POLITECNICO DI TORINO

Corso di Laurea Magistrale in Ingegneria Informatica

Tesi di Laurea Magistrale

Titolo

Relatori:
Prof. Mariagrazia Graziano
Prof. Marco Vacca
Prof. Maurizio Zamboni

Candidata:
Gina Jiang

2017

Acknowledgments

Table of contents

List of figures

Chapter 1

Singular Value Decomposition

1.1 Example of a SVD computation

We have the matrix C as:

$$C = \begin{bmatrix} 5 & 5 \\ -1 & 7 \end{bmatrix}$$

and we want to write it as $C = U\Sigma V^T$.
We have to take into account these 2 equations:

- $C^T C = V\Sigma^T \Sigma V^T$
- $CV = U\Sigma$

We first use the first equation $C^T C = V\Sigma^T \Sigma V^T$, and find the eigenvalues andthe eigenvectors of $C^T C$. The eigenvalues will be the entries of the diagonal entries of $\Sigma^T \Sigma$, while the eigenvectors will be the entries of V Let's compute $C^T \cdot C$

$$C^T \cdot C = \begin{bmatrix} 5 & -1 \\ 5 & 7 \end{bmatrix} \begin{bmatrix} 5 & 5 \\ -1 & 7 \end{bmatrix} = \begin{bmatrix} 26 & 18 \\ 18 & 74 \end{bmatrix}$$

Now we can find the eigenvalues of the matrix $C^T \cdot C$. We know that an eigenvalue of $C^T \cdot C$, is a solutions of the polynomial equation $det(C^T \cdot C - \lambda I) = 0$, where I is the identity matrix.

$$det(C^T \cdot C - \lambda I) = det\left(\begin{bmatrix} 26 & 18 \\ 18 & 74 \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right) =$$

$$= det \begin{bmatrix} 26-\lambda & 18 \\ 18 & 74-\lambda \end{bmatrix} = (26-\lambda)(74-\lambda) - 18 \cdot 18 =$$

$$= \lambda^2 - 100\lambda + 1600$$

Since $\lambda^2 - 100\lambda + 1600$ is a second degree equation, we can solve it using the general formula

$$a\lambda^2 + b\lambda + c$$

$$\lambda = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$$\lambda = \frac{-100 \pm \sqrt{100^2 - 4 \cdot 1600}}{2}$$

which will provide the two solutions $\lambda_1 = 20$ and $\lambda_2 = 80$.
Now we want to find the eigenvector associated to the eigenvalue $\lambda_1 = 20$. We know that this eigenvector $\mathbf{v}$ solve the equation $(C^T \cdot C - \lambda I)\mathbf{v} = 0$

$$(C^T \cdot C - \lambda_1 I)v_1 = C^T \cdot C - 20I)v_1 = \begin{bmatrix} 6 & 18 \\ 18 & 54 \end{bmatrix} \begin{bmatrix} v_{1,1} \\ v_{1,2} \end{bmatrix} = 0$$

We can see it as an equation system

$$\begin{cases} 6 \cdot v_{1,1} + 18 \cdot v_{1,2} = 0 \\ 18 \cdot v_{1,1} + 54 \cdot v_{1,2} = 0 \end{cases}$$

and have $v_{1,1} = -3, \quad v_{1,2} = 1, \quad v_1 = \begin{bmatrix} -3 \\ 1 \end{bmatrix}$.
If we want the matrix V to be othonormal, then each vector of the matrix V has to be a unit vector (i.e. have lenght 1). To make it possible, we divide each entries of the vector by the lenght of the vector. The lenght of v_1 is:

$$|v_1| = \sqrt{v_{1,1}^2 + v_{1,2}^2} = \sqrt{(-3)^2 + 1^2} = \sqrt{10}$$

$$v_1 = \begin{bmatrix} \frac{-3}{\sqrt{10}} \\ \frac{1}{\sqrt{10}} \end{bmatrix}$$

Now we can do the same steps for the second eingenvector

$$(C^T \cdot C - \lambda_2 I)v_2 = C^T \cdot C - 80I)v_2 = \begin{bmatrix} -54 & 18 \\ 18 & -6 \end{bmatrix} \begin{bmatrix} v_{2,1} \\ v_{2,2} \end{bmatrix} = 0$$

$$v_2 = \begin{bmatrix} 1 \\ 3 \end{bmatrix} \Longrightarrow \text{orthonormal} \Longrightarrow v_2 = \begin{bmatrix} \frac{1}{\sqrt{10}} \\ \frac{3}{\sqrt{10}} \end{bmatrix}$$

Now we can write the matrix V

$$V = \begin{bmatrix} v_1 & v_2 \end{bmatrix} = \begin{bmatrix} \frac{-3}{\sqrt{10}} & \frac{1}{\sqrt{10}} \\ \frac{1}{\sqrt{10}} & \frac{3}{\sqrt{10}} \end{bmatrix}$$

and the matrix $\varSigma$. We know that $\varSigma$ is a diagonal matrix, so its transpose matrix is $\varSigma^T = \varSigma$, and doing $\varSigma^T\varSigma$ is the same thing of doing the square of each diagonal entries of $\varSigma$. As said before at the beginning, the eigenvalues are the entries of $\varSigma^T\varSigma$, so to find the matrix $\varSigma$ we just need to make the square root of the eigenvalues and put them in diagonal.

$$\varSigma = \begin{bmatrix} \sqrt{\lambda_1} & 0 \\ 0 & \sqrt{\lambda_2} \end{bmatrix} = \begin{bmatrix} \sqrt{20} & 0 \\ 0 & \sqrt{80} \end{bmatrix} = \begin{bmatrix} 2\sqrt{5} & 0 \\ 0 & 4\sqrt{5} \end{bmatrix}$$

Now we need to find the last part of the SVD, which is the matrix U. To find it we use the second property listed at the beginning of this section

- $CV = U\varSigma$

$$CV = \begin{bmatrix} 5 & 5 \\ -1 & 7 \end{bmatrix} \begin{bmatrix} \frac{-3}{\sqrt{10}} & \frac{1}{\sqrt{10}} \\ \frac{1}{\sqrt{10}} & \frac{3}{\sqrt{10}} \end{bmatrix} = \begin{bmatrix} -\sqrt{10} & 2\sqrt{10} \\ \sqrt{10} & 2\sqrt{10} \end{bmatrix} = U\varSigma$$

To find U, we have to decompose $\begin{bmatrix} -\sqrt{10} & 2\sqrt{10} \\ \sqrt{10} & 2\sqrt{10} \end{bmatrix}$ as the product of U and $\varSigma$

Bibliography

[1] Performance driven FPGA design with an ASIC perspective. Andreas Ehliar, 2009

[2] K.-C. Wu and Y.-W. Tsai, "Structured asic, evolution or revolution?"in ISPD ?04: Proceedings of the 2004 international symposium on Physical design. New York, NY, USA: ACM, 2004, pp. 103?106.

[3] I. Kuon and J. Rose, "Measuring the gap between fpgas and asics",in Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on, 2007.

[4] Actel, "Igloo low-power flash fpgas handbook,"2008. [Online]. Available: http://www.actel.com/documents/IGLOO_HB.pdf

[5] T. Tuan, S. Kao, A. Rahman, S. Das, and S. Trimberger, "A 90nm lowpower fpga for battery-powered applications", in Proceedings of the 2006 ACM/SIGDA 14th international symposium on Field programmable gate arrays, 2006, pp. 3-11

[6] 2013 by Yong Cao, Referencing UIUC ECE408/498AL Course Notes (Lecture 10 - Virginia Tech)

[7] Prefix sum - Parallel algorithm - Wikipedia

[8] Patrick Cozzi,University of Pennsylvania

[9] Qingqing Dang, Shengen Yan, Ren Wu, A fast integral image generation algorithm on GPUs, 2014 IEEE

[10] 2015, Mario Cofano, Mariagrazia Graziano, Maurizio Zamboni, Design of a Logic-in-Memory architecture for massive parallel algorithms

[11] PRACTICAL FAST 1-D DCT ALGORITHMS WITH 11 MULTIPLICATIONS, Christoph Loeffler, Adriaan Lieenberg, and George S. Moschytz

[12] 2015, Filasieno Francesco, Garino Simone, Garlando Umberto, Progettazione di Sistemi a Basso Consumo : Array Sistolico Riconfigurabile

[13] LLM Integer Cosine Transform and Its Fast Algorithm Chi-Keung Fong, Student Member, IEEE, and Wai-Kuen Cham, Senior Member, IEEE

[14] High Order Integer Transform Design for Video Compression,Jie Dong, Member, IEEE, and Yan Ye, Member, IEEE

[15] DIGITAL IMAGE FILTERING By Fred Weinhaus

[16] Integrated system architecture, lecture notes by E. Raviola, 2017

[17] A Guide to Numerical Methods for Transport Equations, Dmitri Kuzmin

[18] DEVELOPPEMENT D'UN CODE DE CALCUL MICROMAGNETIQUE 2D ET 3D: APPLICATION A DES SYSTEMES REELS DE TYPES FILMS, PLOTS ET FILS, Liliana-Daniela BUDA

[19] S. Henikoff, J. G. Henikoff, Amino acid substitution matrices from protein blocks in Proceedings of the National Academy of Sciences of the United States of America, v. 89(22), pp. 10915?10919, November 1992.

[20] T. F. Smith, M. S. Waterman, ?Identi?cation of Common Molecular Subsequences? in Journal of Molecular Biology, v. 147.

[21] A New Non-Restoring Square Root Algorithm and Its VLSI Implementations, Yamin Li and Wanming Chu, International Conference on Computer Design (ICCD?96), October 1996, Austin, Texas, USA