Randomized Tensor-based Algorithm for Image Classification

Ryan Sigurdson
rsigurdson@gmail.com

Department of Mathematics
University of Rochester
Rochester, New York 14627, USA

Carmeliza Navasca
cnavasca@uab.edu

Department of Mathematics
University of Alabama at Birmingham
Birmingham, Alabama 35294, USA

November 26, 2012

Abstract

We present a method for the image classification problem. First, the set of images is organized in a tensor format. Then, we define several classes in terms of subtensors of the same type of images. The method relies on the tensor dimensionality reduction algorithm to create the basis of the subtensor. Our algorithm was tested on the AT&T database of faces. From our experiments, the algorithm successfully classifies unknown images from the measured residual.

1 Introduction

We develop an algorithm for a pattern recognition problem known as the automatic classification of unknown images. Given an unknown image, the goal is to assign the unknown image to a set of predefined classes. This problem is difficult since the variation of the objects within each class is high yet the objects from different classes have less variation. There are many techniques developed for this problem; see [9] and the references therein.

In this work, we adapt a randomized tensor-based formulation for object classification of Eldén and Savas [9] and apply it to the database of faces from the AT&T Laboratories Cambridge [1]. There are 40 different individuals with each individual in 10 different *expressions*. The expressions range from glasses on/off, smiling or frowning, and different lighting with distinct facial expressions. We arrange the images into a tensor of dimension $112 \times 92 \times 400$, with each image as a 112×92 matrix and 400 images along mode-3 (k-axis).

Here we give a short introduction to tensors and higher-order singular value decomposition. A tensor is a multidimensional array. The order of a tensor refers to the dimension of the index set. A matrix is a second-order tensor and a vector is a first-order tensor. Tucker introduced [10, 11] a decomposition (HOSVD), in which a tensor is decomposable into a core tensor $\mathcal{B}$ *multiplied* by a matrix along each mode, i.e.

$$(\mathcal{T})_{ijk} = (\mathcal{B} \bullet_1 \mathbf{U} \bullet_2 \mathbf{V} \bullet_3 \mathbf{W})_{ijk} = \sum_{\hat{i}\hat{j}\hat{k}} \mathcal{B}_{\hat{i}\hat{j}\hat{k}} u_{i\hat{i}} v_{j\hat{j}} w_{k\hat{k}}$$

where $\mathcal{T} \in \mathbb{R}^{I \times J \times K}$ and $\bullet_n$ denotes the n-mode product. The mode products are the left and right multiplications as seen in the matrix SVD: $\mathbf{T} = \mathbf{U} \bullet_1 \mathbf{B} \bullet_2 \mathbf{V}$. Many applications in signal and image processing rely on the tensor multi-dimensionality reduction due to the high-dimensional data with very few significant (e.g. signal source) contributions. There are current methods for low multilinear rank approximation, namely, truncated HOSVD [10], HOOI [2], alternating SDP [7] and randomized generalized CUR decomposition [4]. In this work, we develop a randomized algorithm based on HOOI for the best-(R_1, R_2, R_3) low multilinear rank tensor approximation.

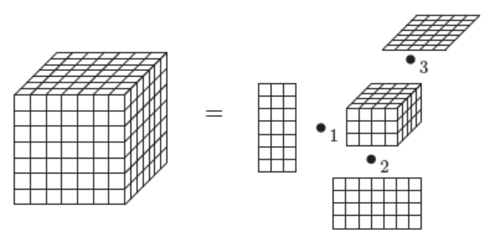

Figure 1: Tucker decomposition (HOSVD)

Randomized algorithms [4, 6] have shown to be powerful tools for approximation of matrix decomposition. In comparison with the standard (deterministic) matrix algorithms, randomization can lead to faster and robust algorithms. Here we focus on randomized versions of low dimensional rank reduction algorithms, namely, the power iteration method for QR decomposition for tensors. Recall the theorem of Eckart and Young [5] which provides the kth low rank matrix approximation of a given $m \times n$ matrix $\mathbf{M}$ in the minimization problem of

$$\hat{\mathbf{M}} = \text{argmin}_{rank(\mathbf{B})<k} \|\mathbf{M} - \mathbf{B}\|_2$$

where $\hat{\mathbf{M}} = \mathbf{U}_k \mathbf{\Sigma}_k \mathbf{V}_k^T$ (truncated SVD) and $\|\mathbf{M} - \hat{\mathbf{B}}\|_2 = \sigma_{k+1}$ (($k+1$)th singular value of $\mathbf{M}$). The truncated SVD can be computed by a variant of the power iteration method for QR decomposition; i.e.

$$\|\mathbf{M} - \mathbf{Q}\mathbf{Q}^T\mathbf{M}\| \approx \min_{rank(\mathbf{B})<k} \|\mathbf{M} - \mathbf{B}\|.$$

The idea is to compute the decomposition into two stages: (1) approximate an orthogonal matrix $\mathbf{Q}$ by oversampling and (2) implement the standard QR factorization using $\mathbf{Q}$ from the first stage. Drineas and Mahoney [4] have developed and applied the column subset selection problem to obtain a k-column $\mathbf{Q}$ in which $\|\mathbf{M} - \mathbf{Q}\mathbf{Q}^T\mathbf{M}\| \leq \sqrt{1 + k(n-k)}\|\mathbf{M} - \hat{\mathbf{M}}\|$ for tensor decomposition.

2 Low Multilinear Rank Tensor

We denote the scalars in $\mathbb{R}$ with lower-case letters $(a, b, \ldots)$ and the vectors with bold lower-case letters $(\mathbf{a}, \mathbf{b}, \ldots)$. The matrices are written as bold upper-case letters $(\mathbf{A}, \mathbf{B}, \ldots)$ and the symbol for tensors are calligraphic letters $(\mathcal{A}, \mathcal{B}, \ldots)$. The subscripts represent the following scalars: $(\mathcal{A})_{ijk} = a_{ijk}$, $(\mathbf{A})_{ij} = a_{ij}$, $(\mathbf{a})_i = a_i$. The superscripts indicate the dimension size.

A matrix $\mathbf{T} \in \mathbb{R}^{I \times J}$ is a second order tensor which has an SVD of $\mathbf{T} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}'$ where $\mathbf{\Sigma} = diag\{\sigma_1, \sigma_2, \cdots, \sigma_p\}$ where $p = \min\{I, J\}$ and $\mathbf{U}$ and $\mathbf{V}$ are orthogonal matrices. The Frobenius norm of $\mathbf{T}$ is defined as $\|\mathbf{T}\|_F = \sqrt{\sigma_1^2 + \sigma_2^2 + \cdots + \sigma_p^2}$. A rank k matrix $\widehat{\mathbf{T}}$ of $\mathbf{T}$, i.e. $k < p = rank(\mathbf{T})$, is defined as $\widehat{\mathbf{T}} = \mathbf{U}_k \mathbf{S}_k \mathbf{V}_k^T = \sum_{i=1}^{k} \sigma_i \cdot u_i v_i'$.

2.1 Tensor SVD

The order of a tensor refers to the cardinality of the index set. A matrix is a second-order tensor and a vector is a first-order tensor.

Definition 2.1 (Tucker mode-n product) *Given a tensor $\mathcal{T} \in \mathbb{R}^{I \times J \times K}$ and the matrices $\mathbf{U_1} \in \mathbb{R}^{\hat{I} \times I}$, $\mathbf{U_2} \in \mathbb{R}^{\hat{J} \times J}$ and $\mathbf{U_3} \in \mathbb{R}^{\hat{K} \times K}$, then the Tucker mode-n products are as follows: $(\mathcal{T} \bullet_1 \mathbf{U_1})_{\hat{i},j,k} = \sum_{i=1}^{I} \mathcal{T}_{ijk} \mathbf{A}_{\hat{i}i}$ (mode-1 product), $(\mathcal{T} \bullet_2 \mathbf{U_2})_{\hat{j},i,k} = \sum_{j=1}^{J} \mathcal{T}_{ijk} \mathbf{B}_{\hat{j}j}$ (mode-2 product) and $(\mathcal{T} \bullet_3 \mathbf{U_3})_{\hat{k},i,j} = \sum_{k=1}^{K} \mathcal{T}_{ijk} \mathbf{C}_{\hat{k}k}$, (mode-3 product).*

The tensor SVD is also referred to multilinear SVD (or higher-order SVD).

Theorem 2.1 (Multilinear SVD [3]) *A third order tensor $\mathcal{T} \in \mathbb{R}^{I \times J \times K}$ can be represented as a product*

$$\mathcal{T} = \mathcal{B} \bullet_1 \mathbf{U_1} \bullet_2 \mathbf{U_2} \bullet_3 \mathbf{U_3}$$

where $\mathbf{U_1} \in \mathbb{R}^{I \times I}$, $\mathbf{U_2} \in \mathbb{R}^{J \times J}$ and $\mathbf{U_3} \in \mathbb{R}^{K \times K}$ are orthogonal matrices. The core tensor $\mathcal{B} \in \mathbb{R}^{I \times J \times K}$ are the matricized subtensors $\mathbf{B}^1_{i=\alpha} \in \mathbb{R}^{J \times K}$, $\mathbf{B}^2_{j=\alpha} \in \mathbb{R}^{I \times K}$ and $\mathbf{B}^3_{k=\alpha} \in \mathbb{R}^{I \times J}$ with the following properties:

- *all-orthogonality:*
  $\langle \mathbf{B}^n_{i_n=\alpha}, \mathbf{B}^n_{i_n=\beta} \rangle = (\sigma^{(1)}_\alpha)^2 \delta_{\alpha,\beta}, \quad \alpha, \beta = 1, \ldots, I_n,$
- *ordering:*
  $\|\mathbf{B}^n_{i_n=1}\|_F \geq \|\mathbf{B}^n_{i_n=2}\|_F \geq \cdots \geq \|\mathbf{B}^n_{i_n=I_n}\|_F \geq 0$

where $\|\mathbf{B}^n_{i_n=\alpha}\|_F = \sigma^{(n)}_\alpha$ for $\alpha = 1, \ldots, I_n$ ($I_1 = I, I_2 = J, I_3 = K$ and $i_1 = i, i_2 = j, i_3 = k$).

The usual inner product of matrices, $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{I \times J}$ is denoted by $\langle \mathbf{A}, \mathbf{B} \rangle = \sum_{ij} b_{ij} a_{ij}$. For third order tensor, there are three sets of singular values: $\sigma^{(1)}_\alpha$'s are mode-1 singular values, $\sigma^{(2)}_\alpha$'s are the mode-2 singular values and $\sigma^{(3)}_\alpha$'s are the mode-3 singular values. The corresponding mode-1, mode-2 and mode-3 singular vectors are $\mathbf{u_1}_\alpha$, $\mathbf{u_2}_\alpha$ and $\mathbf{u_3}_\alpha$, respectively.

2.2 Higher-order Orthogonal Iteration

The higher order orthogonal iteration (HOOI) [2] is an alternating least-squares method for the optimization problem,

$$\begin{aligned} &\min_{\mathcal{B}, \mathbf{U_1}, \mathbf{U_2}, \mathbf{U_3}} \|\mathcal{T} - \mathcal{B} \bullet_1 \mathbf{U_1} \bullet_2 \mathbf{U_2} \bullet_3 \mathbf{U_3}\|_F \\ &= \max_{\mathbf{U_1}, \mathbf{U_2}, \mathbf{U_3}} \|\mathcal{B} \bullet_1 \mathbf{U_1}^T \bullet_2 \mathbf{U_2}^T \bullet_3 \mathbf{U_3}^T\|_F \end{aligned}$$

for approximating low rank matrices $\mathbf{U_1} \in \mathbb{R}^{I \times k_1}$, $\mathbf{U_2} \in \mathbb{R}^{J \times k_2}$, $\mathbf{U_3} \in \mathbb{R}^{K \times k_3}$ and the reduced core tensor $\mathcal{B} \in \mathbb{R}^{k_1 \times k_2 \times k_3}$. Here is the following algorithm based on the power iteration formulation:

1. Obtain initial estimates $\mathbf{U_1}, \mathbf{U_2}, \mathbf{U_3}$
2. while not converged do
   - compute $\mathbf{B}^1 = \mathbf{T}^1 \cdot (\mathbf{U_2} \otimes \mathbf{U_3})^T$ with $\mathbf{U_1} = \text{argmax}\|\mathbf{U_1}^T \cdot \mathbf{B}^1\|$
   - compute $\mathbf{B}^2 = \mathbf{T}^2 \cdot (\mathbf{U_3} \otimes \mathbf{U_1})^T$ with $\mathbf{U_2} = \text{argmax}\|\mathbf{U_2}^T \cdot \mathbf{B}^2\|$
   - compute $\mathbf{B}^3 = \mathbf{T}^3 \cdot (\mathbf{U_1} \otimes \mathbf{U_2})^T$ with $\mathbf{U_3} = \text{argmax}\|\mathbf{U_3}^T \cdot \mathbf{B}^3\|$

   end while

where $\mathbf{T}^m, \mathbf{B}^m$ denote the matricizations of tensors $\mathcal{T}, \mathcal{B}$ in mode m, respectively. The HOOI implementation uses SVD to calculate $\mathbf{U_m}$ at each mode, but QR algorithms can be used as well.

PROTO-ALGORITHM: SOLVING THE FIXED-RANK PROBLEM

Given an $m \times n$ matrix $\boldsymbol{A}$, a target rank k, and an oversampling parameter p, this procedure computes an $m \times (k+p)$ matrix $\boldsymbol{Q}$ whose columns are orthonormal and whose range approximates the range of $\boldsymbol{A}$.

1 Draw a random $n \times (k+p)$ test matrix $\boldsymbol{\Omega}$.
2 Form the matrix product $\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{\Omega}$.
3 Construct a matrix $\boldsymbol{Q}$ whose columns form an orthonormal basis for the range of $\boldsymbol{Y}$.

Figure 2: Proto-type algorithm for a low rank problem [6]

2.3 An Improvement

In Figure 2, the algorithm describes how to obtain initial orthonormal estimates for a low rank matrices $\mathbf{U_m} \in \mathbb{R}^{I_m \times k_m}$. Note that $\mathbf{U_m}\mathbf{U_m}^T$ is a projection matrix onto the subspace spanned by columns of $\mathbf{U_m}$. Then,

$$\mathcal{B} \approx \mathcal{T} \bullet_1 \mathbf{U_1}\mathbf{U_1}^T \bullet_2 \mathbf{U_2}\mathbf{U_2}^T \bullet_3 \mathbf{U_3}\mathbf{U_3}^T.$$

The following theorem describes the performance of Figure 2 with a Gaussian test matrix.

Theorem 2.2 ([6]) *Suppose that $\mathbf{A}$ is a real $m \times n$ matrix. Let k be the target low rank and p be the oversampling parameter where $p \geq 2$ and $k + p \leq \min\{m, n\}$. Execute the proto-type algorithm (see Figure 2) with a Gaussian test matrix (see Figure 2) to obtain an $m \times (k+p)$ matrix $\mathbf{Q}$ with orthonormal columns. Then*

$$\mathbb{E}\|\mathbf{A} - \mathbf{Q}\mathbf{Q}^T\mathbf{A}\| \leq \left(1 + \frac{4\sqrt{k+p}}{p-1} \cdot \sqrt{\min\{m,n\}}\right) \sigma_{k+1}$$

where $\mathbb{E}$ denotes the expectation with respect to the random test matrix and σ_{k+1} is the $(k+1)$th singular value of $\mathbf{A}$.

If the proto-type algorithm is implemented in Step 1 of HOOI, we obtain the initial estimates $\mathbf{U_1}, \mathbf{U_2}, \mathbf{U_3}$. Then we have the following expectations for each mode:

$$\mathbb{E}\|\mathbf{T}^1 - \mathbf{U_1}\mathbf{U_1}^T\mathbf{T}^1\| \leq \left(1 + \frac{4\sqrt{k_1+p_1}}{p_1-1} \cdot \sqrt{\min\{I, JK\}}\right) \sigma^{(1)}_{k_1+1}$$

$$\mathbb{E}\|\mathbf{T}^2 - \mathbf{U_2}\mathbf{U_2}^T\mathbf{T}^2\| \leq \left(1 + \frac{4\sqrt{k_2+p_2}}{p_2-1} \cdot \sqrt{\min\{J, KI\}}\right) \sigma^{(2)}_{k_2+1}$$

and

$$\mathbb{E}\|\mathbf{T}^3 - \mathbf{U_3}\mathbf{U_3}^T\mathbf{T}^3\| \leq \left(1 + \frac{4\sqrt{k_3+p_3}}{p_3-1} \cdot \sqrt{\min\{K, IJ\}}\right) \sigma^{(3)}_{k_3+1}$$

where p_1, p_2, p_3 are the oversampling parameters corresponding to the low multilinear ranks k_1, k_2, k_3. In [6], there is further discussion on how the expectation can provide error bound for the approximation $\|\mathbf{A} - \mathbf{Q}\mathbf{Q}^T\mathbf{A}\|$.

3 Preliminary Results

We use a technique for a facial recognition algorithm via Matlab codes, adapting the the tensor-based technique used by Savas and Eldén [9] in classifying handwritten digits. Using facial images taken from the AT&T Face Database [1], each image is transformed into a matrix of grey scale value ranging from 0 to 256. The images are arranged into a tensor of dimension $112 \times 92 \times 400$, with each image as a 112×92 matrix and 400 images along mode-3.

We then organize a set of 40 predefined classes into subtensors of images of similar types (e.g. facial expression or person). Then, tensor analysis tools are used to extract salient information like the significant basis for each subtensor. For each subtensor, we define a least-squares residual, $\|1 - \sum_{k=1}^{R_3} \langle \mathbf{D}, \mathbf{B}^{\mu}_k \rangle\|$, where $\mathbf{D}$ is the unknown image. See Figure 4 for the algorithm.

The best-(R_1, R_2, R_3) low multilinear rank for the training tensors are computed using the standard HOOI [2]. Deterministic power methods for tensor decomposition have been studied in [2] and [8]. With efficient Matlab codes, normalized residuals and using the best-$(17, 17, 1)$ approximation of subtensors $\mathcal{T}^{\mu} \in \mathbb{R}^{112 \times 92 \times 10}$, the algorithm shows promising results and is able to classify the unknown image correctly. See Figures 5-6.

4 Conclusion and Future Outlook

We propose a randomized tensor-based method for image classification. The tensor formulation easily organizes the set of images. Thus, tensor analysis tools are easily applied. We applied the tensor based image classification algorithm on the AT&T Database.

In our future work, we will implement the randomized algorithm described in Section 2 for calculating the basis of each matrix (randomized HOOI) and provide some theorems which entail the quality of the approximation.

References

[1] AT & T Laboratories Cambridge. `http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html`

[2] L. De Lathauwer, B. De Moor, and J. Vandewalle. "On the Best Rank-1 and Rank-(R1,R2,...,RN) Approximation of Higher-Order Tensors," *SIAM J. Matrix Analysis and Applications*, vol. 21, pp. 1324-1342, 2000.

[3] L. De Lathauwer, B. De Moor, and J. Vandewalle. "A multilinear singular value decomposition," *SIAM*

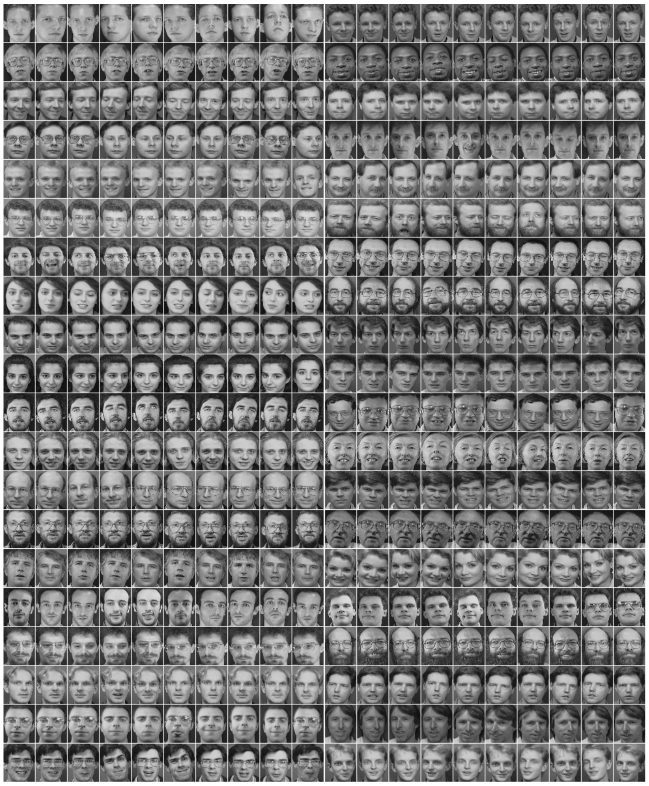

Figure 3: AT&T Face Data [1]. The data is rearranged into subtensors of images of similar types (e.g. facial expression, illumination or individual).

Input: Image tensor $\mathcal{T} \in \mathbb{R}^{112\times 92\times 400}$, unknown image $\mathbf{D}$
Output: $\mu^* \in \{1, 2, \ldots, 40\}$

Training Phase:

Sort the training faces tensor $\mathcal{T}$ into subtensors of the same type, called $\mathcal{T}^\mu \in \mathbb{R}^{112\times 92\times 10}$, $\mu = 1, 2, \ldots, 40$

Compute the best-(R_1, R_2, R_3) low multilinear rank for the trainining tensors $\mathcal{T}^\mu = \mathcal{B}^\mu \bullet_1 \mathbf{U} \bullet_2 \mathbf{V} \bullet_3 \mathbf{W}$

Compute and stored the normalized basis matrices $\mathbf{B}_k^\mu = \mathcal{B}^\mu(:, :, k) \bullet_1 \mathbf{U} \bullet_2 \mathbf{V}$, $k = 1, 2, \ldots, R_3$

Test Phase:

Normalized the unknown image $\mathbf{D}$

Compute the residual for each μ:
$r(\mu) = \|1 - \sum_{k=1}^{R_3} \langle \mathbf{D}, \mathbf{B}_k^\mu \rangle\|$

Calculate the minimum residual over μ;
i.e. $\mu^* = \mathrm{argmin}_\mu r(\mu)$

Classify $\mathbf{D}$ as μ^*.

Figure 4: Classification Algorithm via a Low Multilinear Rank HOOI

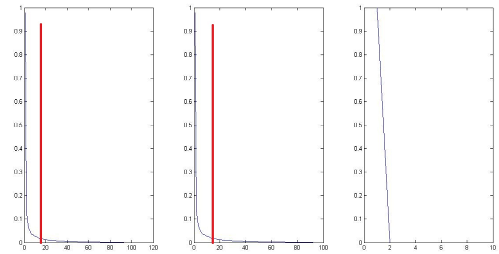

Figure 5: Singular values at each mode, truncated at $k_1 = k_2 = 17, k_3 = 1$. Computed best-$(17, 17, 1)$ low multilinear rank for training subtensors $\mathcal{T}^\mu$.

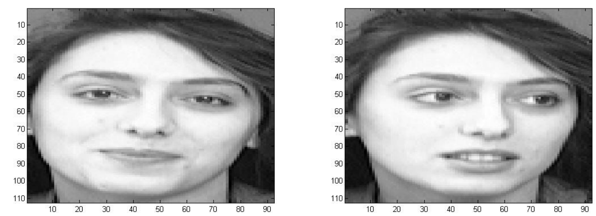

Figure 6: (Left) Test image $\mathbf{D}$. (Right) Return image from a training subtensor which classifies image $\mathbf{D}$.

Journal on Matrix Analysis and Applications, vol. 21, pp. 1253-1278, 2000.

[4] P. Drineas and M. Mahoney. "A randomized algorithm for a tensor-based generalization of the singular value decomposition," *Linear Algebra and Its Applications*, vol. 420, pp. 553-571, 2007.

[5] C. Eckart and G. Young. "The approximation of one matrix by another of lower rank," *Psychometrika*, vol. 1 (3), pp. 211-218, 1936.

[6] N. Halko, P.G. Martinsson and J. Tropp. "Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions," *SIAM Review*, vol. 53(2), pp. 217-288, 2011.

[7] C. Navasca and L. De Lathauwer. "Low Multilinear Rank Tensor Decomposition via Semidefinite Programming." *17th Proceedings of the European Signal Processing Conference (EUSIPCO),* Glasgow, Scotland, August 2009.

[8] P.A. Regalia and E. Kofidis. "The higher-order power method revisited: convergence proofs and effective initialization," *Proc. ICASSP-2000,* Istanbul, Turkey, June 2000.

[9] B. Savas and L. Eldén. "Handwritten digit classification using higher order singular value decompo-

sition," *Pattern Recognition*, vol. 40, pp. 993-1003, 2007.

[10] L.R. Tucker. "The extension of factor analysis to three-dimensional matrices," *Contributions to Mathematical Psychology*, H. Gulliksen and N. Frederiksen, eds., Holt, Rinehardt, & Winston, New York, 1963.

[11] L. R. Tucker. "Some mathematical notes on three-mode factor analysis," *Psychometrika*, vol. 31, pp. 279-311, 1966.

[12] R. Ward. "Methods for sparse analysis of high-dimensional data," *Lecture Notes in IAS Women and Mathematics Program on Sparsity and Computation*, Princeton, May 2011.