2 = \\ &= \left\| \begin{pmatrix} \sigma & w^T \\ 0 & B \end{pmatrix} \begin{pmatrix} \sigma \\ w \end{pmatrix} \right\|_2^2 = (\sigma^2 + w^T w)^2 + \|B w\|_2^2 \geq (\sigma^2 + \|w\|_2^2)^2, \end{aligned}$$

pa vidimo da je

$$\|A_1\|_2^2 (\sigma^2 + \|w\|_2^2) \geq (\sigma^2 + \|w\|_2^2)^2.$$

Dijeljenjem s $(\sigma^2 + \|w\|_2^2)$ dobivamo

$$\sigma^2 = \|A\|_2^2 = \|A_1\|_2^2 \geq \sigma^2 + \|w\|_2^2,$$

što je moguće samo za $w = 0$.

Drugim riječima, vrijedi

$$U_0^T A V_0 = \begin{pmatrix} \sigma & 0 \\ 0 & B \end{pmatrix}.$$

Sada možemo iskoristiti pretpostavku indukcije na matricu B,

$$B = U_1 \Sigma_1 V_1^T$$

pa dobivamo

$$U_0^T A V_0 = \begin{pmatrix} \sigma & 0 \\ 0 & U_1 \Sigma_1 V_1^T \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & U_1 \end{pmatrix} \begin{pmatrix} \sigma & 0 \\ 0 & \Sigma_1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & V_1 \end{pmatrix}^T,$$

odakle odmah slijedi tvrdnja.

Ako želimo biti potpuno precizni, treba još silazno poredati singularne vrijednosti. To se postiže primjenom matrica permutacije P_1 reda m , i P_2 reda n , tako da matrica

$$\Sigma := P_2^T \begin{pmatrix} \sigma & 0 \\ 0 & \Sigma_1 \end{pmatrix} P_1$$

ima silazno poredanu dijagonalu. Lako se vidi da je

$$P_2 = \begin{pmatrix} P_1 & 0 \\ 0 & I_{n-m} \end{pmatrix},$$

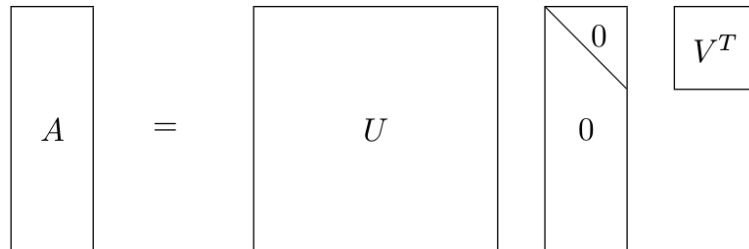
gdje je I_{n-m} jedinična matrica reda $n - m$. Na kraju, znamo da su P_1 i P_2 unitarne matrice, a produkt unitarnih matrica je opet unitarna matrica, pa su

$$U := U_0 \begin{pmatrix} 1 & 0 \\ 0 & U_1 \end{pmatrix} P_2, \quad V := V_0 \begin{pmatrix} 1 & 0 \\ 0 & V_1 \end{pmatrix} P_1,$$

unitarne matrice i vrijedi $A = U \Sigma V^T$.

■

Ilustrirajmo SVD:

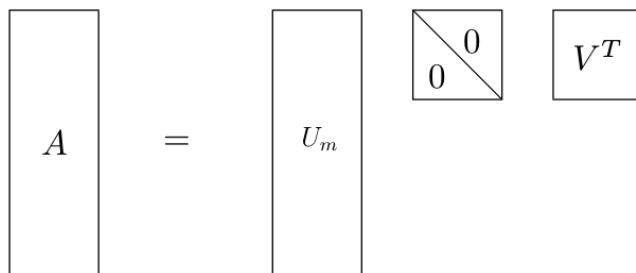


Slika 2.1: $A \in \mathbb{R}^{n \times m}$, $U \in \mathbb{R}^{n \times n}$, $\Sigma \in \mathbb{R}^{n \times m}$, $V^T \in \mathbb{R}^{m \times m}$.

Ako U zapišemo kao $U = (U_m U_m^\perp)$, gdje je $U_m \in \mathbb{R}^{n \times m}$, a Σ zapišemo kao $\begin{pmatrix} \Sigma_m \\ 0 \end{pmatrix}$ gdje je $\Sigma_m \in \mathbb{R}^{m \times m}$, tada dobijemo tanki SVD.

$$A = U_m \Sigma_m V^T,$$

Ilustrirajmo sada tanki SVD:



Slika 2.2: $A \in \mathbb{R}^{n \times m}$, $U_m \in \mathbb{R}^{n \times m}$, $\Sigma_m \in \mathbb{R}^{m \times m}$, $V^T \in \mathbb{R}^{m \times m}$.

Uz navedene oznake lako se vidi da vrijedi

$$A = U_m \Sigma_m V^T = \sum_{i=1}^m \sigma_i u_i v_i^T. \quad (2.2)$$

$$A = \sum_{i=1}^m \sigma_i u_i v_i^T = \left| \begin{array}{c} \text{---} \\ | \end{array} \right. + \left| \begin{array}{c} \text{---} \\ | \end{array} \right. + \dots$$

Definicija 2.1.2. *Stupce matrice $U_m = (u_1, \dots, u_m)$ nazivamo lijevi singularni vektori, a stupce matrice $V = (v_1, \dots, v_m)$ nazivamo desni singularni vektori, brojeve σ_i nazivamo singularne vrijednosti.*

Do sada smo inverz matrice definirali isključivo za kvadratne matrice, a sada želimo taj pojam proširiti na pravokutne matrice i matrice koje nisu punog ranga. Postoje tri ekvivalentna načina na koji se to može načiniti, u nastavku navodimo samo Penroseovu definiciju.

Definicija 2.1.3. Neka je $A \in \mathbb{C}^{n \times m}$, njezin generalizirani inverz je jedinstvena matrica A^+ , takva da vrijedi

$$\begin{aligned} AA^+A &= A, & A^+AA^+ &= A^+, \\ (AA^+)^* &= AA^+, & (A^+A)^* &= A^+A. \end{aligned}$$

Generalizirani inverz ponekad nazivamo i Moore-Penroseov pseudoinverz. Neka je $A = U\Sigma V^T$. Tada Moore-Penrose pseudoinverz za singularnu matricu A možemo izračunati kao $A^+ = V\Sigma^+U^T$, gdje je $\Sigma^+ = \begin{pmatrix} \Sigma_+ & 0 \\ 0 & 0 \end{pmatrix}^+ = \begin{pmatrix} \Sigma_+^{-1} & 0 \\ 0 & 0 \end{pmatrix}$, a Σ_+ dijagonalna matrica sa ne-nul singularnim vrijednostima na dijagonali.

Teorem 2.1.4. Neka je $A = U_m \Sigma_m V^T$ dekompozicija singularnih vrijednosti (SVD) realne matrice A tipa $n \times m$, $n \geq m$. Zapišimo matrice U_m i V iz SVD-a od A u stupčanom obliku

$$U_m = (u_1, \dots, u_m), \quad V = (v_1, \dots, v_m).$$

Onda matricu A (preko SVD-a) možemo pisati kao zbroj matrica ranga 1

$$A = U_m \Sigma_m V^T = \sum_{i=1}^m \sigma_i u_i v_i^T.$$

Matricu A_k , istog tipa kao i A , ranga $\text{rang}(A_k) \leq k < m$, koja je po 2-normi najbliža matrici A , možemo zapisati kao

$$A_k = U_m \Sigma_k V^T = \sum_{i=1}^k \sigma_i u_i v_i^T$$

pri čemu je

$$\Sigma_k = \text{diag}(\sigma_1, \dots, \sigma_k, 0, \dots, 0).$$

Pritom je

$$\|A - A_k\|_2 = \sigma_{k+1}$$

najmanja udaljenost između A i svih matrica ranga najviše k .

Dokaz: Ovo je zapravo, tvrdnja o najboljoj aproksimaciji matrice A matricom nižeg ranga i njihovoj udaljenosti u 2-normi. Prema konstrukciji, A_k ima rang najviše k i vrijedi

$$\|A - A_k\| = \left\| \sum_{i=k+1}^m \sigma_i u_i v_i^T \right\|_2 = \|U_m \text{diag}(0, \dots, 0, \sigma_{k+1}, \dots, \sigma_m) V^T\|_2 = \sigma_{k+1}.$$

Ostaje pokazati da je to i najbliža matrica ranga najviše k matrici A . Neka je B bilo koja matrica istog tipa za koju vrijedi $\text{rang}(B) \leq k$. Onda njezin nul-potprostor ima dimenziju barem $m - k$. S druge strane, potprostor razapet vektorima v_1, \dots, v_{k+1} ima dimenziju $k + 1$, pa sigurno postoji netrivialni vektor koji se nalazi u njegovom presjeku s nul-potprostorom od B . Neka je h pripadni jedinični vektor koji se nalazi u presjeku ta dva potprostora. Onda je $Bh = 0$ i

$$\|A - B\|_2 \geq \|(A - B)h\|_2 = \|Ah\|_2 = \|U_m \Sigma_m V^T h\|_2 = \|\Sigma_m V^T h\|_2 =$$

kako je h linearna kombinacija vektora v_1, \dots, v_{k+1} vrijedi $h = V_{k+1}x$ pri čemu je $\|x\| = 1$ pa dobivamo

$$\begin{aligned} \|\Sigma_m V^T V_{k+1}x\|_2 &= \left\| \Sigma_m \begin{pmatrix} I_{k+1} \\ 0 \end{pmatrix} x \right\|_2 = \\ &= \left\| \begin{pmatrix} \sigma_1 & 0 & \dots & 0 \\ 0 & \sigma_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sigma_m \end{pmatrix} \begin{pmatrix} I_{k+1} \\ 0 \end{pmatrix} x \right\|_2 = \left\| \begin{pmatrix} \sigma_1 & 0 & \dots & 0 \\ 0 & \sigma_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sigma_{k+1} \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 0 \end{pmatrix} x \right\|_2 = \end{aligned}$$

$$\sqrt{(\sigma_1 x_1)^2 + \dots + (\sigma_{k+1} x_{k+1})^2} \geq \sigma_{k+1} \sqrt{x_1^2 + \dots + x_{k+1}^2} = \sigma_{k+1} \|x\|_2 = \sigma_{k+1}.$$

■

2.2 Neke primjene dekompozicije matrice na singularne vrijednosti

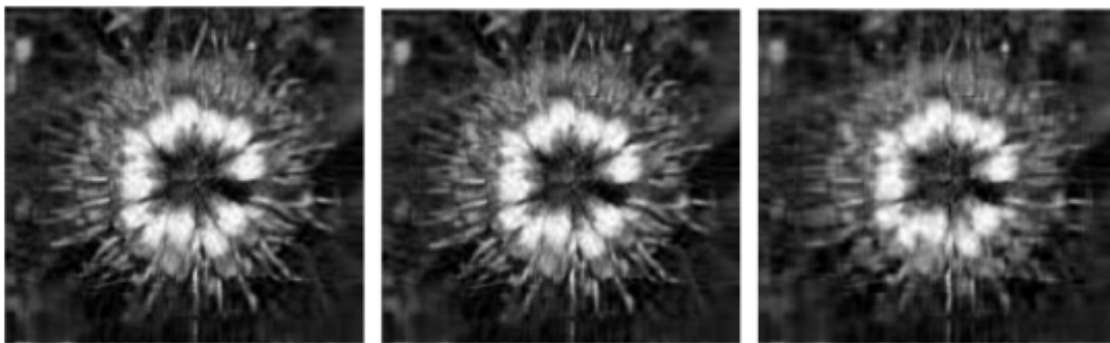
Neka je dana $n \times m$ matrica A s elementima $a_{i,j} \in [0, 1]$. Zamislimo da A predstavlja $m \times n$ grayscale (crno-bijelu) sliku gdje broj na (i, j) mjestu u matrici predstavlja intenzitet svjetlosti piksela na mjestu (i, j) na slici. Ako uzmemo da vrijednost nula

predstavlja crno, a jedan bijelo, zanimljivo je pitanje što će se dogoditi kad napravimo $A = U\Sigma V^T$, a onda u matrici Σ posljednjih nekoliko ne-nul elemenata stavimo na nulu te promotrimo dobivenu sliku A_k , gdje je $A_k = U\Sigma_k V^T = \sum_{i=1}^k \sigma_i u_i v_i^T$, a $\Sigma_k = \text{diag}(\sigma_1, \dots, \sigma_k, 0, \dots, 0)$.



Slika 2.3: Testna slika A dimenzije 376×350 u JPG formatu zauzima 129 kb.

Preciznije, neka je $A = U\Sigma V^T$, SVD matrice A . Prema teoremu 2.1.4, vrijedi da je $A_k = \sum_{i=1}^k \sigma_i u_i v_i^T$ najbolja aproksimacija matrice A matricom ranga k , u smislu $\|A - A_k\| = \sigma_{k+1}$. Da bismo spremili podatke u_1, \dots, u_k te $\sigma_1 v_1, \dots, \sigma_k v_k$ iz kojih možemo dobiti matricu A_k potrebno je pamtit i samo $(m+n) \cdot k$ brojeva budući da su u_i i v_i vektori iz \mathbb{R}^m odnosno iz \mathbb{R}^n . S druge strane, za spremanje matrice A potrebno je pamtit i svih $m \cdot n$ brojeva. Stoga, koristimo A_k kao kompresiju slike A .



Slika 2.4: Kompresija A_k s a) $k = 50$, b) $k = 25$, c) $k = 15$, a zauzimaju redom 35 kb, 18 kb te 11 kb.

Prethodne simulacije su rađene u programskom paketu Matlab. S obzirom na jednostavnost programske izvedbe u nastavku prilažemo i pojednostavljeni kod kojim se čitatelj i sam može uvjeriti u korisnost i elegantnost ove primjene.

```
1 A = imread('test.png');  
2 A = A(:,:,1);  
3 A = double(A);  
4 [U,S,V] = svd(A);  
5 k = 25;  
6 Ak = U(:,1:k)*S(1:k,1:k)*V(:,1:k)';  
7 image(Ak) colormap(gray(256));
```

Poglavlje 3

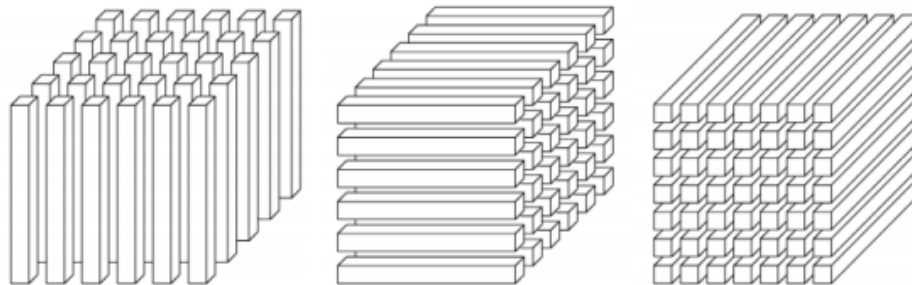
Tenzori

3.1 Uvod

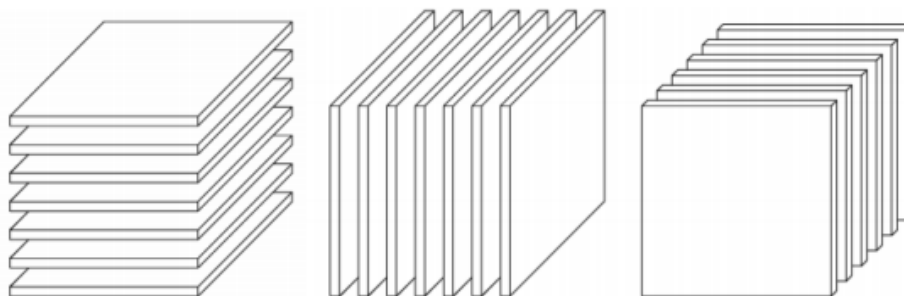
Do sada smo razmatrali linearnu algebru, gdje su glavni objekti vektori i matrice. Njih možemo smatrati kao jednodimenzionalna i dvodimenzionalna polja podataka. Tenzor je višedimenzionalni ekvivalent vektora i matrice i možemo ga predstaviti višedimenzionalnim poljem brojeva. Red tenzora jednak je broju indeksa potrebnih za indeksiranje njegovih elemenata. Primjerice, skalar možemo smatrati tenzorom reda 0, vektor tenzorom reda 1, a matricu tenzorom reda 2. Tenzori reda većeg ili jednakog 3 nazivaju se tenzorima višeg reda.

Preciznije, tenzor reda N je element tenzorskog produkta N vektorskih prostora. Red tenzora $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ je N . Elemente od \mathcal{A} označavamo s $\mathcal{A}_{i_1 i_2 \dots i_n}$, gdje su $1 \leq i_n \leq I_n, n = 1, 2, \dots, N$.

Svaki indeks u tenzoru nazivamo mod, a dimenzija pripadnog moda je broj različitih vrijednosti koje taj indeks može poprimiti. Unutar tenzora možemo indeksirati podtenzore ograničavanjem pojedinih indeksa. Primjerice, za tenzor $\mathcal{A} = (a_{i_1 i_2 i_3}) \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ reda 3 možemo fiksiranjem indeksa u modu 1, 2 ili 3 definirati podtenzore kao $\mathcal{A}_{i_1=n} = \mathcal{A}_{n::}$, $\mathcal{A}_{i_2=n} = \mathcal{A}_{:n:}$ ili $\mathcal{A}_{i_3=n} = \mathcal{A}_{::n}$. Primjetimo da su u tom slučaju dva indeksa slobodna pa su podtenzori zapravo novi tenzori reda 2, odnosno matrice. Posebno za slučaj tenzora reda 3 te podtenzore nazivamo horizontalnim, lateralnim i frontalnim odsječcima. Također možemo definirati i vektore po modu n na način da fiksiramo sve indekse osim jednoga, npr. vektor u modu 1 se dobije kao $\mathcal{A}_{:i_2 i_3}$. Analogno za tenzor reda N možemo definirati podtenzore reda manjeg ili jednakog N . Na sljedećoj slici su prikazani primjeri podtenzora za tenzor reda 3.



Slika 3.1: Primjeri podtenzora. Vektori tenzora u: modu 1, modu 2 i modu 3.



Slika 3.2: Primjeri podtenzora. Horizontalni, lateralni i frontalni odsjeci.

Često je korisno prikazati tenzor u obliku matrice. Stoga definiramo matricu tenzora po modu n kao matricu $A_{(n)}$ koja se dobije tako da vektore iz moda n složimo kao stupce u matricu $A_{(n)}$. Poredak kojim se vektori iz moda n preslikavaju u stupce nije bitan, dok god se isti poredak koristi u svim izračunima.

Matricizacija tenzora (eng. unfold) u modu n od \mathcal{A} označava se s

$$A_{(n)} \in \mathbb{R}^{I_n \times (I_1 \times \dots \times I_{n-1} \times I_{n+1} \times \dots \times I_N)}.$$

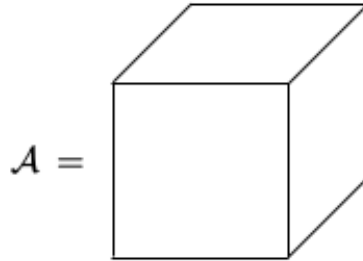
Tenzorski produkt u modu n tenzora $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ s matricom $U \in \mathbb{R}^{J_n \times I_n}$, označen s $\mathcal{A} \times_n U$, je

$$(I_1 \times \dots \times I_{n-1} \times J_n \times I_{n+1} \times \dots \times I_N)$$

tenzor čiji elementi su zadani s

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

Zbog jednostavnosti ograničavamo se na tenzore reda 3, $\mathcal{A} = (a_{ijk}) \in \mathbb{R}^{I_1 \times I_2 \times I_3}$. Takav tenzor možemo vizualizirati na sljedeći način:



Uvedimo sada osnovne pojmove za tenzor reda 3, $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$. Dimenzije tenzora $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ su: I_1 , I_2 i I_3 .

3.2 Osnovni koncepti tenzora

Prvo definiramo skalarni produkt dvaju tenzora istog reda i jednakih dimenzija:

Definicija 3.2.1. *Skalarni produkt dvaju tenzora istih dimenzija $\mathcal{A}, \mathcal{B} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ je suma umnožaka njihovih elemenata, tj.*

$$\langle \mathcal{A}, \mathcal{B} \rangle = \sum_{i,j,k} a_{ijk} b_{ijk}. \quad (3.1)$$

Pripadna norma je

Definicija 3.2.2. *Norma tenzora $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ je kvadratni korijen sume svih kvadrata njegovih elemenata, tj.*

$$\|\mathcal{A}\| = \langle \mathcal{A}, \mathcal{A} \rangle^{1/2} = \left(\sum_{i,j,k} a_{ijk}^2 \right)^{1/2}.$$

Ako gledamo gornje definicije na matricama (dvodimenzionalnim tenzorima) vidimo da je to Frobeniusova norma.

Sljedeće što definiramo je množenje u modu 1 matrice s tenzorom.

Definicija 3.2.3. Množenje u modu 1 tenzora $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ matricom $U \in \mathbb{R}^{I_0 \times I_1}$ u oznaci $\mathcal{A} \times_1 U$, je $I_0 \times I_2 \times I_3$ tenzor dan s

$$(\mathcal{A} \times_1 U)(j, i_2, i_3) = \sum_{k=1}^{I_1} u_{j,k} a_{k,i_2,i_3}.$$

Uspredbe radi, uočimo da za množenje matrica vrijedi

$$A \times_1 U = UA, \quad (UA)(i, j) = \sum_{k=1}^l u_{i,k} a_{k,j}.$$

Znamo da je množenje matrica ekvivalentno množenju svakog stupca iz \mathcal{A} matricom U . Isto vrijedi i za množenje tenzora matricom u modu 1. Svi stupčani vektori iz tenzora \mathcal{A} se množe matricom U .

Slično množenje u modu 2 tenzora s matricom V

$$(\mathcal{A} \times_2 V)(i_1, j, i_3) = \sum_{k=1}^{I_2} v_{j,k} a_{i_1,k,i_3}$$

znači da su svi retci tenzora pomnoženi s matricom V . Opet uočavamo da je množenje u modu 2 matrice s V ekvivalentno matricnom množenju s V^T zdesna,

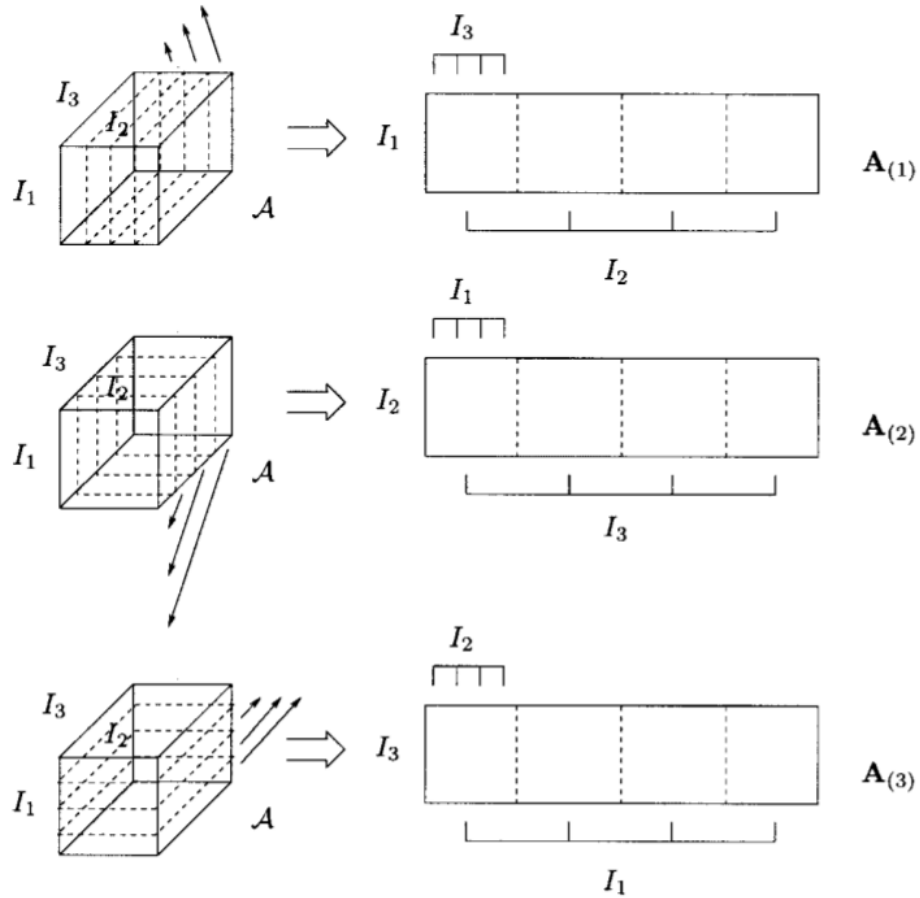
$$A \times_2 V = AV^T;$$

Množenje u modu 3 se dobije analogno.

Ponekad je zgodno matricizirati tenzor u matricu.

Matricizacija tenzora \mathcal{A} je definirana kroz tri moda (koristeći MATLAB-ovu notaciju):

$$\begin{aligned} \text{unfold}_1(\mathcal{A}) := A_{(1)} &:= (\mathcal{A}(:, 1, :), \mathcal{A}(:, 2, :), \dots, \mathcal{A}(:, I_2, :)) && \in \mathbb{R}^{I_1 \times I_2 I_3}, \\ \text{unfold}_2(\mathcal{A}) := A_{(2)} &:= (\mathcal{A}(:, :, 1)^T, \mathcal{A}(:, :, 2)^T, \dots, \mathcal{A}(:, :, I_3)^T) && \in \mathbb{R}^{I_2 \times I_1 I_3}, \\ \text{unfold}_3(\mathcal{A}) := A_{(3)} &:= (\mathcal{A}(1, :, :)^T, \mathcal{A}(2, :, :)^T, \dots, \mathcal{A}(I_1, :, :)^T) && \in \mathbb{R}^{I_3 \times I_1 I_2}. \end{aligned}$$



Slika 3.3: Matricizacija $(I_1 \times I_2 \times I_3)$ tenzora \mathcal{A} na $(I_1 \times I_2 I_3)$ matricu $\mathbf{A}_{(1)}$, $(I_2 \times I_3 I_1)$ matricu $\mathbf{A}_{(2)}$ i $(I_3 \times I_1 I_2)$ matricu $\mathbf{A}_{(3)}$ ($I_1 = I_2 = I_3 = 4$).

Vrijedi i sljedeće:

1. Stupčani vektori od \mathcal{A} su stupčani vektori od $\mathbf{A}_{(1)}$.
2. Retci od \mathcal{A} su stupčani vektori od $\mathbf{A}_{(2)}$.
3. Prostorni vektori od \mathcal{A} su stupčani vektori od $\mathbf{A}_{(3)}$.

Unfold_1 od \mathcal{A} je ekvivalentan dijeljenju tenzora na odsječke $\mathcal{A}(:, i, :)$ (koji su matrice) i slaže ih u dugu matricu $\mathbf{A}_{(1)}$.

Primjer 3.2.4. Neka je $\mathcal{B} \in \mathbb{R}^{3 \times 3 \times 3}$ tenzor, definiran u MATBLABu s

$B(:,:,1) =$	$B(:,:,2) =$	$B(:,:,3) =$
1 2 3	11 12 13	21 22 23
4 5 6	14 15 16	24 25 26
7 8 9	17 18 19	27 28 29

Tada matricizacija u modu 3 $\text{unfold}_3(\mathcal{B})$ daje

```
>> B3 = unfold(B,3)
```

b3 =	1	2	3	4	5	6	7	8	9	
	11	12	13	14	15	16	17	18	19	
	21	22	23	24	25	26	27	28	29	■

Matlab kod za matricizaciju(unfold)

```

1 function [A_mod] = unfold(A, mod)
2
3 A_mod = [];
4 if (max(size(A)) > 3)
5     disp('Funkcija radi samo za tenzor reda 3');
6 else
7     [m, n, p] = size(A);
8
9     switch(mod)
10        case 1
11            for i=1:n
12                A_mod = [A_mod, squeeze(A(:, i, :))];
13            end
14
15        case 2
16            for i=1:p
17                A_mod = [A_mod, A(:, :, i)'];
18            end
19
20        case 3
21
22            if n == 1
23                for i=1:m

```

```

24         B=squeeze(A(i, :, :));
25         A_mod=[A_mod, B];
26     end
27 else
28     for i=1:m
29         A_mod=[A_mod, squeeze(A(i, :, :))'];
30     end
31 end
32
33 otherwise
34     disp('Tensor je reda 3 i mora biti mod 1, 2 ili 3');
35 end
36 end
37
38 end

```

Kao što nam je korisno prikazati tenzor u obliku matrice tako nam je ponekad korisno tenzorificirati matricu odnosno matricu pretvoriti u tenzor. Tenzorifikacija je inverz matricizacije odnosno operator fold je inverz operatora unfold pa vrijedi

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

Matlab kod za tenzorifikaciju(fold)

```

1 function [A_tens] = fold(A_mod, mod, dim_m, dim_n, dim_p)
2
3 for br=1:1
4 switch(mod)
5     case 1
6         [m, n]=size(A_mod);
7         if (m~=dim_m || n~=dim_n*dim_p)
8             disp('dimenzije tenzora nisu dobro zadane (mod 1)')
9             ;
10            A_tens = [];
11            break
12        else
13            A_tens=zeros(dim_m, dim_n, dim_p);

```

```

14         for i=1:dim_n
15             A_tens(:,i,:)=A_mod(:, (i-1)*dim_p+1 : i*
                dim_p);
16
17         end
18     end
19
20     case 2
21         [m, n]=size(A_mod);
22         if (m~=dim_n || n~=dim_m*dim_p)
23             disp('dimenzije tenzora nisu dobro zadane (mod 2)');
24             A_tens=[];
25             break
26         else
27             A_tens=zeros(dim_m, dim_n, dim_p);
28             for i=1:dim_p
29                 A_tens(:, :, i)=A_mod(:, (i-1)*dim_m+1:i*dim_m)';
30             end
31         end
32
33     case 3
34         [m, n]=size(A_mod);
35         if (m~=dim_p || n~=dim_m*dim_n)
36             disp('dimenzije tenzora nisu dobro zadane (mod 3)');
37             A_tens=[];
38             break
39         else
40             A_tens=zeros(dim_m, dim_n, dim_p);
41             for i=1:dim_m
42                 A_tens(i, :, :)=A_mod(:, (i-1)*dim_n+1:i*dim_n)';
43             end
44         end
45     otherwise
46         disp('Tensor je reda 3 i mora biti mod 1, 2 ili 3');
47
48     end
49     end
50     end

```

Korištenjem unfold-fold operacija, možemo formulirati matricno množenje ekviva-

lentno u modu i tenzorskom množenju:

$$\mathcal{A} \times_i U = \text{fold}_i(U \text{ unfold}_i(\mathcal{A})) = \text{fold}_i(UA_{(i)}). \quad (3.2)$$

Dokaz: BSO neka je $i = 1$

$$\begin{aligned} \text{fold}_1(UA_{(1)})(i_1, i_2, i_3) &= \text{fold}_1((UA_{(1)})(i_1, (i_2 - 1) \cdot I_3 + i_3)) \\ &= \text{fold}_1\left(\sum_{j=1}^{I_1} u_{i_1, j} A_{(1)}(j, (i_2 - 1) \cdot I_3 + i_3)\right) = \sum_{j=1}^{I_1} u_{i_1, j} a_{j, i_2, i_3} = (\mathcal{A} \times_1 U)(i_1, i_2, i_3). \end{aligned}$$

■

Matlab kod za tenzorsko mnozenje tmul

```

1 function [B] = tmul(A_tens, U, mod)
2
3 A_mod = unfold(A_tens, mod);
4
5 [r, t] = size(U);
6
7 [k, l] = size(A_mod);
8
9 if t==k
10     [m, n, p] = size(A_tens);
11     switch(mod)
12         case 1
13             B = fold(U * A_mod, mod, r, n, p);
14         case 2
15             B = fold(U * A_mod, mod, m, r, p);
16         case 3
17             B = fold(U * A_mod, mod, m, n, r);
18     end
19 else
20     disp('Dimenzije se ne poklapaju');
21     B=[];
22 end
23
24 end

```

Iz definicije (3.2.3) slijedi da mod i i mod j množenje komutira ako je $i \neq j$:

$$(\mathcal{A} \times_i F) \times_j G = (\mathcal{A} \times_j G) \times_i F = \mathcal{A} \times_i F \times_j G.$$

Dokaz:

BSO neka je $i = 1$, a $j = 2$. Po definiciji (3.2.3) vrijedi:

$$\begin{aligned} (\mathcal{A} \times_1 F)(i_1, i_2, i_3) &= \sum_{k=1}^{I_1} f_{i_1,k} a_{k,i_2,i_3}, \\ (\mathcal{A} \times_2 G)(i_1, i_2, i_3) &= \sum_{k=1}^{I_2} g_{i_2,k} a_{i_1,k,i_3}. \end{aligned}$$

Pa vrijedi:

$$((\mathcal{A} \times_1 F) \times_2 G)(i_1, i_2, i_3) = \sum_{l=1}^{I_2} g_{i_2,l} (\mathcal{A} \times_1 F)_{i_1,l,i_3} = \sum_{l=1}^{I_2} g_{i_2,l} \left(\sum_{k=1}^{I_1} f_{i_1,k} a_{k,l,i_3} \right)$$

Kako $g_{i_2,l}$ ne ovisi o k , može ući pod drugu sumu.

$$= \sum_{l=1}^{I_2} \left(\sum_{k=1}^{I_1} g_{i_2,l} f_{i_1,k} a_{k,l,i_3} \right) = \sum_{k=1}^{I_1} \left(\sum_{l=1}^{I_2} g_{i_2,l} f_{i_1,k} a_{k,l,i_3} \right)$$

Sada $f_{i_1,k}$ ne ovisi o l pa može izaći izvan druge sume.

$$= \sum_{k=1}^{I_1} f_{i_1,k} \left(\sum_{l=1}^{I_2} g_{i_2,l} a_{k,l,i_3} \right) = \sum_{k=1}^{I_1} f_{i_1,k} (\mathcal{A} \times_2 G)_{k,i_2,i_3} = ((\mathcal{A} \times_2 G) \times_1 F)(i_1, i_2, i_3).$$

■

Dvostruko množenje u modu i zadovoljava identitet

$$(\mathcal{A} \times_i F) \times_i G = \mathcal{A} \times_i (GF).$$

Što se lako dokaže koristeći (3.2):

$$\begin{aligned} (\mathcal{A} \times_i F) \times_i G &= (\text{fold}_i(F(\text{unfold}_i(\mathcal{A})))) \times_i G \\ &= \text{fold}_i(G(\text{unfold}_i(\text{fold}_i(F(\text{unfold}_i(\mathcal{A})))))) \\ &= \text{fold}_i(GF \text{ unfold}_i(\mathcal{A})) = \mathcal{A} \times_i (GF). \end{aligned}$$

3.3 Tenzorski SVD (HOSVD)

Matrični SVD može se generalizirati na više načina kako bi dobili tenzorski. Mi promatramo jedan koji se često naziva "Higher order SVD" (HOSVD).

Vidjeli smo da matrice možemo promatrati kao tenzore reda 2. Koristeći tenzorsku notaciju jednadžbu (2.1) iz Teorema 2.1.1 (SVD) sada možemo zapisati kao

$$A = U\Sigma V^T = \Sigma \times_1 U \times_2 V = S \times_1 U^{(1)} \times_2 U^{(2)}$$

isto vrijedi i za tenzore višeg reda. O tome nam više govori sljedeći teorem.

Teorem 3.3.1. (HOSVD) Tenzor $\mathcal{A} \in \mathbb{R}^{l \times m \times n}$ može se zapisati kao

$$\mathcal{A} = \mathcal{S} \times_1 U^{(1)} \times_2 U^{(2)} \times_3 U^{(3)}, \quad (3.3)$$

gdje su $U^{(1)} \in \mathbb{R}^{l \times l}$, $U^{(2)} \in \mathbb{R}^{m \times m}$ i $U^{(3)} \in \mathbb{R}^{n \times n}$ ortogonalne matrice. \mathcal{S} je tenzor istih dimenzija kao \mathcal{A} . Svaka dva odsječka tenzora \mathcal{S} su ortogonalna u smislu skalarnog produkta:

$$\langle \mathcal{S}(i, :, :), \mathcal{S}(j, :, :) \rangle = \langle \mathcal{S}(:, i, :), \mathcal{S}(:, j, :) \rangle = \langle \mathcal{S}(:, :, i), \mathcal{S}(:, :, j) \rangle = 0$$

za $i \neq j$. Singularne vrijednosti u modu 1 su definirane s

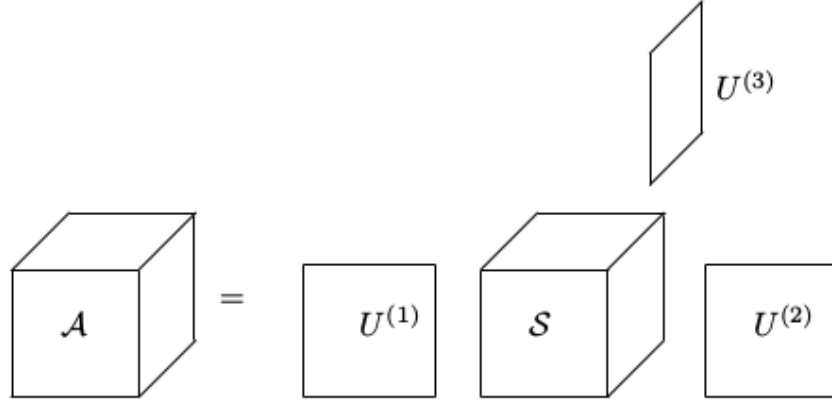
$$\sigma_j^{(1)} = \|\mathcal{S}(j, :, :)\|_F, \quad j = 1, \dots, l$$

i one su sortirane silazno

$$\sigma_1^{(1)} \geq \sigma_2^{(1)} \geq \dots \geq \sigma_l^{(1)}. \quad (3.4)$$

Singularne vrijednosti u ostalim modovima i njihovo sortiranje je analogno.

Ortogonalni tenzor \mathcal{S} nazivamo jezgri tenzor tenzora \mathcal{A} . HOSVD je vizualiziran na sljedećoj slici.



Slika 3.4: Vizualizacija HOSVD-a.

Definicija 3.3.2. *Kroneckerov produkt matrica $A \in \mathbb{R}^{I_1 \times I_2}$ i $B \in \mathbb{R}^{J_1 \times J_2}$ je matrica*

$$A \otimes B = \begin{pmatrix} a_{11}B & \cdots & a_{1I_2}B \\ \vdots & \ddots & \vdots \\ a_{I_11}B & \cdots & a_{I_1I_2}B \end{pmatrix} \in \mathbb{R}^{I_1 J_1 \times I_2 J_2}.$$

Navedimo i neka svojstva Kroneckerovog produkta:

Teorem 3.3.3. *Neka je $A \in \mathbb{R}^{I_1 \times I_2}$, $B \in \mathbb{R}^{J_1 \times J_2}$, $C \in \mathbb{R}^{I_2 \times I_3}$, $D \in \mathbb{R}^{J_2 \times J_3}$. Tada vrijedi*

$$(A \otimes B)(C \otimes D) = AC \otimes BD \in \mathbb{R}^{I_1 J_1 \times I_3 J_3}.$$

Teorem 3.3.4. *Za sve matrice A i B vrijedi $(A \otimes B)^T = A^T \otimes B^T$.*

Teorem 3.3.5. *Ako su $A \in \mathbb{R}^{I_1 \times I_2}$ i $B \in \mathbb{R}^{J_1 \times J_2}$ ortogonalne matrice, tada je i $A \otimes B$ ortogonalna matrica.*

Propozicija 3.3.6. *Neka je $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ i $\mathcal{N} = \{1, \dots, N\}$, ako je $U^{(i)} \in \mathbb{R}^{I_i \times I_i}$ za sve $i \in \mathcal{N}$, tada za svaki $i \in \mathcal{N}$ vrijedi*

$$A_{(i)} = U^{(i)} \cdot S_{(i)} \cdot (U^{(i+1)} \otimes U^{(i+2)} \otimes \cdots \otimes U^{(n)} \otimes U^{(1)} \otimes U^{(2)} \otimes \cdots \otimes U^{(i-1)})^T.$$

Odnosno za tenzor reda 3

$$A_{(1)} = U^{(1)} \cdot S_{(1)} \cdot (U^{(2)} \otimes U^{(3)})^T,$$

$$A_{(2)} = U^{(2)} \cdot S_{(2)} \cdot (U^{(3)} \otimes U^{(1)})^T,$$

$$A_{(3)} = U^{(3)} \cdot S_{(3)} \cdot (U^{(1)} \otimes U^{(2)})^T.$$

U daljnjem ćemo to pisati kao

$$A_{(i)} = U^{(i)} \cdot S_{(i)} \cdot (U^{(i+1)} \otimes U^{(i-1)})^T. \quad (3.5)$$

Više o Kroneckerovom produktu možemo pogledati u [4].

Dokažimo sad Teorem 3.3.1

Dokaz: Pretpostavimo da postoje dva $(l \times m \times n)$ tenzora \mathcal{A} i \mathcal{S} takva da vrijedi

$$\mathcal{S} = \mathcal{A} \times_1 U^{(1)T} \times_1 U^{(2)T} \times_1 U^{(3)T}, \quad (3.6)$$

gdje su $U^{(1)}, U^{(2)}, U^{(3)}$ ortogonalne matrice. Jednadžba (3.3) se po Propoziciji 3.3.6 može napisati u matričnom formatu kao

$$A_{(i)} = U^{(i)} \cdot S_{(i)} \cdot (U^{(i+1)} \otimes U^{(i-1)})^T. \quad (3.7)$$

Sada razmotrimo slučaj gdje je $U^{(i)}$ dobiven iz SVD-a od $A_{(i)}$

$$A_{(i)} = U^{(i)} \Sigma^{(i)} V^{(i)T}, \quad (3.8)$$

u kojem je $V^{(i)}$ ortogonalna i $\Sigma^{(i)} = \text{diag}(\sigma_1^{(i)}, \sigma_2^{(i)}, \dots, \sigma_{I_i}^{(i)})$, gdje je

$$\sigma_1^{(i)} \geq \sigma_2^{(i)} \geq \dots \geq \sigma_{I_i}^{(i)} \geq 0 \quad i \quad I_1 = l, I_2 = m, I_3 = n.$$

Označimo sada s r_i najveći indeks za koji je $\sigma_{r_i}^{(i)} > 0$. Uzevši u obzir da je Kroneckerov faktor u (3.7) ortogonalan (Teorem 3.3.5), iz (3.7) i (3.8) dobivamo da je

$$S_{(i)} = \Sigma^{(i)} \cdot V^{(i)T} \cdot (U^{(i+1)} \otimes U^{(i-1)}).$$

BSO neka je $i = 1$. $A_{(1)}$ je složen od svih stupaca tenzora \mathcal{A} , a to vrijedi i za $S_{(1)}$. To znači da se svi elementi odsječka $\mathcal{S}(i, :, :)$ nalaze u i -tom retku od $S_{(1)}$. Kako je produkt dviju ortogonalnih matrica ortogonalna matrica slijedi da je svaki redak od $S_{(1)}$ produkt retka ortogonalne matrice sa skalarom $\sigma_i^{(1)}$. Skalarni produkt odsječka predstavlja skalarni produkt redaka matrice $S_{(1)}$ koji su međusobno ortogonalni, pa to onda vrijedi i za odsječke, odnosno vrijedi

$$\langle \mathcal{S}(i, :, :), \mathcal{S}(j, :, :) \rangle = \langle \mathcal{S}(:, i, :), \mathcal{S}(:, j, :) \rangle = \langle \mathcal{S}(:, :, i), \mathcal{S}(:, :, j) \rangle = 0 \quad \text{za } i \neq j.$$

Norma odsječka je jednaka 2-normi retka od $S_{(1)}$ pa slijedi da je

$$\sigma_j^{(1)} = \|\mathcal{S}(j, :, :)\|_F, \quad j = 1, \dots, l$$

i

$$\|\mathcal{S}(1, :, :)\| \geq \|\mathcal{S}(2, :, :)\| \geq \dots \geq \|\mathcal{S}(l, :, :)\| \geq 0.$$

Odnosno

$$\sigma_1^{(1)} \geq \sigma_2^{(1)} \geq \dots \geq \sigma_l^{(1)} \geq 0$$

i ako je $r_1 < l$,

$$\|\mathcal{S}(r_1, :, :)\| = \sigma_{r_1+1}^{(1)} = \dots = \|\mathcal{S}(l, :, :)\| = \sigma_l^{(1)} = 0.$$

Konstrukcijom matrica $U^{(i-1)}, U^{(i+1)}$ na isti način kao za $U^{(i)}$, \mathcal{S} zadovoljava sve uvjete HOSVD teorema. Također kao što se vidi iz (3.8), sve matrice $U^{(1)}, U^{(2)}, U^{(3)}$ i tenzor \mathcal{S} koji zadovoljavaju HOSVD teorem mogu se naći pomoću SVD-a matrica $A_{(1)}, A_{(2)}, A_{(3)}$, a \mathcal{S} slijedi iz (3.6).

■

Koristeći tenzorsku dekompoziciju, svaki tenzor reda N možemo prikazati kao produkt

$$\mathcal{A} = \mathcal{S} \times_1 U^{(1)} \times_2 U^{(2)} \times_3 \dots \times_N U^{(N)} \quad (3.9)$$

gdje nam je \mathcal{S} jezgreni tenzor, a U_n , $n = 1, 2, \dots, N$ ortogonalne matrice.

Jednadžba (3.8) i prethodni dokaz zapravo pokazuju kako se može izračunati HOSVD određenog tenzora \mathcal{A} : matrica $U^{(i)}$ u modu i može se direktno naći kao lijeva singularna matrica od matricizacije tenzora \mathcal{A} u modu i . Zbog toga se računanje HOSVD-a tenzora reda N svodi na računanje N različitih matricnih SVD-a matrica formata $(I_n \times I_1 I_2 \dots I_{n-1} I_{n+1} \dots I_N)$.

Nakon toga, jezgreni tenzor \mathcal{S} može se izračunati prebacivanjem matrica singularnih vektora na lijevu stranu u jednadžbi (3.9):

$$\mathcal{S} = \mathcal{A} \times_1 U^{(1)T} \times_2 U^{(2)T} \times_3 \dots \times_N U^{(N)T}.$$

Matlab kod za HOSVD

```

1 function [S, U1, U2, U3] = HOSVD(A_tens)
2
3 [U1, S1, V1] = svd(unfold(A_tens, 1));
4 [U2, S2, V2] = svd(unfold(A_tens, 2));
5 [U3, S3, V3] = svd(unfold(A_tens, 3));
6
7 S=tmul(tmul(tmul(A_tens,U1',1),U2',2),U3',3);
8 end

```

Funkcija $\text{tmul}(A, X, i)$ množi tenzor \mathcal{A} matricom X u modu i , $\mathcal{A} \times_i X$.

Ponekad se događa da je dimenzija od jednog moda veća od produkta dimenzija drugih modova. Pretpostavimo, npr. da je $\mathcal{A} \in \mathbb{R}^{l \times m \times n}$ gdje je $l > mn$. Može se pokazati da tada za tenzor \mathcal{S} vrijedi

$$\mathcal{S}(i, :, :) = 0, \quad i > mn, \quad (3.10)$$

i možemo odbiti dio s nulama tenzora \mathcal{S} te (3.3) napisati kao tanki HOSVD

$$\mathcal{A} = \widehat{\mathcal{S}} \times_1 \widehat{U}^{(1)} \times_2 U^{(2)} \times_3 U^{(3)}, \quad (3.11)$$

gdje je $\widehat{\mathcal{S}} \in \mathbb{R}^{mn \times m \times n}$ i $\widehat{U}^{(1)} \in \mathbb{R}^{l \times mn}$.

3.4 Aproksimacija tenzora HOSVD-om

Matrica se može napisati po uvjetima SVD-a kao suma matrica ranga 1, pogledaj (2.2). Analogno se može izvesti i za tenzore koristeći definiciju tenzorsko-matričnog množenja: tenzor $\mathcal{A} \in \mathbb{R}^{l \times m \times n}$ može biti izražen kao suma od: matrica puta singularni vektor:

$$\mathcal{A} = \sum_{i=1}^n A_i \times_3 u_i^{(3)}, \quad A_i = \mathcal{S}(:, :, i) \times_1 U^1 \times_2 U^{(2)}, \quad (3.12)$$

gdje je $u_i^{(3)}$ stupčani vektor u U^3 . A_i ćemo promatrati kao matricu $\mathbb{R}^{m \times n}$ i kao tenzor $\mathbb{R}^{m \times n \times 1}$.

Ilustrirajmo (3.12) na sljedeći način

$$\text{Cube } \mathcal{A} = \text{Square } A_1 + \text{Square } A_2 + \dots$$

Analogno vrijedi i za ostale modove.

Lako se pokaže da su matrice A_i ortogonalne u smislu skalarnog produkta (3.1):

$$\begin{aligned} \langle A_i, A_j \rangle &= \text{tr}[U^{(2)} \mathcal{S}(:, :, i)^T (U^{(1)})^T U^{(1)} \mathcal{S}(:, :, j) (U^{(2)})^T] \\ &= \text{tr}[\mathcal{S}(:, :, i)^T \mathcal{S}(:, :, j)] = 0. \end{aligned}$$

(Odsječak $\mathcal{S}(:, :, i)$ smo gledali kao matricu i iskoristili unitarnu invarijantnost traga $\text{tr}(AB) = \text{tr}(BA)$.)

Sada se vidi da (3.12) možemo interpretirati na sljedeći način. Svaki odsječak u modu 3 tenzora \mathcal{A} može biti zapisan u uvjetima ortogonalne baze $(A_i)_{i=1}^{r_3}$, gdje je r_3 broj pozitivnih singularnih vrijednosti od \mathcal{A} u modu 3:

$$\mathcal{A}(:, :, j) = \sum_{i=1}^{r_3} z_i^{(j)} A_i, \quad (3.13)$$

gdje je $z_i^{(j)}$ j -ta komponenta od $u_i^{(3)}$. Osim toga, imamo istodobno ortogonalnu faktORIZACIJU od A_i ,

$$A_i = \mathcal{S}(:, :, i) \times_1 U^{(1)} \times_2 U^{(2)},$$

što, zbog sortiranosti (3.4) svih singularnih vrijednosti u modu j za različiti j , ima svojstvo da je "masa" od svakog $\mathcal{S}(:, :, i)$ koncentrirana u gornjem lijevom kutu.

Ako sada skratimo (3.12), odnosno uzmemo prvih k odsječaka iz jezgrenog tenzora \mathcal{S}

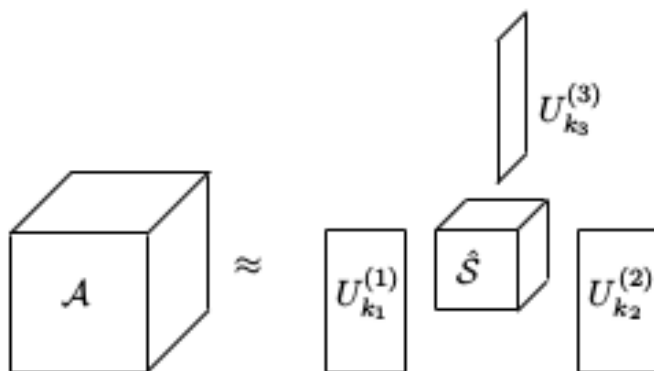
$$\mathcal{A} = \sum_{i=1}^k A_i \times_3 u_i^{(3)}, \quad A_i = \mathcal{S}(:, :, i) \times_1 U^{(1)} \times_2 U^{(2)},$$

za neki k , tada imamo aproksimaciju tenzora (ovdje u modu 3) u uvjetima ortogonalne baze. Vidimo iz (3.13) da svaki odsječak $\mathcal{A}(:, :, j)$ u modu 3 tenzora \mathcal{A} može biti zapisan kao linearna kombinacija ortogonalnih matrica A_j .

Budući da je masa jezgrenog tenzora \mathcal{S} koncentrirana za male vrijednosti od sva tri indeksa, moguće je "stisnuti" podatke u sva tri moda HOSVD-om. Pretpostavimo sada da smo podatke stisnuli u manji tenzor gdje ćemo imati k_i stupaca u modu i . Neka je $U_{k_i}^{(i)} = U^{(i)}(:, 1 : k_i)$ i $\hat{\mathcal{S}} = \mathcal{S}(1 : k_1, 1 : k_2, 1 : k_3)$. Zatim razmotrimo aproksimaciju

$$\mathcal{A} \approx \hat{\mathcal{A}} = \hat{\mathcal{S}} \times_1 U_{k_1}^{(1)} \times_2 U_{k_2}^{(2)} \times_3 U_{k_3}^{(3)}.$$

Ilustrirajmo to na sljedećoj slici:



Na ovaj način smo uz minimalne gubitke uštedjeli memoriju. Tenzor $\mathcal{A} \in \mathbb{R}^{l \times m \times n}$ ima ukupno $l \cdot m \cdot n$ elemenata, dok za njegovu aproksimaciju $\hat{\mathcal{A}}$ moramo pamtit i ukupno $k_1 \cdot k_2 \cdot k_3 + l \cdot k_1 + m \cdot k_2 + n \cdot k_3$ elemenata.

Ako je tenzor $\mathcal{A} \in \mathbb{R}^{1000 \times 1000 \times 1000}$ onda on ima $1000 \cdot 1000 \cdot 1000 = 10^9$ elemenata, dok njegova aproksimacija $\hat{\mathcal{A}}$ za $k_1 = k_2 = k_3 = 100$ ima 10^6 elemenata u $\hat{\mathcal{S}}$, $1000 \cdot 100 = 10^5$ elemenata u $U_{k_1}^{(1)}$, $U_{k_2}^{(2)}$ i $U_{k_3}^{(3)}$, ukupno $1.3 \cdot 10^6$ elemenata. Što znači da umjesto da koristimo 1 000 000 000 elemenata mi koristimo samo 1 300 000 elemenata, a to je ušteda od 997 000 000 elemenata, odnosno umjesto da zauzmemo 3GB memorije zauzimamo samo 3.9MB. Koliko je aproksimacija dobra pogledajmo na sljedećem primjeru.

Primjer 3.4.1. Već smo vidjeli da slike možemo promatrati kao matrice. Isto tako skup slika možemo promatrati kao tenzor reda 3 gdje nam horizontalni odsječci predstavljaju fotografije. Neka je sada \mathcal{A} tenzor koji sadrži 5 istih fotografija.



Slika 3.5: Izvorna fotografija dimenzije 2560 x 1600.

Primjenimo sada HOSVD na tenzor \mathcal{A} i razmotrimo njegovu aproksimaciju $\mathcal{A} \approx \hat{\mathcal{A}} = \hat{\mathcal{S}} \times_1 U_{k_1}^{(1)} \times_2 U_{k_2}^{(2)} \times_3 U_{k_3}^{(3)}$. Za $k_1 = k_2 = 100$, $k_3 = 2$ dobivamo da jedan odsječak tenzora $\hat{\mathcal{A}}$ sadrži sljedeću sliku



Slika 3.6: Komprimirana fotografija dimenzije 2560 x 1600 za $k_1 = k_2 = 100$, $k_3 = 2$.

a za $k_1 = k_2 = 50$, $k_3 = 2$



Slika 3.7: Komprimirana fotografija dimenzije 2560 x 1600 za $k_1 = k_2 = 50$, $k_3 = 2$.

Vidimo da je aproksimacija dobra, a ušteda na memoriji je ogromna. Uočimo još da u memoriju ne spremamo $\hat{\mathcal{A}}$ nego $\hat{\mathcal{S}}$, $U_{k_1}^{(1)}$, $U_{k_2}^{(2)}$ i $U_{k_3}^{(3)}$ jer je $\hat{\mathcal{A}}$ istih dimenzija kao i \mathcal{A} pa ima i isti broj elemenata.

Poglavlje 4

Prepoznavanje Lica Korištenjem HOSVD-a

Ljudska bića su vrlo vješta pri prepoznavanju lica čak i kada izraz lica, osvjetljenje, kut gledanja itd. variraju. Razviti automatske postupke za prepoznavanje lica koji su robusni s obzirom na različite uvjete, izazovan je istraživački problem koji je istražen pomoću nekoliko različitih pristupa. Analiza glavnih komponenti (tj. SVD) popularna je tehnika koja često ide pod imenom "eigenfaces". Međutim, ova metoda je najbolja kada su sve fotografije nastale pod sličnim uvjetima, i ne radi dobro kad se mijenjaju neki faktori okoline.

Nedavno su istraživane metode za multilinearu analizu skupa fotografija. Konkretno, problem prepoznavanja lica razmatran je korištenjem tenzorskog modela, TensorFaces pristupa. I to tako da moduli tenzora predstavljaju drukčiji način gledanja, npr. osvjetljenje ili izraz lica, postalo je moguće poboljšati preciznost algoritma prepoznavanja u usporedbi s PCA (analiza glavnih komponenti eng. principal component analysis) metodom.

U ovom ćemo poglavlju opisati metodu tenzorskog prepoznavanja lica, koja se odnosi na TensorFaces. Budući da se radi o fotografijama, koje se često pohranjuju kao matrice $m \times n$, s m i n reda 100-500, izračuni za svako lice koje treba identificirati su prilično teški. Razmotrit ćemo kako se tenzorski SVD (HOSVD) također može koristiti za smanjenje dimenzije u svrhu smanjenja broja operacija.

4.1 Tenzorski prikaz fotografija

Pretpostavimo da imamo kolekciju fotografija od n_p osoba, gdje je svaka fotografija $m_{i_1} \times m_{i_2}$ matrica s $m_{i_1} m_{i_2} = n_i$ elemenata. Pretpostavljamo da su stupci fotografija složeni tako da svaka fotografija predstavlja vektor u \mathbb{R}^{n_i} . Nadalje pretpostavljamo

da je svaka osoba fotografirana u n_e različitih izraza lica. Često je $n_i \geq 5000$, i obično je n_i znatno veći od n_e i n_p . Takva kolekcija fotografija je spremljena u tenzor:

$$\mathcal{A} \in \mathbb{R}^{n_i \times n_e \times n_p} \quad (4.1)$$

Pozivamo se na različite modove kao što su mod fotografije (eng. the image mode), mod izraza lica (eng. the expression mod) i mod osobe (eng. the person mode), zbog toga koristimo indekse i, e, p u (4.1).

Ako, primjerice, također imamo fotografije svake osobe s različitim osvjetljenjem, kutem gledanja, itd., onda bi mogli predstavljati kolekciju fotografija tenzorom višeg reda. Radi jednostavnosti ovdje promatramo samo tenzore reda 3. Generalizacija tenzorima višeg reda je jasna.

Primjer 4.1.1. *Prethodno smo obradili fotografije od 10 osoba iz baze podataka Yale Face koje smo postavili na 112×78 piksela i pohranili u vektor duljine 8736. Pet fotografija je prikazano na sljedećoj slici.*



Slika 4.1: Osoba 1 s pet različitih izraza lica (iz Yale Face Database).

Svaka osoba je fotografirana u ukupno 11 različitih izraza lica.

Redosljed modova je naravno proizvoljan; u svrhu ilustriranja pretpostavljat ćemo redosljed iz (4.1). Međutim kako bi naglasili proizvoljnost, koristit ćemo oznaku \times_e za množenje tenzora matricom u modu izraza lica (the expression mode), i slično za druge modove. Sada pretpostavljamo da je $n_i \gg n_e n_p$ i pišemo tanki HOSVD (Pogledati Teorem 3.3.1 i (3.11)),

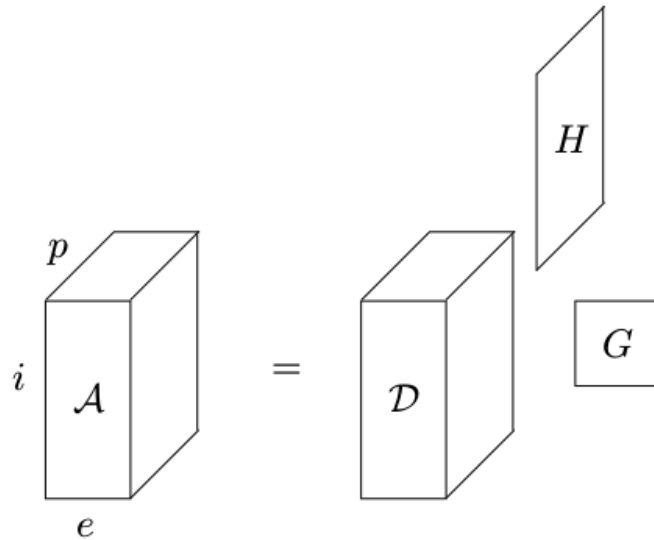
$$\mathcal{A} = \mathcal{S} \times_i F \times_e G \times_p H,$$

gdje je $\mathcal{S} \in \mathbb{R}^{n_e n_p \times n_e \times n_p}$ jezgra tenzora \mathcal{A} , $F \in \mathbb{R}^{n_i \times n_e n_p}$ ima ortogonalne stupce, i $G \in \mathbb{R}^{n_e \times n_e}$ i $H \in \mathbb{R}^{n_p \times n_p}$ su ortogonalne matrice.

HOSVD se može interpretirati na više načina ovisno o tome za što se koristi. Prvo ćemo ilustrirati relaciju

$$\mathcal{A} = \mathcal{D} \times_e G \times_p H,$$

gdje je $\mathcal{D} = \mathcal{S} \times_i F$:



U ovom dijelu rekapituliramo definiciju tenzorsko-matričnog množenja. Množenje u 2-modu ovdje smatramo množenjem u e -modu:

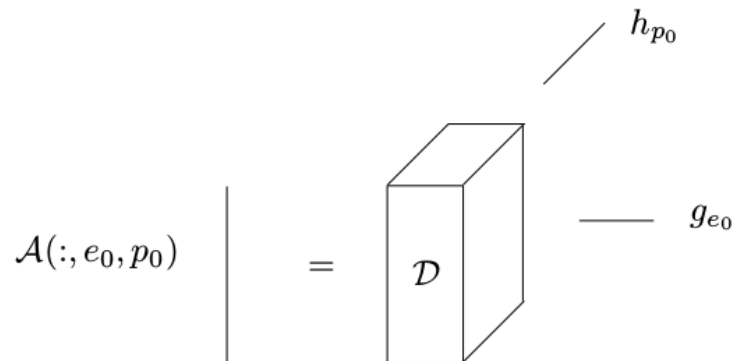
$$(\mathcal{D} \times_e G)(i_1, j, i_3) = \sum_{k=1}^{n_e} g_{j,k} d_{i_1, k, i_3}.$$

Vidimo da fiksiranjem određene vrijednosti parametra ekspresije, tj. stavljanjem $j = e_0$, odgovara korištenju samo e_0 -tog retka od G . Analognim korištenjem u modu osoba dobivamo

$$\mathcal{A}(:, e_0, p_0) = \mathcal{D} \times_e g_{e_0} \times_p h_{p_0}, \quad (4.2)$$

gdje g_{e_0} označava e_0 -ti redak od G i h_{p_0} p_0 -ti redak od H .

Ilustriramo (4.2) na sljedećoj slici:



Opišimo ovo riječima:

Sliku osobe p_0 u izrazu e_0 možemo dobiti množenjem tenzora \mathcal{D} s h_{p_0} i g_{e_0} u njihovim odgovarajućim modovima. Prema tome osoba p_0 je jedinstveno određena vektorom h_{p_0} i izraz e_0 je jedinstveno određen vektorom g_{e_0} , preko bilinearne forme

$$\mathcal{D} \times_e g \times_p h$$

Primjer 4.1.2. *Matlab kod*

```
a=tmul(tmul(D,Ue(4,:),2),Up(6,:),3);
```

vraća 6. osobu u izrazu 4 (sretan)



Slika 4.2: Osoba 6 u izrazu 4 (sretan).

4.2 Prepoznavanje lica

Sada ćemo razmatrati problem klasifikacije kako slijedi:

Za danu fotografiju nepoznate osobe, koju predstavlja vektor u \mathbb{R}^{n_i} odredit ćemo koju od n_p osoba predstavlja ili odlučiti da nepoznata osoba nije u bazi podataka.

Za klasifikaciju pišemo HOSVD (3.3) u sljedećem obliku:

$$\mathcal{A} = \mathcal{C} \times_p H, \quad \mathcal{C} = \mathcal{S} \times_i F \times_e G. \quad (4.3)$$

Za određeni izraz lica e imamo

$$\mathcal{A}(:, e, :) = \mathcal{C}(:, e, :) \times_p H. \quad (4.4)$$

Očito tenzore $\mathcal{A}(:, e, :)$ i $\mathcal{C}(:, e, :)$ možemo promatrati kao matrice, koje označavamo s A_e i C_e . Stoga, za sve izraze imamo linearne relacije

$$A_e = C_e H^T, \quad e = 1, 2, \dots, n_e \quad (4.5)$$

Imajmo na umu da se ista (ortogonalna) matrica H pojavljuje u svih n_e relacija. S $H^T = (h_1 \cdots h_{n_p})$, stupac p iz (4.5) može biti zapisan kao

$$a_p^{(e)} = C_e h_p \quad (4.6)$$

Možemo interpretirati (4.5) i (4.6) na sljedeći način:

Stupac p od matrice A_e sadrži sliku osobe p u izrazu e . Stupci matrice C_e su bazni vektori za izraz e , i red p matrice H , tj. h_p , sadrži koordinate fotografija osobe p u toj bazi. Nadalje isti h_p sadrži koordinate fotografija osobe p u svim izraznim bazama.

Zatim pretpostavimo da je $z \in \mathbb{R}^{n_i}$ slika nepoznate osobe u nepoznatom izrazu (iz n_e) i da ju želimo klasificirati. Taj z ćemo zvati testna slika. Očito, ako je slika osobe p u izrazu e , onda su koordinate vektora z u toj bazi jednake koordinatama vektora h_p . Dakle, možemo klasificirati z računanjem njegovih koordinata u svim bazama ekspresije i provjeravajući, za svaki izraz, podudaraju li se koordinate od z (ili se gotovo podudaraju) s elementima bilo kojeg reda matrice H .

Koordinate od z u izraznoj bazi e mogu se naći rješavanjem problema najmanjeg kvadrata

$$\min_{\alpha_e} \|C_e \alpha_e - z\|_2 \quad (4.7)$$

Klasifikacijski algoritam

```
for  $e = 1, 2, \dots, n_e$ 
  Riješi  $\min_{\alpha_e} \|C_e \alpha_e - z\|_2$ 
  for  $p = 1, 2, \dots, n_p$ 
    if  $\|\alpha_e - h_p\|_2 < \text{tol}$ , onda klasificiraj kao osobu  $p$  i stani.
  end
end
```

Količina posla u ovom algoritmu je velika: za svaku test sliku z moramo riješiti n_e problema najmanjih kvadrata (4.7) s $C_e \in \mathbb{R}^{n_i \times n_p}$.

Poglavlje 5

Primjeri dobiveni MATLAB rutinama

Program smo testirali na 10 osoba fotografiranih u 10 različitih ekspresija(izraza) lica. Fotografije su spremljene u matricu tako da svaki stupac predstavlja jednu fotografiju i to na način da prvih 10 stupaca predstavlja prvu osobu u svih 10 ekspresija, drugih 10 stupaca predstavlja drugu osobu itd. Svaka fotografija je dimenzije 64×64 što znači da je matrica dimenzije 4096×100 .

Pogledajmo sada fotografije tih osoba u prvom izrazu lica:





Slika 5.1: 10 osoba u prvom izrazu lica.

Svaka od tih osoba je fotografirana u 10 različitih izraza lica, pa pogledajmo na prvoj osobi koji su to sve izrazi:

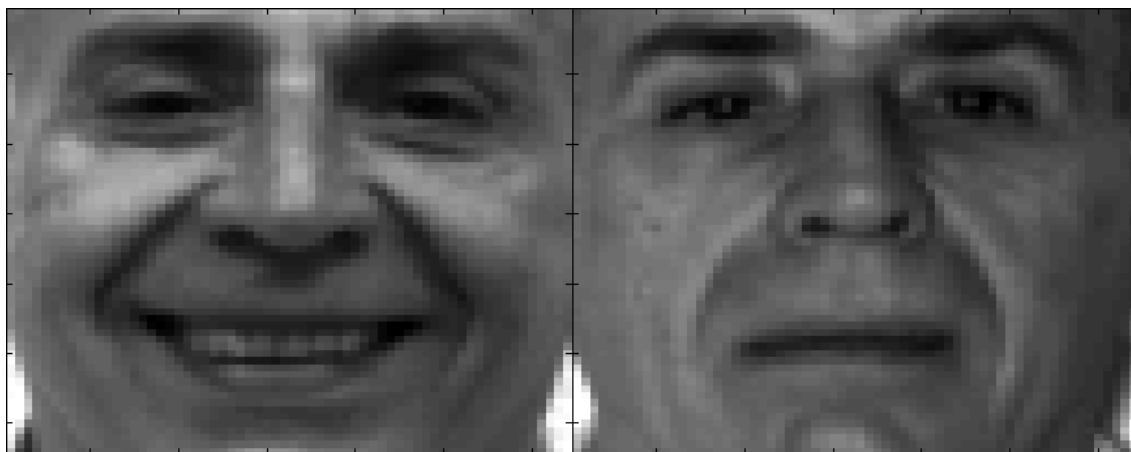


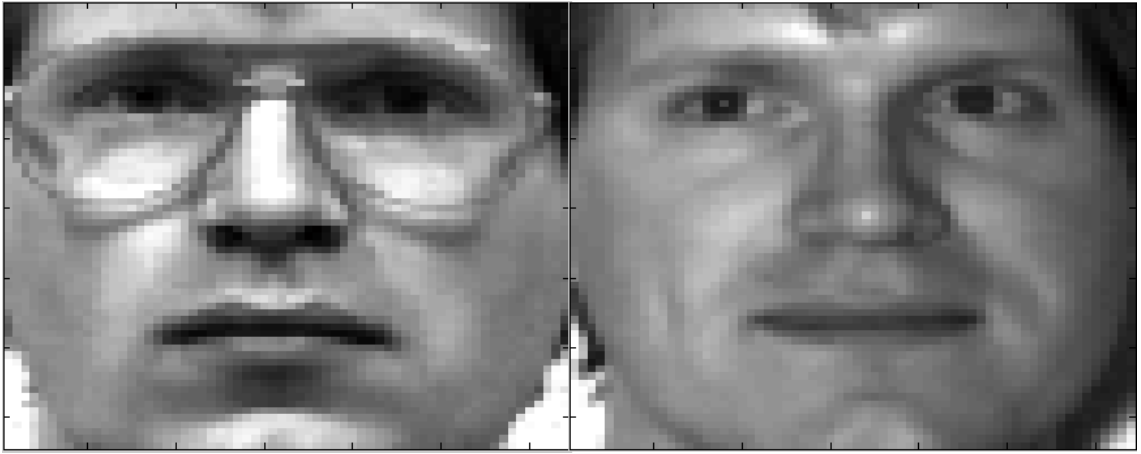
5.1 Klasifikacija

Za potrebe prepoznavanja lica uzeli smo prvih 5 osoba u prve 3 ekspresije lica (15 fotografija) i stavili ih u bazu, te na njima "trenirali" program. Nakon toga smo izbacili te fotografije i na preostalim fotografijama testirali program. Preostalo je po 7 fotografija za svaku osobu koja je u bazi, a po 10 od osoba koje nisu u bazi (ukupno 35 fotografija osoba koje su u bazi i 50 fotografija osoba koje nisu u bazi). Naš cilj je da danu testnu fotografiju program klasificira kao jednu od tih 5 osoba ili da vrati: "Tražena osoba nije u bazi!". Odnosno, ukoliko je testna fotografija jedna od tih 7 fotografija program prepozna tu osobu kao odgovarajuću osobu iz baze, a ako je jedna od 10 fotografija osoba koje nisu u bazi vrati: "Tražena osoba nije u bazi!". Tolerancija u ovom radu iznosi 0.475, a do nje dolazimo eksperimentima.

Uzmimo sada za testnu fotografiju proizvoljnu fotografiju iz skupa od 35 fotografija osoba koje su u bazi i na njoj testirajmo program.

Pogledajmo rezultate na sljedećim slikama: (Lijeva fotografija predstavlja osobu koju želimo klasificirati, a desna klasificiranu osobu.)







Vidimo da je program dobro prepoznao lica. Za sljedeće fotografije je program vratio: "Tražena osoba nije u bazi!".







Što također vidimo da je dobro jer niti jedna od tih osoba nije u prvih 5 osoba na kojima smo trenirali naš program.

Napomenimo na kraju da program nije savršen i da se može dogoditi da krivo klasificira osobu.

Primjer krive klasifikacije: Uzeli smo novu 11. osobu i pokušali je klasificirati.



Klasifikacija očito nije dobra jer to nisu iste osobe!

Da bismo to spriječili možemo smanjiti toleranciju, ali u tom slučaju se može dogoditi da osoba koja je u bazi ne bude prepoznata, odnosno da program vrati: "Tražena osoba nije u bazi!".

Pogledajmo i takav primjer: (Tolerancija u ovom primjeru iznosi 0.35).



Za gornju fotografiju je program vratio: "Tražena osoba nije u bazi!", a trebao ju je klasificirati kao:



Unatoč tome smatramo da je "manja" pogreška ako program ne prepozna osobu koju bi trebao, nego da osoba koja nije u bazi bude prepoznata kao neka osoba koja je u bazi. Iz tih razloga ipak uzimamo manju toleranciju do koje dolazimo eksperimentima.

Bibliografija

- [1] L. Elden, Matrix methods in data mining and pattern recognition, Philadelphia, SIAM., 2007.
- [2] L. De Lathauwer, B. De Moor, and J. Vandewalle. A multilinear singular value decomposition. *SIAM J. Matrix Anal. Appl.*, 21:1253–1278, 2000.
- [3] J. Demmel. *Applied Numerical Linear Algebra*. SIAM . MIT , 1996.
- [4] T. G. Kolda, Multilinear Operators for Higher Order Decompositions, Tech. Report SAND2006-2081, 2006.
- [5] Mladen Rogina, Sanja Singer, Saša Singer. *Numerička analiza*. PMF - Matematički odsjek, Zagreb, 2003.
- [6] Z. Drmač. Bilješke s predavanja iz kolegija Uvod u složeno pretraživanje podataka, PMF - Matematički odsjek, Zagreb, 2016.
- [7] A. Novak, D. Pavlović. *Dekompozicija matrice na singularne vrijednosti i primjene*. math.e, Zagreb, 2013.

Sažetak

U ovom radu opisani su automatski postupci za prepoznavanje lica na fotografijama koji ne ovise o variranju uvjeta poput osvjetljenja, kuta gledanja ili izraza lica. U zadnje vrijeme proučavaju se metode za multilinearu analizu skupine slika. Posebno se za problem prepoznavanja slika razmatra korištenje tenzorskog modela. U tom slučaju modovi tenzora predstavljaju različite uvjete gledanja kao što su osvjetljenje i izrazi lica, i time se povećava preciznost algoritma za prepoznavanje. U radnji ćemo se fokusirati na tenzorsku metodu za prepoznavanje lica. Budući da se bavimo fotografijama koje su spremljene kao $m \times n$ polja, postupci računanja za identifikaciju lica na svakoj slici bili bi prezahtjevni. Tenzorska dekompozicija singularnih vrijednosti (HOSVD) može se koristiti za smanjenje dimenzionalnosti i redukciju broja operacija kod računanja. Na kraju je teorija ilustrirana konkretnim primjerima izraženim u MATLABu.

Summary

This paper describes automatic face recognition procedures for images that do not depend on variations in conditions such as brightness, angle of view, or facial expression. Lately, the methods for multilinear analysis of a group of images are being studied. When it comes to the problem of image recognition, the use of the tensor model has especially been taken into account. In this case, the modes of the tensors represent different viewing conditions such as illumination and facial expressions, thereby increasing the accuracy of the recognition algorithm. The paper focuses on the tensor method for facial recognition. Since we are dealing with images saved as $m \times n$ fields, counting procedures for face identification in each image would be overwhelming. Tensor decomposition of singular values (HOSVD) can be used to reduce dimensionality and reduce number of operations in computing. In the end of the paper, the theory is illustrated by concrete examples expressed in MATLAB.

Životopis

Rođen sam 4.1.1992. u Zagrebu. Nakon završene Osnovne škole Luka 2, upisao sam Srednju školu Sesvete. Godine 2010. upisujem preddiplomski studij matematike na Matematičkom odsjeku PMF-a u Zagrebu, gdje 2015. godine uspisujem diplomski studij Primijenjena matematika.