Simultaneous Tensor Subspace Selection and Clustering: The Equivalence of High Order SVD and K-Means Clustering

Heng Huang
Computer Science and
Engineering Department
University of Texas at Arlington
Arlington, Texas, USA
heng@uta.edu

Chris Ding
Computer Science and
Engineering Department
University of Texas at Arlington
Arlington, Texas, USA
chqding@uta.edu

Dijun Luo
Computer Science and
Engineering Department
University of Texas at Arlington
Arlington, Texas, USA
dluo@uta.edu

Tao Li
School of Computer Science
Florida International University
Miami, Florida, USA
taoli@cs.fiu.edu

ABSTRACT

Singular Value Decomposition (SVD)/Principal Component Analysis (PCA) have played a vital role in finding patterns from many datasets. Recently tensor factorization has been used for data mining and pattern recognition in high index/order data. High Order SVD (HOSVD) is a commonly used tensor factorization method and has recently been used in numerous applications like graphs, videos, social networks, etc.

In this paper we prove that HOSVD does simultaneous subspace selection (data compression) and K-means clustering widely used for unsupervised learning tasks. We show how to utilize this new feature of HOSVD for clustering. We demonstrate these new results using three real and large datasets, two on face images datasets and one on handwritten digits dataset. Using this new HOSVD clustering feature we provide a dataset quality assessment on many frequently used experimental datasets with expected noise levels.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications-Data Mining

General Terms

Algorithms

Keywords

HOSVD, K-Means Clustering, 2DSVD

1. INTRODUCTION

Tensor based dimension reduction has recently been extensively studied for data mining, machine learning, and pattern recognition applications. At the beginning, standard Principal Component Analysis (PCA) and Singular Value Decomposition (SVD) were popular as a tool for the analysis of two-dimensional arrays of data in a wide variety of applications. For example, Deerwester *et al.* [4] and Papadimitriou *et al.* [17] presented SVD based latent semantic indexing for automatic information indexing and retrieval; Billsus and Pazzani [3] used SVD for collaborative filtering method; Sirovich and Kirby used PCA for human facial images [12]; Turk and Pentland [16] proposed the well-known PCA based eigenface method for face recognition; and Alter *et al.* [2] employed SVD to reduce the dimensions of genome-wide expression data.

PCA and SVD work well for data dimension reduction of two-dimensional arrays, but it is not natural to apply them into higher dimensional data, known as high order tensors. Powerful tools have been proposed by Tucker decomposition [19]. In learning and vision area, there are several tensor based methods that have been proposed. Shashua and Levine [18] employed rank-1 to represent images; Yang *et al.* [21] proposed a two dimensional PCA (2DPCA) with column-by-column correlation of image; Ye *et al.* [22, 23] proposed a method called Generalized Low Rank Approximation of Matrices (GLRAM) to project the original data onto a two dimensional space with minimizing the reconstruction error. Ding and Ye proposed a non-iterative algorithm called two dimensional singular value decomposition (2DSVD) [7]. There are several other 3D tensor factorization methods [10] and the equivalence between them and GLRAM/2DSVD has been discussed in paper [11]. For higher dimensional tensors, Vasilescu and Terzopoulos presented High Order Singular Value Decomposition (HOSVD)

[20]. Higher Order Orthogonal Iteration (HOOI) [14] used an iterative method to do tensor decomposition in all indices. Ding *et al.* provided D-1 tensor reduction algorithm and error bound analysis of tensor factorization [6].

Although many studies on tensor factorization appeared in data mining, pattern recognition, and machine learning areas, to our knowledge, this paper is the first one to propose and prove the equivalence between HOSVD and tensor clustering. In previous research, there are several papers that ever explored the relations between unsupervised dimension reduction and unsupervised learning [5, 24].

The main contributions of our paper are as follows:
1) We propose and prove the equivalence of HOSVD and simultaneous subspace selection (compression) and K-means clustering with maintaining global consistency;
2) Experimental results on three public datasets demonstrate our theory;
3) Furthermore, our algorithm provides a dataset quality assessment method to help data mining and machine learning researchers to select the experimental datasets with their expected noise level.

2. TENSOR FACTORIZATION AND CLUSTERING

Consider a set of input data vectors $X = (\mathbf{x}_1, \cdots, \mathbf{x}_n)$ which can be also viewed as a 2D tensor $X = \{X_{ij}\}_{i=1}^{m} {}_{j=1}^{n}$. PCA is the most widely used dimension reduction method by finding the optimal subspace defined (spanned) by the principal directions $U = (\mathbf{u}_1, \cdots, \mathbf{u}_k) \in \Re^{m \times k}$. The projected data point in the new subspace are $V = (\mathbf{v}_1, \cdots, \mathbf{v}_n) \in \Re^{k \times n}$. PCA finds U and V by minimizing

$$\min_{U,V} J_{PCA} = ||X - UV||_F^2 \qquad (1)$$

In PCA or SVD, the Eckart-Young Theorem plays a fundamental role. Eckart-Young Theorem[8] states the optimization problem has PCA/SVD as its global solution and the optimal (minimum) value is the sum of eigenvalues.

$$J_{PCA}^{\mathrm{opt}} = \sum_{m=k+1}^{\min(p,n)} \lambda_m. \qquad (2)$$

where λ_m are eigenvalues of the covariance matrix XX^T.

2.1 High Order SVD

The input data is a 3D tensor: $X = \{X_{ijk}\}_{i=1}^{n_1} {}_{j=1}^{n_2} {}_{k=1}^{n_3}$. The rank-1 and HOSVD factorizations, treating every index uniformly, is

$$\min_{U,V,W,S} J_1 = ||X - U \otimes_1 V \otimes_2 W \otimes_3 S||^2 \qquad (3)$$

where U, V, W are 2d matrices and S is a 3D tensor. Using explicit index, $J_1 = \sum_{ijk} \left(X_{ijk} - \sum_{pqr} U_{ip} V_{jq} W_{kr} S_{pqr} \right)^2$. For HOSVD, $S \in \Re^{k_1 \times k_2 \times k_3}$; for rank-1 decomposition, S is diagonal: $S_{pqr} = m_p$ if $p = q = r$, $S_{pqr} = 0$ otherwise. In many cases, we require that W, U, V are orthogonal: $U^T U = I, V^T V = I, W^T W = I$.

2.2 GLRAM/2DSVD

The input data 3D tensor $X = \{X_{ijk}\}_{i=1}^{n_1} {}_{j=1}^{n_2} {}_{k=1}^{n_3}$ can be viewed as $X = \{X_1, \cdots, X_{n_3}\}$ where each X_i is a 2d matrix (*e.g.* an image) of size $n_1 \times n_2$. Therefore, instead of treating

every index equally as in J_1 of Eq.(3), As in Ye and Ding [7, 22], we leave the data index uncompressed and optimize

$$\min_{U,V,M} J_2 \quad = \quad ||X - U \otimes_1 V \otimes_2 M||^2$$

$$= \quad \sum_{\ell=1}^{n_3} ||X_\ell - U \otimes_1 V \otimes_2 M_\ell||^2$$

$$\text{s.t.} \quad V^T V = I, \ U^T U = I \qquad (4)$$

where $M = \{M_1, \cdots, M_{n_3}\}$. By definition,

$$[U \otimes_1 V \otimes_2 M_\ell]_{ij} = \sum_{p,q} U_{ip} V_{jq} M_{pq\ell}$$

$$= \sum_{p,q} U_{ip} (M_\ell)_{pq} (V^T)_{qj} = (U M_\ell V^T)_{ij}.$$

Thus, a 3D tensor factorization can also be written as

$$\min_{U,V,M_\ell} J_3 \quad = \quad \sum_{\ell=1}^{n_3} ||X_\ell - U M_\ell V^T||^2$$

$$\text{s.t.} \quad V^T V = I, \ U^T U = I. \qquad (5)$$

2.3 Tensor Clustering

Given a set of 1d tensors (vectors) $x_1, x_2, \cdots, x_n$, we can do K-means clustering

$$\min_{\{c_k\}} \sum_{\ell=1}^{n} \min_{1 \le k \le K} ||x_\ell - c_k||^2 = \sum_{k=1}^{K} \sum_{i \in c_k} ||x_\ell - c_k||^2 \qquad (6)$$

where c_k is the centroid vector of cluster c_k. This formalism can be extended to generic tensors. Given a three dimensional tensor X, or, equivalently a set of two dimensional images $X^{(1)}, X^{(2)}, \cdots, X^{(n)}$, the K-means tensor clustering minimizes

$$\min_{\{C_k\}} \sum_{\ell=1}^{n} \min_{1 \le k \le K} ||X^{(\ell)} - C_k||^2 = \sum_{k=1}^{K} \sum_{\ell \in C_k} ||X_k^{(\ell)} - C_k||^2 \qquad (7)$$

where C_k is the centroid tensor of cluster C_k. Now suppose we carried out a 2DSVD (when $d = 3$, otherwise, we need to use a generalized version of 2DSVD) on X into U, V, and $\{M_\ell\}$. Using the distance relationship, the tensor clustering can be done entirely in $\{M_\ell\}$:

$$\min_{\{C_k\}} \sum_{\ell=1}^{n} \min_{1 \le k \le K} ||M^{(\ell)} - C_k||^2 = \sum_{k=1}^{K} \sum_{\ell \in C_k} ||M_k^{(\ell)} - C_k||^2 \qquad (8)$$

where C_k is the centroid tensor of cluster C_k.

3. SIMULTANEOUS COMPRESSION (SUBSPACE SELECTION) AND CLUSTERING

The main purpose of this paper is to provide new insights to understand the relations between HOSVD and tensor clustering. In this section, we first prove HOSVD is equivalent to simultaneous subspace selection and K-means clustering. After that, we provide the algorithm to find the clustering indicators from HOSVD results.

3.1 The Equivalence Theorem

The HOSVD does simultaneous 2DSVD data compression (subspace selection) and K-means clustering. In Eq.3, matrices U and V include the subspaces after projection and

matrix W gives out the clustering results on the data index direction.

THEOREM 1. *The HOSVD factorization of Eq.3 is equivalent to simultaneous 2DSVD data compression of Eq.5 and K-means clustering of Eq.8.*

Proof. We will prove the following:
1) Solution of W in HOSVD is the cluster indicator to K-means clustering (tensor clustering) of Eq.8;
2) (U, V) in 2DSVD of Eq.5 is the same (U, V) in HOSVD of Eq.3;

Given a three dimensional tensor $X_{n_1 n_2 n_3}$, or, equivalently a set of two dimensional images $X^{(1)}, X^{(2)}, \cdots, X^{(n_3)}$ with size $n_1 \times n_2$ (we have $X_{ijl} = X_{ij}^{(l)}$), the objective function of tensor clustering is (please see Eq.8):

$$\min_{\{C_k\}} J_K = \sum_{k=1}^{K} \sum_{\ell \in C_k} ||M_k^{(\ell)} - C_k||^2, \qquad (9)$$

where $M^{(\ell)} = U^T X^{(\ell)} V$. We prove Eq.9 is equivalent to Eq.3.

Eq.9 can be written as:

$$\min_{\{C_k\}} J_K = \sum_{\ell=1}^{n_3} ||M^{(\ell)}||^2 - \sum_{k=1}^{K} \frac{1}{n_k} \sum_{\ell, \ell' \in C_k} \text{Tr}(M^{(\ell) \, T} M^{(\ell')}), \qquad (10)$$

where n_k is the number of images in cluster C_k. The solution of tensor clustering is represented by K non-negative indicator vectors:

$$Q = (\mathbf{q}_1, \cdots, \mathbf{q}_K), \quad \mathbf{q}_k^T \mathbf{q}_l = \delta_{kl}. \qquad (11)$$

where

$$\mathbf{q}_k = (0, \cdots, 0, \overbrace{1, \cdots, 1}^{n_k}, 0, \cdots, 0)^T / \sqrt{n_k} \qquad (12)$$

Let's denote:

$$\begin{aligned} A_{\ell\ell'} &= \text{Tr}(M^{(\ell) \, T} M^{(\ell')}) \\ &= \text{Tr}(V^T X^{(\ell) \, T} U U^T X^{(\ell')} V). \end{aligned} \qquad (13)$$

We rewrite Eq.10 using Eq.11 and Eq.13:

$$\min_Q J_K = \text{Tr}(M^T M) - \text{Tr}(Q^T A Q). \qquad (14)$$

The first item is a constant. Thus min J_K becomes

$$\begin{aligned} \max_Q J_K &= \text{Tr}(Q^T A Q) \\ \text{s.t.} \quad & Q^T Q = I, \end{aligned} \qquad (15)$$

and

$$(QQ^T)_{\ell\ell'} = \begin{cases} 0 & \text{if} \quad M^{(\ell)} \text{ or } M^{(\ell')} \notin C_k \\ 1/n_k & \text{if} \quad M^{(\ell)} \text{ and } M^{(\ell')} \in C_k \end{cases} \qquad (16)$$

The solution to Eq.3 can be derived using the following functions:

$$\max_W \text{Tr}(W^T H W) \text{ s.t. } W^T W = I. \qquad (17)$$

$$\max_U \text{Tr}(U^T F U) \text{ s.t. } U^T U = I. \qquad (18)$$

$$\max_V \text{Tr}(V^T G V) \text{ s.t. } V^T V = I. \qquad (19)$$

where

$$\begin{aligned} F_{ii'} &= \sum_{jj'\ell\ell'} X_{ij\ell} X_{i'j'\ell'} (VV^T)_{jj'} (WW^T)_{\ell\ell'} \\ &= \sum_{\ell\ell'} (X^{(\ell)} VV^T X^{(\ell') \, T})_{ii'} (WW^T)_{\ell\ell'} \end{aligned} \qquad (20)$$

$$\begin{aligned} G_{jj'} &= \sum_{ii'\ell\ell'} X_{ij\ell} X_{i'j'\ell'} (UU^T)_{ii'} (WW^T)_{\ell\ell'} \\ &= \sum_{\ell\ell'} (X^{(\ell)} UU^T X^{(\ell') \, T})_{ii'} (WW^T)_{\ell\ell'} \end{aligned} \qquad (21)$$

$$\begin{aligned} H_{\ell\ell'} &= \sum_{ii'jj'} X_{ij\ell} X_{i'j'\ell'} (UU^T)_{ii'} (VV^T)_{jj'} \\ &= \text{Tr}(V^T X^{(\ell) \, T} UU^T X^{(\ell')} V) \end{aligned} \qquad (22)$$

Obviously, Eq.22 is the same as Eq.13. Therefore, objective function Eq.17 is equivalent to Eq.15. Thus, we have proved HOSVD is doing K-means clustering. The equivalence of (U, V) in 2DSVD and (U, V) in HOSVD will be discussed in Lemma 1 in next section.

3.2 Global Consistence Lemma

Considering the following situation:
A) We do 2DSVD first to obtain $M^{(\ell)}$.
B) We do K-means clustering in the subspace of $\{M^{(\ell)}\}$. This is equivalent to HOSVD as proved in section 3.1.
C) We would like to do 2DSVDs on all K clusters simultaneously, but this would create different subspaces (U_k, V_k) $(k = 1, \cdots, K)$. Instead, we enforce the subspaces (U_k, V_k) to be a single pair of global basis (U, V).
Go back to A) until convergence. After convergence, the results are identical to solutions of HOSVD

Lemma 1: *The converged results (U, V) of steps A)–C) are identical to the (U, V) in solutions of HOSVD.*

Proof. Step C) is solved by K 2DSVDs and solution of each 2DSVD is given by:

$$\max_U \text{Tr}(U^T \tilde{F}_k U) \text{ s.t. } U^T U = I. \qquad (23)$$

$$\max_V \text{Tr}(V^T \tilde{G}_k V) \text{ s.t. } V^T V = I. \qquad (24)$$

where

$$\tilde{F}_k = \frac{1}{n_k} \sum_{\ell \in C_k} (X^{(\ell)} V_k V_k^T X^{(\ell) \, T}), \qquad (25)$$

$$\tilde{G}_k = \frac{1}{n_k} \sum_{\ell \in C_k} (X^{(\ell)} U_k U_k^T X^{(\ell) \, T}), \qquad (26)$$

If we solve for K 2DSVDs separately, (U_k, V_k) will be different based on $\tilde{F}_k$ and $\tilde{G}_k$. When we perform K 2DSVDs simultaneously as step C), the objective functions are:

$$\begin{aligned} \tilde{F} &= \sum_{k=1}^{K} \tilde{F}_k \\ &= \sum_{k=1}^{K} \frac{1}{n_k} \sum_{\ell \in C_k} (X^{(\ell)} VV^T X^{(\ell) \, T}) \\ &= \sum_{\ell} \frac{1}{n_\ell} (X^{(\ell)} VV^T X^{(\ell) \, T}) \end{aligned} \qquad (27)$$

$$\begin{aligned}
\tilde{G} &= \sum_{k=1}^{K} \tilde{G}_k \\
&= \sum_{k=1}^{K} \frac{1}{n_k} \sum_{\ell \in C_k} (X^{(\ell)} U U^T X^{(\ell)\,T}) \\
&= \sum_{\ell} \frac{1}{n_\ell} (X^{(\ell)} U U^T X^{(\ell)\,T}) \qquad (28)
\end{aligned}$$

After convergence, the iterations of A)–C) stop.

Note that, F in Eq.20 is identical to $\tilde{F}$ of Eq.27 using Eq.16. Similar relation holds for G and $\tilde{G}$. Therefore, (U, V) in 2DSVD of Eq.5 are equivalent to (U, V) in HOSVD of Eq.3. □

In summary, we have proved that HOSVD is performing simultaneous 2DSVD and K-means clustering while maintaining global consistence. This is the major contribution of this paper.

3.3 Illustration of Theorem 1

According to Theorem 1, the matrix W in HOSVD solutions is the indicator to clustering results. The clustering indicator can be found by applying clustering method into matrix W. In this paper, we apply K-means clustering method on matrix W to obtain the clustering indicator.

We demonstrate the equivalence theorem by using AT&T face image database [1] (40 subjects, each subject has 10 different images with size 112×96, please see the more detailed data description in section 4) to illustrate the visualization results of confusion matrix $C = WW^T$. Later we will show more experimental results in next section. After performing the HOSVD on a $112 \times 96 \times 400$ image tensor with reduced dimensions as $30 \times 30 \times 400$, we get a 400×30 matrix W and use it to calculate the confusion matrix C. Since HOSVD does both compression and clustering simultaneously, different compression rates affect the clustering results. Here, we choose the commonly used 30×30 as the image size after compression.

In order to get a clear representation, we resize the original 400×400 matrix C to a new 40×40 matrix. Each square represents a 10×10 matrix in the original one and includes all images of one subject. The areas of squares are the sum of values in matrix. Fig. 1 shows the visualization of confusion matrix C. Compared to the small size squares, the large squares on the diagonal denote more images are clustered into one cluster. Therefore, there exists clustering results in matrix W of HOSVD. Here, we use K-means clustering method to find them from the HOSVD results.

The algorithm for finding clustering indicator from HOSVD results is presented in Table 1. Here we only use K-means clustering method to find the clustering results that already exist in HOSVD factorization results. In 2DSVD + K-means clustering, K-means clustering method really processes the non-clustered data into clusters.

The clustering mechanism of HOSVD is similar to use PCA (each image is resized as one vector) to do data projection, but HOSVD uses a more efficient way – tensor factorization.

4. EXPERIMENTAL RESULTS

In this section, we experimentally demonstrate the proposed new insights to understand relations among HOSVD, 2DSVD + K-means clustering, and PCA + K-means clus-
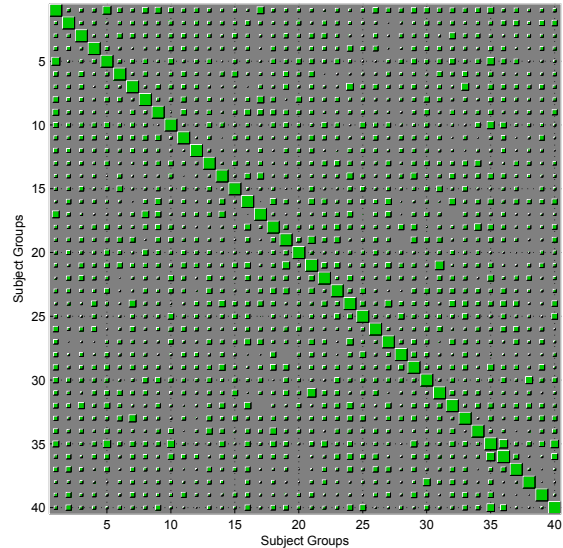
Figure 1: The visualization of confusion matrix C of HOSVD result using AT&T image dataset.

1) Compute HOSVD to get matrix W;
2) Run K-means clustering on matrix W and get clustering indicator.

Table 1: Algorithm to find clustering indicator from HOSVD results.

tering. Two well known datasets will be used: one is AT&T face images dataset and another one is MNIST hand-written digits dataset.

4.1 AT&T Face Image Databases

At first, the benchmark face databases AT&T [1] is used to demonstrate the simultaneous compression (subspace projection) and clustering of HOSVD. In the AT&T database, there are ten different images of each of 40 distinct subjects. For some subjects, the images were taken at different times, varying the lighting, facial expression (open/close eyes, smiling/no-smiling) and facial details (glasses/no glasses). All images were taken against a dark homogeneous background with the subjects in an upright, frontal, position (with tolerance for some side movement).

We explore three methods (PCA + K-means clustering, 2DSVD + K-means Clustering, and HOSVD) using AT&T dataset and the experimental steps are described as follows:
1) **PCA + K-means clustering.** We reshape each image (image size is 102×92) into one vector and all 400 images consist of a 9384×400 matrix. PCA is used to do subspace projection with the same dimensionality as the number of distinct subjects, $X = UV$. K-means clustering is employed on data projection matrix V to obtain clusters (K=40 for AT&T datasets).
2) **2DSVD + K-means Clustering.** 2DSVD is applied to the $102 \times 92 \times 400$ image tensor for data compression with reduced dimensions 30×30. From Eq.5, $X_l = U M_l V$, $l = 1 \cdots 400$). As described in section 2.3, K-means Clustering method is used to cluster M_l with objective function Eq.8.

3) **HOSVD.** We perform HOSVD on the $102 \times 92 \times 400$ image tensor to do simultaneous compression and clustering with reduced dimensions $30 \times 30 \times 40$. After that, K-means Clustering method helps us find the cluster indicator from matrix W.

Since there are 40 distinct subjects in AT&T database, all previous papers using this public dataset consider all 400 images as 40 natural clusters and each cluster includes 10 different images of the same subject. After using three above methods, the final results are represented as 400×40 matrices Q. Each row labeled by one image and each column shows one cluster, *e.g.*, Q_{ij} means image i is clustered into cluster j. We hope to group all ten images of the same subject together to observe how many images of one subject are clustered into the same cluster (the remaining images are clustered into wrong clusters). But in matrix Q the subject label is not identical to the cluster label. In order to demonstrate the clustering results better, we perform the following steps to generate a new 40×40 matrix I:

1) We group every ten images of the same subject together along vertical axis and sum their values together by columns: $I_{ij} = \sum_{k=1}^{10} Q_{i_k j} \delta_{i_k j}$, if image i_k is clustered into cluster j, $\delta_{i_k j} = 1$; otherwise, $\delta_{i_k j} = 0$. As a result, each row in new matrix represents one subject and one column represents one cluster;

2) Using the new matrix I, we build a bipartite graph $G = (V = V_1 \cup V_2, E)$ with two sets of 40 vertices V_1, V_2 and map V_1 to the 40 rows and V_2 to the 40 columns in matrix I;

3) The value of edge $e(i, j)$ in the bipartite graph are defined as the value of $I_{ij}, i, j = 1, \cdots, 40$. Using Hungarian algorithm [13], we do bipartite graph cut with maximizing weight matching.

The result is a new matrix P and the image group number is identical to the cluster number if the images from that group are not clustered into other subjects' clusters. We visualize P_{PCA} in Fig. 2(a), P_{2DSVD} in Fig. 2(b), and P_{HOSVD} in Fig.2(c), respectively. In these three figures, each row illustrates one subject and each column represents one cluster. The green squares show the number of images are clustered into one cluster. The areas of squares are proportional to the number of images clustered together, from 10 to 0. Because most images are clustered into the default subject cluster and few images are clustered into other subjects' clusters, the large squares mostly locate on the diagonals and some small squares are scattered into the other areas.

We also calculate the clustering accuracy of these three methods. The clustering accuracy is defined by the ratio of the number of images that are clustered into the default subject clusters and the number of total images. The number of images with correct labels can be directly computed by summing all the squares' areas on diagonal in each figure of Fig. 2(a), Fig. 2(b), and Fig.2(c). Table 2 illustrates the clustering accuracy comparisons of three methods.

In the beginning, we use all 400 images to do clustering and notice there are some misclustered images for every method. Because images from one subject are not always similar, some of them can even be treated as outliers (noise). Thus, we hope to select the subset of images from original one with fewer outlier images. We consider using the superset of correct clustering results from different methods. After running three methods on 400 images, we select the groups in which at least n (we use $n = 8$ and 10 for two
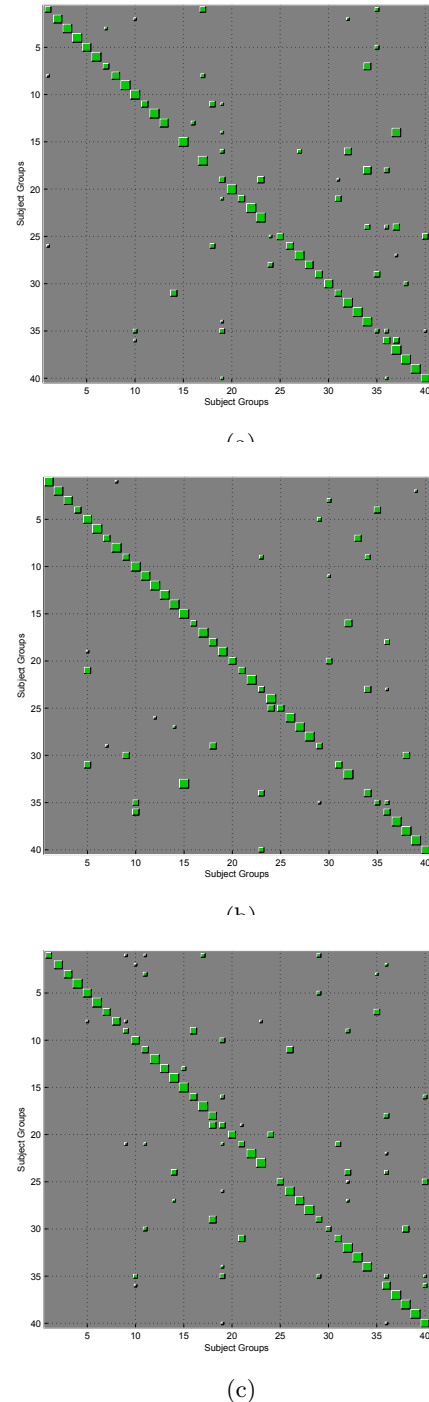
Figure 2: Visualization of (a) PCA + K-means clustering, (b) 2DSVD + K-means clustering, and (c) HOSVD results on AT&T dataset. Each row represents one subject and each column denotes one cluster. The green squares show the number of images clustered into the same cluster.

Figure 4: Sample images of MNIST hand-written digits dataset.

subsets of images selection) images are clustered into the default cluster by at least one of three methods and merge all of them together to create the new dataset. When $n = 8$, we have 30 distinct subjects with 300 images; $n = 10$, we have 22 distinct subjects with 220 images. Three methods are performed on the new datasets again and the clustering accuracy values are summarized into column 3 and 4 in Table 2. Our experimental results show the clustering accuracy is improved when the selection quality of subset images increases.

	All images (400)	Subset images (300)	Subset images (220)
PCA + K-means	70.5%	74.5%	80.2%
2DSVD + K-means	73.5%	78.7%	83.6%
HOSVD	74.0%	80.7%	84.5%

Table 2: The clustering accuracy comparison of PCA + K-means clustering, 2DSVD + K-means clustering, and HOSVD methods using AT&T datasets.

In order to understand the clustering results better, we also pay attention to the images that are not clustered into the default subject cluster. For example, results of three clusters are illustrated into Fig.3. Images on each row are clustered into the same cluster by HOSVD method. Every cluster includes images from more than one subject. The white lines are used to separate the images of different subjects. Because the images from AT&T face database vary in the lighting, facial expression, and facial details, images of some subjects are far away to other images of the same subject and more closer to images of other subjects. The images within the same subject group are not homogeneous. Therefore, if we use distinct subjects as the default labels, the original datasets don't always have the correct cluster labels. We will discuss this issue more in next section.

4.2 MNIST Hand-written Digit Dataset

Here, we present experimental results on the MNIST hand-written digits dataset, which consists of 60,000 training and 10,000 test digits [15]. The MNIST dataset can be downloaded from "http://yann.lecun.com/exdb/mnist/" with 10 classes, from digit "0" to "9". In the MNIST dataset, each image is centered (according to the center of mass of the pixel intensities) on a 28x28 grid. Fig.4 displays sample images of hand-written digits.
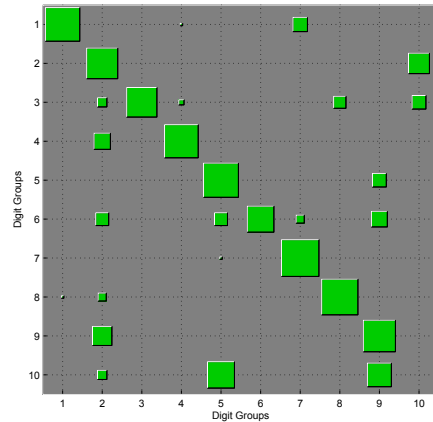

Figure 5: Visualization of HOSVD results on MNIST hand-written digit dataset. Each row is one digit and each column is one cluster. The green squares show the number of digits clustered into the same cluster.

In our experiments, we randomly choose 1000 images (i.e., each digit has 100 images) and apply HOSVD factorization on this $28 \times 28 \times 1000$ tensor. The reduced dimensions are $10 \times 10 \times 10$. Using the same way to plot Fig. 2(a), Fig. 2(b), and Fig.2(c), we visualize the clustering results of HOSVD in Fig.5. Since the areas of squares on diagonal are much larger than the areas of other small squares, the clustering results are pretty good.

5. DATASET QUALITY ASSESSMENT

In our experiments, the clustering results tell us that the images within the same subject group are not always consistent. For example, in face image databases, the same person can have different poses, lights, expression details, and there is also misalignment problem between images. The figures of one person can be more similar to other's than his own images. The digit images also have this situation.

In most high dimensional datasets, there exist outliers leading such data inconsistent problem and confusing the data mining and pattern analysis algorithms. How can we select the high dimensional datasets or subdatasets with fewer outliers? Based on the proposed clustering feature of HOSVD, we provide a dataset quality assessment using HOSVD to help other researchers select a subset of high dimensional data with less noise. Two well known public face image databases will be used to illustrate this new function of HOSVD.

5.1 The AT&T Faces Database

We ever show the examples of different people who are clustered into the same cluster in Fig.3. Here, we draw out several examples in Fig.6 to demonstrate why the images of the same subject are clustered into different clusters during K-means clustering. Six examples are illustrated on each row in Fig.6, respectively. The white lines are used to separate the images of each subject through the clustering (HOSVD) results.

In Fig.6, the subject shown as the first example (first

Figure 3: The clustering results using HOSVD method. Each row shows the images of different subjects are clustered into the same cluster. White lines are used to separate the images of different subjects.

row) has different face directions in his ten images: his jaw changes directions from right to left, and then to center. These different face directions create misalignment errors during image matching. As a result, his ten images are clustered into five different clusters. The subject in the second example has different face directions and different face sizes in these ten images and those differences effect the clustering results. Both third and fourth examples are clustered into two clusters. One has two different face directions and the other one has different face details (the glass). The fifth and sixth examples show the consistent images within the same subject cluster without separation during clustering process.

In order to quantitatively observe the difference between those images, we calculate the average distance of each pair images of the subject. We call it as *dispersion distance* and define it as follows:

$$D_{dispersion} = \sum_{i,j} \frac{||X_i - X_j||^2}{\sum_i ||X_i||^2} / \frac{n(n-1)}{2}, \qquad (29)$$

where X_i and X_j are images of the same subject, and $n = 10$. The dispersion distances of six examples in Fig.6 are 0.56, 0.68, 0.65, 0.51, 0.49, and 0.46. The fifth and sixth examples have small dispersion distances compared to other examples. These dispersion distances help us measure the consistency of high dimensional data under the same category.

So far many data mining and pattern analysis algorithms are tested on the public face databases, *e.g.* AT&T databases, but the noise within those public datasets definitely bothers the evaluations and comparisons of their methods.

Now we show how clustering accuracy changes using different image subsets. At first, the HOSVD method is performed on all images. Based on the results, we select the subjects which at least have n images been clustered into the default subject cluster. When n ranges from 1 to 10, we create ten image subsets. We run HOSVD method again on the selected subset images and the clustering accuracy results are shown in Fig. 7. The numbers of x-axis denote the n values and the values on y-axis are the clustering accuracy. The red line is the clustering accuracy using all 400 images. This figure illustrates how images of the same subject are
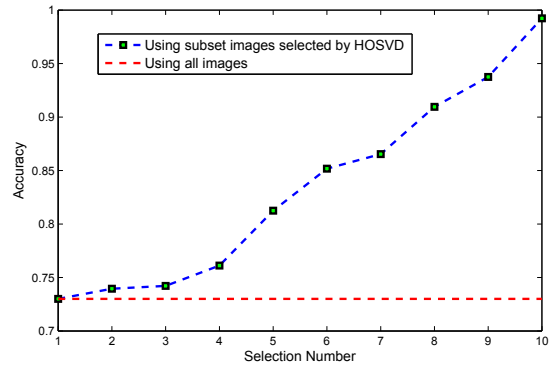
Figure 7: HOSVD results after selecting the subset images for experiments. The values of x-axis denote the n values for subset images selection and the values on y-axis are the clustering accuracy. The red line is the clustering accuracy using all 400 images.

consistent in different subset images. Researchers can select the subset images based on their expected noise level.

5.2 The Yale Face Database B

The other face image database used in our experiment is the combination of extended and original Yale database B [9]. These two databases contain single light source images of 38 subjects (10 subjects in original database and 28 subjects in extended one) under 576 viewing conditions (9 poses x 64 illumination conditions). We fixed the pose. Thus, for each subject, we obtained 64 images under different lighting conditions. The facial areas were cropped into the final images for matching [9], including: 1) preprocessing to locate the faces was applied; 2) original images were normalized (in scale and orientation) such that the two eyes were aligned at the same position. The size of each cropped image in our experiments is 192 × 168 pixels, with 256 gray levels per pixel. Because there is a set of images which are corrupted during the image acquisition [9], we have 31 subjects

Figure 6: The HOSVD results separate the images of several people into different clusters. AT&T face image datasets are used. Each row includes all ten images of the same subject. The white lines are used to separate the images that are clustered into different clusters.
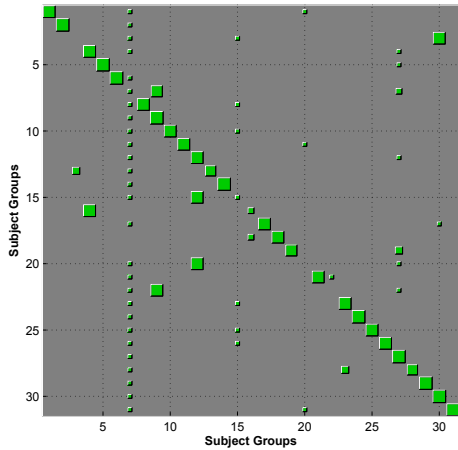

Figure 8: Visualization of HOSVD method using images from the Yale Face Database B.

Figure 9: All 27 images from different subjects are clustered into cluster 7 in Fig.8.

in total without any corrupted images. We randomly select ten illumination conditions for all 31 subjects to create the experimental dataset with 310 images.

We employ HOSVD factorization on the $192 \times 168 \times 310$ tensor with reduced dimensions as $30 \times 30 \times 31$. Using the same way to draw clustering results in section 4.1, we visualize the HOSVD clustering results in Fig.8. Although the main large green squares locate on the diagonal, several
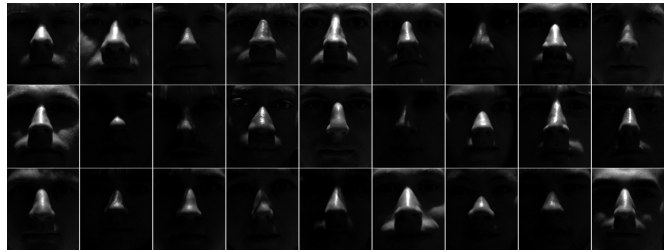
small ones still scatter in other regions. Some images are not consistent with other images of the same subject.

In Fig.8, cluster 7 is an interesting cluster, because there are 27 images from different subjects are clustered into the same cluster. We display all these 27 images in Fig.9. Because these images use dark ambient light, different people's faces are not recognizable. Since they look all similar (dark) and are different to other images of the same subject, we don't believe they are helpful in face images mining and recognition.

6. CONCLUSION

Because numerous data mining and machine learning applications can be handled by matrices or tensors, tensor factorization methods are growing increasingly popular as pow-

erful dimension reduction techniques. In this paper, we first propose a new theorem of HOSVD that it does simultaneous 2DSVD subspace selection (compression) and K-means clustering while maintaining the global consistence. A rigorous proof is provided for this novel insight of relation between HOSVD and tensor clustering. Using the same technique of proving Theorem 1, we can also prove that the ParaFac tensor decomposition is also equivalent to a K-means clustering.

We provide experiments to demonstrate our theoretical results on three public datasets. In experiments, we compare HOSVD method with PCA + K-means clustering and 2DSVD + K-means clustering methods. The experimental results show that HOSVD gives out both data compression and clustering results.

Furthermore, our experimental results suggest that the high dimensional data are not consistent in those public datasets, because of the outliers. In order to select the clean datasets with expected noise level for data mining and machine learning research experiments, we provide a HOSVD based dataset quality assessment method and use it to find interesting results from two well known face image datasets.

Acknowledgments

7. REFERENCES

[1] http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html.

[2] O. Alter, P. O. Brown, and D. Botstein. Singular value decomposition for genome-wide expression data processing and modeling. *Proceedings of the National Academy of Sciences*, 97:10101–10106, 2000.

[3] D. Billsus and M. J. Pazzani. Learning collaborative information filters. In *ICML '98: Proceedings of the Fifteenth International Conference on Machine Learning*, pages 46–54, 1998.

[4] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41:391–407, 1990.

[5] C. Ding and X. He. K-means clustering via principal component analysis. *Proc. of Int'l Conf. Machine Learning*, pages 225–232, 2004.

[6] C. Ding, H. Huang and D. Luo. Tensor Reduction Error Analysis - Applications to Video Compression and Classification. *IEEE Conf. on Computer Vision and Pattern Recognition*, 2008.

[7] C. Ding and J. Ye. Two-dimensional singular value decomposition (2dsvd) for 2d maps and images. *SIAM Int'l Conf. Data Mining*, pages 32–43, 2005.

[8] C. Eckart and G. Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1:183–187, 1936.

[9] A. Georghiades, P. Belhumeur, and D. Kriegman. From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Trans. Pattern Anal. Mach. Intelligence*, 23(6):643–660, 2001.

[10] K. Inoue and K. Urahama. Dsvd: A tensor-based image compression and recognition method. *IEEE Int. Symp. on Circ. and Syst.*, pages 6308–311, 2005.

[11] K. Inoue and K. Urahama. Equivalence of non-iterative algorithms for simultaneous low rank approximations of matrices. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1:154–159, 2006.

[12] Kirby and Sirovich. Application of the kl procedure for the characterization of human faces. *IEEE Trans. Pattern Anal. Machine Intell.*, 12:103–108, 1990.

[13] H. W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, pages 83–97, 1955.

[14] L. D. Lathauwer, B. D. Moor, and J. Vandewalle. On the best rank-1 and rank-(r1, r2, . . . , rn) approximation of higher-order tensors. *SIAM J. Matrix Anal. Appl.*, 21:1324–1342, 2000.

[15] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proc. IEEE*, 86(11):2278–2324, 1998.

[16] T. M. and Pentland. Eigen faces for recognition. *Journal of Cognitive Neuroscience*, 3:71–86, 1991.

[17] C. H. Papadimitriou, P. Raghavan, H. Tamaki, and S. Vempala. Latent semantic indexing: A probabilistic analysis. *In Proceedings of the ACM Conference on Principles of Database Systems*, pages 159–168, 1998.

[18] A. Shashua and A. Levin. Linear image coding for regression and classification using the tensor-rank principle. *IEEE Conf. on Computer Vision and Pattern Recognition*, 2001.

[19] L. Tucker. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31(3):279–311, 1966.

[20] M. Vasilescu and D. Terzopoulos. Multilinear analysis of image ensembles: Tensorfaces. *European Conf. on Computer Vision*, pages 447–460, 2002.

[21] J. Yang, D. Zhang, A. F. Frangi, and J. Yang. Two dimensional PCA: A new approach to appearance based face representation and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(1), 2004.

[22] J. Ye. Generalized low rank approximations of matrices. *International Conference on Machine Learning*, 2004.

[23] J. Ye, R. Janardan, and Q. Li. GPCA: An efficient dimension reduction scheme for image compression and retrieval. *ACM KDD*, pages 354–363, 2004.

[24] H. Zha, C. Ding, M. Gu, X. He, and H. Simon. Spectral relaxation for k-means clustering. *Neural Information Processing Systems*, 14:1057–1064, 2001.