Truncated t-SVD Methods for Facial Recognition

Ning Hao\textsuperscript{1} Misha E. Kilmer\textsuperscript{1} Karen Braman\textsuperscript{2} Randy Hoover\textsuperscript{2}

\textsuperscript{1}Tufts University
\textsuperscript{2}South Dakota School of Mines and Technology

Kilmer and Hao’s work supported by NSF-DMS 0914957
Traditional PCA (aka Eigenfaces)

- Vectorize each image and put it as a column in matrix $D$
- Let $A = D - M$, where $M$ has the mean image in each column
- The principal components are the left singular vectors of $A$
- Truncate $U$ to only $k$ columns.
  \[ U_{:,1:k} U_{:,1:k}^T A_{:,i} \approx A_{:,i} \]  
  saved
- Given input image $B$, let $b = \text{vec}(B)$, reconstructed image is
  \[ b_{rec} = U_{:,1:k} U_{:,1:k}^T (b - \mu) + \mu \]
- Matches by examining expansion coefficients $U_{:,1:k}^T (b - \mu)$ vs. $U_{:,1:k}^T A_{:,k}$.
Tensor-based Approaches

Arrange the data in a multi-way array (tensor) to better exploit features.

Previous Work:

- TensorFaces [Vasilescu and Terzopoulos, 2002, 2003]: Arrange the data in multi-way format, to account for pixels, lighting, illumination, etc. PCA-like factorization via HOSVD [de Lathauwer et al] and/or higher-order Tucker decomposition.

- Tensor-subspace analysis [He, Cai and Niyogi, 2006] which is similar to the approach we advocate images themselves are considered as two-dimensional objects rather than vectorized, but differs in that it uses a graph theoretic (Laplacian based) approach.
Tensor Terminology

Fibers:
- First mode: columns
- Second mode: rows
- Third mode: tubes

Slices:
- Horizontal slices
- Vertical slices
- Frontal slices

Graphics thanks to Tammy Kolda
Goal: t-SVD based PCA-like algorithm

- Each image is *vertical slice* in a third order tensor $\mathcal{D}$
- Let $\mathcal{A} = \mathcal{D} - \mathcal{M}$, where $\mathcal{M}$ has the mean image in each vertical slice.
- Compute $\mathcal{A} = \mathbf{u} \ast \mathbf{s} \ast \mathbf{v}^T$
- First $k$ vertical slices of $\mathbf{u} \rightarrow$ “basis” elements.
- So $\mathbf{u}_{:,1:k,:} \ast \mathbf{u}_{:,1:k,:}^T \ast \mathbf{A}_{:,i,:} \approx \mathbf{A}_{:,i,:}$ \underbrace{{\text{saved}}}_{\text{saved}}$
- Match of input image $B$ based on comparisons of the expansion coefficients, which are now tube fibers!
Toward Defining Tensor-Tensor Multiplication

For $\mathcal{A} \in \mathbb{R}^{m \times p \times n}$, let $A^{(i)} = \mathcal{A}_{\cdot \cdot \cdot \cdot \cdot \cdot i}$. 

\[
\begin{bmatrix}
A^{(1)} \\
A^{(2)} \\
\vdots \\
A^{(n)}
\end{bmatrix} \in \mathbb{R}^{mn \times p}
\]

\[
\begin{bmatrix}
A^{(1)} \\
A^{(2)} \\
\vdots \\
A^{(n)}
\end{bmatrix} \in \mathbb{R}^{m \times p \times n}
\]
The block circulant matrix generated by unfold ($\mathcal{A}$) is

$$\text{circ} (\mathcal{A}) = \begin{bmatrix}
A^{(1)} & A^{(n)} & \ldots & A^{(3)} & A^{(2)} \\
A^{(2)} & A^{(1)} & A^{(n)} & \ldots & \ldots \\
A^{(3)} & A^{(2)} & \ldots & \ldots & \ldots \\
\vdots & \vdots & \ddots & \ddots & \ddots \\
A^{(n)} & \ldots & \ldots & A^{(2)} & A^{(1)}
\end{bmatrix}$$
A block circulant can be block-diagonalized by a (normalized) DFT in the 2nd dimension:

\[(F \otimes I)\text{circ}(\mathcal{A})(F^* \otimes I) = \begin{bmatrix}
\hat{A}^{(1)} & 0 & \cdots & 0 \\
0 & \hat{A}^{(2)} & 0 & \cdots \\
0 & \cdots & \ddots & 0 \\
0 & \cdots & 0 & \hat{A}^{(n)}
\end{bmatrix}\]

Conveniently, an FFT along tube fibers of \(\mathcal{A}\) gives \(\hat{A}\).
Tensor - Tensor Multiplication

[K., Martin, Perrone '08]: For $\mathbf{A} \in \mathbb{R}^{m \times p \times n}$ and $\mathbf{B} \in \mathbb{R}^{p \times q \times n}$, define the \textit{t}-product

$$\mathbf{A} \ast \mathbf{B} \equiv \text{fold} \left( \text{circ} (\mathbf{A}) \cdot \text{unfold} (\mathbf{B}) \right).$$

Result is $m \times q \times n$.

Example: $\mathbf{A} \in \mathbb{R}^{m \times p \times 3}$ and $\mathbf{B} \in \mathbb{R}^{p \times q \times 3}$,

$$\mathbf{A} \ast \mathbf{B} = \text{fold} \left( \begin{bmatrix} A^{(1)} & A^{(3)} & A^{(2)} \\ A^{(2)} & A^{(1)} & A^{(3)} \\ A^{(3)} & A^{(2)} & A^{(1)} \end{bmatrix} \begin{bmatrix} B^{(1)} \\ B^{(2)} \\ B^{(3)} \end{bmatrix} \right).$$

This tensor-tensor multiplication generalizes to higher-order tensors through \textit{recursion} - see Martin et al, 2012.
**Definition**

If $\mathcal{A}$ is $l \times m \times n$, then $\mathcal{A}^T$ is the $m \times l \times n$ tensor obtained by transposing each of the frontal slices and then reversing the order of transposed frontal slices 2 through $n$.

**Example**

If $\mathcal{A} \in \mathbb{R}^{l \times m \times 4}$

$$\mathcal{A}^T = \text{fold} \left( \begin{bmatrix} (A^{(1)})^T \\ (A^{(4)})^T \\ (A^{(3)})^T \\ (A^{(2)})^T \end{bmatrix} \right)$$
Identity and Orthogonality

Definition
The $\ell \times \ell \times n$ identity tensor $I$ is the tensor whose frontal slice is the $\ell \times \ell$ identity matrix, and whose other frontal slices are all zeros.

Definition
$U \in \mathbb{R}^{m \times m \times n}$ is orthogonal if $U^T \ast U = I = U \ast U^T$.

Can show Frobenius norm invariance: $\|U \ast A\|_F = \|A\|_F$. 
Theorem (K. and Martin, 2011)

Let $\mathbf{A} \in \mathbb{R}^{l \times m \times n}$. Then $\mathbf{A}$ can be factored as

$$\mathbf{A} = \mathbf{U} \cdot \mathbf{S} \cdot \mathbf{V}^T$$

where $\mathbf{U}, \mathbf{V}$ are orthogonal $l \times l \times n$ and $m \times m \times n$, and $\mathbf{S}$ is a $l \times m \times n$ $f$-diagonal tensor.
Truncating the t-SVD

\[ B = U_{1:k,1:k} : * S_{1:k,1:k} : * V_{1:k,1:k}^T : \text{satisfies} \]

\[ B = \arg \min_M \| A - B \|_F \]

where \( M = \{ B = X * Y, X \in \mathbb{R}^{l \times k \times n}, Y \in \mathbb{R}^{k \times m \times n} \} \).

\[ \| A - B \|_F = \sum_{i=k+1}^{\min(l,m)} \| S_{i,i} : \|_F \]

The Eckart-Young-like result suggests highest energy terms are first:

\[ A \approx \sum_{i=1}^{k} U_{i,i} : S_{i,i} : V_{i,i}^T : \]
Let $\mathcal{A}$ be $2 \times 2 \times 2$. FFT along tube fibers $\Rightarrow \hat{\mathcal{A}}$ has $\hat{\mathcal{A}}^{(i)}$ as frontal slices (similar hat, superscript, notation for other tensors).

$$
\hat{\mathcal{A}}^{(1)} = \hat{U}^{(1)} \begin{bmatrix} \hat{s}_1^{(1)} & 0 \\ 0 & \hat{s}_2^{(1)} \end{bmatrix} (\hat{V}^{(1)})^H
$$

$$
\hat{\mathcal{A}}^{(2)} = \hat{U}^{(2)} \begin{bmatrix} \hat{s}_1^{(2)} & 0 \\ 0 & \hat{s}_2^{(2)} \end{bmatrix} (\hat{V}^{(2)})^H
$$

IFFT along tube fibers of $\hat{U}, \hat{S}, \hat{V}$ gives $\mathcal{A} = U \ast S \ast V^T$.

$S_{1,1,:}$ is the length-2 tube fiber effectively computed iff

$$
\text{iff t} \left( \begin{bmatrix} \hat{s}_1^{(1)} \\ \hat{s}_1^{(2)} \end{bmatrix} \right)
$$
Since $\mathbf{U}$ is orthogonal, $\mathbf{U}_{:,1:k} \cdot \mathbf{U}_{:,1:k}^T$ defines an orthogonal projector [K., Braman, Hoover, Hao, '13] projecting a centered image onto a “space” with smaller dimension.

Left singular vertical slices of $\mathbf{U}$ as the new basis,

$$\mathbf{A}_{:,j:} \approx \mathbf{U}_{:,1:k} \cdot \mathbf{U}_{:,1:k}^T \cdot \mathbf{A}_{:,j:} = \sum_{t=1}^{k} \mathbf{U}_{:,t:} \cdot \mathbf{c}_{t,j:} \quad (1)$$

where each tube fiber $\mathbf{c}_{t,j:} = \mathbf{U}_{:,t:}^T \cdot \mathbf{A}_{:,j:}, \ t = 1, 2, \ldots, k$.

Each centered image $\mathbf{A}_{:,j:}$ is a $t$-linear (tensor-linear) combination of orthogonal elements $\mathbf{U}_{:,t:}$ for $t = 1, 2, \ldots, k$ with coefficient tubes $\mathbf{c}_{t,j:}$.
Multi-rank

Definition (K., Braman, Hoover, Hao, 2013)
Let $\mathcal{A} \in \mathbb{R}^{l \times m \times n}$. The multi-rank of $\mathcal{A}$ is length $n$ vector consisting of the ranks of all the $\mathcal{A}^{(i)}$, which must be symmetric about the "middle".

Example
Last example, if all $s_{i}^{(j)}$, $1 \leq i, j, \leq 2$ are non-zero, multi-rank is $[2, 2]^T$.

Example
$\mathcal{A} \in \mathbb{R}^{2 \times 2 \times 4}$, multi-rank possible: $[i, j, k, j]^T$, $1 \leq i, j, k \leq 2$. 
Relative Energy and Truncation

\[ s_{i}^{(j)} := S_{i,i,j} \]

\[
\frac{\sum_{i=1}^{n/2+1} \sum_{j=1}^{k_i} (\hat{s}_j^{(i)})^2}{||\hat{\mathbf{A}}(:, :, 1: n/2+1)||^2_F}.
\]

(2)

If \( k_i \,=\, \min(\ell, m) \), energy is 1.

Order \( \hat{s}_j^{(i)} \) in a vector \( \mathbf{q} \) from largest to smallest, find index \( t \) to be the smallest index such that

\[
\sum_{i=1}^{n/2+1} \sum_{j=1}^{k_i} \{ (\hat{s}_j^{(i)})^2 | (\hat{s}_j^{(i)})^2 > q_t^2 \} > ||\hat{\mathbf{A}}(:, :, 1: n/2+1)||^2_F \times .9.
\]

Effect: If \( \hat{S}_{i,i,j} > q_t \) keep it

\( \hat{S}_{i,i,j} \leq q_t \) set to 0.
Define a f-diagonal tensor $\mathcal{D}$ so that the Fourier coefficients along each diagonal tube fiber satisfy, for $i = 1, \ldots, \frac{n}{2} + 1$

$$\hat{d}_j^{(i)} = \begin{cases} 
1 & \text{if } \hat{s}_j^{(i)} \text{ is kept} \\
0 & \text{if } \hat{s}_j^{(i)} \text{ is discarded through procedure above.} 
\end{cases},$$

and set $\hat{d}_j^{(n-i+2)} = \hat{d}_j^{(i)}$ to ensure conjugate symmetry.

Then in the spatial domain, the compressed test set can be written $(\mathcal{U}_{:,ix,:,:} \ast \mathcal{D} \ast \mathcal{U}_{:,ix,:,:}^T) \ast \mathcal{A}$ where the tensor in parenthesis is still an orthogonal projector [K., Braman, Hoover, Hao, ’13].
Numerical Results

In all experiments, measure the **recognition ratio** by

\[
\frac{\text{number of correctly matched images}}{\text{number of test images}}.
\]

We match a testing image to the person whose image's coefficient is closest, in the Frobenius norm, to the coefficient of the testing image. The matches are counted on an individual image basis, and the denominator is with respect to the total number of test images in each trial.
Experiments, EYB Database

Extended Yale B Database, first 20 different illuminations of all 38 people, decimated by 3. Compare our Method 2 to matrix PCA.

In Experiment 1, each trial, randomly select 15 illuminations, use corresponding images for each person as training set. Images for remaining 5 illumination conditions for each person are used for the test set.

In Experiment 2, each trial, randomly select 5 illuminations, use corresponding images of each person as training set. Images for remaining 15 illumination conditions for each person are used for the test.

In both experiments, we run 20 trials and we use the relative energy measure to determine truncation.
Illustration of Differences Between Basis Sets
Comparison of Method I to “Truncated” TensorFaces

Use 10 illuminations of all 38 people EYB database.
TensorFaces is 5th order method (people, pixels, illumination, viewpoint, expression).

Original TensorFaces required the full HOSVD of the tensor - 5 matrix SVDs of unfoldings of tensor, truncation tricky and not optimal. Another variant on truncated Tucker decomposition optimal, require estimations of 5 trunc params apriori.

- One, 3-way version of TensorFaces: people by illumination by pixels
- Compress on the illumination “mode matrix”, similar to our Method I
Experiments, Weizmann

From Weizmann database: 28 different people photographed in 5 viewpoints, 3 illuminations and 3 expressions, decimated by 3.

Implement TensorFaces on the same database also formed as both third order and 5th order tensors, with no truncation.

- Training: 10 people, all 5 viewpoints, the first 2 illuminations, 3 expressions. Testing: 10 people, all 5 viewpoints, the 3 illuminations, 3 expressions

- Training: 10 people, 5 viewpoints, 3 illuminations, the first 2 expressions. Testing: 10 people, 5 viewpoints, 3 illuminations, the 3rd expression
We take 20 t-SVD terms (90% energy recovery) and no truncation in TensorFaces.

<table>
<thead>
<tr>
<th>Experiment</th>
<th>TensorFaces (5th)</th>
<th>TensorFaces (3rd)</th>
<th>T-SVD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Experiment 1</td>
<td>0.8133</td>
<td>0.8033</td>
<td>0.8933</td>
</tr>
<tr>
<td>Experiment 2</td>
<td>0.8933</td>
<td>0.8633</td>
<td>0.9633</td>
</tr>
</tbody>
</table>
Summary and Future Work

- Developed a compressed representation of a database of facial images, reminiscent of matrix-based PCA, truncating t-SVD.
- Further compression achieved via minimizing multi-rank.
- Methods have the benefit of optimality in the Frobenius norm.
- For the same level of compression, recognition rate of our method superior to matrix-based method.
- Qualitatively equivalent recognition rate against TensorFaces for less work.
- Have developed a Krylov-subspace-like biorthogonalization routine that allows us to approximate t-SVD results for potentially less cost.
- Have developed a (pivoted)QR factorization version that is easier to update-downdate.
- Room for higher-order analogues?