Kompresija slike zasnovana na SVD dekompoziciji

Žika Miljković, Miroslav Pavlović
Mentor: prof. dr Vidosav Stojanović

***Sadržaj* – U ovom radu je predstavljen postupak kompresije slike zasnovan Dekompozicijom na Singularne Vrednosti (u daljem tekstu SVD *Singular Value Decomposition*). Slike koje često koristimo u kompjuterskim aplikacijama se teško memorišu i prenose preko interneta. Jedno od mogućih rešenja problema je korišćenje tehnike kompresije, gde je slika predstavljena u vidu matrice, a zatim se nad matricom izvode odgovarajuće operacije. Kompresija slike je postignuta korišćenjem SVD tehnike nad matricom slike. Prednost korišćenja SVD-a je u kompaktnosti energije u slici i njena sposobnost na adaptaciju lokalne statističke promene u slici. Osim toga, SVD može biti primenjen na proizvoljnu, kvadratnu, reverzibilnu ili nereverzibilnu matricu veličine (*m* x *n*). SVD se koristi za kompresiju i redukciju memorijskog prostora na slici.**

***Ključne reči* – Dekompozicija na singularne vrednosti (SVD), kompresija slike *U,S,V* matrice.**

I. UVOD

Danas su razvijene mnoge metode kompresije mirnih slika (sa više nivoa sivog, ili u boji). To su transformacijske kompresije koje su implementirane u JPEG i JPEG2000 standardu. Postupci kompresije slike su neophodni kako bi se smanjilo zauzeće memorije ili potreban kapacitet telekomunikacionih kanala, jer se radi o prenosu ili zapisu ogromne količine podataka potrebnih za reprezentaciju slike. Razvijene su metode koje kompresuju sliku i do 50 puta bez znatnog uticaja na kvalitet slike, a da se pri tome može istolerisati izvestan stepen gubitaka. To su transformacijske kompresije slike sa gubicima [1]. Transformacijska kompresija vrši transformaciju signala, gde dobijeni koeficijenti transformacije ne prikazuju više intenzitet boje i osvetljaj slike već prikazuju frekvencijske komponente. Sa frekvencijskim komponentama se bolje operiše jer znamo kakvo one značenje imaju za sliku. Uklanjajući 50% bitova od visokofrekventnih komponenata slike, slika se degradira za otprilike 5% u odnosu na original. Razvijene su i ispitane različite matematičke operacije: DCT (Diskretna Kosinusna Transformacija), DWT (Diskretna Vejvlet Transformacija) i SVD dekompozicija. DCT kompresija ima svoje nedostatke: blokovska struktura kod velikog stepena kompresije, pojava artifakta oko ivica i gruba kvantizacija boja. Kod DWT kompresije koja otklanja nedostatke JPEG-a zasvovanog na DCT-u postižu se bolji rezultati jer je skoro nemoguće razlikovati sliku po JPEG2000 standardu od originalne. Kod SVD dekompozicije, dekompozicijom originalne matrice (slike) na singularne vrednosti, predstavlja važnu tehniku za kompresiju slike. Nedavno je veoma postala korisna u analizi DNA. Porast mogućnosti računara omogućava sve veće iskorišćavanje ove metode kompresije podataka. Jedna posebna osobina SVD-a je mogućnost primenjene na bilo koju (*m* x *n*) matricu. Ova tehnika rastavlja matricu A na tri matrice $U,S,V,$ tako da je $A=USV^T$, gde su U i V ortogonalne matrice, a S je dijagonalna matrica.

Žika Miljković, Fakultet Tehničkih Nauka u Kosovskoj Mitrovici, Srbija (telefon: 381-64-8611547; e-mail:omiljkovic@gmail.com).
Miroslav Pavlović, Fakultet Tehničkih Nauka u Kosovskoj Mitrovici, Srbija (telefon: 381-64-4457343; email: miroslav.rt@gmail.com).

II. OPIS SVD ALGORITMA

Singularna dekompozicija matrice je jedna od nakorišćenijih dekompozicija u numeričkoj linearnoj algebri. Namena SVD-a je da rastavi matricu A na proizvod $A=USV^T$ [2]-[3]. Matrica U sadrži leve singularne vektore, matrica V sadrži desne singularne vektore, a dijagonalna matrica S sadrži singularne vrednosti. Singularne vrednosti matrice A su razmeštene na glavnoj dijagonali u sledećem poretku

$$\sigma_1 \geq \sigma_2 \geq ...\sigma_{min(m,n)} \geq 0. \quad (1)$$

Proces SVD-a započinjemo tako što izaberemo matricu A koja ima m vrsta i n kolona. Sada, treba da rastavimo matricu A na tri matrice U, S, V^T. Prvo treba da nađemo V. Ako pomnožimo obe strane jednačine $A=USV^T$ sa A^T dobijamo

$$A^T A = (USV^T)^T (USV^T) = VS^T U^T USV^T \quad (2)$$

Kako je $U^T U = I$ dobijamo

$$A^T A = VS^2 V^T \quad (3)$$

Sada je potrebno da dijagonalizujemo $A^T A$. Ako ste primetili, ovo je veoma slično dijagonalizaciji matrice A u $A=Q\Lambda Q^T$. Samo što naša simetrična matrica nije matrica A, već je to $A^T A$. Da bi se našle matrice V i S potrebno je da se nađu sopstvene vrednosti i sopstveni vektori $A^T A$ [4]. Sopstvene vrednosti su kvadrati elemenata S matrice (singularne vrednosti), sopstveni vektori su kolone matrice V (desni singularni vektori). Eliminacija matrice V iz jednačine je veoma slična eliminaciji matrice U. Umesto da množimo sa leve strane sa A^T mi ćemo pomnožiti sa desne strane sa A^T. Ovim dobijamo:

$$AA^T = (USV^T)(USV^T)^T = USV^T VS^T U^T \qquad (4)$$

Kako je $V^T V = I$ dobijamo

$$AA^T = US^2U^T \qquad (5)$$

Ponovo je potrebno pronaći sopstvene vektore, ali ovog puta za $A^T A$. Ovi vektori su kolone matrice U (levi singularni vektori).

Kako je A matrica m x n, S je m x n i

$$A^T A$$

daje matricu dimenzija n x n, i:

$$AA^T$$

daje matricu dimenzija m x m,

$$A = [u_1 ... u_r ... u_m] \cdot \begin{bmatrix} \sigma_1 & & & & \\ & \ddots & & & \\ & & \sigma_r & & \\ & & & \ddots & \\ & & & & 0 \end{bmatrix} \cdot \begin{bmatrix} v_1^T \\ \vdots \\ v_r^T \\ \vdots \\ v_n^T \end{bmatrix} \qquad (6)$$

Gde je U matrica dimenzija m x m, S je m x n, V je n x n.

III. KOMPRESIJA SLIKE KORIŠĆENJEM SVD TEHNIKE

Cilj dekopozicije matrice A u proizvod USV^T je aproksimacija m x n matrice A pomoću daleko manjih vrednosti nego što ih ima originalna matrica.

$$A = \sigma_1 u_1 v_1^T + \sigma_2 u_2 v_2^T + \cdots + \sigma_r u_r v_r^T + 0 u_{r+1} v_{r+1}^T + \cdots \qquad (7)$$

Kako su singularne vrednosti uvek veće od nule, dodavanje zavisnih članova, gde su singularne vrednosti jednake nuli, nema efekta na sliku. Na kraju dobijamo jednačinu sa članovima:

$$A = \sigma_1 u_1 v_1^T + \sigma_2 u_2 v_2^T + \cdots + \sigma_r u_r v_r^T \qquad (8)$$

Možemo dalje aproksimirati matricu izostavljanjem još singularnih članova matrice A. Pošto su singularne vrednosti poređane po opadajućem redosledu, poslednji članovi će imati najmanje uticaja na konačnu sliku. Na ovaj način se smanjuje veličina prostora koja je potrebna za smeštanje slike na računaru.

IV. KORIŠĆENJE SVD-A U MATLAB-U

Matlab je program koji se koristi za tehnike proračuna i vizualizaciju. Matlab nam omogućava da primenimo tehniku SVD-a nad većim matricama [5]. Na primer, mi ćemo upotrebljavati matricu 5 x 5. Korišćenjem komande:

```
A=randint(5,5,50)
```

nam obezbeđuje matricu (A) dimenzija 5 x 5 sa vrednostima između 0 i 50. Naredba **`randint`** predstavlja matricu niza slučajno promenljivih brojeva.

$$A = \begin{bmatrix} 40 & 4 & 7 & 7 & 32 \\ 45 & 13 & 48 & 21 & 1 \\ 6 & 27 & 47 & 45 & 42 \\ 45 & 47 & 24 & 39 & 46 \\ 31 & 48 & 40 & 47 & 33 \end{bmatrix}$$

Sada još jednom možemo iskoristiti moć Matlab-a da bismo izvršili SVD-a nad matricom A. Korišćenjem komandi:

`[U,S,V] = svd(A)` rastavljamo matricu A u proizvod

$A = USV^T$, gde su matrice **`[U,S,V]`** date:

$$U = \begin{bmatrix} -0.2421 & -0.5388 & 0.4541 & 0.5612 & 0.3604 \\ -0.3538 & -0.5620 & -0.7273 & 0.0072 & -0.1728 \\ -0.4644 & 0.5880 & -0.1845 & 0.6158 & -0.1594 \\ -0.5481 & -0.0949 & 0.4775 & -0.3332 & -0.5930 \\ -0.5478 & 0.1976 & -0.0522 & -0.4414 & 0.6806 \end{bmatrix}$$

$$S = \begin{bmatrix} 162.8424 & 0 & 0 & 0 & 0 \\ 0 & 47.0037 & 0 & 0 & 0 \\ 0 & 0 & 41.7566 & 0 & 0 \\ 0 & 0 & 0 & 24.0802 & 0 \\ 0 & 0 & 0 & 0 & 0.7362 \end{bmatrix}$$

$$V = \begin{bmatrix} -0.4301 & -0.8821 & 0.1005 & -0.0917 & 0.1355 \\ -0.4309 & 0.2433 & 0.1752 & -0.7426 & -0.4160 \\ -0.4641 & 0.0534 & -0.7432 & 0.3140 & -0.3617 \\ -0.4738 & 0.3504 & -0.1013 & -0.0810 & 0.7975 \\ -0.4354 & 0.1924 & 0.6298 & 0.5788 & -0.2044 \end{bmatrix}$$

Nakon rastavljanja matrice $A = USV^T$, korišćenjem komande:

```
imshow(A,U,S,V,1,gray)
```

daje nam priliku da vidimo matricu A kao sliku, kao i mogućnost da vidimo svaku pojedinačnu iteraciju nad matricom A. Ova funkcija zahteva malo objašnjenje. Članovi A,U,S i V su same po sebi jasne, ali **`1`** i **`gray`** nisu objašnjavani. Član "**`1`**" startuje program sa prvom iteracijom, a "**`gray`**" koristi mapiranje boje koja je Matlab-om definisana kao siva. Sada, svaki broj matrice odgovara boji koja je Matlab-om definisana kao mapirana boja (siva). Potrebno je da napišemo ovaj program kako bi izvršili kompresiju slike ''Moon'' [5]-[6]-[7].

Korišćenjem komandi:

```
close all
clear all
[A,map]=imread('moon.tif');
B=im2double(A,'indexed');
imshow(B,map)
[u,s,v]=svd(B);
C=zeros(size(B));
    for j=1:k
      C=C+s(j,j)*u(:,j)*v(:,j).';
    end
C=floor(C);
imshow(C,map)
k=find(C<1);
C(k)=1;
```

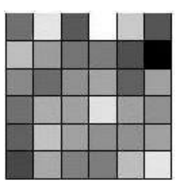
Menjanjem vrednosti promenljive k u for petlji možemo načiniti različite iteracije. Primetićete da postoji sličnost između sledeće dve jednačine:

```
C=C+s(j,j)*u(:,j)*v(:,j).'
```

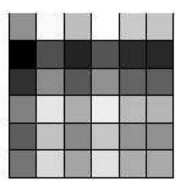
i

$$A = \sigma_1 u_1 v_1^T + r\sigma_2 u_2 v_2^T + \cdots + \sigma_r u_r v_r^T$$

Na prvom primeru slike 1a. matrice vidi se originalna slika kao i slike 1b, 1c, 1d, 1e, i 1f različitih broja iteracija :

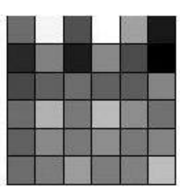


Slika 1a. Originalna matrica

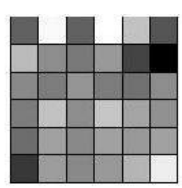


Slika 1b. Prva iteracija

Kao što se može videti prva aproksimacija nije baš dobra, jer se kompresovana slika vizuelno razlikuje od originalne. Ali povećavaćemo broj iteracija dok ne vidimo kako ona treba da izgleda.

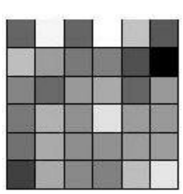


Slika 1c. Druga iteracija

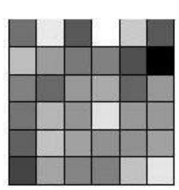


Slika 1d. Treća iteracija

Već od treće iteracije primećuje se dobra vizuelna aproksimacija.



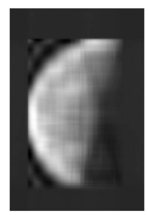
Slika 1e. Četvrta iteracija



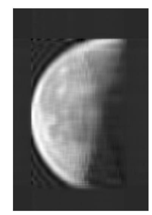
Slika 1f. Peta iteracija

Kao što se može primetiti četvrta iteracija daje približno istu sliku, dok peta iteracija predstavlja kompresovanu sliku koja vizuelno jako liči na originalnu sliku. Sada ćemo videti primenu SVD-a na slici ''Moon'' koja je jedna od standardnih slika Matlab-a, kao drugi primer. I ovde slika 2g. predstavlja originalnu sliku, dok slike 2h, 2i, 2j, 2k i 2l, predstavljaju kompresovane slike različitih broja iteracija. Što je veći broj iteracija dobija se kvalitetnija slika ali je i kompresija manja.

Slika 2g. Originalna slika



Slika 2h. Peta iteracija



Slika 2i. Deseta iteracija

Prvih pet iteracija u stvari daje privid u izgled slike, što predstavlja skromnu aproksimaciju. Ovim se smanjuje smeštajni prostor, u odnosu na originalnu sliku, za 97.67%.

Slika 2j. Dvadeseta iteracija

Slika 2k. Šesdeseta iteracija

Nakon šesdeset iteracija dobijamo dobru aproksimaciju, možemo identifikovati sliku sa znatnim stepenom detalja. Ova slika zahteva 72% manje memorijskog prostora u odnosu na originalnu sliku. Ovo je dobro. Najbolji vizuelni izgled slike ''Moon'' je posle osamdeset iteracija.

Slika 2l. Osamdeseta iteracija

Slika izgleda veoma dobro slično originalnoj dok zauzima samo 37.28% od originalne slike. Uočava se dobar kontrast kao i pojava detalja koji se vide i na originalnoj slici 2g.

V. IZRAČUNAVANJE KOMPRESIJE SLIKE

Sada ćemo izračunati kompresiju slike kako bi razjasnili gore navedene procente kompresije slike dobijene određenim brojem iteracija. Originalna slika ''Moon'' je rezolucije:

$$\boldsymbol{N} = 537 \cdot 358 = 192246$$

gde ***N*** predstavlja broj piksela nekompresovane slike, dok 537 predstavlja vertikalni, a broj 358 horizontalni broj piksela originalne slike. Koeficijent kompresovane slike se izračunava na sledeći način:

$$A = \sigma_1 u_1 v_1^{'} + \sigma_2 u_2 v_2^{'} + ... + \sigma_{20} u_{20} v_{20}^{'} .$$

Svaka u_i sadrži $m = 537$ komponenti;

Svaka v_i sadrži $n = 358$ komponenti;

Svaka σ_i je skalar $\Leftrightarrow$ 1 komponenta;

Kompresovanu sliku označićemo sa ***Nk***.

$$\boldsymbol{Nk} = 20 \cdot (537+358+1) = 17920$$

gde broj 20 predstavlja broj iteracija, a 17920 broj piksela kompresovane slike. U sledećem koraku delimo broj piksela kompresovane slike ***Nk*** i originalne slike ***N*** i dobijamo koeficijent kompresovane slike γ.

$$\gamma = \frac{Nk}{N} \cdot 100 = \frac{17920}{192246} \cdot 100 = 9.32\%$$

$$\gamma = 9.32\%$$

Na kraju, posle 20 iteracija dobijamo približno istu sliku koja iznosi 9.32% kompresovane slike, dok posle 60 iteracija dobijamo dobru kompresovanu sliku koja iznosi 27.97%. U odnosu na kompresovanu sliku sa 20 iteracija, slika sa 60 iteracija izgleda vizuelno mnogo bolje sa primetnom oštrinom i detaljima dok i dalje zauzima malo smeštajnog prostora. Rekonstrukcija slike vršena je vizuelnim načinom.

VI. ZAKLJUČAK

Korišćenje tehnike SVD-a, za kompresiju slike, predstavlja veoma koristan alat za čuvanje memorijskog prostora. Pomoću ove tehnike dobijamo sliku koja se ne razlikuje od originalne slike i koja zauzima do 45% memorijskog prostora u odnosu na originalnu sliku. Pomoću SVD-a moguće je dobiti najprikladnije baze za domen i kodomen linearne transformacije definisane nad matricom $A = USV^T$. SVD-a se ne koristi samo za kompresiju slike već ima i primene u dubinskoj analizi slike, restauraciji loših slika i u matematici, numeričkim izrazima i statistici.

ZAHVALNICA

Zahvaljujem se mentoru prof. dr Vidosavu Stojanoviću i prof. dr Miletu Petroviću na stručnom vođstvu, pomoći i savetima pri izradi ovog rada.

LITERATURA

[1] Miodrag V. Popović, "Digitalna Obrada Slike", Akademska misao, Beograd 2006. pp.138.

[2] Abhiram Ranade, Srinkath S. M., Stayen Kale, "A Variation on SVD Based Image Compression". pp.1.

[3] Jahanzeb Farooq, Michael Osadebay, "Dimensionality Reduction with SVD", "to be published".

[4] W. Keith Nicholson, "Linear Algebra with Application", University of Calgary. Chapter 6 "Eigenvalues and diagolization".

[5] Matlab help.

[6] Michael Kirby, "Application of the SVD to image representation", "to be published".

[7] D.van Alphen, "ECE 455 Lecture 12", "to be published".

ABSTRACT

In this paper we will describe image compression represent on SVD (*Singular Value Decomposition*). It is well known that the images, often used in variety of computer applications, are difficult to store and transmit. One possible solution to overcome this problem is to use a data compression technique where an image is viewed as a matrix and then the operations are performed on the matrix. Image compression is achieved by using Singular Value Decomposition (SVD) technique on the image matrix. The advantage of using the SVD is the property of energy compaction and its ability to adapt to the local statistical variations of an image. Further, the SVD can be performed on any arbitrary, square, reversible and non reversible matrix of *m* x *n* size. In this paper, SVD is utilized to compress and reduce the storage space of an image.

IMAGE COMPRESSION REPRESENT ON SVD (Singular Value Decomposition)

Žika Miljković, Miroslav Pavlović