Proceedings of the ASME 2016 International Manufacturing Science and Engineering Conference
MSEC2016
June 27-July 1, 2016, Blacksburg, Virginia, USA

MSEC2016-8750

PROFILE MONITORING AND FAULT DIAGNOSIS VIA SENSOR FUSION FOR ULTRASONIC WELDING

Weihong Guo*
Rutgers, The State University of New Jersey
Piscataway, NJ, USA

Jionghua (Judy) Jin, S. Jack Hu
University of Michigan
Ann Arbor, MI, USA

ABSTRACT

Sensor signals acquired during the manufacturing process contain rich information that can be used to facilitate effective monitoring of operational quality, early detection of system anomalies, quick diagnosis of fault root causes, and intelligent system design and control. This paper develops a method for effective monitoring and diagnosis of multi-sensor heterogeneous profile data based on multilinear discriminant analysis. The proposed method operates directly on the multi-stream profiles and then extracts uncorrelated discriminative features through tensor-to-vector projection, and thus preserving the interrelationship of different sensors. The extracted features are then fed into classifiers to detect faulty operations and recognize fault types. The developed method is demonstrated with both simulated and real data from ultrasonic metal welding.

KEYWORDS

Profile monitoring, fault diagnosis, tensor decomposition, sensor fusion.

1. INTRODUCTION

The wide applications of low-cost and smart sensing devices along with fast and advanced computer systems have resulted in a rich data environment, which makes a large amount of data available in many applications. Sensor signals acquired during the process contain rich information that can be used to facilitate effective monitoring of operational quality, early detection of system anomalies, quick diagnosis of fault root causes, and intelligent system design and control. In discrete manufacturing and many other applications, the sensor measurements provided by online sensing and data capturing technology are time- or spatial-dependent functional data, also called profile data [1, 2]. In this paper, we are particularly interested in cycle-based profile data, which are collected from repetitive operational cycles of a discrete manufacturing process. Examples of cycle-based profile data include the tonnage signals in stamping processes [3], the pressing force signals in a valve seat assembly operation [4], and the power signals and displacement signals in ultrasonic metal welding [5].

There is extensive research on the modeling and monitoring of cycle-based profile data in the literature [6], a large portion of which focuses on single signal analysis. However, there is a strong industrial interest for multi-signal applications, especially in cases where a single signal does not provide enough information to effectively evaluate the performance of the process. This leads to an increasing demand for multi-sensor fusion methods to analyze the multiple signals captured from different sensors for process monitoring and system diagnostics purposes.

One motivating example is the ultrasonic metal welding process for joining lithium-ion batteries for electric vehicles [7], as illustrated in Figure 1. It is important to develop an online monitoring method to facilitate in-process quality control and fault diagnosis to allow for a faster implementation of corrective actions. In order to have a better understanding of the process, four sensors are installed in the welding machine [8]: the power meter monitors controller power signal, the force sensor measures the clamping force, the displacement sensor measures the displacement between horn and anvil, and the microphone captures the sound during vibration. Table 1 summarizes the sensors, their signal types, and purposes.

* Corresponding author. Tel.: (+1)734-730-5978.
E-mail address: wg152@rci.rutgers.edu.

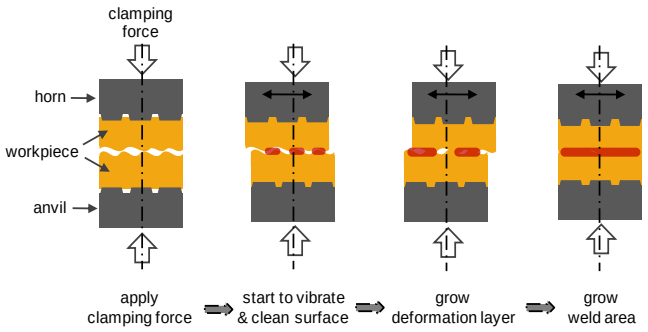


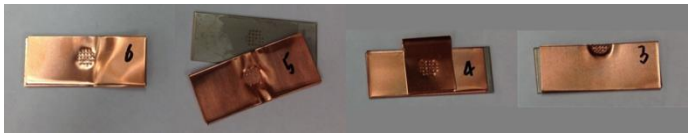
Figure 1. Ultrasonic metal welding process [7]

Table 1. Applied sensors, signal types, and purposes [8]

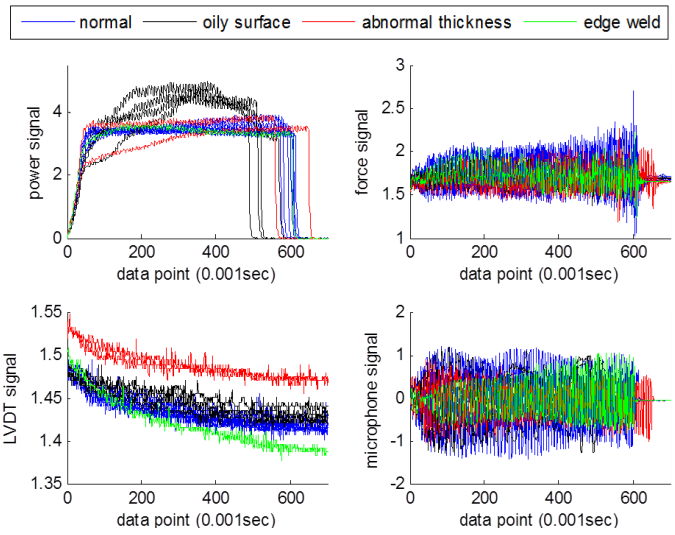
Sensor	Signal type	Purpose
Watt (power) meter	Ultrasonic power output at piezo-ceramic module	Monitor controller power input signal
Force sensor	Clamping force output at piezo-ceramic module	Measure clamping force at the ultrasonic transducer
LVDT sensor	Displacement between horn and anvil	Measure indentation and sheet thickness variation during welding
Microphone	Sound wave form	Detect cracking and slipping during welding

Figure 2(a) shows the welded tabs from the normal welding process and three faulty processes: (1) surface contamination, (2) abnormal thickness, and (3) mislocated/edge weld. Figure 2(b) shows the sensor signals from these processes. In general, the normal welding process produces good welds with strong connections, while the faulty processes tend to create poor quality connections which may have adverse effects on the performance of the battery pack. If samples are contaminated, for example, with oil, there is less friction between the metal layers, causing insufficient vibration at the beginning of the weld. So, the power signal does not rise as fast as a normal weld does. Once oil gets removed by vibration, the power signal picks up. Abnormal welding thickness may be caused by material handling errors, or sheet metal distortion, or operation errors. The displacement signal clearly shows how the displacement between horn and anvil is affected by thicker layers. Mislocated/edge weld may be caused by operation errors or alignment errors. With edge weld, all clamping force is applied to a smaller weld region, resulting in more displacement between horn and anvil towards the end of the weld. It can be seen from Figure 2 that on the one hand each signal contains richer information about product quality and process condition than any single point can provide, and on the other hand a single stream of signals is not informative enough for recognizing the type of faults.

There have been many research efforts on multi-sensor data fusion in manufacturing operations. A large portion of the multi-sensor data fusion methods is based on extracting a single feature from the monitoring signals, e.g., peak value, a weighted summation of signals, etc. The main limitations of this approach include the loss of information involved in the feature extraction process, the loss of sensor-to-sensor correlations, and the problem-dependent nature of the synthesizing scheme. Although profile monitoring techniques have been demonstrated to be more effective than synthetic index-based methods in monitoring processes characterized by repeating patterns [6], only a few authors have studied profile monitoring approaches in the field of sensor fusion [9-11]. Recently, with the fast development of multilinear methods for face recognition, Paynabar *et al.* [12] proposed a multi-channel profile monitoring and fault diagnosis method based on uncorrelated multilinear principal component analysis (UMPCA) [13], whereas Grasso *et al.* [14] investigated the problem of multi-stream profile monitoring using multilinear PCA (MPCA) [15]. Multi-channel profiles are homogeneous, in which all sensors measure the same variable, whereas multi-stream signals are heterogeneous, in which various sensors measure different variables.



(a) Welds from the normal welding process and three faulty processes: surface contamination, abnormal thickness, and mislocated/edge weld (from left to right)



(b) Sensor signals from the normal welding process and three faulty processes

Figure 2. Sensor signals and samples from ultrasonic metal welding processes

In this study, we investigate the use of multilinear extensions of linear discriminant analysis (LDA) to deal with multi-stream signals for the purpose of process monitoring and

fault diagnosis. LDA has been widely used as an effective tool for dimension reduction and discriminant analysis of complex data. Regular LDA is a linear algorithm that can only operate on vectors, thus cannot be directly applied to multi-stream profiles. To apply LDA to multi-stream profiles, these profiles need to be combined and reshaped (vectorized) into vectors first. So, this method is referred to as Vectorized-LDA (VLDA). Applying LDA to this high-dimensional vector creates high computational complexity due to the dimension of scatter matrices. Moreover, vectorization breaks the natural structure and correlation in the original data, e.g., sensor-to-sensor correlation, and potentially loses more useful representations that can be obtained in the original form. Lu *et al*. [16] introduced an uncorrelated multilinear LDA (UMLDA) framework as an alternative to VLDA. UMLDA is a multilinear dimensionality reduction and feature extraction method that operates directly on the multidimensional objects, known as tensor objects, rather than their vectorized versions. The UMLDA extracts uncorrelated discriminative features directly from tensorial data through solving a tensor-to-vector projection. Although MPCA and UMPCA are also multilinear subspace feature extraction algorithms operating directly on the tensorial representations, similar to PCA, they are both unsupervised methods that do not make use of the class information. In manufacturing and many other applications, training samples from various classes can be easily collected in an efficient manner. In these applications, supervised multilinear methods like UMLDA take class information into considerations and thus may be more suitable for fault recognition. Although there is some exploratory research on the applications of UMLDA to image processing on face and gait recognition tasks [16], very little research could be found in the literature on using the UMLDA technique for analyzing multi-stream nonlinear profiles for the purpose of fault detection and diagnosis.

Therefore, the main objective of this paper is to propose a UMLDA-based approach for analyzing multi-stream profiles that considers the interrelationship of different sensors. The features extracted by the proposed UMLDA-based method can effectively discriminate different classes and provide fault diagnosis results. The effectiveness of the proposed method is tested on both simulations and a real-world case study in the ultrasonic metal welding process.

The remainder of this paper is organized as follows. Section 2 presents the method for analysis and dimension reduction of multi-stream profiles using UMLDA. VLDA is also reviewed in this section. Section 3 compares the proposed UMLDA-based method with VLDA and its variants, and other competitor methods including UMPCA-based and MPCA-based methods in the performance of extracting discriminative features and recognizing the type of faults. A case study of ultrasonic metal welding process is given in Section 4. Finally, Section 5 concludes the paper with the discussion of broader impacts.

2. DIMENSION REDUCTION OF MULTI-STREAM SIGNALS USING UMLDA AND VLDA

Multi-way data analysis is the extension of two-way methods to higher-order datasets. This section first reviews the basic notations and concepts in multilinear algebra, and then introduces the implementation of UMLDA and VLDA for the purpose of dimensionality reduction in handling multi-stream signals. More details on the theoretical foundations of the mathematical development of UMLDA based on multilinear algebra can be found in [17-19]. The algorithm we use in this paper for extracting uncorrelated features from tensor data is based on the theories presented in those articles.

2.1. Basic multilinear algebra concepts and tensor-to-vector projection

An L-way array $\mathcal{A}$ is an Lth-order tensor object $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_L}$ such that I_l represents the dimension of the l-mode, $l = 1, \ldots, L$, where the term *mode* refers to a generic set of entities [20]. The l-mode vectors of $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_L}$ are defined as the I_l-dimensional vectors obtained from $\mathcal{A}$ by varying the index i_l $(i_l = 1, \ldots, I_l)$ while keeping all the other indices fixed. In multilinear algebra, a matrix $\mathbf{A}$ can be considered to be a second-order tensor. The column vectors and row vectors are considered as the 1-mode and 2-mode vectors of the matrix, respectively. The l-mode product of a tensor $\mathcal{A}$ by a matrix $\mathbf{U} \in \mathbb{R}^{J_l \times I_l}$, denoted by $\mathcal{A} \times_l \mathbf{U}$, is a tensor with entries $(\mathcal{A} \times_l \mathbf{U})(i_1, \ldots i_{l-1}, j_l, i_{l+1}, \ldots, i_L) = \sum_{i_l} \mathcal{A}(i_1, \ldots, i_L) \cdot \mathbf{U}(j_l, i_l)$. The scalar product of two tensors $\mathcal{A}, \mathcal{B} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_L}$ is defined as $\langle \mathcal{A}, \mathcal{B} \rangle = \sum_{i_1} \sum_{i_2} \ldots \sum_{i_L} \mathcal{A}(i_1, i_2, \ldots, i_L) \cdot \mathcal{B}(i_1, i_2, \ldots, i_L)$.

To project tensorial data into a subspace for better discrimination, there are two general forms of multilinear projection: the tensor-to-tensor projection (TTP) and the tensor-to-vector projection (TVP). The TVP projects a tensor to a vector and it can be viewed as multiple projections from a tensor to a scalar. A tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_L}$ can be projected to a point y through L unit projection vectors $\left\{\mathbf{u}^{(1)^T}, \mathbf{u}^{(2)^T}, \ldots, \mathbf{u}^{(L)^T}\right\}$ as $y = \mathcal{A} \times_1 \mathbf{u}^{(1)^T} \times_2 \mathbf{u}^{(2)^T} \ldots \times_L \mathbf{u}^{(L)^T} = \left\langle \mathcal{A}, \mathbf{u}^{(1)} \circ \mathbf{u}^{(2)} \circ \ldots \circ \mathbf{u}^{(L)} \right\rangle \equiv \langle \mathcal{A}, \mathcal{U} \rangle$, $\mathbf{u}^{(l)} \in \mathbb{R}^{I_l \times 1}$, $\left\|\mathbf{u}^{(l)}\right\| = 1$ for $l = 1, \ldots, L$, where $\|\cdot\|$ is the Euclidean norm for vectors. This multilinear projection $\{\mathbf{u}^{(1)^T}, \mathbf{u}^{(2)^T}, \ldots, \mathbf{u}^{(L)^T}\}$ is called an elementary multilinear projection (EMP), which is the projection of a tensor on a single line (resulting a scalar) and it consists of one projection vector in each mode. The TVP of a tensor object $\mathcal{A}$ to a vector $\mathbf{y} \in \mathbb{R}^P$ in a P-dimensional vector space consists of P EMPs, which can be written as $\left\{\mathbf{u}_p^{(1)^T}, \mathbf{u}_p^{(2)^T}, \ldots, \mathbf{u}_p^{(L)^T}\right\}_{p=1,\ldots,P} = \left\{\mathbf{u}_p^{(l)^T}, l = 1, \ldots, L\right\}_{p=1}^{P}$. The TVP from $\mathcal{A}$ to $\mathbf{y}$ is then written as $\mathbf{y} = \mathcal{A} \times_{l=1}^{L} \left\{\mathbf{u}_p^{(l)^T}, l = 1, \ldots, L\right\}_{p=1}^{P}$, where the pth component of $\mathbf{y}$ is obtained from the pth EMP as $\mathbf{y}(p) = \mathcal{A} \times_1 \mathbf{u}_p^{(1)^T} \times_2 \mathbf{u}_p^{(2)^T} \ldots \times_L \mathbf{u}_p^{(L)^T}$.

In the frame of multi-stream profile data, the simplest L-way array representing the signals is a third-order tensor object $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times M}$ such that I_1 is the number of sensors, I_2 is the number of data points collected on each profile, and M is the number of multi-stream profiles or samples. Note that more articulated datasets may be generated by introducing additional modes, e.g., by adding a further mode to group together different families of sensors.

2.2. The UMLDA approach

Multilinear subspace feature extraction algorithms operating directly on tensor objects without changing their tensorial structure are emerging. Since LDA is a classical algorithm that has been very successful and applied widely in various applications, there have been several variants of its multilinear extension proposed, named multilinear discriminant analysis (MLDA) in general. The projected tensors obtained from MLDA, however, are correlated contrary to classical LDA. To overcome this issue, Lu *et al.* [16] proposed UMLDA, in which a TVP projection is used for projection. In this subsection, we review the UMLDA method proposed in [16].

The derivation of the UMLDA algorithm follows the classic LDA derivation of minimizing the within-class distance and maximizing the between-class distance simultaneously, thus achieving maximum discrimination. A number of EMPs are solved one by one to maximize the discriminant criterion with an enforced zero-correlation constraint. To formulate the UMLDA problem, let $\left\{y_{m_p}, m = 1, \dots, M\right\}$ denote the pth projected scalar features, where M is the number of training samples and y_{m_p} is the projection of the mth sample $\mathcal{A}_m$ by the pth EMP $\left\{\mathbf{u}_p^{(1)^T}, \mathbf{u}_p^{(2)^T}\right\}$: $y_{m_p} = \mathcal{A}_m \times_1 \mathbf{u}_p^{(1)^T} \times_2 \mathbf{u}_p^{(2)^T}$. Adapting the classical Fisher Discriminant Criterion (FDC) to scalar sample, the between-class scatter $S_{B_p}^{\mathbf{y}}$ and the within-class scatter $S_{W_p}^{\mathbf{y}}$ are

$$S_{B_p}^{\mathbf{y}} = \sum_{c=1}^{C} N_c \left(\bar{y}_{c_p} - \bar{y}_p\right)^2, \; S_{W_p}^{\mathbf{y}} = \sum_{m=1}^{M} \left(y_{m_p} - \bar{y}_{c_{m_p}}\right)^2, \quad (1)$$

where C is the number of classes, N_c is the number of samples for class c, c_m is the class label for the mth training sample, $\bar{y}_p = (1/M)\sum_m y_{m_p} = 0$ assuming the training samples are zero-mean, and $\bar{y}_{c_p} = (1/N_c)\sum_{m,c_m=c} y_{m_p}$. Thus, the FDC for the pth scalar samples is $F_p^{\mathbf{y}} = S_{B_p}^{\mathbf{y}} / S_{W_p}^{\mathbf{y}}$. Let $\mathbf{g}_p$ denote the pth coordinate vector, with its mth component $\mathbf{g}_p(m) = y_{m_p}$. The objective of UMLDA is to determine a set of P EMPs that maximize the scatter ratio while producing uncorrelated features. The mathematical formulation of UMLDA can be written as

$$\left\{\mathbf{u}_p^{(1)^T}, \mathbf{u}_p^{(2)^T}\right\} = argmax\, F_p^{\mathbf{y}} \quad (2)$$

$$\text{subject to} \qquad \left\|\mathbf{u}_p^{(1)}\right\| = 1, \; \left\|\mathbf{u}_p^{(2)}\right\| = 1,$$

$$\frac{\mathbf{g}_p^T \mathbf{g}_q}{\|\mathbf{g}_p\| \|\mathbf{g}_q\|} = \delta_{pq}, \qquad p, q = 1, \dots, P$$

where $\delta_{pq} = 1$ for $p = q$ and $\delta_{pq} = 0$ otherwise. The solution to this problem is provided by using the successive determination approach.

The implementation of UMLDA given by Lu *et al.* (2009) for the purpose of face recognition introduces a regularization parameter γ (R-UMLDA). To solve for $\mathbf{u}_p^{(l^*)}$ in the l^*-mode, assuming that $\left\{\mathbf{u}_p^{(l)}, l \neq l^*\right\}$ is given, the tensor samples are projected in these ($L-1$ modes) $\{l \neq l^*\}$ to obtain vectors $\tilde{\mathbf{y}}_{m_p}^{(l^*)} = \mathcal{A}_m \times_{l=1, l\neq l^*}^{L} \left\{\mathbf{u}_p^{(l)^T}, l = 1, \dots, l^*-1, l^*+1, \dots, L\right\}_{p=1}^{P}$. The regularized within-class scatter matrix $\tilde{S}_{W_p}^{(l^*)}$ is defined as

$$\tilde{S}_{W_p}^{(l^*)} = \sum_{m=1}^{M} \left(\tilde{\mathbf{y}}_{m_\mathbf{p}}^{(l^*)} - \bar{\tilde{\mathbf{y}}}_{c_{m_\mathbf{p}}}^{(l^*)}\right)\left(\tilde{\mathbf{y}}_{m_\mathbf{p}}^{(l^*)} - \bar{\tilde{\mathbf{y}}}_{c_{m_\mathbf{p}}}^{(l^*)}\right)^T + \gamma \cdot \lambda_{max}\left(\check{S}_W^{(l^*)}\right) \cdot \mathbf{I}_{I_{l^*}} \quad (3)$$

where $\gamma \geq 0$ is a regularization parameter, $\mathbf{I}_{I_{l^*}}$ is an identity matrix of size $I_{l^*} \times I_{l^*}$, and $\lambda_{max}\left(\check{S}_W^{(l^*)}\right)$ is the maximum eigenvalue of $\check{S}_W^{(l^*)}$, which is the within-class scatter matrix for the $l-mode$ vectors of the training samples, defined as $\check{S}_W^{(l^*)} = \sum_{m=1}^{M}\left(\mathbf{A}_{m(l^*)} - \bar{\mathbf{A}}_{c_m(l^*)}\right)\left(\mathbf{A}_{m(l^*)} - \bar{\mathbf{A}}_{c_m(l^*)}\right)^T$, where $\bar{\mathbf{A}}_{c_m(l^*)}$ is the l^*-mode unfolded matrix of the class mean tensor $\bar{\mathcal{A}}_c = (1/N_c)\sum_{m,c_m=c} \mathcal{A}_m$.

The purpose of introducing the regularization parameter is to improve the UMLDA algorithm under small sample size scenario, where the dimensionality of the input data is high, but the number of training samples for some classes is too small to represent the true characteristics of their classes. This is a common case in small scale production like prototyping or personalized production. This scenario may also occur when a certain type of fault exists but rare, and that the data from that fault case is limited. If the number of training samples is too small, the iterations tend to minimize the within-class scatter towards zero in order to maximize the scatter ratio. Having a regularization parameter in the within-class scatter ensures that during the iteration, less focus is put on shrinking the within-class scatter. The basic UMLDA is obtained by setting $\gamma = 0$.

Based on the observations in [16], the sensitivity of the R-UMLDA to initialization and regularization suggests that R-UMLDA is not a very stable feature extractor and it is good for ensemble-based learning. Regularized UMLDA with aggregation (R-UMLDA-A) is hence introduced to aggregate several differently initialized and regularized UMLDA feature extractors to achieve better classification results. To focus on feature extraction performance, simple aggregation at the matching score level using the nearest-neighbor distance is implemented in R-UMLDA-A. Let A denote the number of R-UMLDA feature extractors to be aggregated. To classify a test sample $\mathcal{A}$, it is projected to A feature vectors $\left\{\mathbf{y}_{(a)}\right\}_{a=1,\dots,A}$

using the A TVPs first. The test sample $\mathcal{A}$ is assigned the label $c^* = \arg min_c\, d(\mathcal{A}, c)$ using the aggregated nearest-neighbor distance.

2.3. The VLDA approach

VLDA is a generalization of LDA to tensor data, which applies the regular LDA to a tensor object reshaped into a vector. In the frame of multi-stream profile data, the third-order tensor object $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times M}$ representing the signals is unfolded slice by slice; the slices are then rearranged into a large two-dimensional matrix $\mathbf{A} \in \mathbb{R}^{I_1 I_2 \times M}$, where I_1 is the number of sensors, I_2 is the number of data points collected on each profile, and M is the number of samples. The classical LDA is then performed on matrix $\mathbf{A}$. What we seek is a transformation matrix $\mathbf{W}$ that maximizes the ratio of the between-class scatter to the within-class scatter

$$\mathbf{W} = argmax\, J(\mathbf{W}) = argmax\, \frac{|\mathbf{W}^T \boldsymbol{S_B} \mathbf{W}|}{|\mathbf{W}^T \boldsymbol{S_W} \mathbf{W}|} \qquad (4)$$

$$\text{subject to} \quad \|\mathbf{w}_i\| = 1, i = 1, \dots, c-1$$

where $\boldsymbol{S_B}$ and $\boldsymbol{S_W}$ are the between-class scatter and within-class scatter, respectively, c is the number of classes. The transformed signal samples can be obtained by $\mathbf{y} = \mathbf{W}^T \mathbf{A}$. More details on the calculation of $\boldsymbol{S_B}$ and $\boldsymbol{S_W}$ using Fisher linear discriminant can be found in [21].

3. Performance Comparison in Simulation

In this section, the performances of the UMLDA and VLDA methodologies are evaluated and compared by means of simulations. The multi-stream signals in simulation are generated in a similar manner as in Grasso *et al.* [14]: a four-stream profile dataset is generated based on three benchmark signals proposed by Donoho and Johnstone [22]. Figure 3 illustrates the three benchmark signals: 'blocks', 'heavysine', and 'bumps', and they are denoted as $\boldsymbol{x}_1, \boldsymbol{x}_2$, and $\boldsymbol{x}_3$, respectively.

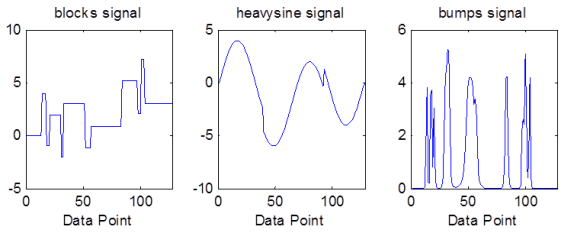


Figure 3. Benchmark signals 'blocks', 'heavysine', and 'bumps'

Let $\chi \in \mathbb{R}^{N \times K \times M}$ denote the third-order tensor object that represents the four-stream profile dataset, where $N = 4$ is the number of streams or sensors, $K = 128$ is the number of data points for all the signals, and M is the number of samples. χ is generated to contain different types of correlation structures: linear correlation (e.g., $\chi_{1,\cdot,m}$ and $\boldsymbol{x}_1$, $\chi_{2,\cdot,m}$ and $\boldsymbol{x}_3$, etc.), curvilinear correlation (e.g., $\chi_{2,\cdot,m}$ and $\boldsymbol{x}_1$, $\chi_{3,\cdot,m}$ and $\boldsymbol{x}_2$, etc.), and no correlation (e.g., $\chi_{3,\cdot,m}$ and $\boldsymbol{x}_1$, $\chi_{4,\cdot,m}$ and $\boldsymbol{x}_3$, etc.). χ is defined as follows:

$$\begin{aligned} \chi_{1,\cdot,m} &= b_{1,m}\boldsymbol{x}_1 + b_{2,m}\boldsymbol{x}_2 + \varepsilon_{1,m} \\ \chi_{2,\cdot,m} &= b_{3,m}\boldsymbol{x}_1^2 + b_{4,m}\boldsymbol{x}_3 + \varepsilon_{2,m} \\ \chi_{3,\cdot,m} &= b_{5,m}\boldsymbol{x}_2^2 + b_{6,m}\boldsymbol{x}_3^2 + \varepsilon_{3,m} \\ \chi_{4,\cdot,m} &= b_{7,m}\boldsymbol{x}_1\boldsymbol{x}_2 + \varepsilon_{4,m} \end{aligned} \qquad (m = 1, \dots, M) \quad (5)$$

where $\varepsilon_{n,m} \sim N(0, 0.5^2)$ is the random noise and $\boldsymbol{b}_m = [b_{1,m}, \dots, b_{7,m}]^T \sim MVN(\boldsymbol{\mu}_b, \Sigma_b)$ is the model parameter vector, $n = 1, \dots, 4, m = 1, \dots, M$. Similar to the dataset used in [14] the following settings are used to generate the dataset:
$\boldsymbol{\mu}_b = [0.2, 1, 1.5, 0.5, 1, 0.7, 0.8]^T$, $\Sigma_b = diag(\sigma_{b_1}^2, \dots, \sigma_{b_7}^2)$
$= diag(0.08, 0.015, 0.05, 0.01, 0.09, 0.03, 0.06)$.

Different out-of-control scenarios are generated to simulate different kinds of deviations from the natural multi-stream pattern. Each out-of-control scenario is associated with an assignable cause. In the context of ultrasonic metal welding (and many other manufacturing processes as well), these assignable causes represent different types of faults, e.g., mislocated weld, sheet metal distortion, surface contamination, etc. In this paper, we assume multiple faults do not occur simultaneously on one part, i.e., a single part has no more than one fault.

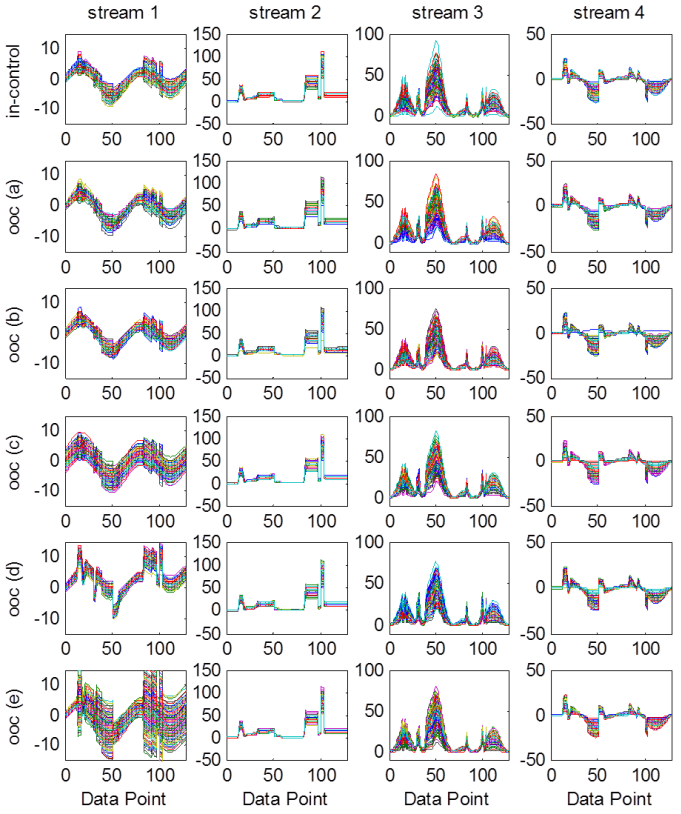


Figure 4. Simulation dataset: 1200 samples in 6 classes

We generated a total of 1200 profile samples with 200 samples in each class: in-control and five out-of-control scenarios (a) – (e). All 1200 samples in $C = 6$ classes are plotted in Figure 4. As indicated by Eq. (5), the four streams of signals are not independent; the correlation structure is complex for profile modeling. Half of these 1200 samples are considered as the training dataset. Specifically, the five out-of-control scenarios are:

(a) mean shift of the 'block' reference signals: $\boldsymbol{x}_1 \rightarrow \boldsymbol{x}_1 + 0.1\sigma_{x_1}\mathbf{1}_{K\times 1}$, resulting in $\tilde{\chi}_{1,\cdot,m} = b_{1,m}(\boldsymbol{x}_1 + 0.1\sigma_{x_1}\mathbf{1}_{K\times 1}) + b_{2,m}\boldsymbol{x}_2 + \varepsilon_{1,m}$, $\tilde{\chi}_{2,\cdot,m} = b_{3,m}\left(\boldsymbol{x}_1 + 0.1\sigma_{x_1}\mathbf{1}_{K\times 1}\right)^2 + b_{4,m}\boldsymbol{x}_3 + \varepsilon_{2,m}$, and $\tilde{\chi}_{4,\cdot,m} = b_{7,m}(\boldsymbol{x}_1 + 0.1\sigma_{x_1}\mathbf{1}_{K\times 1}) + \varepsilon_{4,m}$, where $0.1\sigma_{x_1}$ is the magnitude of the shift, σ_{x_1} is the standard deviation of $\boldsymbol{x}_1$ reference signal, and $\mathbf{1}_{K\times 1}$ is a column vector of ones;

(b) superimposition of a sinusoid term on the 'block' reference signal: $\boldsymbol{x}_1 \rightarrow \boldsymbol{x}_1 + 0.1\sigma_{x_1}\boldsymbol{y}_s$, $\boldsymbol{y}_s$ is a sine function, resulting in $\tilde{\chi}_{1,\cdot,m} = b_{1,m}(\boldsymbol{x}_1 + 0.1\sigma_{x_1}\boldsymbol{y}_s) + b_{2,m}\boldsymbol{x}_2 + \varepsilon_{1,m}$, $\tilde{\chi}_{2,\cdot,m} = b_{3,m}\left(\boldsymbol{x}_1 + 0.1\sigma_{x_1}\boldsymbol{y}_s\right)^2 + b_{4,m}\boldsymbol{x}_3 + \varepsilon_{2,m}$, and $\tilde{\chi}_{4,\cdot,m} = b_{7,m}(\boldsymbol{x}_1 + 0.1\sigma_{x_1}\boldsymbol{y}_s) + \varepsilon_{4,m}$, where $0.1\sigma_{x_1}$, and $\boldsymbol{y}_s$ is the sine function over the domain $[0, K]$, with period K and peak-to-peak amplitude equal to 1;

(c) standard deviation increase of the error term e_1: $\sigma_{\varepsilon_{1.m}} \rightarrow 3\sigma_{\varepsilon_{1.m}}$, resulting in $\tilde{\chi}_{1,\cdot,m} = b_{1,m}\boldsymbol{x}_1 + b_{2,m}\boldsymbol{x}_2 + \tilde{\varepsilon}_{1,m}$, where $\tilde{\varepsilon}_{1,m}{\sim}N(0, (3\times 0.5)^2)$, where $\sigma_{\varepsilon_{1.m}}$ is the standard deviation of the error term $\varepsilon_{1.m}$;

(d) mean shift of the model parameter b_1: $\mu_{b_1} \rightarrow \mu_{b_1} + 5\sigma_{b_1}$, resulting in $\tilde{\chi}_{1,\cdot,m} = \tilde{b}_{1,m}\boldsymbol{x}_1 + b_{2,m}\boldsymbol{x}_2 + \varepsilon_{1,m}$, where $\tilde{b}_{1,m}{\sim}N(\mu_{b_1} + 5\sigma_{b_1}, \sigma_{b_1}^2)$, where $5\sigma_{b_1}$, μ_{b_1} and σ_{b_1} are the mean value and standard deviation of the $w$the model parameter b_1;

(e) standard deviation increase of the model parameter b_1: $\sigma_{b_1} \rightarrow 4\sigma_{b_1}$, resulting in $\tilde{\chi}_{1,\cdot,m} = \tilde{b}_{1,m}\boldsymbol{x}_1 + b_{2,m}\boldsymbol{x}_2 + \varepsilon_{1,m}$, where $\tilde{b}_{1,m}{\sim}N\left(\mu_{b_1}, \left(4\sigma_{b_1}\right)^2\right)$.

Of the five scenarios above, all profiles in streams 1, 2, and 4 are affected in (a) and (b), while in (c), (d), and (e), only the profiles in stream 1 present out-of-control patterns.

Since a large amount of the $\tilde{\varepsilon}_{1,m}$'s generated by $\tilde{\varepsilon}_{1,m}{\sim}N\left(0, \left(3\sigma_{\varepsilon_{1.m}}\right)^2\right)$ in fault (c) would overlap with the $\varepsilon_{1,m}$'s generated by $\varepsilon_{1,m}{\sim}N(0, \sigma_{\varepsilon_{1,m}}^2)$ in the in-control class, and that the $\tilde{b}_{1,m}$'s generated by $\tilde{b}_{1,m}{\sim}N\left(\mu_{b_1}, \left(4\sigma_{b_1}\right)^2\right)$ in fault (e) would greatly overlap with the $b_{1,m}$'s generated by $b_{1,m}{\sim}N(\mu_{b_1}, \sigma_{b_1}^2)$ in the in-control class, faults (c) and (e) would be very difficult to separate from the in-control class.

3.1. Methods in comparison

The general framework of profile monitoring and fault diagnosis using multi-stream signals is illustrated in Figure 5. For multilinear methods like UMLDA, the multi-stream signals can be directly represented in a tensor object, and then the tensor is normalized so that the training samples are in the same dimension and zero-mean. For linear methods like VLDA, the multi-stream signals need to be vectorized to a matrix, and then followed by normalization. Feature extraction method, e.g., UMLDA or VLDA, then produces vector features that can be fed into standard classifiers for classification. The output is a tensor class labels which represents 'normal' or some fault type.

Performance comparison is conducted in two levels: (1) feature extraction performance, and (2) classification performance. To compare feature extraction performance, we use the following four multilinear and three linear methods to extract features: regularized UMLDA (R-UMLDA), regularized UMLDA with aggregation (R-UMLDA-A), UMPCA, MPCA, VLDA, uncorrelated LDA (V-ULDA), and regularized LDA (V-RLDA). The feature vectors obtained are then fed into the nearest-neighbour classifier (NNC) with the Euclidean distance measure for classification.

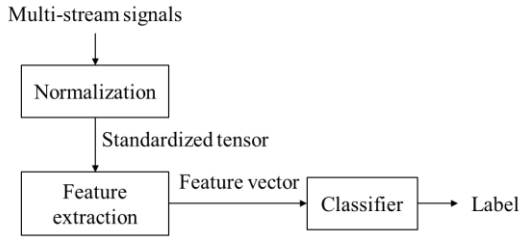


Figure 5. Framework of profile monitoring and fault diagnosis using multi-stream signals

In R-UMLDA, the regularization parameter γ is empirically set to $\gamma = 0.001$. Since the tensor object $\chi \in \mathbb{R}^{4\times 128\times M}$, one R-UMLDA will extract up to 4 features. In R-UMLDA-A, up to $A = 20$ differently initialized and regularized UMLDA feature extractors are combined with each producing up to 4 features, resulting in a total of 80 features. The regularization parameter ranges from 10^{-7} to 10^{-2}.

UMPCA and MPCA are unsupervised multilinear methods that seek a set of projections to maximize the variability captured by the projected tensor. UMPCA will produce up to 4 features which are uncorrelated, while MPCA will produce as many as approximately 80 features which are correlated in order to capture at least 99% of the variation in each mode. Details on the theoretical development of UMPCA and MPCA can be found in [13, 15].

In addition to VLDA, two more linear methods are included in comparison, V-ULDA and V-RLDA. V-ULDA and V-RLDA improve LDA on undersampled problems and small sample size problems, respectively. Each method will project to up to $C - 1$ features with C being the number of classes. Details on the theoretical development of ULDA and RLDA can be found in [23, 24].

3.2. Simulation results

Using the procedures described in Section 2 and Section 3.1, regularized UMLDA is applied to the generated data. In UMLDA, the eigentensors corresponding to the $p$the EMP, $\mathbf{U}_p \in \mathbb{R}^{4\times128}$, $p = 1,2,3,4$ are obtained by $\mathbf{u}_p^{(1)} \circ \mathbf{u}_p^{(2)}$, where $\mathbf{u}_p^{(1)} \in \mathbb{R}^{4\times1}$ and $\mathbf{u}_p^{(2)} \in \mathbb{R}^{128\times1}$. Figure 6 shows $\mathbf{U}_p$, $p = 1,2,3,4$, obtained from the training dataset in a single simulation run of Case A. Each row of $\mathbf{U}_p$ corresponds to one signal stream. As can be seen from Figure 6, the eigenvectors corresponding to the first EMP show an efficient discrimination against streams 1 and 4, whereas those corresponding to the second EMP show a strong discrimination against stream 2. The eigenvectors corresponding to the third and fourth EMPs show weak discriminations against stream 4, whereas limited useful information is extracted from stream 3 for discriminant analysis. These results are exactly compatible with the data generation model, thus implying that R-UMLDA can effectively extract information for discriminant analysis about multi-stream profiles.

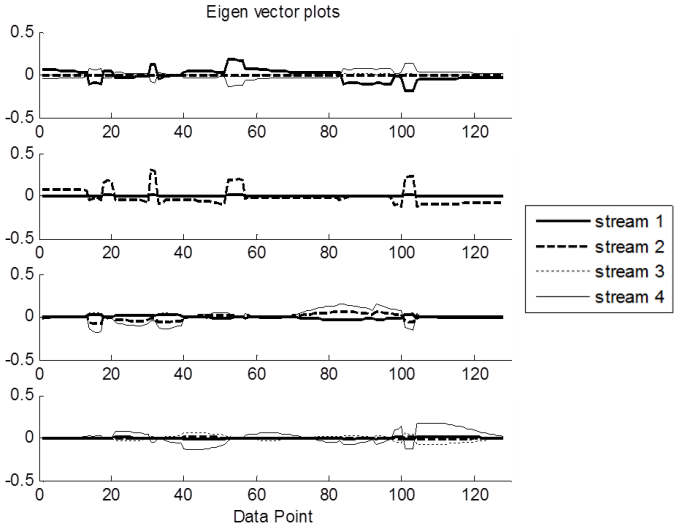


Figure 6. Eigentensors from R-UMLDA in simulation

Using the first p EMPs ($p = 1,2,3,4$), multi-stream profiles can be projected to p uncorrelated features, which are then fed into the nearest-neighbour classifier (NNC). The classification performance in the testing dataset is shown in Figure 7 and Table 2. Figure 7 plots the following detailed results against the number of features used:

- correct classification rate: $\sum_{m=1}^{M_{test}} I(\hat{c}_m = c_m)/M_{test}$, where $\hat{c}_m$ is the predicted class for sample m, c_m is the true class, and M_{test} is the number of testing samples;
- correct passing rate: $\sum_{m=1}^{M_{test}} I(\hat{c}_m = 0 | c_m = 0)/M_{test}$, where '0' indicates the 'normal' class;
- correct detection rate: $\sum_{m=1}^{M_{test}} I(\hat{c}_m > 0 | c_m > 0)/M_{test}$, where $c > 0$ indicates a fault class;
- true fault classification rate: $\sum_{m=1}^{M_{test}} I(\hat{c}_m = c_m | c_m > 0)/M_{test}$;
- rate of true detection but wrong fault classification: $\sum_{m=1}^{M_{test}} I(\hat{c}_m \neq c_m | \hat{c}_m > 0, c_m > 0)/M_{test}$.

As can be seen in Figure 7, the first two features extracted by R-UMLDA are the most powerful features in classification. Adding the third and fourth features slightly helps improve the correct classification rate.

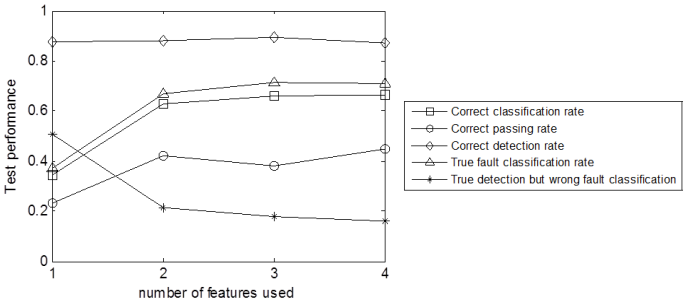


Figure 7. Classification performance of NNC for R-UMLDA features in testing dataset

Table 2. Confusion matrix of NNC for R-UMLDA features in testing dataset

1 feature	**Classified as**					
Actual	Normal	Fault (a)	Fault (b)	Fault (c)	Fault (d)	Fault (e)
Normal	23	23	23	25	1	5
Fault (a)	18	25	25	26	0	6
Fault (b)	18	27	26	24	0	5
Fault (c)	17	29	21	23	1	9
Fault (d)	1	0	0	1	77	21
Fault (e)	8	12	12	19	15	34
2 features	**Classified as**					
Actual	Normal	Fault (a)	Fault (b)	Fault (c)	Fault (d)	Fault (e)
Normal	42	0	5	41	1	11
Fault (a)	0	98	1	0	0	1
Fault (b)	4	1	74	17	0	4
Fault (c)	38	0	12	41	0	9
Fault (d)	0	0	0	1	81	18
Fault (e)	17	0	2	28	13	40
3 features	**Classified as**					
Actual	Normal	Fault (a)	Fault (b)	Fault (c)	Fault (d)	Fault (e)
Normal	38	0	1	46	1	14
Fault (a)	0	98	1	0	0	1
Fault (b)	0	1	98	0	0	1
Fault (c)	38	0	2	44	1	15
Fault (d)	0	0	0	2	79	19
Fault (e)	16	0	0	29	17	38
4 features	**Classified as**					
Actual	Normal	Fault (a)	Fault (b)	Fault (c)	Fault (d)	Fault (e)
Normal	45	0	0	40	0	15
Fault (a)	0	98	1	0	0	1
Fault (b)	0	0	99	1	0	0
Fault (c)	49	0	1	33	1	16
Fault (d)	0	0	0	2	80	18
Fault (e)	16	0	0	25	15	44

More detailed classification results with respect to the number of features fed into the classifier are shown in the confusion matrices in Table 2. From Table 2, we can easily observe an improvement in classification accuracy when two or more EMPs are used instead of using only the first one. We also notice that when two or more features are used, most of the classification errors come from separating the in-control class, fault (c), and fault (e) from each other. This observation is exactly compatible with the data generation model, based on which we have expected that faults (c) and (e) are the most difficult classes to separate from the in-control class.

Applying the competitor methods described in Section 3.1, Figure 8 shows the classification performance in terms of correct classification rate of NNC for various feature extraction methods in Case A testing dataset. The plotted results are the average correct classification rates in 100 simulation runs. In Figure 8, the curves with triangle markers correspond to classification performance for UMPCA and MPCA features. It is obvious that these results are significantly worse than LDA-based methods, regardless of the number of features used. This agrees with our understanding of PCA-based feature extractors which do not make use of the class information and only seek projections to maximize the captured variability instead of class discrimination.

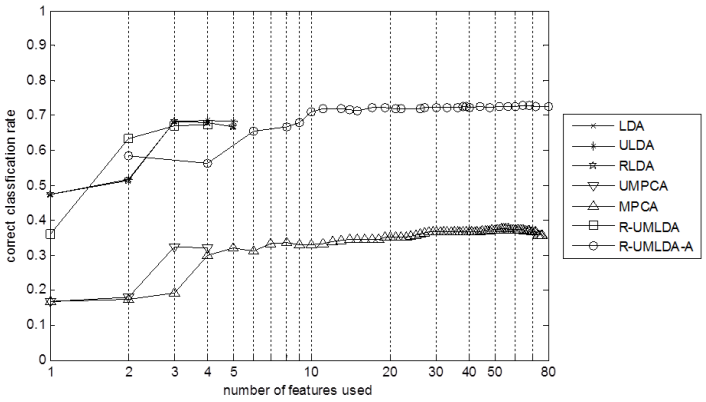


Figure 8. Classification performance of NNC for various feature extractors in testing dataset

The curves with cross, star, and asterisk markers in Figure 8 correspond to vectorized LDA methods (including LDA, ULDA, and RLDA), whereas the curves with square and circle markers correspond to UMLDA methods. It can be seen from Figure 8 that the first two features extracted by R-UMLDA are the most powerful features in classification. Beyond the first two features, the performance from R-UMLDA varies very slowly with an increased number of features used. The first three features extracted by vectorized LDA methods are also powerful, but the improvement from using the first two R-UMLDA features is not significant.

The best correct classification rate is achieved using R-UMLDA-A. Figure 8 shows that R-UMLDA-A outperforms all other algorithms. This demonstrates that aggregation is an effective procedure and there is indeed complementary discriminative information from differently regularized R-UMLDA feature extractors.

4. CASE STUDY IN MULTI-LAYER ULTRASONIC METAL WELDING

The ultrasonic metal welding example introduced at the beginning of this paper is analyzed in this section to demonstrate the proposed method for multi-stream profile monitoring and fault diagnosis. Ultrasonic welding is a critical process for joining lithium-ion batteries for electric vehicles. In this case study, welding experiments of joining three layers of copper with 1 layer of nickel plated copper are investigated. The clamping pressure is 34 psi and the vibration amplitude is 40 μm.

As described in the introduction, Figure 2(a) shows the welded tabs from the normal welding process and three faulty processes: (1) surface contamination, (2) abnormal thickness, and (3) mislocated/edge weld. Figure 2(b) shows signals associated with these welds from four sensors. These sensor signals provide rich information about the product quality and process condition. Both R-UMLDA and VLDA methods are trained using 8 normal samples, 2 samples with fault 1 (oily surface), 1 sample with fault 2 (abnormal thickness), and 1 sample with fault 3 (edge weld).

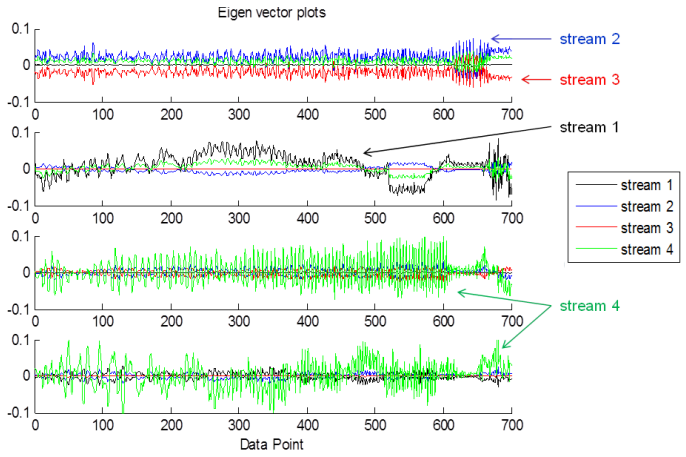


Figure 9. Eigentensors from R-UMLDA in ultrasonic metal welding

Using one R-UMLDA feature extractor with $\gamma = 0.001$, the eigentensors corresponding to the four EMPs are shown in Figure 9. It can be seen from this figure that the eigentensors corresponding to the first EMP show an efficient discrimination and strong negative correlation in streams 2 and 3. The eigentensors corresponding to the second EMP show a strong discrimination in stream 1, whereas those corresponding to the third and fourth EMPs deliver similar information on discrimination in stream 4.

After training UMLDA and VLDA, the feature extractors and NNC are applied to five testing samples: 2 from the normal process, 2 from fault 1, and 1 from fault 2. Figure 10 plots the classification performance in terms of correct classification rate

of NNC for UMLDA and VLDA in the testing samples. For the five testing samples, it can be seen that R-UMLDA-A can easily achieve 100% correct classification using only four features, while R-UMLDA achieves 80%. The vectorized LDA methods, however, do not perform as well as UMLDA. The features extracted by RLDA achieve the same level of classification accuracy as R-UMLDA, whereas LDA and ULDA extract much weaker features. The results indicate that UMLDA-based methods, especially R-UMLDA-A, outperforms VLDA methods (including LDA, ULDA, and RLDA) in detecting abnormal processes and fault diagnosis.

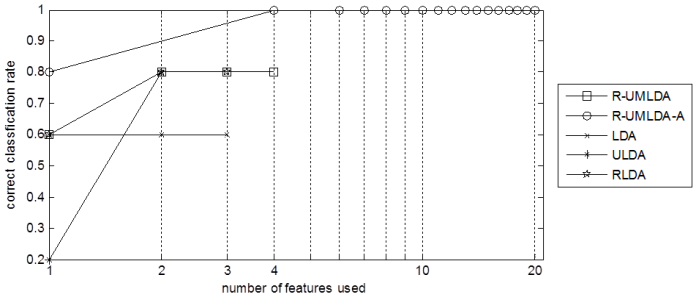


Figure 10. Classification performance of NNC for UMLDA and VLDA in ultrasonic welding

5. CONCLUSION

In this paper, based on UMLDA, we proposed a method for effective analysis of multi-sensor heterogeneous profile data. With various sensors measuring different variables, information from each sensor, sensor-to-sensor correlation, and class-to-class correlation should all be considered. A simulation study was conducted to evaluate the performance of the proposed method and its performance superiority over VLDA and other competitor methods. The results showed that the features extracted by VLDA and competitor methods are not as powerful as UMLDA in discriminating profiles and classification. We also applied both UMLDA and VLDA to a real case study of a multi-layer ultrasonic metal welding process for the purpose of process characterization and fault diagnosis. The results indicate that UMLDA outperforms VLDA in not only detecting the faulty operations but also classifying the type of faults.

In the future, several remaining issues in this framework will be studied in more depth, such as the impact of the number of training samples on UMLDA-based feature extraction, and the possibility of improving classification performance in fault diagnosis using ensemble learning with R-UMLDA. More comprehensive case study will be performed in the future as we collect more samples from welding experiments. Developing tensor-based methods for monitoring manufacturing processes with vision technology will be an interesting topic for future research. Furthermore, the extension of the developed method to online process monitoring and online learning would be an interesting development.

ACKNOWLEDGMENTS

This research is supported by the National Science Foundation and General Motors Collaborative Research Lab in Advanced Vehicle Manufacturing at The University of Michigan.

REFERENCES

[1] Woodall, W.H., Spitzner, D.J., Montgomery, D.C. and Gupta, S. (2004) Using Control Charts to Monitor Process and Product Quality Profiles. *Journal of Quality Technology*, 36, 309-320.

[2] Woodall, W.H. (2007) Current Research on Profile Monitoring. Produção, 17, 420-425.

[3] Jin, J and Shi, J. 1999. Feature-preserving Data Compression of Stamping Tonnage Information Using Wavelets. *Technometrics*, 41, 327-339.

[4] Paynabar, K and Jin, J. (2011) Characterization of Non-Linear Profiles Variations Using Mixed-Effect Models and Wavelets. *IIE Transactions*, 43: 4, 275-290

[5] Lee, S. S., Shao, C., Kim, T. H., Hu, S. J., Kannatey-Asibu, E., Cai, W. W., Spicer, J.P., Wang, H. and Abell, J. A. (2014). Characterization of Ultrasonic Metal Welding by Correlating Online Sensor Signals With Weld Attributes. *Journal of Manufacturing Science and Engineering*, 136, 051019.

[6] Noorossana, R., Saghaei, A and Amiri, A. (2012) Statistical Analysis of Profile Monitoring. New York: Wiley.

[7] Lee, S.S., Kim, T.H., Hu, S.J., Cai, W.W. and Abell, J.A. (2010) Joining Technologies for Automotive Lithium-ion Battery Manufacturing – A Review. *Proceedings of ASME 2010 International Manufacturing Science and Engineering Conference*, 541-549.

[8] Hu, S.J. (2011) *Technical Report: On-line Quality Monitoring System for Ultrasonic Battery Tab Welding*, General Motors Collaborative Research Lab at the University of Michigan, Ann Arbor, MI.

[9] Kim, J., Huang, Q., Shi, J. and Chang, T.-S. (2006) Online Multichannel Forging Tonnage Monitoring and Fault Pattern Discrimination Using Principal Curve. J*ournal of Manufacturing Science and Engineering*, 128, 944-950.

[10] Amiri, A., Zou, C. and Doroudyan, M.H. (2013) Monitoring Correlated Profile and Multivariate Quality Characteristics. *Quality and Reliability Engineering International*, 30, 133-142.

[11] Chou, S. H., Chang, S.I. and Tsai, T.R. (2014) On Monitoring of Multiple Non-linear Profiles. *International Journal of Production Researc*h, 52, 3209-3224.

[12] Paynabar, K., Jin, J., and Pacella, M. (2013) Monitoring and Diagnosis of Multichannel Nonlinear Profile Variations Using Uncorrelated Multilinear Principal Component Analysis. *IIE Transactions*, 45, 1235-1247.

[13] Lu, H., Plataniotis, K.N. and Venetsanopoulos, A.N. (2009) Uncorrelated Multilinear Principal Component Analysis for Unsupervised Multilinear Subspace Learning. *IEEE Transactions on Neural Network*s, 20, 1820–1836.
[14] Grasso, M, Colosimo, B.M. and Pacella, M. (2014) Profile Monitoring Via Sensor Fusion: The Use of PCA Methods for Multi-Channel Data. *International Journal of Production Research*, 52, 6110-6135.
[15] Lu, H., Plataniotis, K.N. and Venetsanopoulos, A.N. (2008) MPCA: Multilinear Principal Component Analysis of Tensor Objects. *IEEE Transactions on Neural Networks*, 19, 18–39.
[16] Lu, H., Plataniotis, K.N. and Venetsanopoulos, A.N. (2009) Uncorrelated Multilinear Discriminant Analysis With Regularization and Aggregation For Tensor Object Recognition. *IEEE Transactions on Neural Network*s, 20, 103-123.
[17] De Lathauwer, L., De Moor, B. and Vandewalle, J. (2000) A Multilinear Singular Value Decomposition. SIAM Journal on Matrix Analysis and Applications, 21, 253–1278.
[18] Kolda, T. G., and Bader, B. W. (2009). Tensor Decompositions and Applications. *SIAM review*, *51*(3), 455-500.
[19] Acar, E., and Yener, B. (2009) Unsupervised Multiway Data Analysis: A Literature Survey. *IEEE Transactions on Knowledge and Data Engineering*, 21, 6–20.
[20] Kiers, H. A. L. (2000). Towards a Standardized Notation and Terminology in Multiway Analysis. *Journal of Chemometrics*, 14, 105–122.
[21] Duda, R.O., Hart, P.E. and Stork, D.G. (2012) Pattern Classification. John Wiley & Sons.
[22] Donoho, D. L. and I. M. Johnstone, I.M. (1994) Ideal Spatial Adaptation by Wavelet Shrinkage. *Biometrik*a, 81, 425–455.
[23] Ye, J. (2005) Characterization of A Family of Algorithms for Generalized Discriminant Analysis on Undersampled Problems. *Journal of Machine Learning Research*, 6, 483–502.
[24] Ye, J., Xiong, T., Li, Q., Janardan, R., Bi, J., Cherkassky, V., and Kambhamettu, C. (2006) Efficient Model Selection for Regularized Linear Discriminant Analysis. *Proceedings of the 15th ACM International Conference on Information and Knowledge Management*, 532-539.