Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16)

Common and Discriminative Subspace Kernel-Based Multiblock Tensor Partial Least Squares Regression

Ming Hou[1], **Qibin Zhao**[2,3], **Brahim Chaib-draa**[1] and **Andrzej Cichocki**[2,4]
[1]Laval University, Quebec, Canada
[2]RIKEN Brain Science Institute, Wako, Japan
[3]Shanghai Jiao Tong University, Shanghai, China
[4]Skolkovo Institute of Science and Technology, Moscow, Russia
ming.hou.1@ulaval.ca, qbzhao@brain.riken.jp, brahim.chaib-draa@ift.ulaval.ca, cia@brain.riken.jp

Abstract

In this work, we introduce a new generalized nonlinear tensor regression framework called kernel-based multiblock tensor partial least squares (KMTPLS) for predicting a set of dependent tensor blocks from a set of independent tensor blocks through the extraction of a small number of common and discriminative latent components. By considering both common and discriminative features, KMTPLS effectively fuses the information from multiple tensorial data sources and unifies the single and multiblock tensor regression scenarios into one general model. Moreover, in contrast to multilinear model, KMTPLS successfully addresses the nonlinear dependencies between multiple response and predictor tensor blocks by combining kernel machines with joint Tucker decomposition, resulting in a significant performance gain in terms of predictability. An efficient learning algorithm for KMTPLS based on sequentially extracting common and discriminative latent vectors is also presented. Finally, to show the effectiveness and advantages of our approach, we test it on the real-life regression task in computer vision, i.e., reconstruction of human pose from multiview video sequences.

Introduction

Over the past few years, tensor-variate regression approaches have attracted a great amount of attention and have been successfully applied in a variety of fields such as chemometrics (Smilde, Westerhuis, and Boque 2000), computer vision (Zhao et al. 2014; Guo, Kotsia, and Patras 2012), signal processing (Cichocki et al. 2015; Hou, Wang, and Chaib-draa 2015; Zhao et al. 2013a; Eliseyev and Aksenova 2013), industrial batch process (Luo, Bao, and Gao 2015) and medical imaging data analysis (Zhou, Li, and Zhu 2013; Li, Zhou, and Li 2013; Hou and Chaib-draa 2015).

Among these approaches, the N-way partial least squares (NPLS) regression approach, proposed by (Bro 1996), is a natural multi-way extension of standard partial least squares (PLS) (Abdi 2010; Zhao, Zhang, and Cichocki 2014) for modeling a linear relationship between a response tensor and a predictor tensor. The key idea of NPLS is to search for a set of latent variables by performing a simultaneous canonical/parallel (CP) factor analysis decomposition (Harshman 1970) of both independent tensor $\mathscr{X}$ and dependent tensor $\mathscr{Y}$, ensuring that the latent components from $\mathscr{X}$ and $\mathscr{Y}$ have the maximum pairwise covariance. However, due to the fact that the NPLS is established based on the CP model, it exhibits several drawbacks such as inferior fitness ability and slow convergence rate when dealing with higher order tensorial data.

The higher-order partial least squares (HOPLS) (Zhao et al. 2013a), on the other hand, attempts to conduct a multilinear projection of both tensors onto a new latent space using the orthogonal block Tucker decomposition (De Lathauwer, De Moor, and Vandewalle 2000; Tucker 1963), with constraint that the extracted latent variables explain as much as possible the covariance between $\mathscr{X}$ and $\mathscr{Y}$. Benefiting from the advantages of Tucker over CP model, HOPLS promises to provide a better predictive ability as well as a highly flexible model that optimally balances between fitness and complexity. To further enhance the predictability, an extension of HOPLS, named kernel-based tensor partial least square (KTPLS), was introduced in (Zhao et al. 2013b) based on the combination of kernel machines with tensor decomposition, providing an attractive solution to the nonlinear tensor regression problem.

Recently, advancing technologies have been producing massive tensorial data streams from multiple sources or modalities that couple in a common domain, and thus can be analyzed jointly, e.g., the linked EEG and fMRI from neuroimaging data analysis. Most of studies have focused on the single-block situation and the predictability is always restricted due to the limited amount of information contained in the single data source. If we simply apply the above methods and average the predictions among the blocks, then the obtained result will not outperform the best singleton case, since the common features of all data blocks are neglected. Such progress highlights a growing need for the development and application of tensor regression techniques by considering multiple predictor and response tensor blocks. For this purpose, Smilde et al. proposed the multiway multiblock covariates regression model (Smilde, Westerhuis, and Boque 2000), also termed as MMCR, to quantitatively analyze a collection of predictor and response tensor blocks. Generally, MMCR handles all the predictor blocks in a hierarchical way that a super latent factor matrix is defined based on all the individual latent factor matrices to make the final prediction. Nevertheless, MMCR may suffer from low predic-

tive ability due to the restriction of linearity, and thus is inadequate in terms of predictability in a situation where the nonlinear dependencies exist between the response and predictor tensor blocks. Moreover, Westerhuis (Westerhuis, Kourti, and MacGregor 1998) suggested that multilinear multiblock PLS like MMCR is equivalent to PLS model with all variables combined into a large $\mathscr{X}$-block in terms of predictive ability. Besides, the developed algorithm for solving MMCR is based on an alternating least square (ALS) style approach (Carroll and Chang 1970), which is known to be a suboptimal procedure with slow convergence rate.

To address these limitations, we present a new generalized nonlinear tensor regression framework, namely kernel-based multiblock tensor partial least squares (KMTPLS), that serves as an extension to KTPLS (Zhao et al. 2013b) by incorporating the kernel concept into the context of multiblock tensor regression. In this paper, our contributions are (i) introducing a generalized nonlinear framework that effectively fuses the information from multiple tensorial data sources and integrates single and multiblock tensor regression scenarios into one general model using both common and discriminative features; (ii) addressing the nonlinear dependencies between multiple response and predictor tensor blocks successfully by combining kernel machines with joint Tucker decomposition, which leads to a further enhanced predictive power; (iii) developing an efficient algorithm for KMTPLS based on sequentially extracting common and discriminative latent vectors, which can easily scale to a number of blocks; (iiii) it is the first work that applies multiblock tensor regression approach to the multiview or multimodal human motion estimation problem in computer vision.

Notation and Background

Higher-order tensors (Kolda and Bader 2009; Cichocki 2013) are represented as $\mathscr{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_D}$ in calligraphy letters, where D is the order of $\mathscr{X}$. Matrices denoted by boldface capital letters $\mathbf{X}$ are the tensors of order 2. We denote vectors using boldface lower-case letters $\mathbf{x}$. The ith entry of a vector $\mathbf{x}$ is denoted by x_i, and the (i,j) entry of a matrix $\mathbf{X}$ is denoted by x_{ij}. Likewise, we denote the entry $(i_1, i_2, ..., i_D)$ of Dth-order tensor $\mathscr{X}$ as $x_{i_1 i_2 \ldots i_D}$. The *d-mode vector* of tensor $\mathscr{X}$ is a vector of $\mathbb{R}^{I_d}$, which is obtained by varying the index I_d while keeping other indices fixed. The *d-mode matricization* of $\mathscr{X}$, denoted as $\mathbf{X}_{(d)} \in \mathbb{R}^{I_d \times I_1 \cdots I_{d-1} I_{d+1} \cdots I_D}$, is the process of rearranging the d-mode vectors into the columns of the resulting matrix. Particularly, the vectorization of tensor $\mathscr{X}$ is denoted as $vec(\mathscr{X})$. The *d-mode product* of tensor $\mathscr{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_D}$ and matrix $\mathbf{A} \in \mathbb{R}^{J_d \times I_d}$ is denoted by

$$\mathscr{Y} = \mathscr{X} \times_d \mathbf{A} \in \mathbb{R}^{I_1 \times \cdots \times I_{d-1} \times J_d \times I_{d+1} \times \cdots \times I_D}. \quad (1)$$

Kernel-based Multiblock Tensor Partial Least Squares (KMTPLS)

As mentioned in the previous section, the main objective of KMTPLS is to predict a set of dependent tensor blocks from a set of independent tensor blocks through the extraction of a small number of common and individual latent components followed by a regression step against them. Without the loss of generality, we consider a (M_1+1)th-order independent tensor block $\mathscr{X}_1 \in \mathbb{R}^{N \times I_1 \times \cdots \times I_{M_1}}$, a (M_2+1)th-order independent tensor block $\mathscr{X}_2 \in \mathbb{R}^{N \times J_1 \times \cdots \times J_{M_2}}$ and a $(L+1)$th-order dependent tensor block $\mathscr{Y} \in \mathbb{R}^{N \times K_1 \times \cdots \times K_L}$, which can be obtained by concatenating N pairs of observations $\{(\mathscr{X}_1^{(n)}, \mathscr{X}_2^{(n)}, \mathscr{Y}^{(n)})\}_{n=1}^{N}$ that couple in the first mode with the same sample size. Similar to KTPLS (Zhao et al. 2013b), the tensorial input and output data points $\mathscr{X}_1^{(n)}$, $\mathscr{X}_2^{(n)}$ and $\mathscr{Y}^{(n)}$ are mapped into the high-dimensional feature space $\mathscr{H}$ using a nonlinear transformation ϕ as follows:

$$\phi : \mathscr{X}^{(n)} \rightarrow \phi(\mathscr{X}^{(n)}) \in \mathbb{R}^{H_1 \times \cdots \times H_M}. \quad (2)$$

We thus have $\phi(\mathscr{X}_1)$, $\phi(\mathscr{X}_2)$ and $\phi(\mathscr{Y})$ for the corresponding blocks, which for simplicity can be denoted as Φ_1, Φ_2 and Ψ, respectively. Unlike KTPLS, we now perform the Tucker decompositions of Φ_1, Φ_2 and Ψ jointly in $\mathscr{H}$ by taking both common and individual features into account

$$\Phi_1 = \mathscr{G}_{\mathscr{X}_1} \times_1 [\mathbf{T}_{com} | \mathbf{T}_{dis_1}] \times_2 \mathbf{P}_1^{(1)} \times \cdots \times_{M_1+1} \mathbf{P}_1^{(M_1)} + \mathscr{E}_{\mathscr{X}_1} \quad (3)$$

$$\Phi_2 = \mathscr{G}_{\mathscr{X}_2} \times_1 [\mathbf{T}_{com} | \mathbf{T}_{dis_2}] \times_2 \mathbf{P}_2^{(1)} \times \cdots \times_{M_2+1} \mathbf{P}_2^{(M_2)} + \mathscr{E}_{\mathscr{X}_2} \quad (4)$$

$$\Psi = \mathscr{G}_{\mathscr{Y}} \times_1 [\mathbf{U}_{com} | \mathbf{U}_{dis_1} | \mathbf{U}_{dis_2}] \times_2 \mathbf{Q}^{(1)} \times \cdots \times_{L+1} \mathbf{Q}^{(L)} + \mathscr{E}_{\mathscr{Y}} \quad (5)$$

$$\mathbf{U}_{dis_1} = \mathbf{T}_{dis_1} \mathbf{D}_{dis_1} + \mathbf{E}_{dis_1} \quad (6)$$

$$\mathbf{U}_{dis_2} = \mathbf{T}_{dis_2} \mathbf{D}_{dis_2} + \mathbf{E}_{dis_2} \quad (7)$$

$$\mathbf{U}_{com} = \mathbf{T}_{com} \mathbf{D}_{com} + \mathbf{E}_{com}, \quad (8)$$

where $\{\mathscr{G}_{\mathscr{X}_1}, \mathscr{G}_{\mathscr{X}_2}, \mathscr{G}_{\mathscr{Y}}\}$ stand for the core tensors and $\{\mathbf{P}_1^{(m_1)}, \mathbf{P}_2^{(m_2)}, \mathbf{Q}^{(l)}\}$ denote the corresponding loading factor matrices. $\{\mathbf{T}_{com}, \mathbf{U}_{com}\}$ are defined as the common latent factor matrices while $\{\mathbf{T}_{dis_1}, \mathbf{U}_{dis_1}\}$ and $\{\mathbf{T}_{dis_2}, \mathbf{U}_{dis_2}\}$ correspond to the discriminative ones. Note that each pair of latent factor matrices is connected by an inner relation, namely the diagonal matrices $\mathbf{D}_{com}$, $\mathbf{D}_{dis_1}$ and $\mathbf{D}_{dis_2}$, assuming that $\mathbf{U}$ is linearly approximated by $\mathbf{T}$. $\{\mathscr{E}_{\mathscr{X}_1}, \mathscr{E}_{\mathscr{X}_2}, \mathscr{E}_{\mathscr{Y}}\}$ and $\{\mathbf{E}_{dis_1}, \mathbf{E}_{dis_2}, \mathbf{E}_{com}\}$ represent the residuals. In this way, Φ_1 and Φ_2 are simultaneously correlated with Ψ by having the largest covariance between latent factors $\mathbf{T}_{com}$ and $\mathbf{U}_{com}$. Meanwhile, Φ_1 singly associates with Ψ by maximizing covariance between latent factors $\mathbf{T}_{dis_1}$ and $\mathbf{U}_{dis_1}$. Likewise, the individual connection between Φ_2 and Ψ is characterized via the maximum covariance between $\mathbf{T}_{dis_2}$ and $\mathbf{U}_{dis_2}$. In essence, the common part $\mathbf{T}_{com}$ captures the variation that is present in all predictor blocks and hence describes what all blocks have in common. The discriminative part $\mathbf{T}_{dis_1}$ and $\mathbf{T}_{dis_2}$, on the other hand, explain the variation only in their own individual tensor block. Rather than explicitly estimating all the high-dimensional core tensors and loading factors, we only need to compute all the latent factors described in (3)-(8). To this end, for the common latent factor $\mathbf{T}_{com} = [t_1, ..., t_R]$ the optimization objective derived from (Zhao et al. 2013b) for our 2-block setting turns out to be

$$\max_{\{\mathbf{w}_{1r}, \mathbf{w}_{2r}, \mathbf{v}_r\}} [cov_1(\mathbf{t}_r, \mathbf{u}_r) cov_2(\mathbf{t}_r, \mathbf{u}_r)]^2, \; r = 1, ..., R,$$

$$s.t. \; \mathbf{t}_r = \Phi_{1(1)} \mathbf{w}_{1r} = \Phi_{2(1)} \mathbf{w}_{2r}, \; \mathbf{u}_r = \Psi_{1(1)} \mathbf{v}_r, \quad (9)$$

where we aim to optimize two covariance cov_1 and cov_2 at the same time, yielding the maximum product of pair-wise covariance of latent vectors. Here $\mathbf{w}_{1r}$, $\mathbf{w}_{2r}$ and $\mathbf{v}_r$ serve as weight vectors. In the case of discriminative part $\mathbf{T}_{dis_1}$ ($\mathbf{T}_{dis_2}$), the objective is similar except that only the cov_1 (cov_2) should be taken into consideration.

The strategy for solving the above optimization problem consists in sequentially extracting by deflation R pairs of the latent vectors $\{\mathbf{t}_r, \mathbf{u}_r\}$ to incorporate the information of Φ_1 and Φ_2 into the model simultaneously. The KMTPLS is summarized in Algorithm 1, which consists of two major stages. In stage 1 (line 4-15), during each inner iteration the latent vector $\mathbf{t}_r$ is first updated by the information of Φ_1 at line 7 and then immediately followed an update by the information of Φ_2 at line 9. Having extracted a new pair of $\{\mathbf{t}_r, \mathbf{u}_r\}$, the deflations with respect to $\Phi_{1(1)}\Phi_{1(1)}^T$, $\Phi_{2(1)}\Phi_{2(1)}^T$ and $\Psi_{(1)}\Psi_{(1)}^T$ are executed from line 12 to 14, removing the calculated variation from the corresponding blocks. The repeated extraction procedure stops when the desired R number of vector pairs are obtained (line 15). Stage 1 means the common contribution from Φ_1 and Φ_2 is collaboratively responsible for Ψ. Substituting these inner steps into each other, stage 1 is in fact equivalent to solving the following eigenvalue problems

$$\Phi_{1(1)}\Phi_{1(1)}^T\Psi_{(1)}\Psi_{(1)}^T\Phi_{2(1)}\Phi_{2(1)}^T\Psi_{(1)}\Psi_{(1)}^T\mathbf{t}_r = \lambda\mathbf{t}_r \tag{10}$$

$$\mathbf{u}_r = \Psi_{(1)}\Psi_{(1)}^T\mathbf{t}_r. \tag{11}$$

It is worth noting that $\Phi_{1(1)}\Phi_{1(1)}^T$ and $\Phi_{2(1)}\Phi_{2(1)}^T$, containing only the inner products between vectorized tensorial data points, can be substituted by kernel Gram matrices $\mathbf{K}_{\mathscr{X}_1}$ and $\mathbf{K}_{\mathscr{X}_2}$ respectively. Likewise, the $\Psi_{(1)}\Psi_{(1)}^T$ is also replaced with $\mathbf{K}_{\mathscr{Y}}$. Hence, the previous deflation step in Algorithm 1 for $\Phi_{1(1)}\Phi_{1(1)}^T$ (line 12) finally becomes

$$\mathbf{K}_{\mathscr{X}_1} \leftarrow (\mathbf{I}-\mathbf{t}_r\mathbf{t}_r^T)\mathbf{K}_{\mathscr{X}_1}(\mathbf{I}-\mathbf{t}_r\mathbf{t}_r^T), \tag{12}$$

and the previous formulation (10)-(11) can be rewritten as

$$\mathbf{K}_{\mathscr{X}_1}\mathbf{K}_{\mathscr{Y}}\mathbf{K}_{\mathscr{X}_2}\mathbf{K}_{\mathscr{Y}}\mathbf{t}_r = \lambda\mathbf{t}_r \tag{13}$$

$$\mathbf{u}_r = \mathbf{K}_{\mathscr{Y}}\mathbf{t}_r. \tag{14}$$

Thereafter, we continue to extract the discriminative latent pairs $\{\mathbf{T}_{dis_1}, \mathbf{U}_{dis_1}\}$ and $\{\mathbf{T}_{dis_2}, \mathbf{U}_{dis_2}\}$, which are used to explain the variation of each individual block, from the deflated Φ_1, Φ_2 and Ψ obtained in the first stage. In stage 2, we follow the similar extracting pattern as stage 1 except that $\{\Phi_1, \Psi\}$ and $\{\Phi_2, \Psi\}$ are considered separately, implying two separate contributions are from Φ_1 and Φ_2 for the final response Ψ, respectively.

Eventually, with all the latent factors $\mathbf{T}_{com}$, $\mathbf{T}_{dis_1}$, $\mathbf{T}_{dis_2}$ and $\mathbf{U}_{com}$, $\mathbf{U}_{dis_1}$, $\mathbf{U}_{dis_2}$ in hand, the new response $\mathscr{Y}^{new}$ can be jointly predicted from the test points $\mathscr{X}_1^{new}$ and $\mathscr{X}_2^{new}$ by modifying the prediction formula in (Rosipal and Trejo 2002) to be

$$\mathbf{y}^{(new)T} = \alpha\mathbf{k}_{\mathscr{X}_1^{new}}^T\mathbf{U}_1(\mathbf{T}_1^T\mathbf{K}_{\mathscr{X}_1}\mathbf{U}_1)^{-1}\mathbf{T}_{dis_1}^T\mathbf{Y}_{(1)}$$
$$+(1-\alpha)\mathbf{k}_{\mathscr{X}_2^{new}}^T\mathbf{U}_2(\mathbf{T}_2^T\mathbf{K}_{\mathscr{X}_2}\mathbf{U}_2)^{-1}\mathbf{T}_{dis_2}^T\mathbf{Y}_{(1)}, \tag{15}$$

where $(\mathbf{k}_{\mathscr{X}_1^{new}})_n = k(\mathscr{X}_1^{new}, \mathscr{X}_1^{(n)})$ and $(\mathbf{k}_{\mathscr{X}_2^{new}})_n = k(\mathscr{X}_2^{new}, \mathscr{X}_2^{(n)})$. $\mathbf{T}_1 = [\mathbf{T}_{com}|\mathbf{T}_{dis_1}]$, $\mathbf{T}_2 = [\mathbf{T}_{com}|\mathbf{T}_{dis_2}]$, $\mathbf{U}_1 = [\mathbf{U}_{com}|\mathbf{U}_{dis_1}]$ and $\mathbf{U}_2 = [\mathbf{U}_{com}|\mathbf{U}_{dis_2}]$. The hyperparameter α varying from 0 to 1 indicates the relative importance of each block, and thus can be heuristically determined according to their individual predictive ability. Notice that here $\mathbf{y}^{(new)}$ is in a vector form and should be reformulated into a tensor form $\mathscr{Y}^{new}$ by refolding.

We would like to make several remarks about our KMTPLS approach. It can actually be regarded as a generalized kernel-based nonlinear tensor regression framework that includes the KTPLS as a special case by setting the number of latent components from other tensor blocks to be zero. Additionally, this framework enables us to straightforwardly extend both model and algorithm to more general cases in which more than two predictor tensor blocks and one response tensor block are involved. Furthermore, our model, based on common and discriminative features, is superior to MMCR with respect to the interpretability. Inheriting the advantage from KTPLS, the framework is capable of flexibly adapting different types of specially defined tensor kernel functions, namely $(\mathbf{K}_{\mathscr{X}_1})_{nn'} = k_{x_1}(\mathscr{X}_1^{(n)}, \mathscr{X}_1^{(n')})$, $(\mathbf{K}_{\mathscr{X}_2})_{nn'} = k_{x_2}(\mathscr{X}_2^{(n)}, \mathscr{X}_2^{(n')})$ and $(\mathbf{K}_{\mathscr{Y}})_{nn'} = k_y(\mathscr{Y}^{(n)}, \mathscr{Y}^{(n')})$, to different individual tensor blocks, leading to a better performance than that of combining all variables in one large $\mathscr{X}$-block.

Note that our algorithm is motivated by the classical NIPALS (Rosipal and Trejo 2002), which has proved to be extremely robust for solving the eigenvalue related problems, providing a robust procedure for iteratively estimating latent components in our setting. Therefore, we can reliably extract a desired number of pairs of latent components up to the point when the rank of the corresponding unfolded tensor block is reached. The computational complexity of extraction as well as deflation of a single pair of latent vectors scales as $\mathscr{O}(n^2)$ while the overall cost is linearly proportional to the total number of desired latent components in T blocks, i.e., $R_{com} + R_{dis1} + \cdots + R_{disT}$. Hence, such sequential strategy of extraction of one pair of latent vectors at a time by deflation is more efficient than MMCR (Smilde, Westerhuis, and Boque 2000) that performs a simultaneous calculation of all factor matrices. We should mention that the cost of establishing the kernel matrix, depending on the specially designed tensor kernel function, should also be taken into account.

Experimental Results

In all experiments, the MMCR and KMTPLS were used for comparison including the settings of single and multiple input blocks for KMTPLS model. For simplicity, the polynomial kernel function of second degree was employed. The predictive performance was quantitatively evaluated by means of the root mean squares of prediction (RMSEP) (Kim et al. 2005) and the Q index that is defined as $Q = 1 - \|\mathscr{Y} - \hat{\mathscr{Y}}\|_F / \|\mathscr{Y}\|_F$, where $\hat{\mathscr{Y}}$ is the prediction of $\mathscr{Y}$.

Algorithm 1 Our Kernel-based Multiblock Tensor Partial Least Squares (KMTPLS)

Input: observations of Φ_1, Φ_2 and Ψ; number of desired latent vectors R, R_1 and R_2
Output: common matrices $\mathbf{T}_{com}$, $\mathbf{U}_{com}$ and discriminative matrices $\mathbf{T}_{dis_1}$, $\mathbf{U}_{dis_1}$, $\mathbf{T}_{dis_2}$, $\mathbf{U}_{dis_2}$
Initialize: randomly initialize $\mathbf{u}_r$, $\mathbf{u}_{1r_1}$ and $\mathbf{u}_{2r_2}$
/* **Stage 1: extract common latent factor $\mathbf{T}_{com}$, $\mathbf{U}_{com}$** */
repeat
  repeat
    $\mathbf{t}_r = \Phi_{1(1)}\Phi_{1(1)}^T\mathbf{u}_r,\ \mathbf{t}_r \leftarrow \mathbf{t}_r/\|\mathbf{t}_r\|$
    $\mathbf{u}_r = \Psi_{(1)}\Psi_{(1)}^T\mathbf{t}_r,\ \mathbf{u}_r \leftarrow \mathbf{u}_r/\|\mathbf{u}_r\|$
    $\mathbf{t}_r = \Phi_{2(1)}\Phi_{2(1)}^T\mathbf{u}_r,\ \mathbf{t}_r \leftarrow \mathbf{t}_r/\|\mathbf{t}_r\|$
    $\mathbf{u}_r = \Psi_{(1)}\Psi_{(1)}^T\mathbf{t}_r,\ \mathbf{u}_r \leftarrow \mathbf{u}_r/\|\mathbf{u}_r\|$
  until convergence
  deflate $\Phi_{1(1)}\Phi_{1(1)}^T$ matrix: $\Phi_{1(1)}\Phi_{1(1)}^T \leftarrow (\Phi_{1(1)} - \mathbf{t}_r\mathbf{t}_r^T\Phi_{1(1)})(\Phi_{1(1)} - \mathbf{t}_r\mathbf{t}_r^T\Phi_{1(1)})^T$
  deflate $\Phi_{2(1)}\Phi_{2(1)}^T$ matrix: $\Phi_{2(1)}\Phi_{2(1)}^T \leftarrow (\Phi_{2(1)} - \mathbf{t}_r\mathbf{t}_r^T\Phi_{2(1)})(\Phi_{2(1)} - \mathbf{t}_r\mathbf{t}_r^T\Phi_{2(1)})^T$
  deflate $\Psi_{(1)}\Psi_{(1)}^T$ matrix: $\Psi_{(1)}\Psi_{(1)}^T \leftarrow (\Psi_{(1)} - \mathbf{t}_r\mathbf{t}_r^T\Psi_{(1)})(\Psi_{(1)} - \mathbf{t}_r\mathbf{t}_r^T\Psi_{(1)})^T$
until R pairs of latent vectors of $\{\mathbf{T}_{com}, \mathbf{U}_{com}\}$ or the rank condition is met
/* **Stage 2: extract discriminative latent factors $\mathbf{T}_{dis_1}$, $\mathbf{U}_{dis_1}$, $\mathbf{T}_{dis_2}$, $\mathbf{U}_{dis_2}$** */
repeat
  repeat
    $\mathbf{t}_{1r_1} = \Phi_{1(1)}\Phi_{1(1)}^T\mathbf{u}_{1r_1},\ \mathbf{t}_{1r_1} \leftarrow \mathbf{t}_{1r_1}/\|\mathbf{t}_{1r_1}\|$
    $\mathbf{u}_{1r_1} = \Psi_{(1)}\Psi_{(1)}^T\mathbf{t}_{1r_1},\ \mathbf{u}_{1r_1} \leftarrow \mathbf{u}_{1r_1}/\|\mathbf{u}_{1r_1}\|$
  until convergence
  deflate $\Phi_{1(1)}\Phi_{1(1)}^T$ matrix: $\Phi_{1(1)}\Phi_{1(1)}^T \leftarrow (\Phi_{1(1)} - \mathbf{t}_{1r_1}\mathbf{t}_{1r_1}^T\Phi_{1(1)})(\Phi_{1(1)} - \mathbf{t}_{1r_1}\mathbf{t}_{1r_1}^T\Phi_{1(1)})^T$
  deflate $\Psi_{(1)}\Psi_{(1)}^T$ matrix: $\Psi_{(1)}\Psi_{(1)}^T \leftarrow (\Psi_{(1)} - \mathbf{t}_{1r_1}\mathbf{t}_{1r_1}^T\Psi_{(1)})(\Psi_{(1)} - \mathbf{t}_{1r_1}\mathbf{t}_{1r_1}^T\Psi_{(1)})^T$
  repeat
    $\mathbf{t}_{2r_2} = \Phi_{2(1)}\Phi_{2(1)}^T\mathbf{u}_{2r_2},\ \mathbf{t}_{2r_2} \leftarrow \mathbf{t}_{2r_2}/\|\mathbf{t}_{2r_2}\|$
    $\mathbf{u}_{2r_2} = \Psi_{(1)}\Psi_{(1)}^T\mathbf{t}_{2r_2},\ \mathbf{u}_{2r_2} \leftarrow \mathbf{u}_{2r_2}/\|\mathbf{u}_{2r_2}\|$
  until convergence
  deflate $\Phi_{2(1)}\Phi_{2(1)}^T$ matrix: $\Phi_{2(1)}\Phi_{2(1)}^T \leftarrow (\Phi_{2(1)} - \mathbf{t}_{2r_2}\mathbf{t}_{2r_2}^T\Phi_{2(1)})(\Phi_{2(1)} - \mathbf{t}_{2r_2}\mathbf{t}_{2r_2}^T\Phi_{2(1)})^T$
  deflate $\Psi_{(1)}\Psi_{(1)}^T$ matrix: $\Psi_{(1)}\Psi_{(1)}^T \leftarrow (\Psi_{(1)} - \mathbf{t}_{2r_2}\mathbf{t}_{2r_2}^T\Psi_{(1)})(\Psi_{(1)} - \mathbf{t}_{2r_2}\mathbf{t}_{2r_2}^T\Psi_{(1)})^T$
until R_1 pairs of latent vectors of $\{\mathbf{T}_{dis_1}, \mathbf{U}_{dis_1}\}$ or the rank condition is met, and R_2 pairs of latent vectors of $\{\mathbf{T}_{dis_2}, \mathbf{U}_{dis_2}\}$ or the rank condition is met

Utrecht Multi-Person Motion Database

In this section, we first applied our approach to a real-life tensor regression task, i.e., estimation of articulated 3D human pose positions from multiple video streams, to systematically show the advantages of KMTPLS. The dataset was taken from the Utrecht Multi-Person Motion (UMPM) benchmark (Van Der Aa et al. 2011), which provides synchronized motion capture data and video sequences from multiple viewpoints. More specifically, the video sequences of 30-60 seconds were captured using 4 Basler calibrated color cameras with a resolution of 644×484 pixels at 50 fps. In addition, to obtain the 3D ground truth information, they used a Vicon MoCap system to record the 3D positions of 37 reflective markers attached to each subject at a frame rate of 100 fps for describing the movements of head, shoulders, elbows, wrists, knees and ankles etc. The activities of natural motions of humans in daily life such as walking, jogging, balancing were captured within a region of 6m×6m. For our test, we employed the intensity image sequences, with a downsized resolution of 32×24 pixels, as the input of algorithms. Hence, each video sequence can be naturally represented as a 3rd-order predictor tensor (i.e., frames × width × hight). As for the ground truth, the mocap data was down-sampled at 50 fps, and thus can be expressed as a 3rd-order tensor (i.e., samples × 3D positions × markers). For each scenario, we split the video sequence into two different partitions, i.e., a training set from the first 1/3 part and a test set from the remaining 2/3 part, respectively. The cross-validation applied on training set was performed to select all the desired tuning parameters. In this experiment, we chose 3 principal cameras that form an equilateral triangle including the front camera (F), the left camera (L) and the right camera (R), and we mainly focused on the 2-block situation while the multiple input blocks cases were studied in the following section. To be fair, the maximal possible number of latent vectors that can be extracted from each individual block was set as 10 for both KMTPLS and MMCR.

Table 1 summarizes the prediction performance from KMTPLS and MMCR. Clearly, our method significantly outperformed MMCR in almost all cases in terms of predictive accuracy, especially in the "grab" scenario where the largest performance gaps were 0.1839 and 178.4 with respect to Q and RMSEP when the front camera and the right camera were integrated. Meanwhile, as also shown in Figure 1, we can see that the considerably improved predictability were achieved by all 2-block situations in comparison to the corresponding single block cases. For example, in the "table" scene, the optimal Q obtained by the left and the right cameras were 0.5995 and 0.5906, while it was boosted to 0.6847 as we effectively fuse two cameras using KMTPLS.

To investigate the effect of the α on the predictive performance, Q is given in Figure 2 for varying alpha value at the retained ranks. We may notice that the optimal α, which is the measurement of involvement in the final prediction, can somehow roughly reflect the relatively predictive power of each individual block. Specifically, the optimal α in the "grab" scene were 0.1 and 0.2 for the "F-R" and "L-R", indicating the model intended to attach more importance to the camera "R" with the best singleton predictive accuracy of 0.7789. For visualization, Figure 3 further demonstrates the ground truth and the predicted 3D front angle trajectories for the test sequence using the front and left cameras. We can see that our KMTPLS "L-R" achieves much better accuracy than both KMTPLS "L" and MMCR "L-R" cases.

Berkeley Multimodal Human Action Database

In order to validate the superiority of KMTPLS in the context of several cameras from multiple views, the second

UMPM		KMTPLS						MMCR		
Scenario	Appraisal	F	L	R	F-L	F-R	L-R	F-L	F-R	L-R
Grab	Rank	(10)	(6)	(10)	(6,4,1)	(8,1,2)	(8,0,2)	(3,10,2)	(3,10,8)	(3,10,1)
	Q	0.7141	0.7240	0.7789	**0.7690**	**0.7950**	**0.7995**	0.6237	0.6111	0.6497
	RMSEP	241.4	230.0	185.5	**192.7**	**172.8**	**169.7**	347.7	351.2	292.7
Triangle	Rank	(9)	(9)	(10)	(8,2,2)	(1,8,0)	(7,1,3)	(4,9,1)	(4,10,1)	(5,8,2)
	Q	0.7599	0.7462	0.7227	**0.7801**	**0.7629**	**0.7691**	0.7168	0.7195	0.7324
	RMSEP	202.5	208.2	232.3	**181.3**	**200.2**	**190.6**	241.3	239.9	224.5
Table	Rank	(10)	(9)	(10)	(1,9,8)	(1,9,0)	(9,0,1)	(3,9,9)	(5,10,1)	(4,7,2)
	Q	0.6340	0.5995	0.5906	**0.6910**	0.6414	**0.6847**	0.6713	**0.6626**	0.6761
	RMSEP	298.0	319.3	325.9	**247.4**	291.2	**250.7**	267.2	**277.2**	260.9

Table 1: Performance comparison of KMTPLS and MMCR for the optimal Rank, Q and RMSEP and on UMPM.

MHAD			Jumping in the Place			Bending		
Subject	Method	Blocks	Q	RMSEP	Time (ms)	Q	RMSEP	Time (ms)
S6	KMTPLS	1	0.6836 (0.0056)	209.3 (4.2)	**38** (19)	0.6991 (0.0059)	188.7 (4.4)	**71** (12)
		2	**0.6955** (0.0043)	**200.7** (2.9)	70 (27)	**0.7076** (0.0030)	**182.8** (2.3)	122 (15)
		3	**0.6986** (0.0030)	**199.3** (1.9)	92 (58)	**0.7112** (0.0021)	**179.9** (1.6)	173 (13)
		4	**0.7000** (0.0025)	**198.5** (1.6)	139 (81)	**0.7129** (0.0015)	**179.0** (1.2)	324 (44)
	MMCR	2	0.6717 (0.0050)	218.3 (3.0)	1616 (999)	0.6524 (0.0179)	217.3 (11.1)	4510 (2201)
		3	0.6762 (0.0027)	215.4 (1.6)	2530 (1235)	0.6602 (0.0045)	212.8 (3.1)	8154 (3077)
		4	0.6790 (0.0019)	213.7 (1.1)	2954 (788)	0.6630 (0.0029)	210.8 (1.8)	13116 (3974)
S10	KMTPLS	1	0.7141 (0.0057)	196.8 (4.3)	**45** (13)	0.7329 (0.0055)	168.9 (3.9)	**49** (22)
		2	**0.7230** (0.0047)	**190.4** (3.5)	74 (43)	**0.7449** (0.0029)	**160.2** (2.2)	110 (26)
		3	**0.7253** (0.0038)	**188.3** (2.8)	134 (63)	**0.7493** (0.0029)	**157.3** (1.3)	149 (21)
		4	**0.7263** (0.0027)	**187.9** (2.0)	195 (85)	**0.7516** (0.0015)	**155.6** (1.1)	199 (27)
	MMCR	2	0.6930 (0.0047)	211.3 (3.4)	2248 (2175)	0.7010 (0.0100)	188.9 (6.3)	8118 (3676)
		3	0.6967 (0.0021)	208.7 (1.5)	2771 (2551)	0.6945 (0.0164)	193.0 (10.5)	11575 (4390)
		4	0.6989 (0.0010)	207.2 (0.8)	2985 (2408)	0.6941 (0.0133)	193.4 (8.7)	15540 (4837)
S12	KMTPLS	1	0.7059 (0.0065)	198.9 (5.3)	**16** (6)	0.6755 (0.0060)	195.3 (4.3)	**69** (16)
		2	**0.7199** (0.0032)	**189.2** (2.3)	31 (12)	**0.6841** (0.0020)	**189.5** (1.4)	138 (19)
		3	**0.7240** (0.0029)	**186.3** (2.2)	41 (13)	**0.6859** (0.0015)	**188.8** (1.0)	226 (38)
		4	**0.7265** (0.0016)	**184.3** (1.2)	69 (23)	**0.6867** (0.0015)	**188.4** (1.0)	314 (45)
	MMCR	2	0.6909 (0.0051)	208.1 (3.4)	1689 (983)	0.6541 (0.0032)	208.1 (1.7)	3256 (1318)
		3	0.6943 (0.0041)	205.4 (2.5)	2568 (1039)	0.6564 (0.0033)	206.8 (1.3)	7289 (2180)
		4	0.6959 (0.0036)	204.2 (2.1)	4280 (2136)	0.6585 (0.0020)	205.5 (1.0)	10649 (2925)

Table 2: Performance comparison of KMTPLS and MMCR for the averaged Q, RMSEP and learning time on MHAD.

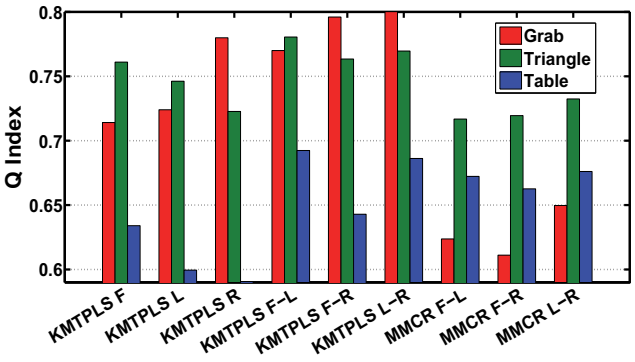

Figure 1: Performance comparison of KMTPLS and MMCR for the Q versus the different combination of cameras.

experiment was carried out on the Berkeley Multimodal Human Action Database (MHAD) (Ofli et al. 2013) that consists of temporally synchronized data simultaneously recorded by several different systems, including an optical motion capture system, four multi-view stereo camera clusters, two Microsoft Kinect cameras.

The MHAD comprises 11 actions performed by 12 subjects, with each subject performing 1 or 5 repetitions of each action per recording. We chose the 2 most challengeable actions with 5 repetitions that involve dynamics in both upper and lower extremities, namely "jumping in place" and "bending-hands up all the way down". In this experiment, we were particularly interested in how the predictive performance is affected by the number of predictor blocks in the process of fusion. To this end, we selected 4 of C1 cameras from 4 clusters as well as 2 of C2 cameras from clusters L1 and L2, resulting in a total number of 6 cameras from all 12 cameras. Note that each time we chose a subset of these 6 cameras as candidates to jointly make the prediction, we therefore have "6 *choose* T" (C_6^T) different combinations for the T-block situation. Similar to the previous

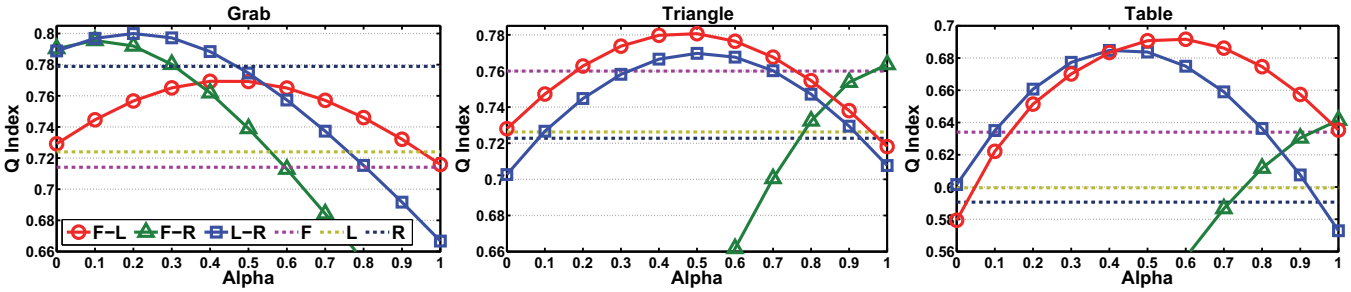

Figure 2: Performance comparison of KMTPLS for the Q when using the optimal rank versus the relative importance α.

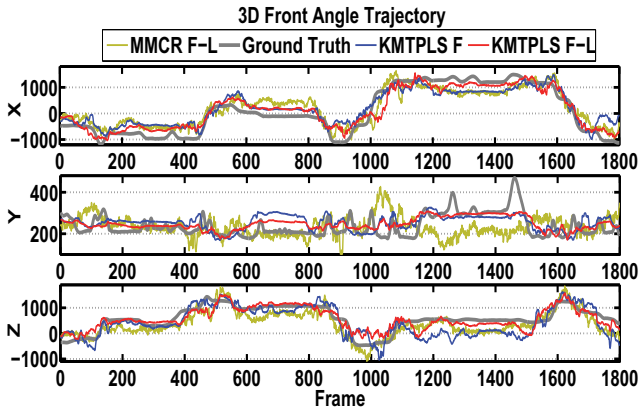

Figure 3: Visualization of ground truth and the trajectories predicted by MMCR and KMTPLS in the "table" scenario.

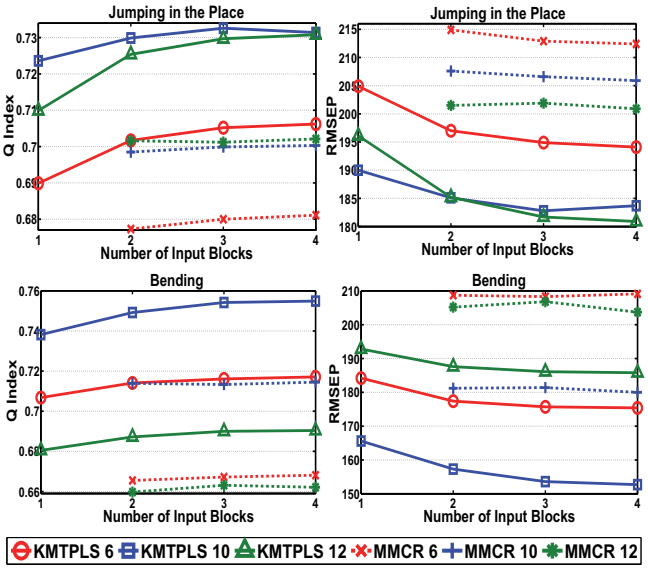

Figure 4: Performance comparison of KMTPLS and MMCR for the best Q, RMSEP versus the number of input blocks.

experiment, both video sequence and mocap data are represented as 3rd-order tensors. In the case of the ground truth, the 3D positions of 43 LED markers affixed to different parts of the body were captured within the space of 2m×2m. We took the first recording of each action as training set and second recording as test set. The optimal number of common and discriminative latent vectors was selected by cross-validation on the training set. The maximal number of latent vectors from each block for both models was fixed as 8. Finally, the total number of predictor tensor blocks T grew up to 4, and the importance parameter α for each block was simply fixed to be $1/T$.

The averaged prediction results as well as the learning time of the two methods are compared in Table 2. As was expected, KMTPLS exhibited much better performance than MMCR with respect to all the measurements. In particular, for the "bending" action, the averaged improvement by KMTPLS over MMCR when using 4 blocks were 0.0499, 0.0575 and 0.0282 in terms of Q, and 31.8, 37.8 and 17.1 in terms of RMSEP for subjects "s6", "s10" and "s12", respectively. On the other hand, KMTPLS shows a consistently enhanced predictability as the number of blocks increases from 1 up to an optimal number, demonstrating the effectiveness of our fusion strategy. Specifically, compared with 1-block KMTPLS, the improved accuracy of action "jumping" with respect to Q by 4-block KMTPLS accumulated to 0.0164, 0.0122 and 0.0206 for subjects "s6", "s10" and "s12".

Figure 4 illustrates the best prediction results among the C_6^T combinations in each T-block case. We also see that the predictive enhancement becomes smaller as more blocks were incorporated into the model, which was then followed by a slight decrease when the number of blocks exceeded an optimal threshold. For instance, in Figure 4, the optimal number of blocks for subject "s10" in "jumping" action is 3. This reflects the fact that adding too many blocks, on the other hand, may increase the chances of overfitting or bring noise into the model, leading to a degraded predictive ability.

Conclusion

We have proposed a generalized nonlinear tensor regression framework KMTPLS that effectively fuses the information from multiple tensorial data sources and unifies the single and multiblock tensor regression into one model using both common and discriminative features. Compared with multilinear model MMCR (Smilde, Westerhuis, and Boque 2000), our approach can successfully deal with the nonlinear

dependencies between multiple response and predictor tensor blocks by combining kernel machines with joint Tucker decomposition, resulting in significantly enhanced predictive power.

The experimental results on both UMPM and MHAD databases have demonstrated the advantages of our method when applied to the real-world tensor regression task in computer vision. For future work, we would like to mention that performance could be further improved at the cost of running time for designing and adapting more sophisticated tensorial kernel functions by exploiting more structural information of tensorial data than simply using polynomial or Gaussian kernel functions.

Acknowledgments

This work was supported in part by JSPS KAKENHI (Grant No. 15K15955), the National Natural Science Foundation of China (Grant No. 61202155) and the Natural Science and Engineering Council of Canada (NSERC).

References

Abdi, H. 2010. Partial least squares regression and projection on latent structure regression (PLS regression). *Wiley Interdisciplinary Reviews: Computational Statistics* 2(1):97–106.

Bro, R. 1996. Multiway calidration. multilinear PLS. *Journal of chemometrics* 10:47–61.

Carroll, J. D., and Chang, J.-J. 1970. Analysis of individual differences in multidimensional scaling via an N-way generalization of eckart-young decomposition. *Psychometrika* 35(3):283–319.

Cichocki, A.; Mandic, D.; De Lathauwer, L.; Zhou, G.; Zhao, Q.; Caiafa, C.; and Phan, H. A. 2015. Tensor decompositions for signal processing applications: From two-way to multiway component analysis. *Signal Processing Magazine, IEEE* 32(2):145–163.

Cichocki, A. 2013. Tensor decompositions: a new concept in brain data analysis? *arXiv preprint arXiv:1305.0395*.

De Lathauwer, L.; De Moor, B.; and Vandewalle, J. 2000. A multilinear singular value decomposition. *SIAM journal on Matrix Analysis and Applications* 21(4):1253–1278.

Eliseyev, A., and Aksenova, T. 2013. Recursive n-way partial least squares for brain-computer interface. *PloS one* 8(7):e69962.

Guo, W.; Kotsia, I.; and Patras, I. 2012. Tensor learning for regression. *Image Processing, IEEE Transactions on* 21(2):816–827.

Harshman, R. A. 1970. Foundations of the parafac procedure: Models and conditions for an "explanatory" multi-modal factor analysis.

Hou, M., and Chaib-draa, B. 2015. Hierarchical tucker tensor regression: Application to brain imaging data analysis. In *Image Processing (ICIP), 2015 IEEE International Conference on*. IEEE.

Hou, M.; Wang, Y.; and Chaib-draa, B. 2015. Online local gaussian process for tensor-variate regression: Application to fast reconstruction of limb movements from brain signal. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE.

Kim, H.; Zhou, J. X.; Morse III, H. C.; and Park, H. 2005. A three-stage framework for gene expression data analysis by L1-norm support vector regression. *International journal of bioinformatics research and applications* 1(1):51–62.

Kolda, T. G., and Bader, B. W. 2009. Tensor decompositions and applications. *SIAM review* 51(3):455–500.

Li, X.; Zhou, H.; and Li, L. 2013. Tucker tensor regression and neuroimaging analysis. *arXiv preprint arXiv:1304.5637*.

Luo, L.; Bao, S.; and Gao, Z. 2015. Quality prediction based on HOPLS-CP for batch processes. *Chemometrics and Intelligent Laboratory Systems* 143:28–39.

Ofli, F.; Chaudhry, R.; Kurillo, G.; Vidal, R.; and Bajcsy, R. 2013. Berkeley mhad: A comprehensive multimodal human action database. In *Applications of Computer Vision (WACV), 2013 IEEE Workshop on*, 53–60. IEEE.

Rosipal, R., and Trejo, L. J. 2002. Kernel partial least squares regression in reproducing kernel hilbert space. *The Journal of Machine Learning Research* 2:97–123.

Smilde, A. K.; Westerhuis, J. A.; and Boque, R. 2000. Multiway multiblock component and covariates regression models. *Journal of Chemometrics* 14(3):301–331.

Tucker, L. R. 1963. Implications of factor analysis of three-way matrices for measurement of change. *Problems in measuring change* 15.

Van Der Aa, N.; Luo, X.; Giezeman, G.; Tan, R.; and Veltkamp, R. 2011. Utrecht multi-person motion (UMPM) benchmark. Technical report, Citeseer.

Westerhuis, J. A.; Kourti, T.; and MacGregor, J. F. 1998. Analysis of multiblock and hierarchical PCA and PLS models. *Journal of chemometrics* 12(5):301–321.

Zhao, Q.; Caiafa, C. F.; Mandic, D. P.; Chao, Z. C.; Nagasaka, Y.; Fujii, N.; Zhang, L.; and Cichocki, A. 2013a. Higher order partial least squares (HOPLS): a generalized multilinear regression method. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 35(7):1660–1673.

Zhao, Q.; Zhou, G.; Adali, T.; Zhang, L.; and Cichocki, A. 2013b. Kernel-based tensor partial least squares for reconstruction of limb movements. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, 3577–3581. IEEE.

Zhao, Q.; Zhou, G.; Zhang, L.; and Cichocki, A. 2014. Tensor-variate gaussian processes regression and its application to video surveillance. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, 1265–1269. IEEE.

Zhao, Q.; Zhang, L.; and Cichocki, A. 2014. Multilinear and nonlinear generalizations of partial least squares: an overview of recent advances. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 4(2):104–115.

Zhou, H.; Li, L.; and Zhu, H. 2013. Tensor regression with applications in neuroimaging data analysis. *Journal of the American Statistical Association* 108(502):540–552.