Tensorial Data Modeling and Analysis for Manufacturing Process Quality Control Using Sensor Data

By

Yaser Zerehsaz

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Industrial and Operations Engineering)
in The University of Michigan
2017

Doctoral Committee:

Professor Jionghua (Judy) Jin, Chair
Assistant Professor Eunshin Byon
Professor Elizaveta Levina
Assistant Professor Matthew Plumlee

Yaser Zerehsaz

yzereh@umich.edu

ORCID iD: 0000-0002-2248-3651

To my wife Mozhde

ACKNOWLEGMENTS

I would like to thank my advisor, Professor Judy Jin, for her forbearance, support and motivation enabling me to accomplish my PhD studies. The valuable insight that I have gained in my PhD program under Professor Jin's supervision is ineffable. The ability to systematically integrating the domain knowledge regarding the problem under the study with statistical knowledge is a magnificent skill which I was not acquainted with before I initiated my PhD with Professor Jin.

Special thanks go to my other committee members, Professor Elizaveta Levina, Professor Eunshine Byon, and Professor Matthew Plumlee for their constructive comments and tremendously helpful suggestions.

I would also like to thank Professor Matt Reed, for the great collaboration that I had with him in the first year of my PhD.

My special gratitude also goes to my previous advisors in my master studies, Professor Rassoul Noorossana and Professor Abbas Saghaei who taught me the fundaments of scientific research. I would like to thank my undergraduate advisor, Professor Kamran Paynabar, who was my main inspiration to pursue my graduate studies.

Best of thanks goes to my mother, Salimeh, whose generosity is definitely the most valuable keepsake which I have ever been given in my life. I would like to thank my father, Hassan, for all his sacrifices which I will not be able to compensate for.

I would also like to thank all my unforgettable friends in Ann Arbor. I am certain that without them, I would not be able to survive.

Finally, I would like to thank my wife, Mozhde. Without her, I would never be able to apprehend happiness, composure and self-confidence.

TABLE OF CONTENTS

LIST OF TABLES

.

LIST OF FIGURES

ABSTRACT

One of the main engines driving the new era of industrial data analytics is advanced sensing technologies and distributed computing. Although a vast amount of data can be garnered with the aid of these sensing and computing technologies, several challenges aggravating the data analysis arise at the same time. One important question to pose is how to develop the new data analysis and modeling methodologies that can be applied to the new set of complex datasets containing functional and tensorial data as instances moving away from the conventional data structures.

The objective of this dissertation is to study and address three major challenges typically occurring when working with complex multi-stream and functional sensing data. First, while the classical data analysis tools are designed to work with single-sensor data represented by matrices or vectors, sometimes the multisensory data are more efficiently represented by high-order arrays to preserve the original structure of the data. A dataset is said to have a multi-stream structure, meaning that it contains more than two informative dimensions. In Chapter 2 of this dissertation, a high-order-based monitoring method is suggested for monitoring a tensorial dataset of tool wear measurements. In addition to achieving high monitoring performance, the developed tensor-based chart is capable of providing correlation pattern analysis. This is useful in discerning assignable causes of unusual patterns of tool wear to enhance the process diagnosis ability.

The second challenge addressed in this dissertation is to deal with outlier observations that are inevitable when collecting sensor data. Chapter 3 develops a new robust decomposition method that can handle the outlier observations. The critical point in this study is that the proposed method is the first decomposition technique considering both correlated noise components and outlier observations. The correlated noise can be seen in time series, longitudinal and image data. The chapter starts with developing a robust low-order matrix decomposition method, and it proceeds to extending the concepts and mathematical formulas to a high-order tensor setting. The method is applied to a dataset with the purpose of surface defect monitoring using real billet images taken from a hot rolling process.

The third challenge studied in this dissertation is how to find a mathematical relationship between a quality response variable and some process variables that are not necessarily scalar or functional. Indeed, some of the predictors are in the form of tensors in addition to the regular functional or scaler predictors. The challenge is how to estimate the parameters of such general regression models. Chapter 4 focuses on developing a flexible yet parsimonious model for predicting scalar response variable utilizing some tensorial and functional predictors. The developed model is called functional linear regression with tensorial predictor (FLRTP). The advantage of this methodology compared to classical functional data analysis and linear regression methods is that it can handle both functional and tensorial predictors without performing vectorization on the tensorial predictors. This is helpful since the multi-stream structure of the predictor is preserved and the number of parameters to be estimated is kept at a reasonable amount. The performance of all methods is evaluated using simulation and real-world studies.

CHAPTER 1

INTRODUCTION

1.1 Motivation

The advent of advanced sensing and computing techniques creates a great opportunity of garnering a substantial amount of information during manufacturing system operations. On one hand, this is a magnificent breakthrough, for it provides practitioners with a more apposite viewpoint regarding the various aspects of manufacturing process operations and product quality. On the other hand, the analysis of such significant amount of complicated sensing data during continuous production has become a considerable challenge. Examples of such complex datasets include multi-stream sensing signals for tooling conditional monitoring, image data for non-contact product quality inspection, multi-stream sensing signals as the functional or tensorial predictors used for quality prediction in a manufacturing process.

Studying the *variation pattern* of process sensing signals or product quality measurements is tremendously helpful in providing essential insight regarding the unobtrusive root causes of process operational variability and poor product quality. That serves as an essential basis for manufacturing system monitoring and diagnosis to ensure product quality. Moreover, *process modeling* of the relationship between product quality and process variables can provide valuable insights about the newly established processes. This can help reduce process ramp-up time and ensure quality control at an early production stage.

Although there are several methodologies that are often utilized for variation pattern analysis and process modeling, most existing methods are not always or directly applicable for multi-stream sensor data. In fact, there are several critical issues that must be accounted for. The first issue is that the collected dada are usually introducible in more than one dimension represented by a multi-stream, high-order array data structure. In these cases, if a simple low-order (one-dimensional vector or two-dimensional matrix) representation is used to represent the dataset by stacking up the high dimensional data attributes, it would change the original cross-correlation structure among those higher dimensional attributes; thus leading to a loss of data cross-correlation information among different dimensions. Therefore, this dissertation will study a general method for variation patterns analysis, which can be applied to both a low-order matrix representation and a high order multi-stream data representation.

The second issue is that, in practice, the measurements of sensing signals are often mixed with complex noises that are not independently and identically distributed (i.i.d). These sorts of noise components are called *structured* noises with either columnwise or rowwise dependencies or both. Such structured noises are often observed in spatiotemporal measurements, time series data, network structured data, and image sensor data (Allen et al. 2014), etc. For instance, the background noise in a set of consecutive images captured in a short time interval has a spatiotemporal correlation structure. In most of the existing variation pattern analysis methods, the assumption of i.i.d. noise components is so crucial that if violated, the extracted features would not be appropriate to reflect the true signal characteristics; thus leading to an incorrect decision. Furthermore, there are always some *outlier* observations (samples) which are differently distributed from the anticipated distribution of the bulk of other normal data. In the most existing variation pattern analysis methods, if keep those outliers in the existing analysis methods, they would severely affect the variation pattern analysis results (Huber 1981). Therefore, this dissertation will study a new robust variation analysis method to be insensitive to outlier samples in the dataset while addressing the problem of structured noise simultaneously.

The third critical issue is how to find the relationship between a quality response variable and a set of unconventional predictors that contain process sensing signals that depend on a spatial, temporal or spatiotemporal index. The functional data analysis (FDA) methods have emerged to suggest a variety of approaches, like functional linear regression for modeling these variables when they play the role of a response variable or predictor. As it will be discussed later, in some applications, in addition to the functional predictors, there are some tensorial predictors (predictors in the format of high-order arrays). Treating these predictors as functional variables (or scalar variables) will not only increase the dimensionality of the problem but also break the high-order structure of the multi-stream data. Therefore, this dissertation will suggest a new modeling approach considering both tensorial and functional predictors to predict scalar response variables.

In summary, this dissertation aims to develop new methodologies that can be applied for variation pattern analysis and monitoring for multi-stream sensing data when they consist of outliers and structured noise. Furthermore, it will also develop a new approach for modeling a scalar quality response using a regression model consisting of complex predictors. The three major research topics to be studied in this dissertation are:

(1) to develop a new monitoring method using a high-order decomposition that can effectively model the multi-stream sensor data to achieve a superior monitoring and fault diagnosis performance compared to the other traditional control charts using a stack-up low-order decomposition.

(2) to develop generic data dimension reduction methods that are robust to the outliers and are capable of considering structured noise for both low-order and high-order representations. The advantage of the proposed methods can be seen in two folds: one is that there is no need for data cleaning process, and the other is having modeling flexibility without restrictive assumptions on the i.i.d. noise components and outlier-free data. The high-order decomposition method can be utilized for dimension reduction and feature extraction for multi-stream datasets. This is useful when applying a low-order decomposition method is ineffective due to the multi-stream structure of the data and high dimensionality.

(3) to develop a generic, parsimonious regression model for predicting scalar quality response variables with mixed types of predictors including functional, tensorial and scalar variables.

1.2 Overview of Dissertation

In this section, a brief overview of the dissertation research topics will be provided. The section is organized by giving specific manufacturing process examples to illustrate the research motivation, objectives and challenges. A brief discussion of the methodology to be employed for addressing the research challenges is provided.

1.2.1 Feature Extraction Using High-order Decomposition for Tool Wear Monitoring

Ultrasonic welding process is a rather new technology developed for performing high-quality welds when working with light and thin workpieces having different materials. For instance, the lithium-ion battery cells assembled in electric vehicles use ultrasonic welding to join copper and aluminum materials. The core factor in ultrasonic welding process is high-frequency vibration. Figure 1-1 schematically demonstrates the ultrasonic welding mechanism.

The thin metal sheets to be welded are held together by a clamping force perpendicular to the welding surface. The vibrational movements augment oscillating shear parallel to the workpieces. The material deformation is a result of the combination of clamping force and oscillating shears.

Ultrasonic metal welding machine typically consists of a controller, a transducer, a booster, a horn and an anvil. The horn and anvils are considered as the major tools consisting of pyramid-shaped teeth called knurls. The vibrations are transmitted to the workpieces using these knurls, and, hence, the wear of knurls over time is inevitable.

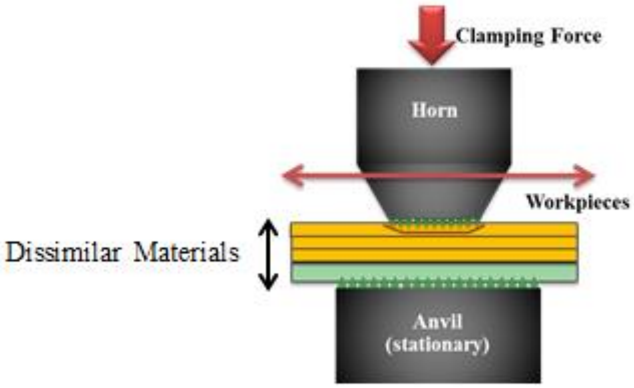


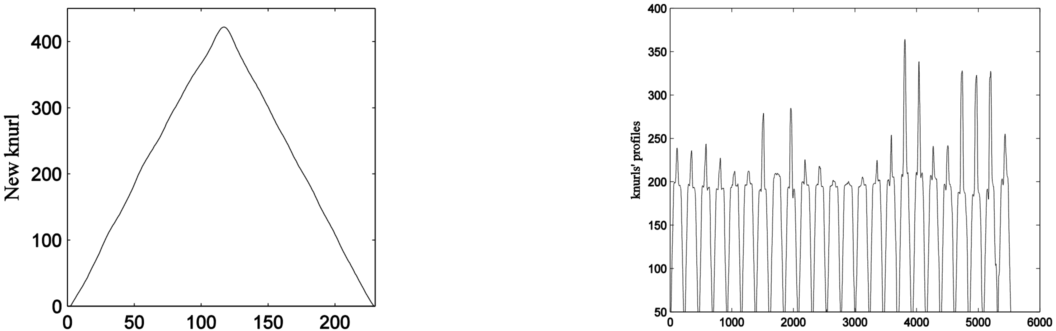
Figure 1-1: Ultrasonic welding machine (Shao et al. 2013)

The tool wear in ultrasonic welding has drastic effects on the quality of welds since the vibrations and the clamping force are not appropriately transmitted to the materials through the worn knurls. Defective welds can cause the malfunction of the whole battery pack (Shao et al., 2016). Therefore, an efficient monitoring system is needed to quickly detect the knurls' wear during the welding operation. Figure 1-2 (a) shows a set of normal knurls with healthy tips and complete body shape. In contrast, the knurls in Figure 1-2 (b) are worn completely.

The tool wear in ultrasonic welding can be monitored based on the changes in the profiles of the knurls' shape. The knurls profiles measurements are used for assessing tool wear. Figure 1-3 shows a cross-sectional view of the measurements regarding (a) a row of a new knurl and (b) a row of worn knurls corresponding to Figure 1-2 (a) and (b), respectively. The measurements are all made using a 3D laser scanning device.

(a) new knurl (b) worn knurl

Figure 1-2: Pyramid-shape knurls



(a) new knurl (b) a row of knurls' profiles

Figure 1-3: knurls' profiles measurements

Although there are several methods developed for tool wear monitoring, the process control for ultrasonic welding is more challenging due to the following three issues that will be addressed in Chapter 2 of this dissertation. The first challenge is how to systematically identify the unusual wear pattern of knurls in different rows, which can be related to the root causes of a malfunction in the welding operation. One effective way to analyze the tool wear pattern and the associated root causes is to study the cross correlation of knurls' profiles among different rows of the anvil. High correlation between two rows, for instance, means that these rows are worn in a similar way, which is high likely linked to underlying common root causes. To study the correlation structure, we can apply SVD method to the data and scrutinize the singular vectors, but the data must be represented by a matrix form. That means either the rows or columns of knurls of an anvil must be stacked up in a matrix. This will lead to losing the spatial correlation structure among different rows (columns) of the data; consequently, the data correlation structure is not fully captured.

The second challenge is how to ensure that the monitoring method is able to quickly detect tool wear characterized by knurls' profiles changes. Some of the methods in the literature, like fast Fourier transform (FFT) fail to detect such slight wear in the peaks of knurls particularly when not all knurls are worn in a row or a column of knurls in an anvil. To overcome these shortcomings, a high-order array is used to represent the dataset. The high-order SVD (HOSVD) method is then applied to the array to analyze the correlation structure of the rows. In the second step, a multivariate control chart (T^2 chart) is constructed based on the features extracted using the HOSVD method to monitor the changes in the knurls' profiles shapes.

A simulation study is done to evaluate the performance of the proposed method in both capturing the correlation structure of the data and monitoring performance in detecting the knurls' wear. The method is applied to a real tool wear dataset to illustrate its advantages over the existing methods lacking this diagnostic capability due to their inability of representing multi-stream data.

1.2.2 Robust Generalized SVD with Correlated Noise and Outliers

As discussed in Section 1.1, high dimensionality is an indispensable issue in analyzing and monitoring image data. One way to tackle high dimensionality is to reduce the data dimension by extracting a low-dimensional set of the relevant features for process monitoring. SVD forming the basis of PCA is an effective way to decompose a low-order array of data. The estimated singular vectors can project the original high dimensional dataset to a low-dimensional space which is easier to work with. The counterpart of SVD in the high-order settings is the

HOSVD method that is used for dimension reduction. Although SVD (HOSVD) is a nonparametric method in the sense that it does not require any distribution assumption for the observations, it imposes some assumptions about the noise components (random error terms) of the observations, that if violated, the results will be unreliable. This dissertation would firstly introduce the problem for low-order arrays and then extend it to a high-order formulation.

To specifically illustrate the point, the dataset is assumed to be represented by matrix $\mathbf{X} = \mathbf{M} + \mathbf{E}$, where $\mathbf{M} = \sum_{j=1}^{k} \mathbf{a}_j \mathbf{b}_j^T$ is the true signal matrix that must be recovered or estimated from $\mathbf{X}$, the jth left and right singular vectors are denoted by $\mathbf{a}_j$ and $\mathbf{b}_j$, respectively, and $\mathbf{E}$ is the noise matrix embedded in the true signal $\mathbf{M}$.

The SVD method tries to estimate the true matrix $\mathbf{X}$ by minimizing the L_2 norm of the residuals matrix $\mathbf{R} = \mathbf{X} - \mathbf{M}$. One key assumption in SVD is that the columns and rows of the $\mathbf{E}$ matrix must be independent; otherwise, the SVD method does not provide accurate estimates. For this reason, Allen et al. (2014) proposed the generalized SVD (GSVD) method replacing the L_2 norm with a weighted norm (similar to weighted least square problem) to account for structured noise matrix $\mathbf{E}$. The weighted residuals matrix $\mathbf{R}_w$ is given as $\mathbf{R}_w = \widetilde{\mathbf{Q}}^T \mathbf{R} \widetilde{\mathbf{C}}$ where $\mathbf{Q}$ and $\mathbf{C}$ are the row-wise and column-wise weights, respectively, $\mathbf{Q} = \widetilde{\mathbf{Q}} \widetilde{\mathbf{Q}}^T$ and $\mathbf{C} = \widetilde{\mathbf{C}} \widetilde{\mathbf{C}}^T$. The GSVD method, as a result, minimizes the L_2 norm of $\mathbf{R}_w$ matrix.

If a high-order method like HOSVD is used for feature extraction, similar to SVD, the problem will be to minimize the L_2 norm of the tensor of residuals $\mathcal{R} = \mathcal{X} - \mathcal{M}$, where $\mathcal{X}$ is the tensor representing the dataset and $\mathcal{M} = \mathbf{v}^{(1)} \circ \mathbf{v}^{(2)} \circ \ldots \circ \mathbf{v}^{(N)}$ is an approximation of $\mathcal{X}$. Using the L_2 norm in the objective function, HOSVD suffers the same drawbacks as SVD for structured noise. The generalized high-order SVD (GHOSVD) is developed (Allen et al. 2012) to consider the multi-steam correlation structure in the noise tensor. Moreover, similar to GSVD, the GHOSVD lacks the ability to handle outlier observations. This is true due to the fact that the objective function in the GHOSVD method is the L_2 norm of the weighted tensor of residuals, defined as $\|\mathcal{X} - \mathcal{M}\|_{\mathbf{C}_1,\mathbf{C}_2,\ldots,\mathbf{C}_N}$ with $\mathbf{C}_1, \mathbf{C}_2, \ldots, \mathbf{C}_N$ be the weights matrices for all N modes (dimensions) of data.

A potential problem affecting the performance of both SVD (HOSVD) and GSVD (GHOSVD) methods is the existence of outliers in the dataset, which is an inevitable problem in most practical cases. The outliers are some observations distributed differently from the majority of observations. The difference among outliers and regular observations is defined with regard to the context being studied. In the low-order decomposition context, if it is presumed that the jth left singular vector $\mathbf{a}_j$ is known, then the GSVD (SVD) problem changes to a linear regression problem with the right singular vector $\mathbf{b}_j$ (coefficients) to be estimated. From the linear regression literature, the outliers in the decomposition context can be defined as the observations lying distant from the linear relationship between the predictors ($\mathbf{a}_j$) and the response variable ($\mathbf{X}$). The large distance between the outliers and the fitted line is reflected by the residuals matrix $\mathbf{R}_w(\mathbf{R})$, and the L_2 norm overemphasizes this distance. The first right singular vector $\hat{\mathbf{b}}_1$, as a result, will be heavily pulled by the outliers. Since other singular vectors are perpendicular to the first one, they will all be affected likewise. The same problem also exists in the high-order setting.

One example of the aforementioned problem can be found in the image-based surface defect monitoring for hot rolling products. The semi-finished products, such as ingots, slabs, billets, etc., are intermediate workpieces requiring further operations to be final products. For example, billets are pieces of metal with a desired rounded shape as shown in Figure 1-4 (a). One critical quality characteristic in billets is to have a surface free of defects usually appearing in the form of cracks (openings) on the surface of billets. In the literature of image-based monitoring, PCA (SVD) is widely used for reducing the dimension and extracting the image features used for monitoring (Lin et al. 2008, Lu and Tsai 2005, Liu and MacGregor 2006, Yan et al. 2015). These methods are not applicable in some cases due to following three challenges. Firstly, in some datasets like images, it is reasonable to assume that the noise components (matrix **E**) have one-way (or two-way) dependencies due to the spatial correlation of the pixel intensities in a neighborhood.


(a) normal billet
(high quality image L_2 norm = 1235.8)

(b) outlier billet
(overexposed L_2 norm = 4386.9)

Figure 1-4: A normal and outlier billet image with associated $\mathbf{L_2}$ norms

Secondly, when cameras are used to capturing the images in harsh production conditions, sometimes the captured images are abnormal (overexposed/underexposed) as shown in Figure 1-4 (b). The abnormality of such images can be characterized by the changes in the mean vector and/or covariance matrices (row-wise and/or column-wise covariance matrices) of the noise matrix **E**. Hence, it is reasonable to consider these low-quality images with a poor visibility as outliers since they lie distant from the majority of normal images. This distance can be modeled by the difference among the mean and covariance matrix of the outlier noise components and those of the normal noise components.

Thirdly, the analysis sometimes has to be performed on a group of m collected samples of $n \times p$ images where $m \ll np$. When setting up an imaged-based monitoring system to monitor the products surface quality or the defects, the monitoring baseline of image features must be estimated using a group of collected images at the initial Phase I. For image feature extraction, a conventional technique is to vectorize all images to have a stack-up matrix with each row representing one image. The next step is to apply a decomposition method like SVD or GSVD to obtain the singular vectors and to compute the baseline features. The first drawback of using a stack-up low-order array is that it leads to a high dimensional matrix and the number of rows (m) will be less than the number of columns (np). In these cases, both GSVD and SVD methods will not be efficient. The second drawback is that when vectorizing an image, the spatial correlation of the pixels is broken, and as a result, the singular vectors will not accurately reflect the true correlation structure. In this dissertation, a high-order array is used to represent the data, and then, a new robust high-order decomposition method considering the structured noise is developed.

As discussed, the first two problems will dampen the performance of the SVD (GSVD) and any other least-square based model fitting procedure, like HOSVD. The effect of outliers can be seen in the L_2 norms computed for both the normal and outlier billet images in Figure 1-4 (a) and Figure 1-4 (b), respectively. The L_2 norm of the outlier image is clearly larger than that of the normal image, and this will unfavorably affect the singular vectors estimates. To overcome this shortcoming, the first part of Chapter 3 of this dissertation focuses on developing a *robust* generalized SVD method that is insensitive to the outliers; thus giving estimates for the main features. The word “robust” in this context means having a decomposition method not affected by the outliers in the dataset. To the best of our knowledge, the proposed RGSVD method is the first robust decomposition method that can account for outlier observations and structured noise at the same time. The regular decomposition methods are either sensitive to outliers or unable to model structured noise.

To account for the third problem, one naïve way is to rewrite the high-order objective function in a two-dimensional format over each mode, and then apply the RGSVD method on each of the N (number of modes in the data) sub-problems. A potential problem in this case is that the RGSVD method can handle the outliers when the whole sample (whole row in a matrix) is contaminated. The method crashes when some of the elements in a row are outliers, or when reformatting a high-order objective function, there will be elementwise outliers in some of the sub-problems. For this reason, the aforementioned approach will not be effective. The second part of the third chapter of this dissertation extends the RGSVD method to high-order setting. The robust GHOSVD method (RGHOSVD) is developed and the capabilities of the proposed method are demonstrated using a simulation study and a real dataset.

1.2.3 Functional Linear Regression with Tensorial Predictor

Finding the relationship between the quality of products with different associated process variables can be specifically helpful for process quality control. In practice, there are some situations where process variables cannot be considered as scalar variables; i.e., a variable is a function of time, space, etc. It is critical to consider the functional structure between this variable and its temporal (spatial) index.

In this dissertation, the friction stir blind riveting (FSBR) process is used as an example to illustrate the need for the proposed model. FSBR is relatively a new joining technology combining two conventional joining operations, namely, the friction stir riveting and blind riveting. A blind rivet is stirred inside the workpieces by a pre-specified feed rate, spindle speed and the configuration of the materials (the sequence of materials), which are called process setup variables. The penetration force which is a function of blind rivet location inside the workpieces is measured and recorded using a sensor. In addition, the temperature of the environment is captured using infrared cameras. The last two variables are not controllable and are called process sensing variables. At the end of the process, the mandrel is pulled up inside the workpieces to fasten them and the notch is broken at the end. The quality of the joined workpiece is tested using a standard tensile test where the maximum tolerated load right before the join is disassembled is used as the quality metric.

An important objective of modeling the FSBR process is to study how the product quality is affected by the variables involved in the process. To find a Mathematical model of the FSBR process is to regress the maximum tensile load on process set up variables (feed rate, spindle speed and configuration) and process sensing variables (penetration force and the environment's temperature obtained from infrared images).

In the above FSBR process, there are three types of predictors including the process setup variables as scalar predictors, the penetration force as a functional predictor, and the environment's temperature (infrared images) as tensorial predictors. As mentioned before, treating the tensorial variables as either a stack-up scalar or functional variables will lead to inaccurate estimates. Although in the literature there exist several papers considering the problem of regressing continuous or categorical response variables on tensorial predictors (please see Guo et al. 2012, Zhou et al. 2013, Li et al. 2013), none of the proposed models take into account both the functional and tensorial predictors. The existing algorithms, as a result, will not be able to handle the functional linear regression with a tensorial predictor.

Chapter 4 of this dissertation tries to develop a new model considering both tensorial and functional predictors. A new algorithm is also developed to compute the regression coefficients.

The performance of the developed method is evaluated using both a simulation study and a real-world dataset in the FSBR process.

1.3 Outline of Dissertation

In this dissertation, new methodologies are proposed for variation pattern analysis; data dimension reduction, feature extraction and process modeling. The features extracted using the proposed decomposition methods can be used for process monitoring. The developed process modeling approach that is developed can be used for predicting the product quality based on process setups and sensing signals. In each chapter, the performance of the proposed methods is evaluated through simulations and real case studies. The organization of the dissertation is shown in Figure 1-5.

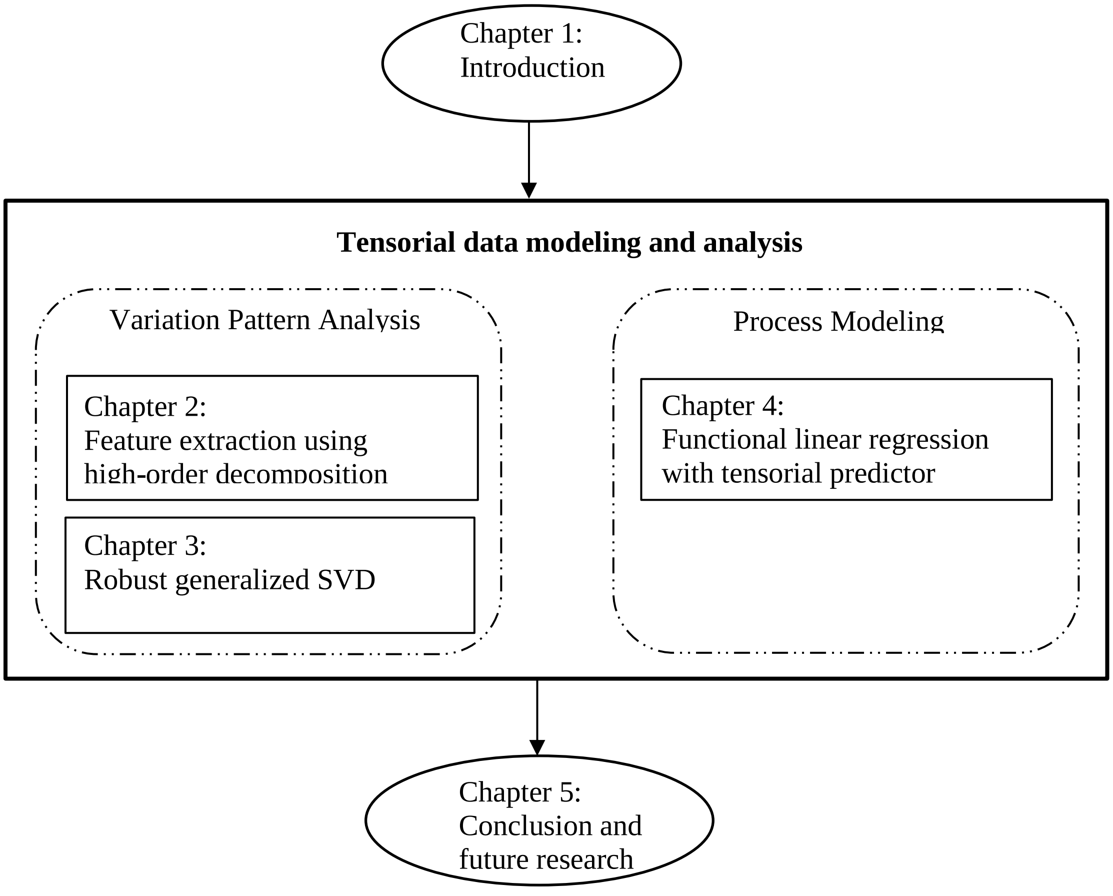


Figure 1-5: Outline of dissertation

References:

1. Allen, G. I., (2012), “Regularized Tensor Factorizations and Higher-Order Principal Components Analysis,” *Rice University Technical Report No. TR2012-01*, arXiv:1202.2476.

2. Allen, G. I., Grosenick, L., and Taylor, J., (2014), “A Generalized Least Squares Matrix Decomposition,” *Journal of the american statistical association*, *Theory & methods*, vol. 109, no. 505, pp. 145-159.

3. l, W., Kotsia, I., and Patras, I., (2012), “Tensor learning for regression. Image Processing,” *IEEE Transactions*, 21(2):816–827.

4. Li X., Zhou H., Li L., (2013), “Tucker tensor regression and neuroimaging analysis,” Available at arXiv:1304.5637.

5. Lin, H. –D., Chung, C. –Y., and Lin, W. –T., (2008), “Principal component analysis based on wavelet characteristics applied to automated surface defect inspection,” *WSEAS Transactions on Computer Research*, vol. 3, no. 4, pp. 193-202.

6. Liu, J. J., MacGregor, J. F., (2006), “Estimation and monitoring of product aesthetics: application to manufacturing of “engineered stone” countertops,“ *Machin vision and applications*, vol. 16, no. 6, pp. 374-383.

7. Ramsay, J. O. and Silverman, B. W., (2002), “Applied Functional Data Analysis,” Springer, New York. MR1910407.

8. Shao, C., Kim, T. H., Jin, J. J., Hu, S. J., Spicer, J. P., and Abell, J. A., (2016), “Tool wear monitoring for ultrasonic metal welding of lithium-ion batteries,” *ASME Journal of Manufacturing Science and Engineering*, 138(5), pp. 051005.

9. Shao, C., Paynabar, K., Kim, T. H., Jin, J. J., Hu, S. J., Spicer, J. P., ... & Abell, J. A., (2013), "Feature selection for manufacturing process monitoring using cross-validation," *Journal of Manufacturing Systems*, 32(4), pp. 550-555.

10. Huber, P. J., (1981), "Robust Statistics," John Wiley & Sons, New York.

11. Yan, H., Paynabar, K., and Shi, J., (2015), "Anomaly Detection in Images with Smooth Background **v**ia Smooth-Sparse Decomposition," *Technometrics*, DOI:10.1080/00401706.2015.1102764.

12. Zerehsaz, Y., Jin, J., "Robust Generalized Decompositions with Applications in Image-based Monitoring," *working paper*.

13. Zerehsaz, Y., Shao, C., Jin, J., (2016), "Tool Wear Monitoring in Ultrasonic Welding Using High-order Decomposition," *Journal of Intelligent Manufacturing*, DOI: 10.1007/s10845-016-1272-4.

14. 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.

CHAPTER 2

FEAUTURE EXTRACTION USING HIGH-ORDER DECOMPOSITION FOR TOOL WEAR MONITORING

2.1 Introduction

Ultrasonic welding has been well used for joining lithium-ion battery cells in electric vehicle manufacturing. Ultrasonic welding tool wear has a significant impact on the weld quality of lithium-ion batteries (Shao et al., 2016), because the failure of a single weld may lead to the malfunction of the entire battery pack. To ensure the weld quality, a commonly used preventive maintenance practice in most plants is to replace tools when the number of welding operations reaches a preset limit. Such limit is often set very conservatively, resulting in waste of useful tool life. Therefore, an accurate tool wear monitoring algorithm is critically needed to reduce the unnecessary maintenance cost induced by inappropriate tool replacements.

Tool wear monitoring has received tremendous attention over the last several decades. Most of papers have focused on machining, where some sensing signals, such as vibration, force, acoustic emission, and electric current are often collected and analyzed. A data transform, like fast Fourier transformation (FFT), wavelet decomposition, principal component analysis (Jolliffe 2005), and dominant feature identification (DFI) can be applied to extract relevant features for tool life prediction, classification or tool wear monitoring. For instance, Zhou et al. (2011) use the DFI method to extract the features from an acoustic emission signal. An autoregressive moving average (ARMA) model was then utilized for predicting tool wear in a ball-nose cutter of a high-speed milling machine. Shi and Gindy (2007) employed the PCA method on multiple sensor signals represented by a data matrix for the purpose of extracting important features related to tool wear. A least squares support vector machine (LS-SVM) method was further used to build a tool wear prediction model based on the extracted features. The methodology was applied to predict the wear in the teeth of a high-speed steel broaching tool. Li et al. (1999) applied a discrete wavelet transform to extract the features from acoustic emission and electric current signals. The extracted features were used for detecting the breakage of the tool in a drilling machine. A thorough review on different tool wear monitoring techniques can be found in Abellan and Subiron (2010).

Tool wear monitoring of ultrasonic welding is more challenging than that of machining processes, as the former tool wear mechanism is more complicated and has not been thoroughly understood. In ultrasonic welding, high-frequency energy is used to produce acoustic sound waves. The generated vibration produces oscillating shears between thin metal sheets held together under a clamping force perpendicular to the interface between the workpieces. The clamping force and the oscillation lead to deforming the material and forming the final joining. Because of the high frequency oscillation and workpiece material deformation, a small relative movement of the contacting surfaces between the tool (anvil in Figure 2-1) and the workpiece is inevitable. This interaction will lead to anvil wear during the repeated welding operations.

Figure 2-1 shows major components in an ultrasonic welding machine consisting of a controller, a transducer, a booster, a horn and an anvil. Both horn and anvil contain pyramid-shaped teeth called knurls. The tool wear in ultrasonic welding is characterized based on the profile change of knurls’ shape. Figure 2-2 (a) shows an optical image of knurls in an anvil. The highlighted area (4 rows) shows the major welding area that is underneath the horn pads. Figure 2-2 (b) gives the cross-sectional view of the knurl profiles at the highlighted area.

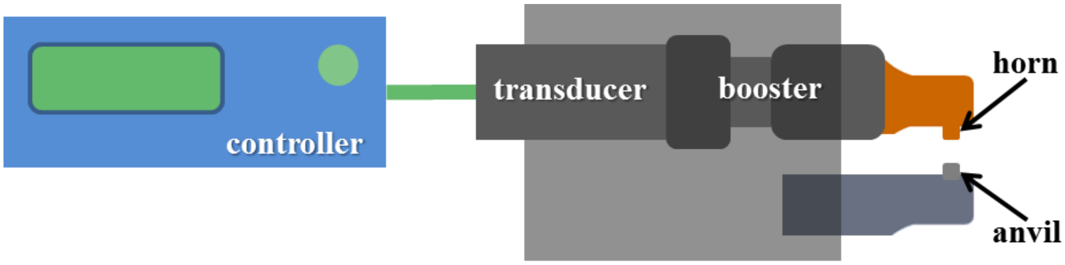
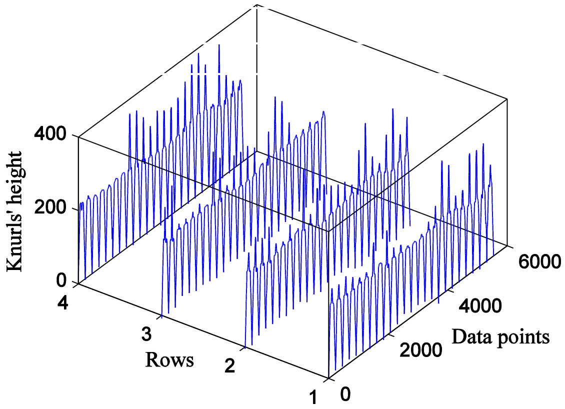


Figure 2-1: Ultrasonic welding system (Shao et al. 2013)

(a) Optical image of an anvil



(b) Cross-sectional view of knurls profiles at the highlighted 4 rows
Figure 2-2: Optical image of an anvil and knurl profiles

The first purpose of this chapter is to develop an effective monitoring strategy for early detection of tool wear based on the knurls profile measurements (e.g. Figure 2-2 (b)) in an ultrasonic welding process. The second aim of the chapter is to analyze tool wear variation pattern and to identify unusual patterns caused by the misalignment of the anvil with respect to the horn. This would help improve the tool setup to increase the tool life. In the remainder of this section, existing research papers related to tool wear monitoring for ultrasonic metal welding will be reviewed; moreover, the research challenges in using multidimensional data to early detect tool wear and unusual wear pattern will be summarized.

Some previous research was conducted to use indirect process sensing signals for the tool wear monitoring in the ultrasonic welding. Shao et al. (2014) studied the relationship between the online vibration signals and tool conditions for predicting the remaining lifetime of an anvil. They constructed a prediction model using the FFT method to extract the dominant frequency of vibration signals as the response variable and the number of welds performed by the anvil as the predictor. The remaining tool life was predicted based on the prediction of the dominant frequency feature exceeding a preset threshold. This approach has two major drawbacks: one problem is that the preset threshold is sensitive to the materials and operating conditions. The other drawback is that even if the indirect process sensing signal shows dramatic change when the knurls are worn, it is impossible to analyze the spatial cross-correlation among knurls on the anvil. In practice, the information of wear pattern in the anvil would help find the root causes of unusual tool wear at an early stage to prevent severe production loss.

Shao et al. (2016) used the FFT method to extract the monitoring features from the knurls profiles measurements. They subsequently built various classifiers on the extracted features to classify each anvil to one of four wear statuses predefined based on operating experience and engineering knowledge. Instead of their classification approach, another alternative way is to simply build a monitoring control chart based on the extracted FFT features (Kisic et al. 2015). However, some early slight wear appearing on the knurls' profile shoulders (Figure 2-3) or the knurls' peaks may not significantly affect the FFT features. Furthermore, if only one or a few knurls are worn in the same column of knurls, and the rest of knurls are normal, the FFT features would not significantly change. Therefore, the FFT features are not sensitive in detecting the early or local wear of an anvil.

In addition to the FFT features, two other features, which can be directly extracted from the knurl profile, were suggested by Shao et al. (2016) for classification. The first feature was the variability of the knurls' heights. A decreasing trend in the knurls' height variability indicates that the knurls start to wear off. The second feature was the average of the widths of the right and left shoulders as shown in Figure 2-3. The larger these shoulders are the more the knurls are worn. As it is shown in Subsection 2.3.2, using these simple features for tool wear monitoring is not always effective. This inefficacy reveals itself, for instance, when the peaks are slightly worn, or small wear occurs at only one side of a knurl (one shoulder). For the aforementioned reasons, an alternative method with the ability of detecting slight wear or asymmetric wear in the shape of knurls is needed (please see Subsection 2.3.2 for performance comparison).

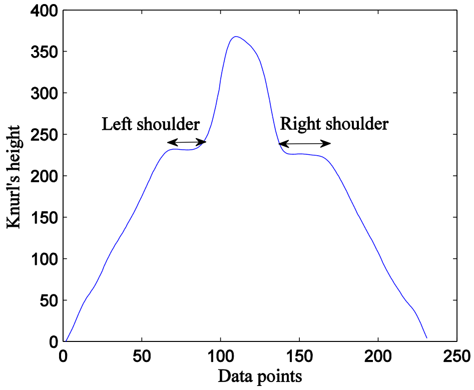
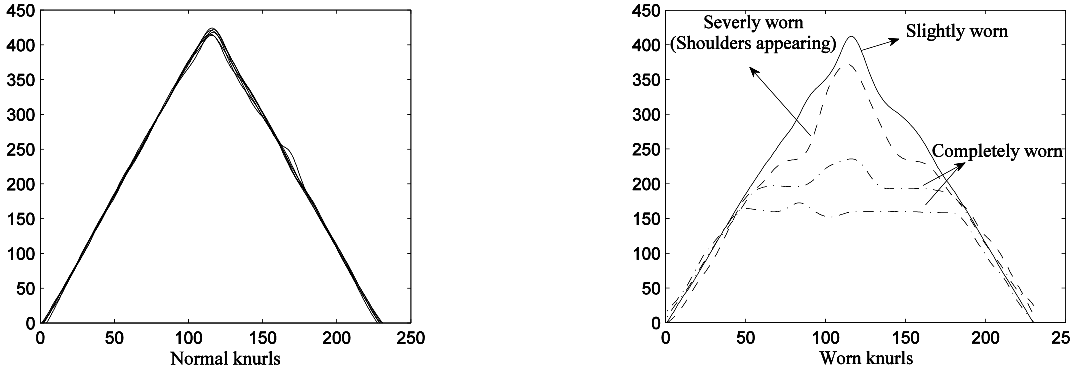


Figure 2-3: A knurl with shoulders appearing on both sides

There are two key challenges in establishing an appropriate methodology for monitoring and analysis of tool wear in an ultrasonic metal welding process: the monitoring method must be able to promptly detect slight wear in the shape of knurls. The other challenge is to have the ability to systematically analyze spatial cross-correlation among different rows of knurls profiles.

Wear pattern analysis is very useful in identifying the associated root causes. For example, as shown in Figure 2-2, if the first two rows have more wear than that of Row 3 and Row 4, this may indicate a misalignment of the anvil with respect to the horn. Early correction of such a tool misalignment problem will help save production loss.

To understand the shape difference between normal and worn knurls, a set of normal knurls profiles and worn knurls profiles are shown in Figure 2-4 (a) and Figure 2-4 (b), respectively. It can be clearly seen that normal knurls have a consistent profile shape change that is quite different from that of worn knurls. Therefore, the monitoring method can be set based on the change pattern (variation pattern) of the knurls profile shapes.



(a) Overlap of four healthy knurls on an anvil (b) Overlap of four worn out knurls on an anvil

Figure 2-4: Illustration of normal and worn knurl profiles

PCA using singular value decomposition is one possible solution to analyze the multivariate variation pattern. In order to use PCA, the dataset must be represented by a matrix. For the tool wear analysis, the knurls’ profiles measured at one specific sampling time on an anvil should be sequentially stacked up and represented by a row vector. Hence, different rows of the matrix represent different anvil samples. As discussed before, the spatial correlation among the knurls at different rows of an anvil should be considered in the analysis. However, stacking up all knurls’ profiles will destroy such a spatial cross-correlation structure. To overcome this shortcoming, a high-order array called tensor is suggested in the chapter to represent the multidimensional structure of tool wear data. Specifically, the ultrasonic tool wear dataset is represented by a three-way tensor including three dimensions defined as follows: the first dimension is the positional row index of knurls, the second dimension is the positional column index of knurls, and third dimension is the data point index of each knurl profile. Using such a multidimensional representation, the spatial correlation structure among different rows is clearly preserved. Analysis of this spatial correlation pattern can facilitate process fault diagnosis. For instance, if the knurls profiles wear in the first and second rows are positively correlated, it means that the knurls in these two rows are getting worn with a similar pattern. On the other hand, a negative correlation between two rows indicates that one row is getting worn faster than the other row. This unusual wear pattern is an indicator for the existence of a tool installation problem.

Several authors have favored the use of a multidimensional array (tensor) representation when the dataset has more than two dimensions (He et al. 2005; Paynabar et al. 2013; Yan et al. 2015). In this chapter, a high-order decomposition method called high-order singular value decomposition (HOSVD) is used to factorize the tensor representing the data and extract effective monitoring features afterwards. A multivariate monitoring chart will be constructed based on those extracted features.

To sum up, the method suggested in this chapter aims at systematically analyzing the variation pattern of knurls profiles wear and extracting effective monitoring features for tool wear monitoring and root cause inference. In order to capture the spatial cross-correlation structure of the knurls among different rows, a high-order array is employed to represent the knurls' profile measurements. For variation pattern analysis of such a tensorial dataset, the

HOSVD method is employed. The HOSVD method can systematically check whether spatial correlation exists among rows of knurls in the anvil (please see Subsection 2.3.1). Furthermore, a T^2 control chart is constructed to monitor the extracted features based on the HOSVD results. This will be discussed in detail in Subsection 2.2.2.

The reminder of this chapter is organized as follows: Section 2.2 provides a brief review of different strategies including a high-order array for representing the data; additionally, the HOSVD method and the multivariate T^2 control chart constructed based on the extracted monitoring features are discussed. A simulation study is conducted in Section 2.3 to show the superiority of the HOSVD method to the PCA and FFT methods in two aspects, namely, (1) effectively capturing the true variation pattern of the data, and (2) accurately detecting slight/early wear in the knurls. A case study is also performed in Section 2.4 to demonstrate the effectiveness of the proposed method.

2.2 A Brief Review of Tensor Data Representation and the HOSVD-based T^2 Control Chart

In this section, firstly, different data representation strategies are discussed. The basic notation relating to the tensor representation used throughout the chapter is introduced in Subsection 2.2.1 followed by some useful multilinear algebraic operations. Secondly, the HOSVD method and the T^2 control chart constructed upon the extracted features from HOSVD are elaborated in Subsection 2.2.2.

2.2.1 Data Representation and Basic High-order Algebraic Operations

Table 2-1 lists all the symbols used in this chapter. Two fairly different strategies are used to represent the tool wear data. A simple matrix representation called low-order representation is a classical way to represent a dataset. A matrix $\mathbf{X} \in \mathbb{R}^{24\times 924}$ is used to represent the knurls' profiles shown in Figure 2-2. In the following analyses, this is called Representation I. The number of rows in this matrix indicates that each anvil has 24 columns of knurls, and the number of columns ($4 \times 231 = 924$) represents the stacked-up rows of knurls' profiles with each knurl profile having 231 data points. As discussed before, the spatial cross-correlation among different rows of knurls on an anvil breaks down in Representation I. Alternatively, a different low-order representation $\mathbf{Y} \in \mathbb{R}^{96\times 231}$ denoted as Representation II can also be used, where each knurl profile is considered as a row in matrix **Y**, and 96 ($24 \times 4 = 96$) rows of matrix **Y** correspond to stacked-up rows of 24 columns of knurls. In this way, the cross-correlations among different rows and different columns are mixed together and cannot be distinguished.

Table 2-1: List of symbols

x	Scalers
$\mathbf{x}$	Vectors
$\mathbf{X}$	Matrices
$\mathbf{X}^{(n)}$	the nth matrix in a group of matrices
$\boldsymbol{\mathcal{X}}^{I_1\times I_2\times\ldots\times I_N}$	An N-dimensional (Nth-order) tensor
I_n	the number of elements in the nth order
$x_{i_1 i_2 \ldots i_n}$	$i_1 i_2 \ldots i_n$th element in tensor $\boldsymbol{\mathcal{X}}^{I_1\times I_2\times\ldots\times I_N}$

The tool wear dataset shown in Figure 2-1 (b) can be, furthermore, represented by a 3rd-order tensor $\boldsymbol{\mathcal{X}}^{4\times 231\times 24}$ with 4 rows as Mode 1, 231 data points of each knurl profile as Mode 2, and 24 columns of knurls on an anvil as Mode 3. Some commonly used multilinear algebraic operations for tensors are introduced as follows:

(1) *Tensor matricization*: It is used to transform a tensor into a matrix. Specifically, for mode-n matricization, each row vector of the new matrix is obtained by stacking up the elements of all other modes, and the number of rows is equal to I_n, i.e., the resultant matrix $\mathbf{X}^{(n)}$ has the dimension of $I_n \times I_1 I_2 \ldots I_{n-1} I_{n+1} \ldots I_N$. That is, when matricizing a tensor, the $i_1 i_2 \ldots i_N$th

element in the tensor $\boldsymbol{\mathcal{X}}^{I_1 \times I_2 \times \ldots \times I_N}$ is mapped to element (i_n, j) of matrix $\mathbf{X}^{(n)}$, where $j = 1 + \sum_{\substack{k=1 \\ k \neq n}}^{N} (i_k - 1) J_k$ and $J_k = \sum_{\substack{m=1 \\ m \neq n}}^{k-1} I_m$.

(2) *Tensor vectorization*: Tensor vectorization is simply the process of rearranging the tensor $\boldsymbol{\mathcal{X}}^{I_1 \times I_2 \times \ldots \times I_N}$ to produce a vector $\mathbf{v}$ with size $I_1 I_2 \ldots I_N$.

(3) *Tensor-matrix product*: It generally yields a new tensor. Specifically, mode-n product of a tensor $\boldsymbol{\mathcal{X}}^{I_1 \times I_2 \times \ldots \times I_N}$ by a matrix $\mathbf{U} \in \mathbb{R}^{J \times I_n}$ is denoted by $\boldsymbol{\mathcal{Y}} = \boldsymbol{\mathcal{X}} \times_n \mathbf{U} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_{n-1} \times J \times I_{n+1} \times \ldots \times I_N}$. Each element of the resulting new tensor is obtained as

$$\mathcal{Y}_{i_1 i_2 \ldots i_{n-1} j i_{n+1} \ldots i_N} = \sum_{i_n=1}^{I_n} x_{i_1 i_2 \ldots i_N} u_{j i_n} \qquad \text{(2-1)}$$

where $u_{j i_n}$ is element (j, i_n) in matrix $\mathbf{U}$.

(4) *Tensor-vector product*: It is a special case of tensor-matrix product. Specifically, the mode-n product of a tensor $\boldsymbol{\mathcal{X}}^{I_1 \times I_2 \times \ldots \times I_N}$ by a vector $\mathbf{v} \in \mathrm{R}^{I_n}$ is defined as $(\boldsymbol{\mathcal{X}} \times_n \mathbf{v})_{i_1 i_2 \ldots i_{n-1} i_{n+1} \ldots i_N} = \sum_{i_n}^{I_n} x_{i_1 i_2 \ldots i_N} v_{i_n}$, where v_{i_n} is the i_nth element of vector $\mathbf{v}$. This mode-n tensor-vector product changes the tensor order from N into $N - 1$. For a comprehensive review regarding the multilinear algebra and high-order decomposition methods, the interested readers can refer to Kolda and Bader (2009).

2.2.2 HOSVD Method and HOSVD-based T^2 Control Chart Construction

The general idea in HOSVD known as Tucker decomposition is to factorize a tensor $\boldsymbol{\mathcal{X}}^{I_1 \times I_2 \times \ldots I_N}$ into a core tensor denoted by $\boldsymbol{\Psi}^{k_1 \times k_2 \times \ldots \times k_N}$ using a set of factor matrices $\mathbf{V}^{(n)} \in \mathbb{R}^{I_n \times k_n}$; n = 1, 2,…, N, where $k_n < I_n$ is the number of components in mode n. In the context of ultrasonic welding dataset, a tensor $\boldsymbol{\mathcal{X}}^{4 \times 231 \times 24}$ is used to represent the dataset. The factor matrices $\mathbf{V}^{(1)} \in \mathbb{R}^{4 \times k_1}$, $\mathbf{V}^{(2)} \in \mathbb{R}^{231 \times k_2}$ and $\mathbf{V}^{(3)} \in \mathbb{R}^{24 \times k_3}$ are the matrices of singular vectors for Mode 1, Mode 2 and Mode 3, respectively. Note that the HOSVD method enables the practitioners to choose different number of components k_n for each mode n, and the number of components can be determined based on a predefined threshold of the explained variability. To obtain the optimal orthogonal singular vectors for HOSVD, the objective function defined in Eq. (2-2) is to minimize the L_2 norm of the residuals which are the difference between the original tensor $\boldsymbol{\mathcal{X}}$ and the approximated tensor $\widehat{\boldsymbol{\mathcal{X}}} = \boldsymbol{\Psi} \times_1 \mathbf{V}^{(1)} \times_2 \mathbf{V}^{(2)} \times_3 \dots \times_N \mathbf{V}^{(N)}$ (De Lathauwer et al. 2000); i.e.,

$$\text{Minimize } \left\| \boldsymbol{\mathcal{X}} - \boldsymbol{\Psi} \times_1 \mathbf{V}^{(1)} \times_2 \mathbf{V}^{(2)} \times_3 \dots \times_N \mathbf{V}^{(N)} \right\|_2^2, \text{ or equivalently}$$

$$\text{Maximize } \boldsymbol{\Psi} = \boldsymbol{\mathcal{X}} \times_1 {\mathbf{V}^{(1)}}^T \times_2 {\mathbf{V}^{(2)}}^T \times_3 \dots \times_N {\mathbf{V}^{(N)}}^T$$

$$\text{s.t: } {\mathbf{V}^{(n)}}^T \mathbf{V}^{(n)} = \mathbf{I};\ n = 1, 2, \dots, N, \qquad (2\text{-}2)$$

Unlike the regular low-order SVD, there is no closed-form solution for the high-order decomposition problem in Eq. (2-2). In practice, the problem is solved iteratively by fixing all factor matrices but one factor matrix in mode n; $n = 1, 2, \dots, N$, and obtaining the solution for mode-*n's* factor matrix. This procedure can be repeated for all other factor matrices in different modes. Figure 2-5 gives the high-order orthogonal iterations (HOOI) algorithm proposed by De Lathauwer et al. (2000).

```
Initialize $\mathbf{V}^{(n)} \in \mathbb{R}^{I_n \times k_n}$, for n =1, 2, ..., N
Repeat until convergence or maximum iteration reached
    For n = 1, 2, ..., N do
    $\boldsymbol{\mathcal{G}} = \boldsymbol{\mathcal{X}} \times_1 {\mathbf{V}^{(1)}}^T \dots \times_{n-1} {\mathbf{V}^{(n-1)}}^T \times_{n+1} {\mathbf{V}^{(n+1)}}^T \dots \times_N {\mathbf{V}^{(N)}}^T \in \mathbb{R}^{k_1 \times k_2 \times \dots k_{n-1} \times I_n \times k_{n+1} \times \dots k_N}$
    matricize tensor $\boldsymbol{\mathcal{G}}$, apply SVD, and choose $\mathbf{V}^{(n)}$ to be the left leading $k_n$ singular vectors
    End for
    $\boldsymbol{\Psi} = \boldsymbol{\mathcal{X}} \times_1 {\mathbf{V}^{(1)}}^T \times_2 {\mathbf{V}^{(2)}}^T \dots \times_N {\mathbf{V}^{(N)}}^T$
Convergence criterion: in each iteration k, stop if for all n =1, 2, ..., N, $\left\| \mathbf{V}_k^{(n)} - \mathbf{V}_{k-1}^{(n)} \right\|_2 < \varepsilon$ ,
where $\varepsilon$ is a small value
Return $\boldsymbol{\Psi}, \mathbf{V}^{(1)}, \mathbf{V}^{(2)}, \dots, \mathbf{V}^{(N)}$
End procedure
```

Figure 2-5: High-order orthogonal iteration algorithm (De Lathauwer et al. 2000)

After decomposing the original tensor of data $\boldsymbol{\mathcal{X}}^{4\times231\times24}$ ($I_1 = 4$, $I_2 = 231$, and $I_3 = 24$), the monitoring features can be obtained by projecting the centralized tensor $\boldsymbol{\mathcal{X}}^{4\times231\times24}$ using the knurls profiles factor matrix $\mathbf{V}^{(2)} \in \mathbb{R}^{231\times k_2}$; that is, the tensor of scores is obtained as

$$\boldsymbol{\mathcal{Z}} = \boldsymbol{\mathcal{X}} \times_2 \mathbf{V}^{(2)^T} \tag{2-3}$$

where $\boldsymbol{\mathcal{Z}} \in \mathbb{R}^{4\times k_2\times24}$ is the tensor of scores. The number of components $k_n; n = 1, 2, 3$, can be determined based on the explained variance defined as $\frac{\|\boldsymbol{\Psi}\|_2^2}{\|\boldsymbol{\mathcal{X}}\|_2^2}$, where $\boldsymbol{\Psi} = \boldsymbol{\mathcal{X}} \times_1 \mathbf{V}^{(1)^T} \times_2 \mathbf{V}^{(2)^T} \ldots \times_N \mathbf{V}^{(N)^T}$ is the core tensor, $\boldsymbol{\mathcal{X}}$ is the original tensor representing the data, $\|\boldsymbol{\mathcal{X}}\|_2^2$ is the squared Frobenious norm of tensor $\boldsymbol{\mathcal{X}}$, and it is defined as $\|\boldsymbol{\mathcal{X}}\|_2^2 = \sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \ldots \sum_{i_N=1}^{I_N} x_{i_1 i_2 \ldots i_N}^2$. Tensor $\boldsymbol{\mathcal{Z}}$ can be transformed into a matrix $\mathbf{Z} \in \mathbb{R}^{24\times4k_2}$ via tensor matricization over Mode 3. Rows of matrix $\mathbf{Z}$ are considered as samples used to estimate the mean vector and covariance matrix of the extracted features.

As discussed, the construction of the HOSVD-based T^2 control chart involves two steps. (1) Using the algorithm in Figure 2-5 to estimate the matrix of singular vectors (factor matrices) based on the data from normal anvils. The features are computed as in Eq. (2-3), and a $24 \times 4k_2$ matrix of features is obtained. (2) An appropriate control chart is constructed for monitoring the extracted features. Since there is usually more than one feature, a multivariate T^2 control chart is set up to monitor these $p = 4k_2$ extracted features. The details will be shown in the case study provided in Section 2.5. Let $\mathbf{z}_i \in \mathbb{R}^{4k_2\times1}$ denote the *i*th row of matrix $\mathbf{Z}$, then the T_i^2 statistic can be computed as

$$\boldsymbol{T}_i^2 = (\mathbf{z}_i - \bar{\mathbf{z}})^T \mathbf{S}^{-1} (\mathbf{z}_i - \bar{\mathbf{z}}); i = 1, 2, \ldots, I_3 = 24, \tag{2-4}$$

where the jth element of mean vector $\bar{\mathbf{z}}$ is $\frac{\sum_{i=1}^{24} z_{ij}}{24}$, z_{ij} is the jth element of the vector $\mathbf{z}_i$ for $j = 1,2, \ldots, p = 4k_2$, and $\mathbf{S} \in \mathbb{R}^{p\times p}$ is the sample covariance matrix of features estimated using in-control 24 samples represented by matrix $\mathbf{Z}$ (Hotelling 1947). Provided that the features follow a normal distribution, the upper control limit is computed as $\mathrm{UCL} = \frac{p(\,I_3+1)(I_3-1)}{I_3(I_3-p)} f_{\alpha,p,I_3-p}$ in Phase II and $\mathrm{UCL} = \frac{(I_3-1)^2}{I_3} \beta_{\alpha, p/2, (I_3-p-1)/2}$ for Phase I, where f_{α,ν_1,ν_2} is the $(1-\alpha)100^{\text{th}}$ percentile of an F distribution with ν_1 and ν_2 degrees of freedom, and $\beta_{\alpha,a,b}$ is the $(1-\alpha)100^{\text{th}}$ percentile of a Beta distribution with parameters a and b (Tracy, Young, and Mason 1992). In case the features do not follow a normal distribution, the upper control limit can be obtained using the empirical distribution of T_i^2's. When the PCA transform is used for a low-order data representation, a similar procedure is followed for constructing a T^2 chart using the transformed PC features.

2.3 Performance Comparison of the HOSVD, PCA and FFT Methods

In general, there are two key aspects to be considered when analyzing tool wear data in an ultrasonic welding process. The first important issue is studying the correlation structure of the knurls across different rows of an anvil. This is helpful in understanding the knurls' wear pattern on an anvil. For this purpose, Subsection 2.3.1 shows the superiority of HOSVD to PCA in capturing the true correlation structure of the data. The second favorable aspect is that the monitoring method can effectively detect worn knurls at an early wear level. Subsection 2.3.2 provides a performance comparison among four control charts established based on the HOSVD, PCA, FFT and the knurls' height methods.

2.3.1 Comparison of HOSVD and PCA in Capturing the Correlation Structure

In this subsection, a surrogated simulation study is conducted to compare the performance of HOSVD and PCA in explaining the spatial cross-correlation among the rows of knurls on an anvil. Since FFT does not provide any information regarding the variation pattern of the data, it is excluded from this comparison. A set of tensorial data represented by tensor

$\boldsymbol{\mathcal{X}} \in \mathbb{R}^{4\times231\times24}$ is generated to represent an anvil having 4 rows of knurls with each row containing 24 columns of knurls. Each knurl is represented by a profile having 231 data points. The elements of tensor $\boldsymbol{\mathcal{X}}$ are denoted by x_{ijk} corresponding to the jth point of a knurl profile located on Row i and Column k on an anvil. The simulation condition is set such that Row 1 and Row 2 have the same high variance. Row 3 and Row 4 have equal low variances. Furthermore, Row 1 and Row 2 are positively correlated while there is a negative correlation between Row 3 and Row 4. These parameters are specifically represented by

$$\sigma_{R_1}^2 = \sigma_{R_2}^2 > \sigma_{R_3}^2 = \sigma_{R_4}^2,\ \rho_{R_1R_2} > 0 \text{ and } \rho_{R_3R_4} < 0 \qquad \text{(2-5)}$$

where $\sigma_{R_i}^2$ is the variance of Row i, and $\rho_{R_iR_j}$is the correlation between Row i and Row j. In this simulation, these parameters are specified as $\sigma_{R_1}^2 = \sigma_{R_2}^2 = 100 > \sigma_{R_3}^2 = \sigma_{R_4}^2 = 60$, $\rho_{R_1R_2} = \rho$, $\rho_{R_3R_4} = -\rho$, and $\rho = 0.5$. In order to simulate the knurls profiles, a mixed-effect model is used, which is defined as

$$\mathbf{y}_{\boldsymbol{k}}^{(\boldsymbol{i})} = \mathbf{B}^{(\boldsymbol{i})}\left(\boldsymbol{\beta}^{(\boldsymbol{i})} + \mathbf{r}_{\boldsymbol{k}}^{(\boldsymbol{i})}\right) + \boldsymbol{\varepsilon}_{\boldsymbol{k}}^{(\boldsymbol{i})} \qquad \boldsymbol{k} = \mathbf{1}, \mathbf{2}, \dots, \mathbf{24};\ \boldsymbol{i} = \mathbf{1}, \mathbf{2}, \mathbf{3}, \mathbf{4} \qquad \text{(2-6)}$$

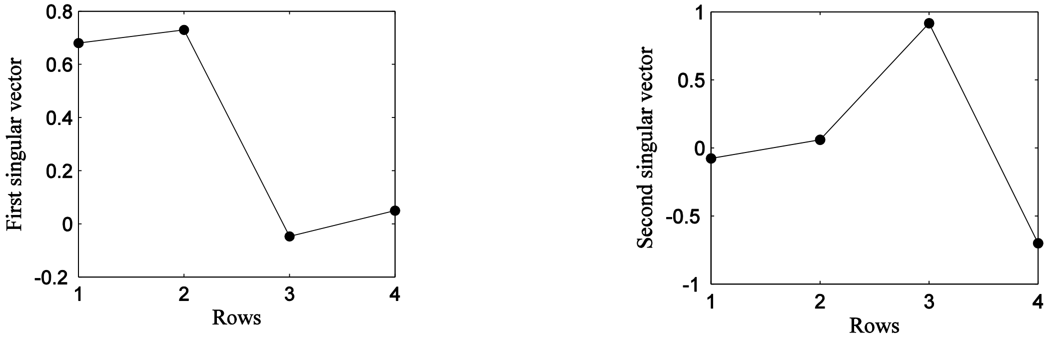
where $\mathbf{y}_k^{(i)}$ is the 231×1 vector of the kth knurl profile in Row i, $\mathbf{B}^{(i)}$ is the 231× L matrix of B-spline basis values with L knots. The vector of fixed-effect coefficients is denoted by $\boldsymbol{\beta}^{(i)} \in \mathbb{R}^{L\times1}$, and it is computed as follows: a set of normal knurls is selected from the dataset, and each knurl profile is regressed on B-spline basis values stored in matrix $\mathbf{B}^{(i)}$. The coefficients are computed using the least-square method. The mean vector of the computed coefficients is considered as the vector of fixed-effect B-spline coefficients $\boldsymbol{\beta}^{(i)}$. $\mathbf{r}_k^{(i)} \in \mathbb{R}^{L\times1}$ is the vector of random-effect coefficients that are normally distributed with a zero mean vector and the covariance matrix $\mathbf{U} \in \mathrm{R}^{L\times L}$ with elements given in Eq. (2-5). The vector of random errors $\boldsymbol{\varepsilon}_k^{(i)} \in \mathbb{R}^{231\times1}$ is assumed to follow a normal distribution with a zero mean vector and the diagonal covariance matrix $\sigma_\varepsilon^2\mathbf{I}$. In this chapter, the random errors' variability is $\sigma_\varepsilon^2 = 0.1$ and $\mathbf{I} \in \mathbb{R}^{231\times231}$ is an identity matrix. An important property of the mixed-effect model in Eq. (2-6) is that it has a high flexibility in modeling the cross-correlation among different rows of an anvil.

Both the HOSVD and PCA methods are applied to the simulated data. Table 2-2 gives the percentage of explained variance by the first three components of each method. PCA I and PCA II denote the PCA method applied to Representation I ($\mathbf{X} \in \mathbb{R}^{24\times924}$) and Representation II ($\mathbf{Y} \in \mathbb{R}^{96\times231}$) defined in Subsection 2.2.1, respectively.

Table 2-2: Percentage of explained variance by the components of HOSVD and PCA methods

	Method		
Number of components	HOSVD (%)	PCA I (%)	PCA II (%)
1	30.22	27.33	40.33
2	24.66	23.40	28.3
3	12.05	15.61	21.01
Total explained variability	66.93	66.34	89.64

Figure 2-6 (a) shows the first singular vector of HOSVD method for Mode 1, where Row 1 and Row 2 have the largest weights among all 4 rows. This result is expected because the first two rows are set to have the highest variability in the simulation. Moreover, the simulated positive correlation between Row 1 and Row 2 is well captured by this singular vector. The second singular vector computed by HOSVD is plotted in Figure 2-6 (b). It can be observed that the HOSVD method gives almost zero weights to Row 1 and Row 2, while Row 3 and Row 4 have a large weight with an opposite sign reflecting the negative correlation between Row 3 and Row 4.



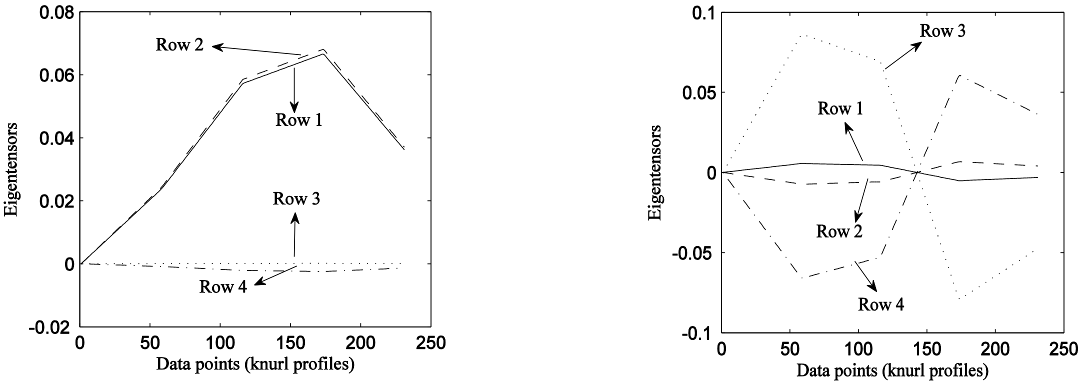
(a) First Mode-1 singular vector

(b) Second Mode-1 singular vector

Figure 2-6: First and second singular vectors computed by HOSVD method

Figure 2-7 uses the eigentensors of HOSVD to illustrate the variation pattern in Mode 1 and Mode 2 simultaneously, where the eigentensor for the lth component is calculated as $\mathbf{v}^{(1)} \circ \mathbf{v}^{(2)}$ with $\mathbf{v}^{(1)} \in \mathbb{R}^{4\times 1}$ and $\mathbf{v}^{(2)} \in \mathbb{R}^{231\times 1}$ corresponding to the lth columns of factor matrices $\mathbf{V}^{(1)}$and $\mathbf{V}^{(2)}$, respectively, and " $\circ$ " denoting the outer product between two vectors. As expected, in the first eigentensor (Figure 2-7), Row 1 and Row 2 have the highest weights with a positive correlation, while in the second eigentensor shown in Figure 2-7 (b), Row 3 and Row 4 are highlighted with a negative correlation.

For comparison, the PCA method is further applied to the simulated dataset with Representation I ($\mathbf{X} \in \mathbb{R}^{24\times 924}$) and Representation II ($\mathbf{Y} \in \mathbb{R}^{96\times 231}$). The first and second eigenvectors estimated by PCA I are plotted in Figure 2-8 (a) and Figure 2-8 (b), respectively.

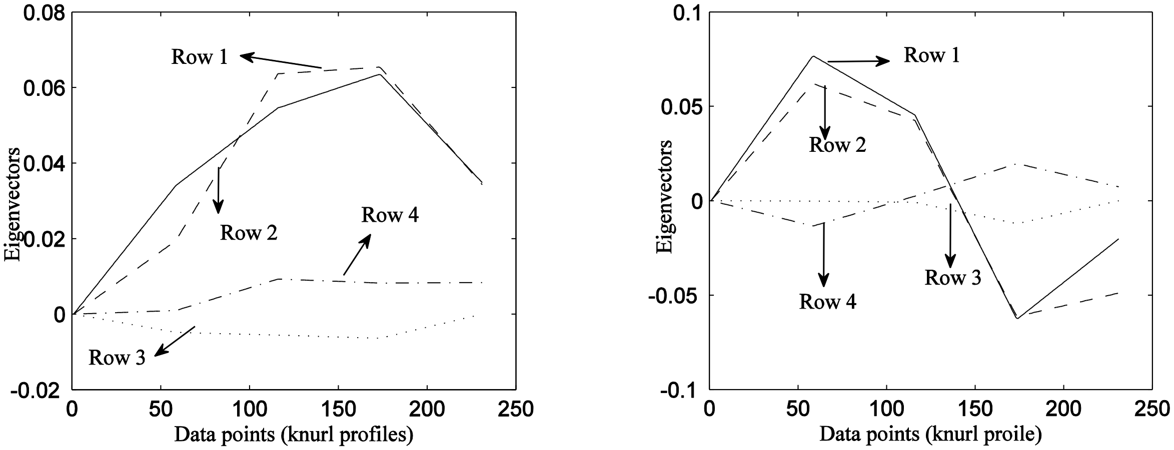


(a) First eigentensor

(b) Second eigentesor

Figure 2-7: Eigentesors estimated by the HOSVD method

Different from the HOSVD's results in Figure 2-7, Figure 2-8 shows that both eigenvectors of PCA I reflect the large variability of Row 1 and Row 2, and the negative correlation between Row 3 and Row 4 is not obviously reflected by the second eigenvector. This shows that PCA cannot fully capture the correlation of Rows 3 and 4.



(a) First eigenvector obtained using PCA (b) Second eigenvector obtained using PCA

Figure 2-8: First and second eigenvectors estimated by PCA I

Figure 2-9 shows the first and second eigenvectors of PCA II results. As expected, PCA II cannot provide any information regarding the correlation structure of the rows due to the way the data are represented.

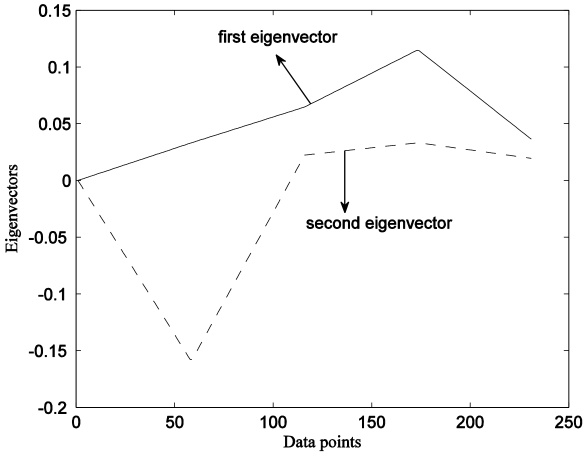
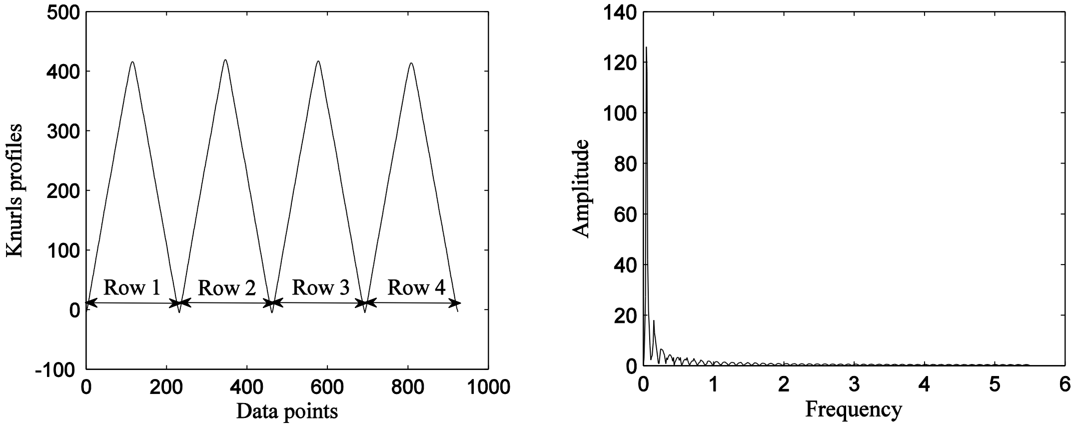


Figure 2-9: First and second eigenvectors estimated by PCA II

The above comparison suggests that the HOSVD method has been able to more effectively capture the true spatial cross-correlation structure among the rows of knurls on an anvil. In practice, this correlation structure is sensitive to the relative position of the anvil to the workpiece/horn in the ultrasonic welding machine, and it can reflect the fact whether the anvil is properly set up or not. Therefore, the HOSVD method is suggested in this chapter to monitor the tool wear with a diagnostic capability of identifying the anvil's misalignment problem.

2.3.2 Monitoring Performance Comparison

A 3D microscope was utilized to measure the tool surfaces, and the knurl heights were extracted from the surface measurements. In this subsection, four different methods for monitoring the tool wear data are compared: (1) the HOSVD-based T^2 control chart, (2) PCA-based control chart, (3) FFT-based control chart, and (4) a control chart to monitor the knurls' height. The FFT method is applied to each column containing four knurls on an anvil in order to extract the frequency-domain features for representing the repeated pattern. Frequency-domain features are the amplitudes corresponding to the dominant frequencies obtained from FFT method applied to each column of knurls profiles. Figure 2-10 (a) shows one column of 4 knurls profiles, and the frequency-domain profile is plotted in Figure 2-10 (b) with sampling frequency set to 5.5. For the fourth method, to calculate the height of the knurls profiles, firstly, a general baseline is determined as the lowest point of knurls profiles on an anvil. Secondly, using this baseline, for the normal knurls, the height of the knurls is calculated as the distance between the peak (the highest point on the knurl profile) and the baseline point. For a worn knurl with shoulders as shown in Figure 2-3, the height is the difference between the lower shoulder and the baseline. Finally, for a completely worn knurl, the height is the distance between the flat line on the top and the baseline.

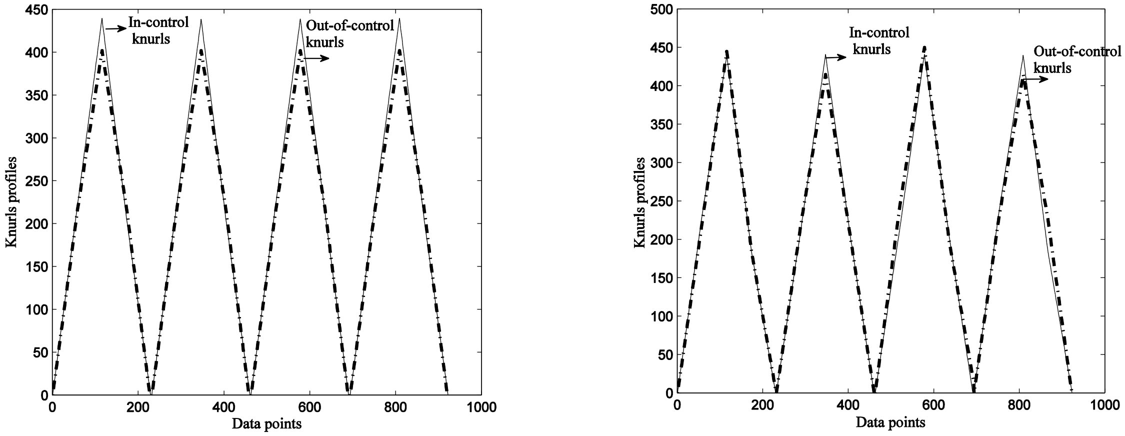


(a) Knurls profiles in one column (b) Frequency-domain profile

Figure 2-10: A column of knurls for FFT and FFT result

The monitoring performance of all methods is assessed using a simulation study performed in two different scenarios. In the first scenario (Scenario I) as shown in Figure 2-11 (a), all four rows of knurls located in one column are worn, while in Scenario II as shown in Figure 2-11 (b), two rows of knurls are completely healthy and the other two rows of knurls are slightly worn. The in-control knurls are all simulated using the mixed-effect model in Eq. (2-6). A set of slightly-worn knurls are used to calculate the fixed-effect coefficients in the mixed-effect model to simulate the out-of-control knurls. Four different values of correlation coefficient $\boldsymbol{\rho} =$ 0.1, 0.5, 0.7 and 0.95 are considered for the cross-correlation among the rows. The cross-correlation between Row 1 and Row 2 is positive while Row 3 and Row 4 have negative correlation. Either the correlation between Row 1 and Row 3 or the correlation between Row 2 and Row 4 is negligible. A set of 400 anvils is simulated using the above-mentioned procedure. Each anvil contains 4 rows with each row having 24 columns of knurls. Note that in Scenario II as shown in Figure 2-11 (b), two out of four rows in each column are arbitrarily selected. Then, in each simulation run, a set of slightly-worn knurls are simulated. This way of the simulation enables us to generate a completely random wear pattern on the anvils.

As discussed in Subsection 2.2.1, the dataset can be represented in three ways. Based on Representation I, a 9600×924 matrix can represent the whole simulated dataset. Following Representation II, the dataset is represented using a 38400×231 matrix. The third way is to use a tensor representation with the dimension of $4 \times 231 \times 9600$. The PCA method is applied to above two matrix representations.



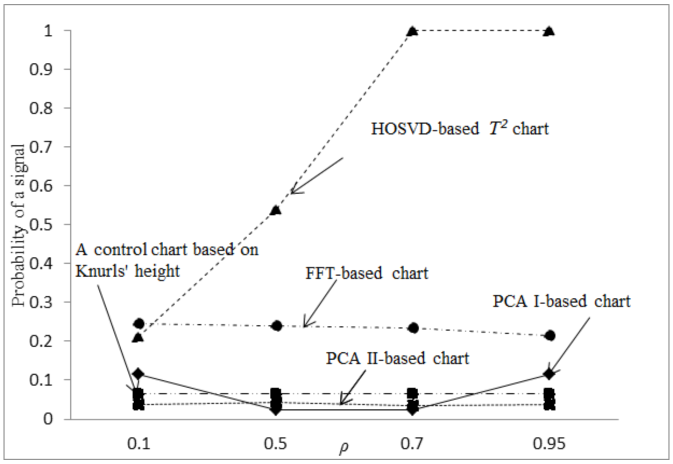
(a) Scenario I: all knurls are worn (b) Scenario II: two out of four knurls are worn

Figure 2-11: Simulated in-control and out-of-control knurls in a column of anvil

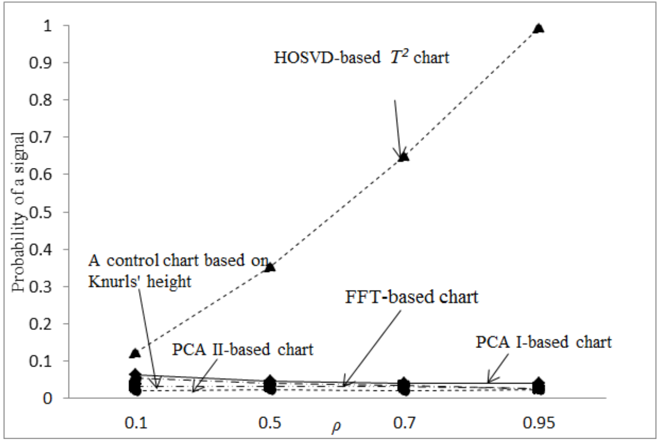
For PCA I method, 7 components are used for constructing the monitoring chart since they are sufficient to cover 90% of the data variability. For PCA II method, four PC features can explain about 90% of the data variability. When the HOSVD method is applied to the tensor of data directly, in order to represent 90% of the variability of the data, we need to use 4 components for the row’s mode, 3 for the knurls profiles mode, and 6 for the columns of knurls mode. For the FFT method, since the first frequency feature is sufficient to explain about 99% of the signal’s energy, only this feature is used for constructing the associated control chart. After extracting the monitoring features using each method, a T^2 control chart is used for monitoring the features extracted from the PCA and HOSVD methods. A Shewhart control chart is used for monitoring the magnitude of the dominant frequency in the FFT method. In addition, the knurls’ heights are monitored using a Shewhart control chart.

To compare the monitoring performance, in each simulation run, 400 in-control anvils are generated, and the Type-I error rate of each method is computed as the proportion of out-of-control samples to all simulated samples. In this study, 10000 simulation runs are conducted. The final estimation of the Type-I error rate is computed by averaging the Type-I error rates of the 10000 runs' results. The upper control limit is then set in such a way resulting in an overall Type-I error rate equal to 0.5%. The probability of triggering an alarm for detecting worn knurls, i.e., detection power, is used as a criterion for comparing the detection performance of aforementioned three methods.

Figure 2-12 (a) gives the detection power of all four methods under different correlation coefficients ρ in Scenario I. The HOSVD-based method outperforms all other methods except when the cross-correlation among the rows is $\rho = 0.1$. In this case, the FFT method performs slightly better than HOSVD-based T^2 chart. The detection power of the HOSVD-based T^2 chart increases with the increase of the cross-correlation since the tensor data representation can truly preserve the actual data cross-correlation structure among the rows as discussed in Subsection 2.3.1. The two PCA-based charts perform almost similarly in detecting the out-of-control situation and their performance is also not affected significantly by the amount of correlation among the rows. As expected, the control chart constructed based on the knurls' heights has shown poor performance in detecting slightly-worn knurls. In Scenario II, it is observable in Figure 2-12 (b) that the HOSVD-based T^2 chart significantly outperforms all other methods. Based on these findings, it can be concluded that the HOSVD-based method is a more reasonable option for monitoring the tool wear data when a strong cross-correlation exists.



(a) Scenario I



(b) Scenario II

Figure 2-12: Probability of signal with regard to different correlation coefficients

2.4 Case Study

This section presents the results of applying the HOSVD-based method for monitoring the wear of knurls using a real production dataset. The dataset consists of five anvils: Anvil 1 is new, and Anvils 2~5 are ordered with increasing wear levels. The HOSVD method is used to extract the features followed by a T^2 control chart constructed based on the extracted features. To explain 90% of the data variability, the number of components in Mode 1, Mode 2 and Mode 3 are chosen as $k_1 = 4$, $k_2 = 3$ and $k_3 = 2$, respectively. Hence, the tensor scores $\mathcal{Z}$ given in Eq. (2-3) is a $4 \times 3 \times 24$ tensor, and the matrix of extracted monitoring features is represented by $\mathbf{Z} \in \mathbb{R}^{24 \times 12}$ for constructing the T^2 control chart Figure 2-13 shows the T^2 control chart for all five anvils. Each point on the control chart represents the T^2 statistic computed for a column of knurls on an anvil; as a result, a set of 24 points represents one anvil. The control limit and the parameters are estimated using the new Anvil 1 with all the 24 T^2 statistics being within the control limit. Since some of the knurls on Anvil 2 are slightly worn, there are some out-of-control points in the chart. In Anvil 3, most of knurls are severely and completely worn; as a result, all points are out of control, and this is the same for Anvil 4 and Anvil 5. The aforementioned observations show that the proposed control chart can effectively detect the worn knurls at different statuses.

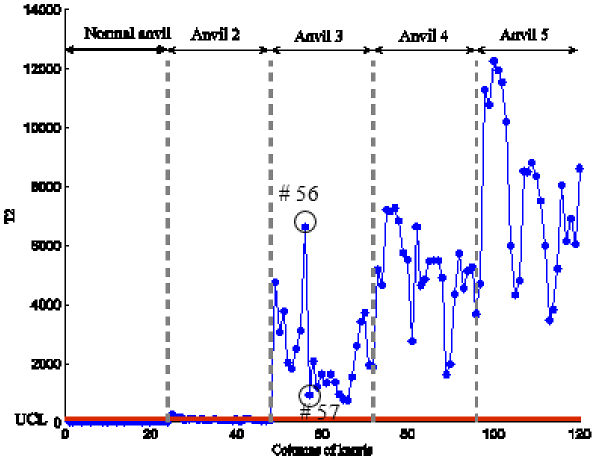
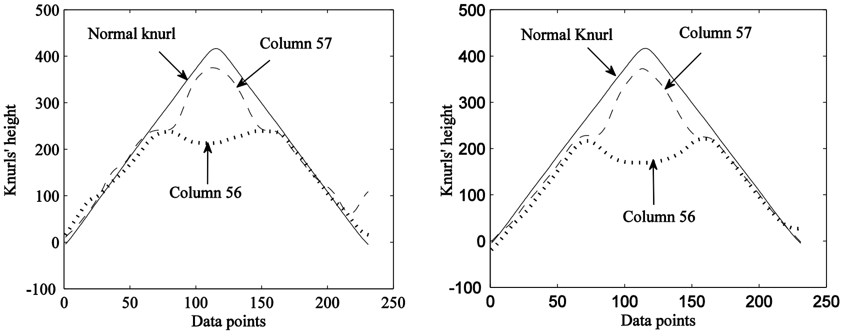
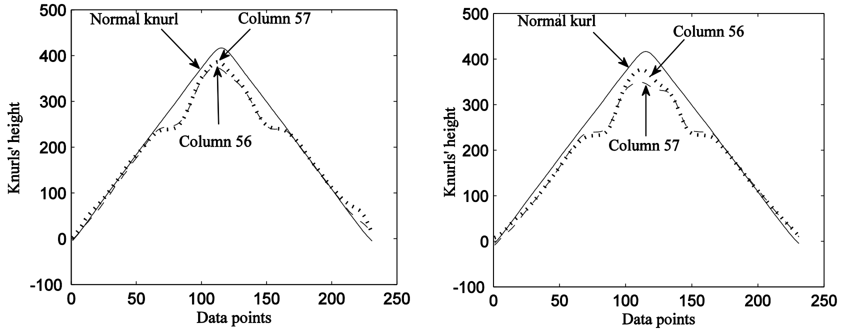


Figure 2-13: $\mathbf{T^2}$ control chart to monitor knurls' wear in all five anvils

To further verify the accuracy of the proposed method, Column 56's knurls are compared to those of Column 57 marked in Figure 2-13. Since the 56th T^2 statistic is much higher than the 57th T^2 statistic, it is expected that the knurls in the 56th Column are more worn than those in the 57th Column. Figure 2-14 compares the knurls for Column 56, Column 57 and a normal knurl without any wear, in which the 4 rows of knurls are compared separately in Figure 2-14 (a)~(d), respectively. Figures 2-14 (a) and 2-14 (b) clearly show that the knurls in Row 1 and Row 2 of Column 56 are completely worn. In contrast, Column 57's knurls are only slightly worn. The knurls of Row 3 and Row 4 shown in Figures 2-14 (c) and 2-14 (d) do not show a significant difference in the wear levels between Column 56 and Column 57. Hence, it is concluded that the severe wear level in Row 1 and Row 2 is responsible for the high value of T^2 statistic in Column 56.



(a) Row 1 (b) Row 2



(c) Row 3 (d) Row 4

Figure 2-14: Visual comparison of Column 56 versus Column 57

As discussed in Subsection 2.3.1, one of the strengths of the HOSVD method is its superior capability in the analysis of the variation pattern for the tensor data, or specifically in this case, explicit analysis of the cross-correlation among different rows of knurls in an anvil. This can help provide information regarding the knurls' wear pattern on the anvil and show whether the anvil is misaligned or not. As an example, Figure 2-15 shows completely worn knurls in Row 1 and Row 2 while less wear can be seen in Row 3 and Row 4. The proposed HOSVD method is applied to this anvil data using the tensor representation. The resultant Mode-1 singular vectors are plotted in Figure 2-16, in which the first singular vector gives an average of all four rows while the second singular vector clearly shows the negative cross-correlation of

Rows 1 and 2 with Rows 3 and 4. This negative cross-correlation, as pointed out, reflects the unusual wear pattern on the anvil due to the misalignment in the anvil installation.

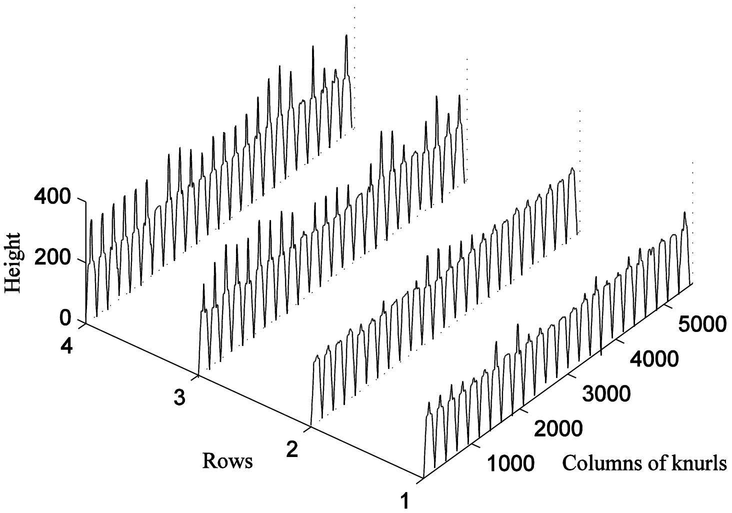
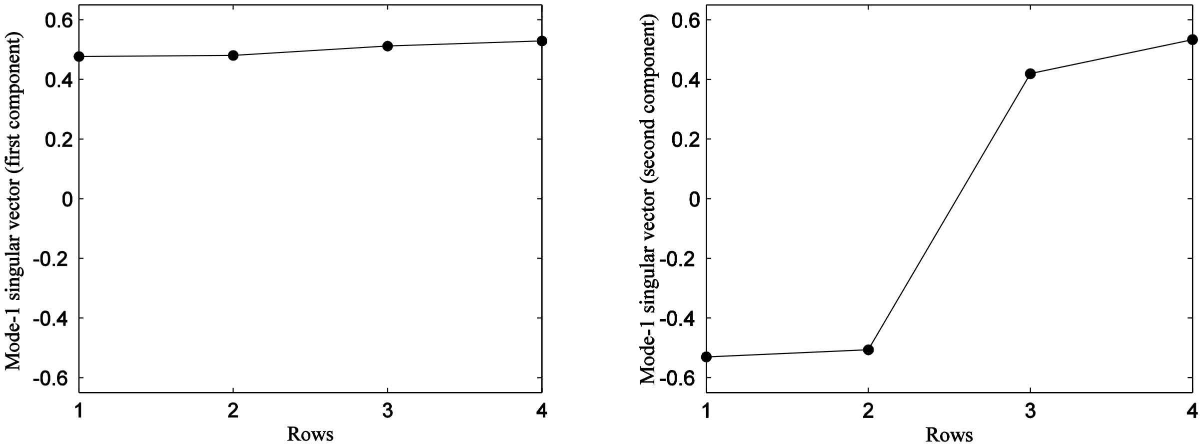


Figure 2-15: Visualization of a misaligned anvil



(a) first component (b) Second component

Figure 2-16: Mode-1 singular vectors obtained using HOSVD method

References:

1. Abellan-Nebot, J. V., Subirón, F. R., (2010), “A review of machining monitoring systems based on artificial intelligence process models,” *The International Journal of Advanced Manufacturing Technology*, 47(1–4), pp.237–257.

2. He, X., Cai D., and Niyogi P., (2005), “Tensor Subspace Analysis,” *Advances in neural information processing systems*, 18 (NIPS), MIT Press.

3. Hotelling, H., (1947), “Multivariate quality control,” *Techniques of statistical analysis*, 1947, pp 114-184

4. Jolliffe, I., (2005), “Principal component analysis,” *Wiley Online Library*.

5. Kisić, E., Durović, Z., Kovačević, B., Petrović, V., (2015) “Application of T^2 Control Charts and Hidden Markov Models in Condition-Based Maintenance at Thermoelectric Power Plants,” *Hindawi corporation, Shock and Vibration*, 2015, Article ID 960349, 11 pages.

6. Kolda, T., and Bader, B., (2009), “Tensor decompositions and applications,” *SIAM Rev*, 51(3), pp.455-500.

7. Lathauwer, L. de., Moor, B. de., and Vandewalle, J., (2000), “On the best rank-1 and rank-(R1, R2,...,Rn) approximation of high order tensors,” *SIAM Journal of Matrix Analysis and Applications*, 21(4), pp.1324–1342.

8. Li, X., Dong, S., and Yuan, Z., (1999), “Discrete Wavelet Transform for Tool Breakage Monitoring,” *International Journal of Machine Tools and Manufacture*, 39 (12), pp. 1935–1944.

9. Mason, R. L., Tracy, N. D., Young, J. C., (1995), “Decomposition of T2 for Multivariate Control Chart Interpretation,” *Journal of Quality Technology*, 27 (2), pp. 109-119.

10. Paynabar, K., Jin, J. J., and Pacella, M., (2013), “Monitoring and Diagnosis of Multichannel Nonlinear Profile Variations Using Uncorrelated Multilinear Principal Component Analysis,” *IIE Transactions*, 45(11), pp. 1235-1247.

11. Shao, C., Guo, W., Kim, T. H., Jin, J. J., Hu, S. J., Spicer, J. P., and Abell, J. A., (2014), “Characterization and monitoring of tool wear in ultrasonic metal welding,” 9*th International Workshop on Microfactories*, Honolulu, Hawai‘i, October 5–8, pp. 161–169.

12. Shao, C., Kim, T. H., Jin, J. J., Hu, S. J., Spicer, J. P., and Abell, J. A., (2016), “Tool wear monitoring for ultrasonic metal welding of lithium-ion batteries,” *ASME Journal of Manufacturing Science and Engineering*, 138(5), pp. 051005.

13. Shao, C., Paynabar, K., Kim, T. H., Jin, J. J., Hu, S. J., Spicer, J. P., ... & Abell, J. A., (2013), “Feature selection for manufacturing process monitoring using cross-validation,” *Journal of Manufacturing Systems*, 32(4), pp. 550-555.

14. Shi, D., Gindy, N. N., (2007), “Tool Wear Predictive Model Based on Least Squares Support Vector Machines,” *Mechanical Systems and Signal Processing*, 21(4), pp. 1799–1814.

15. Yan, H., Paynabar K., and Shi, J., (2015), “Image-Based Process Monitoring Using Low-Rank Tensor Decomposition,” *IEEE Transactions on Automation Science and Engineering*, PP(99): pp. 1–12.

16. Zhou, J. H., Pang, C. K., Zhong, Z. W., and Lewis, F. L., (2011), “Tool wear monitoring using acoustic emissions by dominant-feature identification,” *IEEE Transactions on Instrumentation and Measurement*, 60(2), pp. 547–559.

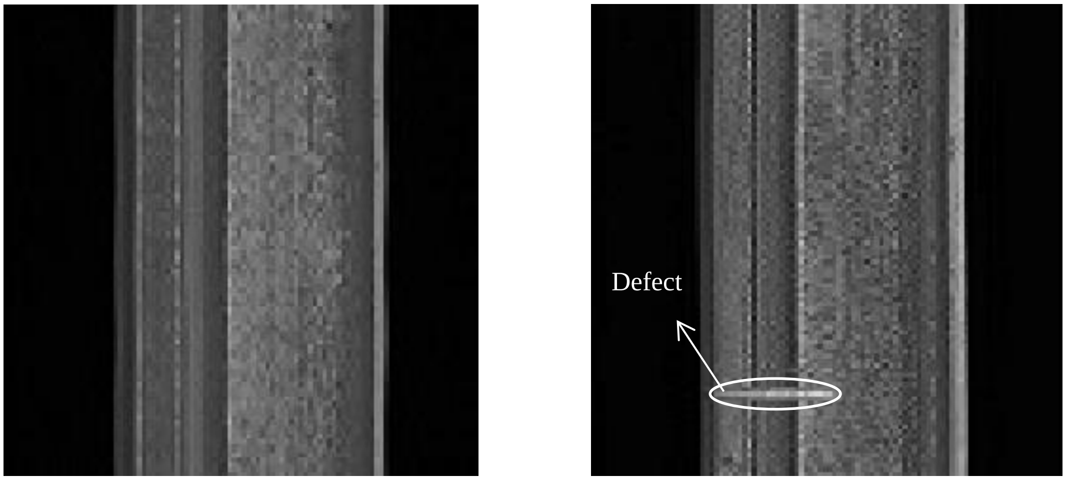
CHAPTER 3

ROBUST GENERALIZED SVD WITH CORRELATED NOISE AND OUTLIERS

3.1 Introduction

Using images for monitoring the quality of a process is nowadays appealing due to the rich information that can be provided by these images about a product. In many industrial applications, the type of information typically needed for quality control is mainly about some characteristics of the products, such as geometry, aesthetic features, surface defects, etc. Analyzing images can provide updated information regarding the critical quality characteristics with high precision. One practical example of image analysis for process quality monitoring can be found in billets defect monitoring. Billets are manufactured by solidifying the liquefied metal in a hot rolling process. The billets are further operated to be final products. A critical feature that must be inspected to ensure that the billets are qualified for use in the next stage is the quality of the surface which must be free of defects (see Figure 3-1).

Although human visual inspection conventionally was an admitted strategy for finding the defects, it is a cumbersome procedure, because the number of images that must be evaluated is very large due to the high production rate. This problem affects both the accuracy and speed of the evaluations. Having such problems in human-vision inspections led to the emergence of machine-vision system (MVS) which is basically a computer performing all the required steps for image analyses including data acquisition, preprocessing, feature extraction and monitoring (Megahed et al., 2011).



(a) a faultless billet (b) a defective billet

Figure 3-1: Images of billets

All MVSs consist of some common steps that must be performed sequentially. Data acquisition is basically the utilization of cameras in order to capture the images. After the data-acquisition step, a preprocessing step which typically includes noise reduction, compression and contrast increase is performed on the images. In the next step, the images are analyzed in order to extract the useful features, and finally a control chart is used for monitoring the extracted features. It is worth mentioning that in some cases the purpose of image analysis is to locate the defects and to determine the boundaries of the defects on an image (Yan et al., 2015). However, this chapter focuses mainly on developing a general methodology for feature extraction and dimension reduction, which can be a vehicle for image-based monitoring.

Several techniques are developed in the imaged-based process monitoring literature with a vast variety of applications. Megahed et al. (2011) provided a thorough review about the methods typically used for image-based monitoring. In some applications where the geometric features of a product must be monitored, image processing techniques can be applied as an intermediate tool for extracting the required features. For instance, using edge detection methods, Tan et al. (1996) extracted the length, width, and the area of extruded food like corn puffs from some images. X-bar control charts, subsequently, were used to monitor the extracted features.

Nembhard et al. (2003) used an integrated method for monitoring the color quality in plastic products. An EWMA control chart is used to monitor hue as a metric representing the color quality, which is determined using some images taken from a strip of high-density polyethylene tape.

Another interesting application of image-based monitoring is in the flotation process widely used for concentrating metal-bearing mineral in an ore. The color and structure of froth flotation on the surface can be indicators for the type of minerals in the froth and some flotation characteristics, such as degree of mineralization. Liu et al. (2005) applied the principal component analysis (PCA) method to the image data and extracted the first two important principal components (PC). A masking process was developed in the PC scores' space to extract the color features corresponding to the froth. Wavelet decomposition method was subsequently applied to the PC features to determine the structural features of the froth, like the histogram of bubble size. To detect any abnormality in the new images, a control chart is used to monitor the residuals obtained from a new image and the image reconstructed using the PCA model in Phase I.

A Hotelling's T^2 control chart was suggested by Liu and MacGregor (2006) to monitor aesthetic and visual appearance of products like countertop stones. The idea was to apply the wavelet texture analysis method based on a 2D discrete wavelet transform with the purpose of extracting the textural features. A dimension reduction method such as PCA was afterwards employed to reduce the dimension of extracted features. The extracted PC scores were monitored using the T^2 control chart. With a rather similar purpose, Lin (2007) took advantage of wavelet decomposition method to extract the textural features of the images taken from surface barrier layer chips of ceramic capacitators. A T^2 control chart was constructed based upon the extracted features for monitoring the surface for detecting ripple defects. Lin et al. (2008) combined the wavelet decomposition method with PCA in order to detect the defects in a light-emitting diode (LED) chip and concluded that the combined wavelet-PCA chart has favorable monitoring abilities. Using singular value decomposition (SVD), Lu and Tsai (2005) removed the background texture of LCD panels by reconstructing the images using the eigenvectors corresponding to the small eigenvalues. An X-bar control chart was subsequently used for monitoring the defects on the surface of LCD panels.

As it can be seen in the image-based monitoring literature, the decomposition methods, such as SVD (PCA) play an essential role in forming the basis for process monitoring. The rationale behind using a decomposition method for image data analysis in addition to dimension reduction can be explained based on the fact that the singular vectors are informative sources for analyzing the variation pattern of the images' pixel intensities. Pixel intensity reflects the gray intensity in the texture of an image, and it usually ranges from 0 to 255. The low intensity of a pixel shows that the color is almost black while high intensity in a pixel means that the color is closer to white in that pixel. The important point is that when there is no defect in a specific area of an image, the pixel intensities should not dramatically vary from one pixel to another; as a result, the corresponding singular vector must demonstrate the random (natural) variability of the pixel intensities in that area depending on the amount and type of noise in the image. However, the pixel intensities around the defective area shows higher variability, and the associated singular vector must demonstrate this fact by assigning higher loads on the involved variables (Figure 3-2).The discrepancy in the singular vector space of the defective and faultless areas gives the motive for basing the defect monitoring scheme on the features extracted by a decomposition method, such as SVD.

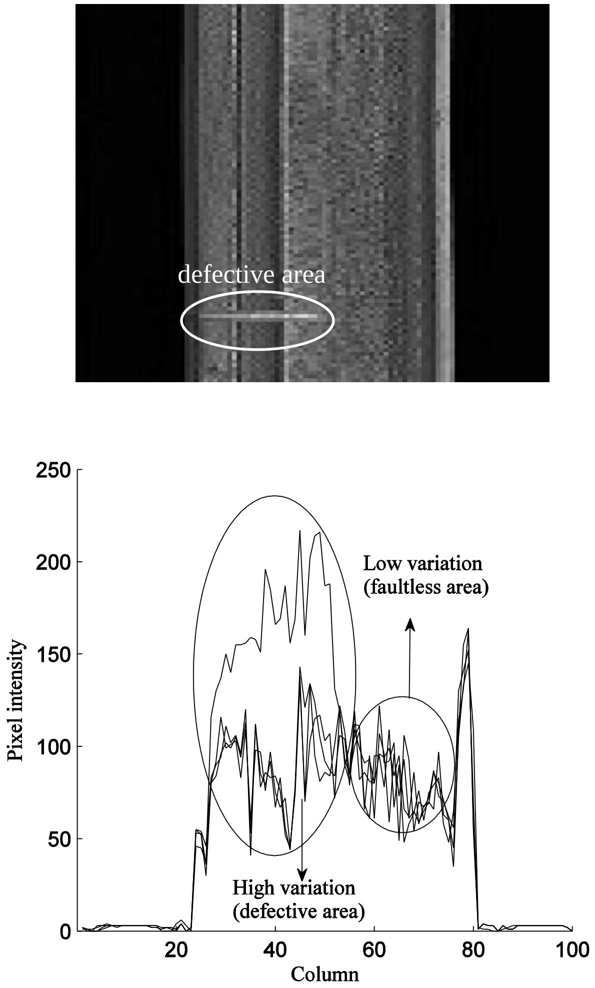
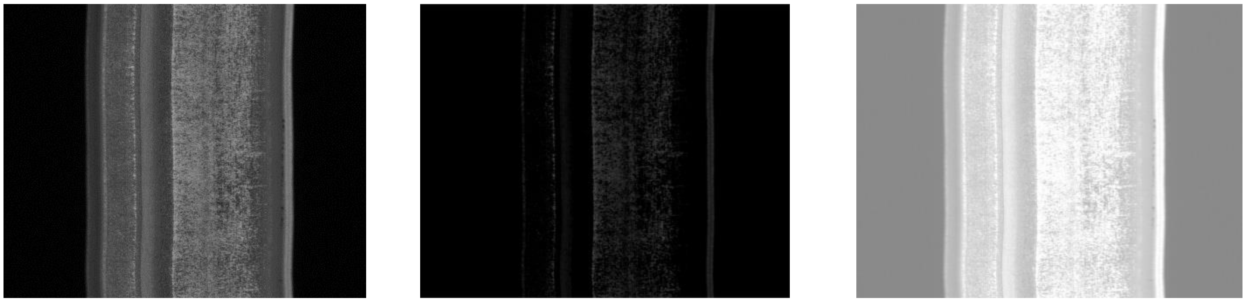


Figure 3-2: Pixel intensities’ variation patterns for normal and defective areas

Due to some issues, such as sensor errors, technical faults in data acquisition equipment, abrupt system failures, inappropriate illuminations, etc., some of the images might have unfavorable visibility. These abnormal images are typically called outliers whose existence is problematic in the sense that they drastically influence the accuracy of the estimates (Figure 3-3). In the context of decomposition methods like SVD, since some images are used for estimating a baseline singular vector in phase I, which will be used as a reference for online monitoring in Phase II, the abnormal images in the initial dataset will misrepresent the true singular vector, and the characteristics of a normal image will not be correctly captured.



(a) normal image (b) underexposed image (too dark) (b) overexposed image (too bright)

Figure 3-3: Different types of outlier images

Many authors have addressed the problem of outliers when using the SVD or PCA methods (Maronna and Yohai 2008, Hubert et al. 2005). One possible way to overcome this problem is to detect and eliminate all the outlier points. However, this is a cumbersome task even if it is possible due to the massive amount of data. In addition, some of the images might not be very unclear; thus not detectable, but they still affect the estimates of the singular vectors. These images are not usually detected by the outlier detection methods. An alternative yet not efficient strategy is to substitute the covariance matrix by a robust scatter matrix like minimum covariance determinant and decompose the scatter matrix, e.g., Naga and Antille (1990), Croux and Haesbroeck (2000). This approach is restricted to low-dimensional data because of the high computational cost for calculating a high-dimensional covariance matrix. Another way to address the issue of outliers is to optimize a robust dispersion estimate of the variability. These methods are generally called projection pursuit (please see Croux and Ruiz-Gazen 2005, Maronna (2005), and Ke and Kanade 2005). A rather different approach was proposed by Hubert et al. (2005). The proposed method was called ROBPCA, and it combines the projection pursuit method with a robust scatter matrix estimate.

When working with outliers, usually the whole sample is considered as an outlier; i.e., the whole row in the data matrix is contaminated. This type of outliers is called rowwise outliers. The problem of outliers sometimes, however, appears in a rather different format. In the second type, some but not all the elements of a row are outliers. These types of outliers are called elementwise outliers, and they frequently happen in high dimensional images. Liu et al. (2003) developed a regularized robust function using Huber function to account for the elementwise outliers. Having developed a robust M-estimation algorithm, De la Torre and Black (2001) used a gradient descent algorithm to solve the robust PCA problem considering elementwise outliers in images. Rey (2007) addressed the problem of elementwise contamination and proposed weighted singular value decomposition. Maronna and Yohai (2008) proposed an M-estimation algorithm following the regression MM estimates developed by Yohai (1987). Their proposed approach can account for both rowwise and elementwise outliers.

All the aforementioned robust decomposition methods are based on either SVD or PCA method. It is worth to mention that a crucial assumption of the SVD method is that the noise components in the data are independently distributed. In the imaging context, this means that the noise for one specific pixel is independent from those in other pixels. This assumption does not necessarily hold in imaging owing to the fact that the noise components in the neighboring pixels are spatially correlated. Under these circumstances, the regular PCA method fails to capture the true structure of the dataset (true variation pattern) due to the presence of correlation in the noise. Allen et al. (2014) proposed a generalized PCA (GPCA/GSVD) method to effectively consider the correlation among the noise components in the dataset. A detailed elaboration on the GSVD method will be provided in Section 3.2. Similar to the SVD method, the GSVD method is sensitive to the presence of outliers in the dataset, and the estimated singular vectors will be different from the true singular vectors if the data are contaminated. Hence, a robust GSVD method must be developed in order to account for the outliers in the dataset.

This chapter suggests a robust generalized singular value decomposition method that can be used for decomposition, dimension reduction and feature extraction purposes. The proposed method accounts for the spatial/spatiotemporal correlation structure in the noise components; as a result, denoising (prewhitening) is not necessary. It, furthermore, automatically accounts for the outliers, so there is no need for prior data analysis and outlier removal before applying the method.

In order to apply the low-order decomposition methods, such as SVD, GSVD, and RGSVD to extract the image features in Phase I, each image must be reordered and represented as a vector. This will lead to a matrix with each row representing an image collected in Phase I. The baseline features for Phase II monitoring are extracted from this stack-up matrix. Although frequently used in practice, this strategy has some drawbacks. First, the dimension of the resulting matrix will be high, and as it will be discussed later, this issue increases the running time of the algorithm used for obtaining the solutions of the decomposition. Second, the spatial correlation structure which typically exists in the images is broken, leading to inaccurate estimates of the baseline features. For these reasons, a high-order array (tensor) is suggested to represent the images in Phase I. The advantages of employing a high-order array are (1) it preserves the correlation structure of the images, and (2) it avoids the high dimensionality issue caused by vectorizing the images. A robust generalized high-order SVD (RGHOSVD) method is developed, which can be used for decomposing the tensor of image data in the presence of outliers and spatially correlated noise components (see Section 3.4 for further details).

To sum up, the major strategies that can be used for image-based monitoring are control charts constructed upon the features extracted using (1) wavelet decomposition methods followed by PCA for further dimension reduction, and (2) A decomposition method, like SVD for feature extraction followed by a control chart to monitor the features. In both strategies, the outlier images and the correlated noise in the images affect the accuracy of the estimates. As a result, the baseline features which are estimated in Phase I will be far from the true features; thus leading to poor monitoring performance in Phase II.

This chapter is laid out as follows. Section 3.2 explains the GSVD method. Section 3.3 introduces the robust GSVD (RGSVD) method along with an iterative algorithm to obtain the solutions. In Section 3.4, the RGHOSVD method is proposed for high-order feature extraction, and an algorithm is developed for solving the problem. A comprehensive simulation study is conducted to investigate the accuracy of the estimates in different decomposition methods; moreover, the monitoring performance of the image-based monitoring charts is further evaluated in Section 3.5. Section 3.6 applies the RGHOSVD method (which proves to be the most effective method) to monitor the defects on the surface of billets in a real image dataset in rolling bar process.

3.2 Model Formulation

If we assume that m images are collected in Phase I with each image represented by a matrix $\mathbf{Y} \in \mathbb{R}^{n \times p}$, then each image is modeled as

$$\mathbf{Y} = \sum_{l=1}^{L} a_l \boldsymbol{\alpha}_l \boldsymbol{\beta}_l^T + \boldsymbol{\epsilon} \tag{3-1}$$

where L is the number of components (number of signals), a_l is the intensity of the lth signal, $\boldsymbol{\alpha}_l \in \mathbb{R}^n$ and $\boldsymbol{\beta}_l \in \mathbb{R}^p$ are the lth left and right singular vectors (signals), respectively, $\boldsymbol{\epsilon} \in \mathbb{R}^{n \times p}$ is the noise matrix with a matrix-variate distribution i.e. $\boldsymbol{\epsilon} \sim \boldsymbol{F}\left(\overleftrightarrow{\mathbf{0}}, \Delta, \boldsymbol{\Sigma}\right)$, where $\boldsymbol{F}$ is an unknown distribution, $\overleftrightarrow{\mathbf{0}}$ is a zero-valued mean matrix, $\Delta \in \mathbb{R}^{n \times n}$ is the rowwise covariance matrix, and $\boldsymbol{\Sigma} \in \mathbb{R}^{p \times p}$ is the columnwise covariance matrix. To apply a decomposition method for feature extraction in Phase I, all the m images are vectorized and stacked up in a matrix $\mathbf{X} \in \mathbb{R}^{m \times np}$ modeled as

$$\mathbf{X} = \sum_{l=1}^{L} d_l \mathbf{u}_l \mathbf{v}_l^T + \mathbf{E} \tag{3-2}$$

where d_l is the intensity of the lth signal, $\mathbf{u}_l \in \mathbb{R}^m$ represents the lth left singular vector of matrix $\mathbf{X}$, $\mathbf{v}_l \in \mathbb{R}^{np}$ denotes the lth right singular vector, and $\mathbf{E} \in \mathbb{R}^{m \times np}$ is the noise matrix with correlated columns. Each row of the noise matrix $\mathbf{E}$ in Eq. (3-2) follows a vectorized matrix-variate distribution with zero mean vector and covariance matrix $\boldsymbol{\psi} \in \mathbb{R}^{np \times np}$ which is the Kronecker product of the rowwise and columnwise covariance matrices of the matrix **Y** (Gupta and Nagar 1999); i.e. $\boldsymbol{\psi} = \Delta \otimes \boldsymbol{\Sigma}$ where $\otimes$ denotes the Kronecker product. It is assumed that the noise components in one image are independent from those in other images. That is, the rows of matrix **E** are independent of each other. However, the correlated noise in the columns of matrix **E** if not correctly modeled, it adversely affects the estimation of the singular vectors.

To overcome the problem of correlated columns in matrix **E**, which typically occurs in datasets involving imaging, longitudinal data, time series etc., a weighted SVD method was proposed by Allen et al. (2014). The main strategy was to replace the L_2 norm in SVD method by a $Q\,\&\,R$ norm in order to account for rowwise and columnwise correlations in the noise matrix. The $Q\,\&\,R$ norm of a matrix **A** is defined as $||\mathbf{A}||_{Q,R}^2 = Tr(\mathbf{QARA}^T)$, where **Q** and **R** are called quadratic operators weighting the elements of matrix **A** based on the correlation structure of the elements. The stronger the correlation between two rows (columns) is, the larger their corresponding weight in the quadratic operator matrix $\mathbf{Q}(\mathbf{R})$ is. In this sense, the quadratic operators can be regarded as covariance matrices and estimated from the data. However, this might not be the most efficient solution to determine the quadratic operators when dealing with spatiotemporal data.

To estimate the quadratic operators in such applications, some ideas, like the covariance functions can be borrowed from the spatial statistics literature to estimate the covariance matrices. For spatially correlated data, the main assumption is that if two points are spatially close to each other, they are most likely have similar characteristics; thus having high correlation. As a result, the covariance function which significantly depends on the distance between the variables is a reasonable option for estimating the covariance matrix. The Matern, rational, piecewise polynomial, exponential and squared exponential covariance functions are all examples of parametric covariance functions typically used as estimators for the covariance matrix of spatially correlated data. One issue that must be considered when using the covariance function is how to choose the parameters in the function. In practice, depending on the application, the parameters are determined by optimizing an appropriate criterion, like the total variability explained by the extracted features.

One important point to mention is that the quadratic operators must be determined based on the matrices of images $\mathbf{Y}$'s. The final quadratic operator used in the solution is a columnwise quadratic operator $\mathbf{C} \in \mathbb{R}^{np \times np}$ weighting the columns of matrix $\mathbf{X}$. In other words, the columnwise quadratic operator is defined as $\mathbf{C} = \mathbf{Q} \otimes \mathbf{R}$, where $\mathbf{Q} \in \mathbb{R}^{n \times n}$ and $\mathbf{R} \in \mathbb{R}^{p \times p}$ are the rowwise and columnwise quadratic operators of the image matrix $\mathbf{Y}$, respectively. The rowwise quadratic operator of matrix $\mathbf{X}$ is an $m \times m$ identity matrix since the rows of matrix $\mathbf{X}$ are assumed to be independent. Having obtained the quadratic operator $\mathbf{C}$, the solution of the problem in Eq. (3-2) is computed by an iterative block-wise algorithm (Allen et al. 2014). The estimation of the quadratic operators will be further discussed in Section 3.3.

3.3 Robust Generalized Singular Value Decomposition Method (RGSVD)

In this section, the RGSVD method is proposed for decomposing the matrix of images **X**. In the linear regression literature, there are several robust procedures that can be employed to estimate the parameters when the data contain outliers. The appropriate method can be selected by considering two essential properties of the resulting estimates. The first one is the degree of robustness of an estimator, which is typically measured by the breakdown-point criterion. The breakdown-point of an estimator is defined as the smallest fraction of outliers that can break the estimator down in the sense of providing inaccurate estimates.

An estimator with high breakdown point is favorable since it shows more robustness to the existence of outliers. Maximum-likelihood-type estimates (M estimates) proposed by Huber (1973) to estimate the regression parameters tend to have breakdown point of zero when there are some outliers in the predictors. High breakdown-point estimators, such as least trimmed squares (LTS) or least median of squares (LMS), on the other hand, suffer from low efficiency, as the second important property of the robust estimators, meaning that in case there are no outliers in the data the estimates are inaccurate. Rousseeuw and Yohai (1984) developed the S estimator by minimizing an M-estimate of the regression residual scale. A potential problem of the S estimator reveals itself when the errors in the regression model are normally distributed. In this case, the estimates are inefficient. In other words, these estimates cannot achieve high efficiency and high breakdown point simultaneously. The MM estimators are initially proposed by Yohai (1987) with the purpose of overcoming the shortcomings of other robust estimators. In this chapter, the approach of Yohai (1987) is developed to estimate the right and left singular vectors in GSVD and GHOSVD problems.

There are three major steps in obtaining the MM estimators (1) start with high breakdown-point but not necessarily efficient estimates (initial solutions), (2) obtain a robust M-estimate of the residual scale, and (3) attain the final M estimates known as MM estimates using the initial solutions and the robust scale estimator computed in Step 1 and Step 2, respectively. In the following subsections, a detailed elaboration is provided for each of the steps.

3.3.1 Initial estimates

To obtain high breakdown-point (50%) initial estimates for the left and right singular vectors, the repeated median method proposed by Siegel (1982) is utilized. The method was mainly used to provide a nonparametric estimate of the slope parameter in linear regression when the data is contaminated. A similar approach can be employed to obtain the initial estimates for the left and right singular vectors in the RGSVD method. Consider a single-component $(L = 1)$ version of the model in Eq. (3-2) with $\mathbf{u}$ and $\mathbf{v}$ to be estimated. If either $\mathbf{u}$ or $\mathbf{v}$ is known, estimating the other singular vector is similar to determining the slope of a linear regression model. Therefore, the repeated median method can be employed in this framework for singular vector estimation. For each column $j; j = 1, 2, \dots, np$, of matrix **X** define $med_i; i = 1,2, \dots, m,$ as the median of the slopes $\frac{x_{ij}-x_{lj}}{u_i-u_l}$; $l \neq i = 1,2 \dots, m$. Then, the jth element of the right singular vector **v** can be obtained as

$$v_j = \text{median}\{med_1, med_2, \dots, med_m\}; j = 1, 2, \dots, np, \tag{3-3}$$

Similarly, the ith element of the left singular vector **u** with known and fixed **v** can be computed as

$$u_i = \text{median}\left\{med_1, med_2, \dots, med_{np}\right\}; i = 1, 2, \dots, m, \tag{3-4}$$

where med_j; $j = 1, 2, \dots, np$, is defined for each row i, $i = 1,2, \dots, m$, as the median of the slopes $\frac{x_{ij}-x_{ip}}{v_j-v_p}$; $p \neq j = 1, 2, \dots, np$. Starting with an arbitrary vector for $\mathbf{u}$, we update $\mathbf{v}$ using Eq. (3-3). Then the left singular vector $\mathbf{u}$ is updated in a similar way as in Eq. (3-4). Since this algorithm does not converge, after N ($N = 1000$) iterations the solutions minimizing $\left\|\mathbf{X} - \widehat{\mathbf{X}}\right\|$ with $\widehat{\mathbf{X}} = \mathbf{u}\mathbf{v}^T$ can be considered as the initial estimates for the singular vectors. This algorithm provides initial estimates with high breakdown-point.

3.3.2 M-estimate of the residuals' scale

The second step of the MM-estimate method aims at acquiring a robust estimate of the residuals' scale. If the ith row of matrix $\mathbf{X}$ is denoted by $\mathbf{x}_i$, the objective function of the GSVD method (Allen et al. 2014) can be rewritten as

$$\sum_{i=1}^{n} \|\mathbf{x}_i - \boldsymbol{u}_i \mathbf{v}_l^T\|_C^2 \tag{3-5}$$

where $\|\mathbf{A}\|_C^2 = Tr(\mathbf{A}\mathbf{C}\mathbf{A}^T)$, and $\mathbf{C}$ is the column-wise quadratic operator. Based on the objective function in Eq. (3-5), the ith residual can be defined as $r_i = \|\mathbf{x}_i - u_i \mathbf{v}_k^T\|_C^2$ with scale parameter $\tau(r)$. The M-estimate of the residuals' scale is defined as the solution of

$$\frac{1}{m}\sum_{i=1}^{m} f_0\left(\frac{r_i}{\tau}\right) = \delta \tag{3-6}$$

where δ is a constant and can be defined as $E_\phi f_0(r) = \delta$ with ϕ standing for standard normal distribution, and $f_0(\cdot)$ is a bounded loss function with the following properties (Huber 1981):

(1) f_0 is symmetric, continuously differentiable, and $f_0(0) = 0$.

(2) There exists $a > 0$ such that f_0 is strictly increasing on $[0, a]$ and constant on $[a, \infty)$.

(3) $\frac{\delta}{f(a)} = 0.5$.

As shown by Huber (1981), the last condition guarantees that the scale estimator in Eq. (3-6) has a 50% breakdown point. The reason for standardizing the residuals by their scale τ in Eq. (3-6) is to ensure that the M-estimator will be scale equivariant meaning that changing the scale of the residuals does not change the estimates. Using the initial estimates for the left and right singular vectors, an M-scale estimate would be the solution of Eq. (3-6). In this chapter, we use the Tuckey bisquare function f_0 defined as

$$f_0\left(\frac{r}{\tau}\right) = min\left\{1,1-\left(1-\frac{\mathrm{r}}{\mathrm{\tau}}\right)^3\right\} \tag{3-7}$$

There are several robust loss functions, such as Welsch, Cauchy, Huber etc. However, the performance of the MM estimator is not significantly affected by the choice of the loss function as long as it satisfies Properties 1 to 3.

3.3.3 Analytical Solution for the RGSVD Problem

The MM estimates are defined as the solution to the single-component RGSVD problem defined as

$$\text{Minimize} \sum_{i=1}^{m} f_1\left(\frac{\left\|\mathbf{x}_i-u_i\mathbf{v}^T\right\|_C^2}{\tau}\right) \tag{3-8}$$

St:

$$\mathbf{v}^T\mathbf{C}\mathbf{v}=1$$

where $f_1(\cdot) \leq f_0(\cdot)$, and must satisfy the first and second properties stated in Subsesction 3.3.2.

Theorem 3-1: Assume that f_0 and f_1 are selected appropriately, the optimal solutions to the RGSVD problem in Eq. (3-8) denoted by $\mathbf{u}^*$ and $\mathbf{v}^*$ (left and right singular vectors, respectively) are computed as

$$\mathbf{v}^* = \frac{\sum_{i=1}^{m} u_i\mathbf{x}_i\omega_i}{\sqrt{\left[\sum_{i=1}^{m} u_i\mathbf{x}_i\omega_i\right]^T\mathbf{C}\sum_{i=1}^{m} u_i\mathbf{x}_i\omega_i}} \;\&\; u_i^* = \frac{\mathbf{x}_i\mathbf{C}\mathbf{v}}{\sqrt{\sum_{i=1}^{m}(\mathbf{x}_i\mathbf{C}\mathbf{v})^2}};\; i\; = 1,2,\dots,m, \tag{3-9}$$

where $\omega_i = \frac{df_1(r_i)}{dr_i};\; i\; = 1,2,\dots,m$. The proof of the theorem can be found in Appendix 3-A. There is no closed-form solution to the RGSVD problem; however, one possible way is to assume that one of the singular vectors is known and solve the problem for the other singular vector. This procedure can be repeated until a convergence criterion is satisfied. In the next subsection, a blockwise power algorithm with deflation approach is proposed for obtaining the RGSVD problem's solutions.

3.3.4 Computational Algorithm

A blockwise algorithm is proposed to obtain the solutions for the RGSVD problem. The power method with deflation approach is used to estimate the singular vectors (Golub and Van loan 1996). The idea of the deflation approach is that after estimating the singular vectors for the lth component, the original matrix of data $\mathbf{X}$ is approximated by matrix $\widehat{\mathbf{X}} = \mathbf{u}_l {\mathbf{v}_l}^T$, and the singular vectors are obtained for the residual matrix $\mathbf{X} - \widehat{\mathbf{X}}$. It is straightforward to show that this will give the $(l+1)$th singular vectors.

Lemma 3-1: The objective function in the RGSVD problem in Eq. (3-8) is a nonconvex function. Lemma 1 implies that the power algorithm converges to a local optimum solution (see Appendix 3-B). Figure 3-4 demonstrates the proposed algorithm for solving the RGSVD problem.

```
Initialize the algorithm using $\mathbf{u}_0$ and $\mathbf{v}_0$ computed based on the repeated median method introduced in Subsection 3.1
For $l = 1, 2, \dots, L$ components:
a. Set $\widehat{\mathbf{X}} = \mathbf{X}$, and repeat until $e_1^k < \varepsilon, e_2^k < \varepsilon$, and $e_3^k < \varepsilon$ for each iteration $k$
        1. Compute $r_i^k = \|\mathbf{x}_i - u_i \mathbf{v}^T\|_C^2; \; i = 1,2,\dots,m$
        2. Obtain $\tau^k$ as the solution of $\frac{1}{m}\sum_{i=1}^m f_0\left(\frac{r_i^k}{\tau^k}\right) = \delta$
        3. Calculate $\omega_i^k$'s ; $i = 1,2 \; \dots \; m$
        4. Update the singular vectors as
        $\mathbf{v}^k = \frac{\sum_{i=1}^m u_i^k \mathbf{x}_i \omega_i^k}{\sqrt{\left[\sum_{i=1}^m u_i^k \mathbf{x}_i \omega_i^k\right]^T \mathbf{C} \sum_{i=1}^m u_i^k \mathbf{x}_i \omega_i^k}}$ & $u_i^k = \frac{\mathbf{x}_i \mathbf{C} \mathbf{v}^k}{\sqrt{\sum_{i=1}^m \left(\mathbf{x}_i \mathbf{C} \mathbf{v}^k\right)^2}}; \; i = 1, 2, \dots, m$
        5. Compute $e_1^k = |\tau^k - \tau^{k-1}|$, $e_2^k = \|\mathbf{v}^k - \mathbf{v}^{k-1}\|_2$, and $e_3^k = \|\mathbf{u}^k - \mathbf{u}^{k-1}\|_2$.
b. $\widehat{\mathbf{X}} = \widehat{\mathbf{X}} - \mathbf{u}_l {\mathbf{v}_l}^T$
```

Figure 3-4: Blockwise power algorithm with deflation for solving RGSVD problem

3.4 Extension to Robust Generalized High-order Decomposition (RGHOSVD)

As discussed, one drawback of using the stack-up RGSVD method is that the resulting matrix $\mathbf{X} \in \mathbb{R}^{m\times np}$ has a large number of columns. Furthermore, the spatial correlation structure within each image is broken when vectorizing and stacking up the images in one matrix. To avoid such complications, several authors suggest using a high-order array or a tensor to represent the data. In this case, the RGSVD method must be extended to be applicable to the high-order arrays. A brief introduction to multilinear algebra is given in Subsection 3.4.1. The robust generalized high-order SVD method and an algorithm to obtain the solutions are provided in Subsection 3.4.2.

3.4.1. Basic High-Order Notation and Algebraic Operations

A tensor is a high-order array used to represent a dataset with more than two dimensions. Each dimension of a tensor is called a mode or order. For example, a vector is a 1st order tensor, and a matrix is a 2nd order tensor. An Nth order tensor is denoted by $\mathcal{X}^{I_1\times I_2\times\ldots\times I_N}$ with I_n giving the number of elements in mode n. In the rolling-bar dataset, the images can be all represented by a 3rd order tensor denoted by $\mathcal{X}^{n\times p\times m}$ where n is the number of rows, p is the number of columns in the image, and m denotes the number of images collected in Phase I. Sometimes it is useful to work with a subarray of a tensor. For instance, the $(N-1)$th order subarrays of tensor $\mathcal{X}^{I_1\times I_2\times\ldots\times I_N}$ are denoted by $\mathcal{X}^{:,:,\ldots,:,i_N}$; $i_N = 1,2,\ldots,I_N$, where a colon means that all the elements of a specific mode are included in the subarray. Some useful algebraic operations are (Kolda and Bader 2009):

Tensor to matrix multiplication: the mode-n product of a tensor $\mathcal{X}^{I_1\times I_2\times\ldots\times I_N}$ by a matrix $\mathbf{U} \in \mathbb{R}^{J\times I_n}$ is denoted by $\mathcal{Z} = \mathcal{X} \times_n \mathbf{U}$. Each element of tensor $\mathcal{Z} \in \mathbb{R}^{I_1\times I_2\times\ldots\times I_{n-1}\times J\times I_{n+1}\times\ldots\times I_N}$ is obtained as

$$z_{i_1 i_2 \ldots i_{n-1} j i_{n+1} \ldots i_N} = \sum_{i_n}^{I_n} x_{i_1 i_2 \ldots i_N} u_{j i_n} \tag{3-10}$$

where u_{ji_n} is the (j, i_n)th element of matrix $\mathbf{U}$.

Norm of a tensor: the norm of a tensor $\mathcal{X}^{I_1 \times I_2 \times \ldots \times I_N}$ is the square root of sum of squares of all the elements $x_{i_1 i_2 \ldots i_N}$; $i_\mathrm{n} = 1, 2, \ldots, I_n$, and $n = 1,2, \ldots, N$. That is,

$$\|\mathcal{X}\| = \sqrt{\sum_{i_1}^{I_1} \sum_{i_2}^{I_2} \ldots \sum_{i_\mathrm{N}}^{I_N} x_{i_1 i_2 \ldots i_N}^2} \tag{3-11}$$

The inner product of tensors: the inner product of two tensors $\mathcal{X}^{I_1 \times I_2 \times \ldots \times I_N}$ and $\mathcal{Y}^{I_1 \times I_2 \times \ldots \times I_N}$ is the sum of the elementwise products of all the elements, i.e.

$$\langle \mathcal{X}, \mathcal{Y} \rangle = \sum_{i_1}^{I_1} \sum_{i_2}^{I_2} \ldots \sum_{i_\mathrm{N}}^{I_N} x_{i_1 i_2 \ldots i_N}\, y_{i_1 i_2 \ldots i_N} \tag{3-12}$$

3.4.2. Robust Generalized High-Order SVD Solutions

The GHOSVD method is proposed by Allen et al. (2012) and it is a generalization of the GSVD method. The image data can be represented by a 3rd order tensor; however, in this subsection, the RGHOSVD method is developed for a generic Nth order tensor. For example, when the color of an image carries important information regarding the quality of the product, in addition to the spatial resolution of the pixels, it is useful to consider the color as the third mode, which forms a 4th order tensor of image data. A general Nth order tensor of image data $\mathcal{X}^{I_1 \times I_2 \times \ldots \times I_N}$ is modeled as

$$\mathcal{X} = \mathcal{G} \times_1 \mathbf{V}^{(1)} \times_2 \mathbf{V}^{(2)} \times_3 \ldots \times_N \mathbf{V}^{(N)} + \mathcal{E} \tag{3-13}$$

Where $\mathcal{G}^{L \times L \times \ldots \times L}$ is a core tensor, $\mathbf{V}^{(n)} \in \mathbb{R}^{I_n \times L}$ $(n = 1,2, \ldots, N)$ is the factor matrix for the nth mode with L as the number of components, and $\mathcal{E}^{I_1 \times I_2 \times \ldots \times I_N}$ is the tensor of noise components. It is assumed that $\mathcal{E}^{I_1 \times I_2 \times \ldots \times I_N}$ has a tensor-variate distribution with tensor mean zero and mode-n covariance matrix $\Delta^{(n)}$; $n = 1,2, \ldots, N$. The last mode in tensor $\mathcal{X}$ represents independent samples of images. That is, the mode-N covariance matrix $\Delta^{(N)}$ is a diagonal matrix. the single-component RGHOSVD problem is defined as

$$\text{Minimize } f\left(\frac{\left\|\mathcal{X}-\mathbf{v}^{(1)}\circ\mathbf{v}^{(2)}\circ\ldots\circ\mathbf{v}^{(N)}\right\|^2_{\mathbf{C}_1,\mathbf{C}_2,\ldots,\mathbf{C}_N}}{\xi}\right) \tag{3-14}$$

S.t:

$$\mathbf{v}^{(n)^T}\mathbf{C}_n\mathbf{v}^{(n)} = 1;\, n = 1, 2, \ldots, N-1$$

where $\mathbf{v}^{(n)} \in \mathbb{R}^{I_n}$ is the mode-*n* singular vector, $\mathbf{C}_n \in \mathbb{R}^{I_n\times I_n}$ is the quadratic operator for mode *n*, and ξ is a robust M-estimate of the residuals' scale. In case the last mode represents different sample images, the quadratic operator $\mathbf{C}_N$ can be set to the identity matrix since different images are assumed to be independent. The objective function in Eq. (3-14) can be rewritten as $\sum_{i_N=1}^{I_N} f\left(\frac{\left\|\mathcal{X}_{::\cdots:i_N}-v_{i_N}^{(N)}\mathbf{v}^{(1)}\circ\mathbf{v}^{(2)}\circ\ldots\circ\mathbf{v}^{(N-1)}\right\|^2_{\mathbf{C}_1,\mathbf{C}_2,\ldots,\mathbf{I}}}{\xi}\right)$, where $\mathcal{X}_{::\cdots:i_n}$ is a $(N-1)$th order subarray of tensor $\mathcal{X}$, and $v_{i_N}^{(N)}$ is the i_Nth element of the mode-N singular vector $\mathbf{v}^{(N)}$.

Theorem 3-2: The solution to the RGHOSVD problem in Eq. (3-14) can be obtained as (See Appendix 3-B). Let's define $\mathbf{b} \in \mathbb{R}^{I_n}$ as

$$\mathbf{b} = \sum_{i_N=1}^{I_N} \mathcal{X}_{::\cdots:i_N} \times_1 \mathbf{C}_1\mathbf{v}^{(1)} \times_2 \ldots \times_{n-1} \mathbf{C}_{n-1}\mathbf{v}^{(n-1)} \times_{n+1} \mathbf{C}_{n+1}\mathbf{v}^{(n+1)} \ldots \times_N \omega_{i_N} v_{i_N}^{(N)}$$

$$\mathbf{v}^{(n)} = \frac{\mathbf{b}}{\sqrt{\mathbf{b}^T\mathbf{C}_n\mathbf{b}}} \;\&\; v_{i_N}^{(N)} = \frac{\mathcal{X}_{::\cdots:i_N}\times_1\mathrm{C}_1\mathrm{v}^{(1)}\times_2\ldots\times_{N-1}\mathrm{C}_{N-1}\mathrm{v}^{(N-1)}}{\sqrt{\sum_{i_N=1}^{I_N}\left[\mathcal{X}_{::\cdots:i_N}\times_1\mathrm{C}_1\mathrm{v}^{(1)}\times_2\ldots\ldots\times_{N-1}\mathrm{C}_{N-1}\mathrm{v}^{(N-1)}\right]^2}} \tag{3-15}$$

$$i_N = 1, 2, \ldots, I_N;\, n = 1, 2, \ldots, N-1$$

Similar to the RGSVD method, a blockwise algorithm based on tensor power method with deflation is suggested to compute the singular vectors. A high breakdown-point robust solution is needed to initialize the algorithm; hence, in this section, the repeated median method is extended.

Consider a single-component ($L = 1$) form of Eq. (3-13) expressed as $\mathcal{X} = \mathbf{v}^{(1)} \circ \mathbf{v}^{(2)} \ldots \circ \mathbf{v}^{(N)} + \mathcal{E}$. Each element of tensor $\mathcal{Y}$ is given by

$$x_{i_1 i_2 \ldots i_N} = v_{i_1}^{(1)} v_{i_2}^{(2)} \ldots v_{i_{n-1}}^{(n-1)} v_{i_{n+1}}^{(n+1)} \ldots v_{i_N}^{(N)} v_{i_n}^{(n)} + \varepsilon_{i_1 i_2 \ldots i_N} \tag{3-16}$$

$$i_n = 1, 2, \ldots, I_n; n = 1, 2, \ldots, N$$

Provided that all singular vectors are known except for $\mathbf{v}^{(n)}$, the model in Eq. (3-16) turns into a simple linear regression model with unknown coefficient $v_{i_n}^{(n)}$. The i_nth element of the mode-n robust initial singular vector is defined as

$$v_{i_n}^{(n)} =$$

$$median_{\substack{i_m=1,2,\ldots,I_m \\ m\neq n=1,2,\ldots,N}} \left\{ median_{\substack{j_m\neq i_m=1,2,\ldots,I_m \\ m\neq n=1,2,\ldots,N}} \left\{ \frac{x_{i_1 i_2 \ldots i_n \ldots i_N} - x_{j_1 j_2 \ldots i_n \ldots j_N}}{v_{i_1}^{(1)} v_{i_2}^{(2)} \ldots v_{i_{n-1}}^{(n-1)} v_{i_{n+1}}^{(n+1)} \ldots v_{i_N}^{(N)} - v_{j_1}^{(1)} v_{j_2}^{(2)} \ldots v_{j_{n-1}}^{(n-1)} v_{j_{n+1}}^{(n+1)} \ldots v_{j_N}^{(N)}} \right\} \right\}$$

$$i_n = 1,2 \ldots I_n; n = 1,2, \ldots, N \tag{3-17}$$

where $v_{i_n}^{(n)}$ is the initial, robust estimate of the i_nth element of the mode-n singular vector $\mathbf{v}^{(n)}; n = 1,2, \ldots, N$. The initial solution is used to start the algorithm provided in Figure 3-5. One advantage of the blockwise tensor power algorithm over the regular power algorithm (RGSVD problem) is having faster running time. It is shown in Appendix 3-D that for each iteration of the algorithm, the computational complexity is of order $O(n^2pm + p^2nm)$, which is a fourth-degree polynomial. On the other hand, the complexity of the blockwise power algorithm is a fifth-degree polynomial with order $O(n^2p^2m)$. Although the number of iterations of the algorithm depends on some parameters, like the signal intensity, the amount of noise in the data etc., in similar conditions, the blockwise tensor power algorithm terminates with a higher speed (see Section 3.5, Table 3-2).

```
Initialize the algorithm with $\mathbf{v}_0^{(n)}$; $n = 1,2,\dots,N$,
For $l = 1,2,\dots,L$, components:
a. Set $\widehat{\mathcal{X}} = \mathcal{X}$, and repeat until $e^k < \varepsilon, e_n^k < \varepsilon; n = 1,2,\dots,N$, for each iteration $k$
        1. Compute $r_{i_N}^k = \left\| \mathcal{X}_{::\cdots:i_N} - v_{i_N}^{(N)} \mathbf{v}^{(1)} \circ \mathbf{v}^{(2)} \dots \circ \mathbf{v}^{(N-1)} \right\|_{C_1,C_2,\dots,I}^2$; $i_N = 1,2,\dots,I_N$
        2. Obtain $\xi^k$ as the solution of $\frac{1}{I_N}\sum_{i_N=1}^{I_N} f_0\left(\frac{r_{i_N}^k}{\xi^k}\right) = \delta$.
        3. Calculate $\omega_{i_N}^k$'s ; $i_N = 1,2,\dots,I_N$
        4. Update the singular vectors as
                $\mathbf{b} = \sum_{i_N=1}^{I_N} \mathcal{X}_{::\cdots:i_N} \times_1 \mathbf{C}_1\mathbf{v}_k^{(1)} \times_2 \dots \times_{n-1} \mathbf{C}_{n-1}\mathbf{v}_k^{(n-1)} \times_{n+1} \mathbf{C}_{n+1}\mathbf{v}_k^{(n+1)} \dots \times_N \ \omega_{i_N}^k v_{ki_N}^{(N)}$
                $\mathbf{v}_k^{(n)} = \frac{\mathbf{b}}{\sqrt{\mathbf{b}^T\mathbf{C}_n\mathbf{b}}}$ & $v_{ki_N}^{(N)} = \frac{\mathcal{X}_{::\cdots:i_N} \times_1 \mathbf{C}_1\mathbf{v}_k^{(1)} \times_2 \dots \times_{N-1} \mathbf{C}_{N-1}\mathbf{v}^{(N-1)}}{\sqrt{\sum_{i_N=1}^{I_N}\left[\mathcal{X}_{::\cdots:i_N} \times_1 \mathbf{C}_1\mathbf{v}_k^{(1)} \times_2 \dots \times_{N-1} \mathbf{C}_{N-1}\mathbf{v}_k^{(N-1)}\right]^2}}$
                $n = 1,2,\dots,N-1; i_N = 1,2,\dots,I_N$
        5. Compute $e^k = |\xi^k - \xi^{k-1}|$, $e_n^k = \left\|\mathbf{v}_k^{(n)} - \mathbf{v}_{k-1}^{(n)}\right\|_2$; $n = 1,2 \dots N$
b. $\widehat{\mathcal{X}} = \mathcal{X} - \mathbf{v}_l^{(1)} \circ \mathbf{v}_l^{(2)} \circ \dots \circ \mathbf{v}_l^{(N)}$
```

Figure 3-5: The blockwise tensor power method with deflation for solving RGHOSVD problem

3.5 A Simulation Study

In this section, first, the accuracy of the estimates is appraised and compared to that of other methods when the noise in the images is correlated and there are outlier images in the dataset. Second, the monitoring performance of the proposed methodologies in detecting the defective images is compared with some frequently-used methods in the literature.

3.5.1 A Performance Comparison to Evaluate the Accuracy of the Estimates

The accuracy of singular vectors estimates for five decomposition methods is compared under different scenarios. The methods that are considered for estimation are SVD, GSVD, Robust SVD (Marrona 2005), RGSVD and RGHOSVD. The image matrix $\mathbf{Y}$ is simulated using Eq. (3-1) with one component; i.e. $L = 1$. The left singular vector $\boldsymbol{\alpha} \in \mathbb{R}^{30}$ is generated using a sinusoidal signal $\boldsymbol{\alpha} = \cos(\pi\mathbf{x}_l)$ with $\mathbf{x}_l$ taking $n = 30$ equidistant values in $(0,1)$. For the right singular vector $\boldsymbol{\beta} \in \mathbb{R}^{30}$, a triangular signal is used to generate the vector $\boldsymbol{\beta} = \begin{cases} \frac{|\mathbf{x}_r - 0.5|}{0.5} & -0.5 < \mathbf{x}_r < 0.5 \\ 0 & \mathrm{o.w} \end{cases}$. To construct the correlated noise matrix $\boldsymbol{\epsilon} \in \mathbb{R}^{30\times 30}$, first, a white noise matrix $\mathbf{Z} \in \mathbb{R}^{n\times p}$ is simulated with independent elements z'_{ij} $(i = 1,2,\dots,n =$

$30;\ j = 1,2, \dots, p = 30)$ following normal distribution with mean zero and variance $\sigma^2 = 0.01$. Second, the noise matrix $\boldsymbol{\epsilon}$ is computed as $\boldsymbol{\epsilon} = \Delta^{\frac{1}{2}}\mathbf{Z}\boldsymbol{\Sigma}^{\frac{1}{2}}$, where $\Delta^{\frac{1}{2}}$ and $\boldsymbol{\Sigma}^{\frac{1}{2}}$ are any matrix square roots (Cholesky decomposition) of Δ and $\boldsymbol{\Sigma}$, respectively.

To model the correlation structure in the noise matrix, the Toeplitz correlation structure is admitted. In the Toeplitz correlation model, the assumption is that the closer the two vectors lie in a subspace, the higher the correlation between these vectors is. This structure is employed to have a more realistic simulation of the images where the noise components are spatially correlated, and the correlation is high in a small neighborhood. The elements of the Toeplitz matrix $\mathbf{H} \in \mathbb{R}^{p\times p}$ ($\mathbf{H} \in \mathbb{R}^{n\times n}$) are defined as

$$h_{ij} = h_{i+1j+1} = h_{i-j};\ h_0 = 1; i,j = 1,2 \dots p\ (i,j = 1,2 \dots n), \tag{3-18}$$

Using the Toeplitz matrix calculated in in Eq. (3-18), we obtain the column-wise covariance matrix as

$$\boldsymbol{\Sigma} = \theta^{\mathbf{H}} = \begin{bmatrix} \theta & \theta^2 & \dots & \theta^p \\ \theta^2 & \theta & \dots & \theta^{p-1} \\ \vdots & \vdots & \ddots & \vdots \\ \theta^p & \theta^{p-1} & \dots & \theta \end{bmatrix} \tag{3-19}$$

where $0 \le \theta \le 1$ controls the amount of autocorrelation in the noise. The row-wise covariance matrix Δ is generated in a similar way. Figure 3-6 shows two simulated images before and after adding the noise shown in Figure 3-6 (a) and Figure 3-6 (b), respectively.

(a) without noise (b) after adding noise

Figure 3-6: Simulated images

Fifty images ($m = 50$) are simulated using Eq. (3-1). All the images are vectorized and stacked up in matrix $\mathbf{X} \in \mathbb{R}^{50\times 900}$. The SVD, RSVD, GSVD and RGSVD methods are applied to decompose matrix $\mathbf{X}$ and estimate the right singular vectors. For the high-order setting, a 3rd order tensor $\mathcal{X}^{30\times 30\times 50}$ is used to represent the images. The RGHOSVD, GHOSVD and high-order SVD (HOSVD) are the high-order decomposition methods utilized to estimate Mode-1 and Mode-2 singular vectors.

To account for the correlated noise in the images, the row-wise and column-wise quadratic operators denoted by $\mathbf{Q}$ and $\mathbf{R}$ **(**$\mathbf{C}_1$ and $\mathbf{C}_2$ for the high-order decomposition**)** must be estimated using faultless images with clear visibility. One important assumption is that the structure (the distribution and the distributional parameters) of the noise components does not dramatically change from one image to another. In this chapter, it is assumed that all the images are taken at almost the same environmental conditions by one hot-eye camera from a fixed angle, distance and direction with respect to the subject. All images' properties on the quality, size, brightness, etc., are previously set in the camera and kept unchanged for all images. These conditions help ensure all the images share a reasonably similar noise structure. The quadratic operators are both estimated using the γ-exponential covariance functions with parameters determined to maximize the explained variability by the first component. The ijth element of the row-wise quadratic operator $q_{ij}; i,j = 1,2,\dots,n = 30$, estimated by the γ-exponential covariance function is defined as

$$\hat{q}_{ij} = e^{-\left(\frac{d_{ij}}{\eta}\right)^{\gamma}};\ i,j = 1,2,\dots,n = 30 \tag{3-20}$$

where d_{ij} is the distance between row i and row j in matrix $\mathbf{Y}$, γ controls the smoothness of the function and $\eta > 0$ is the scale parameter that can be adjusted to achieve a desired amount of explained variability by the first component. When $\gamma = 2$, the γ-exponential covariance function is called squared exponential function which is the most widely-used kernel function for estimating the covariance function (Rasmussen and Williams 2005). The column-wise quadratic operator is determined in a similar way as

$$\hat{r}_{st} = e^{-\left(\frac{d_{st}}{\nu}\right)^{\gamma}};\ s,t = 1,2,\dots,p = 30 \tag{3-21}$$

where $\hat{r}_{st}$ is the stth element of the quadratic operator $\mathbf{R}$, d_{st} is the distance between the sth and tth columns in matrix $\mathbf{Y}$, and ν is the scale parameter. The column-wise quadratic operator for matrix **X** is calculated as

$$\hat{\mathbf{C}} = \hat{\mathbf{Q}} \otimes \hat{\mathbf{R}} \tag{3-22}$$

In this chapter, γ is set to 2 to have smooth quadratic operators, and the scale parameters are set to $\eta = \nu = 10$ resulting in at least 70% explained variability by the first component of the GSVD, RGSVD, GHOSVD and RGHOSVD methods.

As pointed out, the white noise matrix $\mathbf{Z}$ is simulated using a multivariate normal distribution with mean vector zero and covariance matrix $\sigma^2\mathbf{I}$. To generate outlier images, the white noise matrix is simulated with different mean vector and/or covariance matrix. If P is the percentage of outlier images (contamination percentage) in the m collected images, then each white noise matrix $\mathbf{Z}$ is obtained by sampling (n times) from a p-dimensional multivariate normal mixture distribution defined as

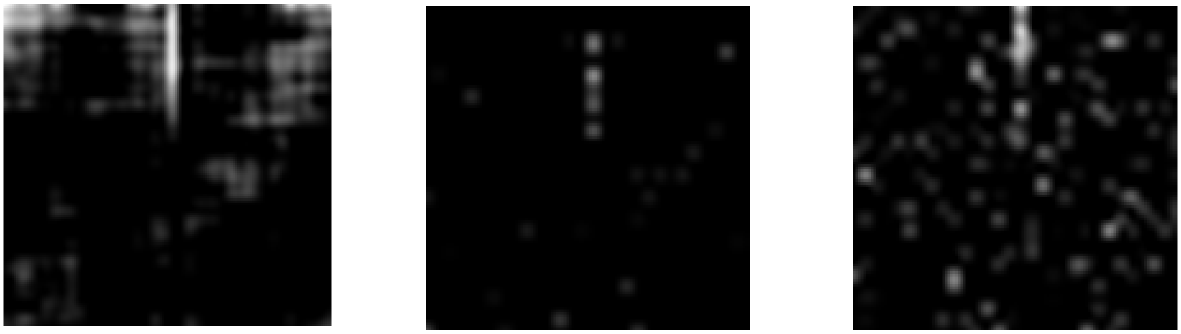
$$P\mathcal{N}_p(\boldsymbol{\mu},\ \delta^2\sigma^2\mathbf{I}) + (1-P)\mathcal{N}_p\left(\vec{\mathbf{0}}, \sigma^2\mathbf{I}\right) \tag{3-23}$$

Where $P = 0.1$, $\boldsymbol{\mu} = \vec{\mathbf{0}} + \lambda\vec{\mathbf{1}}$, $\delta = 1.2$, $\lambda = \{\pm 0.5, \pm 1\}$, $\vec{\mathbf{0}}$ and $\vec{\mathbf{1}}$ are p-dimensional vectors of zeros and ones, respectively. The structured noise matrix containing the outliers is calculated as $\boldsymbol{\epsilon} = \Delta^{\frac{1}{2}}\mathbf{Z}\boldsymbol{\Sigma}^{\frac{1}{2}}$. Figure 3-7 gives outlier images generated under different scenarios. As it is noticeable, negative shifts in the mean of the noise distribution underexposes the images (Figure 3-7 (b) and Figure 3-7 (c)) while the positive mean drifts overexpose the image as shown in Figure 3-7 (e).

After generating the image data, all methods are applied to the data and the singular vectors are estimated. The accuracy of the estimates is evaluated using the canonical angle metric. Let $\mathcal{W}$ and $\mathcal{T}$ denote two subspaces defined over real or complex numbers. If $\dim(\mathcal{W}) = k$, $\dim(\mathcal{T}) = l$, and $q = \min(k, l)$, the canonical angles between the subspaces $\mathcal{W}$ and $\mathcal{T}$ is given by

$$\varphi_i \in \left[0, \frac{\pi}{2}\right] = min_{\mathbf{w}_i \in \mathcal{W}}\left\{min_{\mathbf{t}_i \in \mathcal{T}}\left\{\arccos\left(\frac{\mathbf{w}_i^T\mathbf{t}_i}{\sqrt{\mathbf{w}_i^T\mathbf{w}_i}\sqrt{\mathbf{t}_i^T\mathbf{t}_i}}\right)\right\}\right\};\ i = 1,2,\dots,q \tag{3-24}$$

where $\mathbf{w}_i$ and $\mathbf{t}_i$ are called principle vectors. Based on this metric, $cos(\varphi_i) \in [0,1]$ is used as a criterion to measure the closeness of the estimated factor matrix $\widehat{\mathbf{V}}$ to the true factor matrix $\mathbf{V}$. The more accurate the estimates are, the larger $cos(\varphi_i)$ is.



(a) $\lambda = 0, \delta = 1$ (normal image) (b) $\lambda = -1, \delta = 1$ (c) $\lambda = -1, \delta = 1.2$

(d) $\lambda = 0, \delta = 1.2$ (e) $\lambda = 1.5, \delta = 1.2$

Figure 3-7: Simulated outlier images

Note that the high-order decomposition methods are able to estimate the left and right singular vectors denoted by $\hat{\mathbf{v}}_l^{(1)}$ and $\hat{\mathbf{v}}_l^{(2)}$, respectively. However, the low-order decomposition methods estimate the right singular vector $\hat{\mathbf{v}}_l$ corresponding to the matrix of stacked-up images $\mathbf{X}$. For the high-order methods, $\hat{\mathbf{v}}_l$ is computed as $\hat{\mathbf{v}}_l = \hat{\mathbf{v}}_l^{(1)} \otimes \hat{\mathbf{v}}_l^{(2)}$ to facilitate the comparison between the low-order and high-order decomposition methods.

Table 3-1: Canonical angles computed for estimated right singular vector with parameters $\lambda = \pm 1$; $P = 0.1$, and $\theta = 0.9$

	Signal intensity d				
Method	**0.8**	**0.9**	**1**	**1.5**	**2**
SVD	0.1224 (0.0103)	0.1598 (0.0174)	0.1964 (0.0199)	0.9518 (0.0049)	0.9821 (0.0010)
GSVD	0.2082 (0.0189)	0.2626 (0.0243)	0.3096 (0.0254)	0.5144 (0.0330)	0.7820 (0.0278)
RSVD	0.1800 (0.0159)	0.3229 (0.0329)	0.4667 (0.0327)	0.9660 (0.0023)	0.9847 (0.0007)
RGSVD	0.9168 (0.0035)	0.9347 (0.0024)	0.9507 (0.0020)	0.9773 (0.0008)	0.9866 (0.0006)
HOSVD	0.2254 (0.0529)	0.5356 (0.0656)	0.6720 (0.0619)	0.8745 (0.0450)	0.9565 (0.0274)
GHOSVD	0.0433 (0.0108)	0.1184 (0.0375)	0.4330 (0.0630)	0.8992 (0.0410)	0.9607 (0.0255)
RGHOSVD	**0.9823** (0.0018)	**0.9849** (0.0015)	**0.9891** (0.0011)	**0.9943** (0.0006)	**0.9967** (0.0003)

Table 3-1 shows the canonical angles $\left(cos(\varphi)\right)$ computed for both low-order and high-order decomposition methods used to estimate the first right singular vector $\hat{\mathbf{v}}$. The most effective method in each condition (each signal intensity) is highlighted in Table 3-1. Both the RGHOSVD and RGSVD methods outperform other decomposition methods in providing highly accurate estimates for the right singular vector. The RGHOSVD method is preferable to RGSVD particularly when the signal intensity is low. For high values of signal intensity ($d \geq 2$ here) all methods perform similarly with RGHOSVD slightly outperforming the others.

As discussed, another important aspect of every algorithm is time spent to generate the results. To compare convergence rate of the high-order-based algorithm with the blockwise power method (low-order-based algorithm) the running time of the algorithms are computed and compared. Table 3-2 gives the average running time ratio $t_r = \frac{t_L}{t_H}$ for different image dimensions and signal intensities, where t_L is the running time for the RGSVD method, and t_H represents the time spent to obtain the solutions using the RGHOSVD method.

Table 3-2: Average running time ratio $\mathbf{t_r}$

	Signal intensity d	
$(\boldsymbol{n}, \boldsymbol{p}, \boldsymbol{m})$	0.8	2
(60,60,50)	1.3	3.5
(70,70,50)	5.8	2.6
(80,80,50)	10.1	3.6
(100,100,50)	18.3	5.02
(150,150,50)	5.8	6
(60,60,100)	3.2	1.4
(70,70,100)	4.1	2.6
(80,80,100)	6.2	3.1
(100,100,100)	6.7	2.7
(150,150,100)	4.2	3.2

As it is notable in Table 3-2, the blockwise tensor power algorithm converges faster than the blockwise power algorithm in all cases. This is more obvious when the signal intensity is relatively weak ($d = 0.8$).

3.5.2 Monitoring Performance Comparison for Detecting Defective Images

In this subsection, the monitoring performance of several image-based monitoring methods is evaluated and compared to the proposed methodologies. Using Eq. (3-1), $m = 50$

faultless 30×30 images are simulated (see Figure 3-7 (a) for a faultless image). Similar to Subsection 3.5.1, 10% of the data are assumed to be contaminated i.e. $P = 0.1$. The outlier images are generated by changing λ from 0 to ± 0.5 and/or inflating the variability of noise components i.e. $\delta = 1.2$. Furthermore, θ is set to 0.9, $\sigma^2 = 0.01$, and $d = 5$. The in-control dataset is used in Phase I to estimate the covariance matrix of the baseline image features. For the out-of-control data, three different types of defects are considered in order to have more realistic defect types which might exist on the surface of billets in the rolling-bar dataset. Figures 3-8 (b) ~ 3-8 (d) give transverse, longitudinal and rectangular defects, respectively.

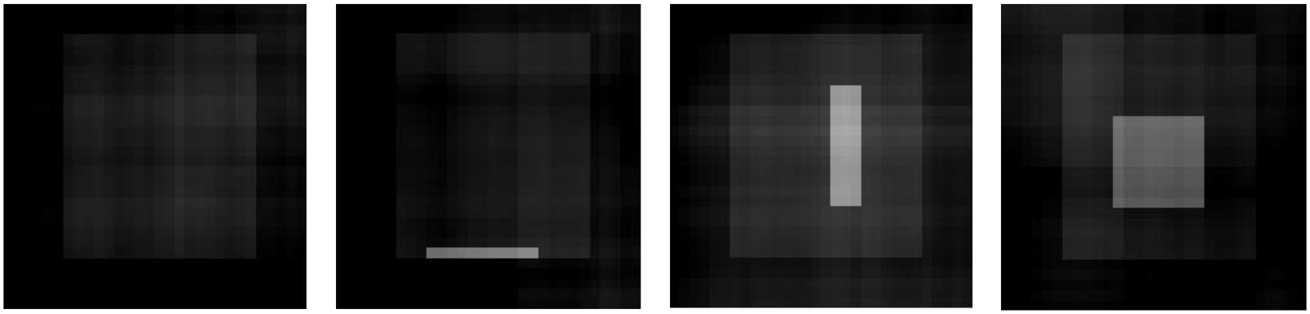

(a) Faultless image (b) Transverse defect (c) Longitudinal defect (d) Rectangular defect

Figure 3-8: Simulated defective and faultless images

For the decomposition methods, the features can be computed by projecting the centralized matrix or tensor of the data using the estimated singular vectors. For example, the lth extracted feature can be defined as

$$\mathbf{lf}_l = \mathbf{X}\hat{\mathbf{v}}_l; l = 1,2,\dots,L, \qquad (3\text{-}25)$$

where $\mathbf{lf}_l \in \mathbb{R}^m$ is the lth extracted feature, $\hat{\mathbf{v}}_l$ is the lth estimated right singular vector using Phase I image data represented by matrix $\mathbf{X}$, and L is the number of components. The extracted features have mean vector zero and diagonal covariance matrix $\sum_{\text{lf}} \in \mathbb{R}^{L \times L}$ with the lth diagonal element defined as the sample variance of $\mathbf{lf}_l$. For the high-order decomposition methods, the features are computed as

$$hf_l = \mathcal{X}_{::i_3} \times_1 \hat{\mathbf{v}}_l^{(1)} \times_2 \hat{\mathbf{v}}_l^{(2)}; l = 1,2 \dots L, i_3 = 1, 2, \dots, I_3, \qquad (3\text{-}26)$$

where hf_l is the lth extracted feature, $\mathcal{X}_{::i_3}$ is the i_3th two-dimensional subarray of the tensor $\mathcal{X}$, $\hat{\mathbf{v}}_l^{(1)}$ and $\hat{\mathbf{v}}_l^{(2)}$ are the lth estimated left and right singular vectors, respectively.

In addition to the regular decomposition techniques, a control chart based on wavelet transform is used for image-based monitoring. A discrete wavelet transform (Daubechies 1990) is used to estimate the wavelet coefficients for each row of matrix **X** in Phase I. The PCA method is further applied to the matrix of coefficients and the PC scores are computed. A Hotelling's T^2 control chart can be used to monitor the mean vector of the extracted features. After ensuring that all points (m images in Phase I) are in-control statistically, we can monitor the ith image using T_i^2 statistic given by

$$T_i^2 = (\mathbf{g})^T {\textstyle\sum}_g^{-1}(\mathbf{g}); i = 1,2, \dots \qquad (3\text{-}27)$$

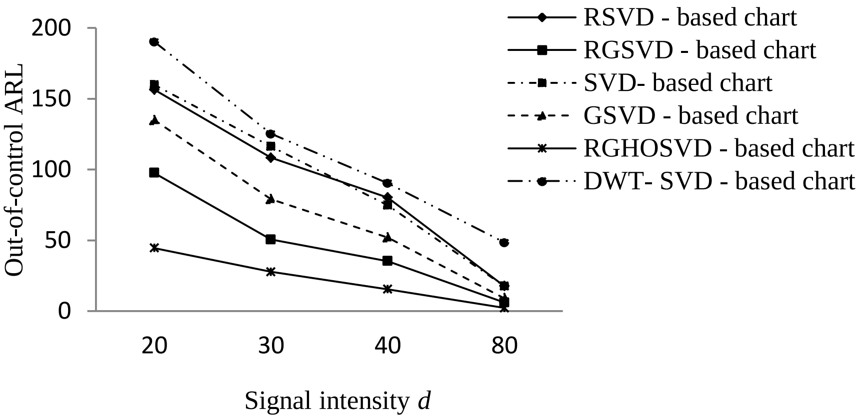
where $\mathbf{g} \in \mathbb{R}^{L\times 1}$ is the vector of features calculated in Phase II, and $\mathbf{\Sigma}_g$ is the covariance matrix of the baseline features estimated in Phase I. If a low-order decomposition method is used in Phase I, $\mathbf{g} = \mathbf{lf}$ is calculated using Eq. (3-25) and $\Sigma_g = \mathbf{\Sigma}_{lf}$. Otherwise, Eq. (3-26) is used to compute $\mathbf{g} = \mathbf{hf}$, and $\mathbf{\Sigma}_g = \mathbf{\Sigma}_{hf}$. Provided that the features follow a normal distribution, the upper control limit is computed in Phase I as $\text{UCL} = \frac{L(m)(m-2)}{(m-1)(m-1-L)} f_{\alpha,L,m-1-L}$, where f_{α,ν_1,ν_2} is the $(1-\alpha)$ 100th percentile of an F distribution with ν_1 and ν_2 degrees of freedom. In case the features do not follow a normal distribution, the upper control limit can be obtained using the empirical distribution of T_i^2's ; $i = 1, \dots, m$, computed in Phase I. When the number of components is $L = 1$, the T^2 chart boils down to a simple Shewhart control chart used for monitoring individual observations.

The monitoring performance is evaluated using the out-of-control average run length (ARL) criterion. The in-control ARL (ARL_0) is set to 200 for all methods. To calculate the ARL values, the simulations are performed in two phases. In Phase I, all the images are assumed to be faultless with some outlier images. The feature extraction methods are applied and the baseline features are obtained. The number of components is set to one $L = 1$, which results in at least 70% explained variability for all methods. The control limits are computed to achieve 0.5% Type-I error rate ($\mathrm{ARL}_0 = 200$). To achieve $\mathrm{ARL}_0 = 200$, the control limits for each chart are set to an arbitrary value and the run length variable (RL) is calculated for each simulation run which starts with generating a series of faultless images and terminates when an alarm is triggered by the associated control chart (See Table 3-3). After 10000 simulation runs, we calculate ARL as $\mathrm{ARL} = \frac{\sum_{j=1}^{10000} \mathrm{RL}_j}{10000}$. In Phase II, the simulated images contain one of the defects shown in Figures 3-8 (b) ~ 3-8 (d). The out-of-control ARL is calculated as described above.

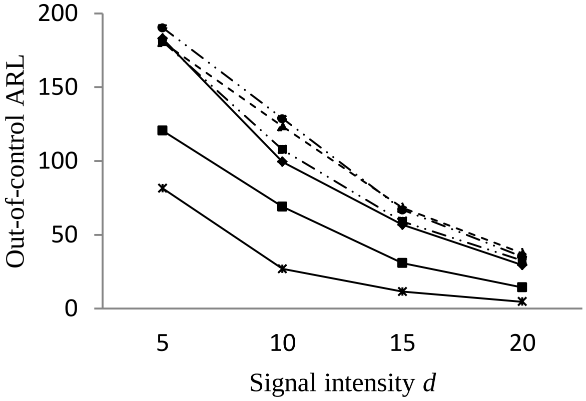
Table 3-3: Lower control limits (LCL) and upper control limits (UCL) obtained using simulation

Feature extraction method	**(LCL, UCL)**	**Calculated ARL_0**
RGHSVD	(-1.58, 1.58)	198.7
RGSVD	(-7, 7)	199.9
GSVD	(-2.8, 2.8)	202.3
RSVD	(-7.5, 7.5)	198.6
SVD	(-7.55, 7.55)	201.5
DWT - PCA	(-7.8, 7.8)	200.5

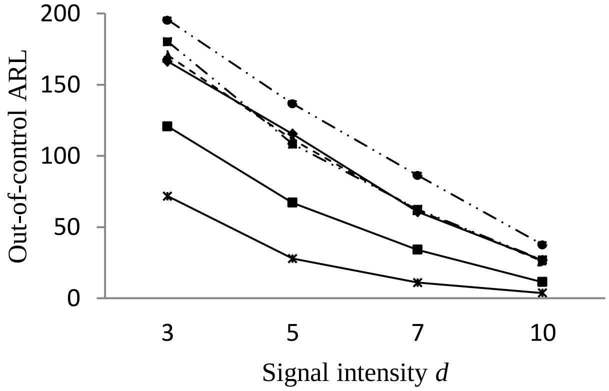
The out-of-control ARLs for detecting transverse, longitudinal and rectangular defects are plotted in Figure 3-9 (a) ~ Figure 3-9 (c), respectively. For all different types of defects, the RGHOSVD – based chart uniformly outperforms other methods in detecting the defects. After the RGHOSVD-based chart, the RGSVD – based method is superior to other methods for all types of defects. Clearly, all methods perform more similarly when the signal is strong.



(a) Transverse defect



(b) Longitudinal defect



(c) Rectangular defect

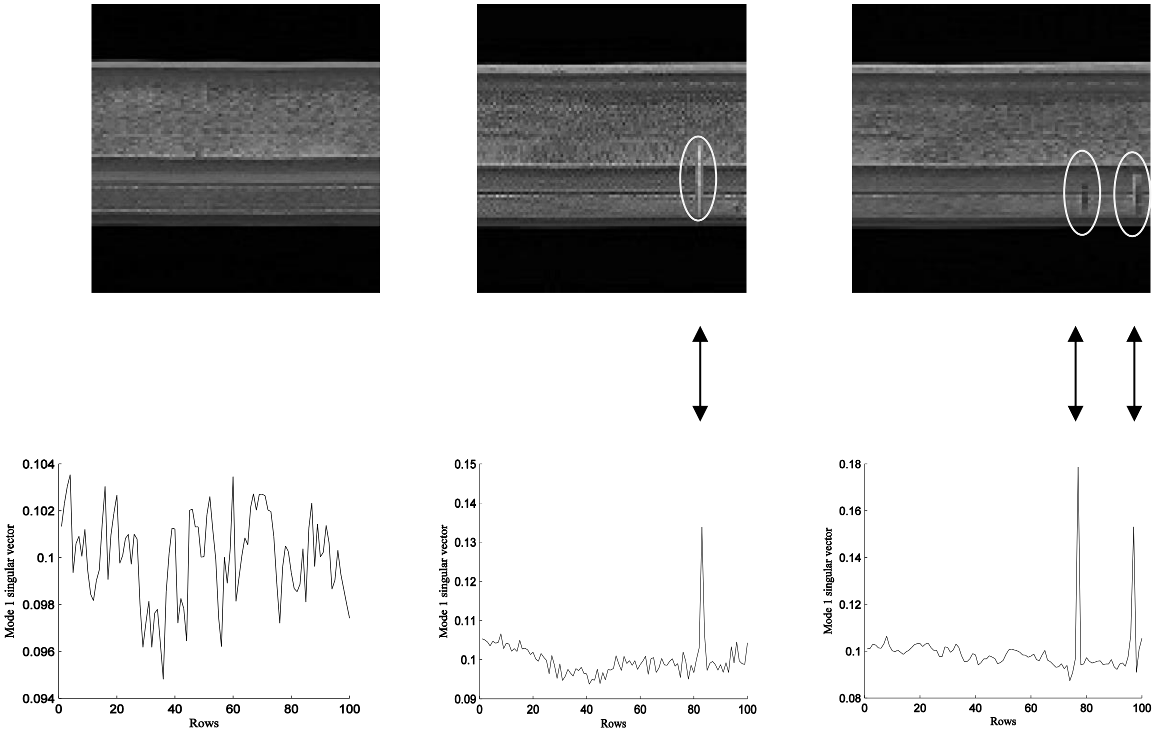
Figure 3-9: Out-of-control ARL comparisons for detecting different types of defects

3.6 Case study

This section presents the results of applying the RGHOSVD method for image-based defect monitoring in semi-finished casting products. The semi-finished casting products, such as ingots, slabs, billets, etc. are intermediate castings needing more operations to be considered as finished products. Billets are pieces of metal with a desired rounded or squared cross section, and they are further processed to produce bar stock or wire as final products. One critical quality characteristic in the billets is to have a surface free of defects usually appearing in the form of cracks (openings) on the surface of billets (see Figure 3-1 (a)). If a defect is in the direction of the metal's extraction, it is called a longitudinal defect, while those defects which are across the extraction are called transverse defects (Figure 3-1 (b)).

Since billets are semi-finished products, it is necessary to detect the defective units at an early stage of production to avoid the additional replacement costs incurred due to substituting a final defective product. As a result, a control chart to monitor the defects on the surface of billets is required. Using a hot-eye camera, a vision sensing system is set up to take snapshots of billets at short time intervals. The RGHOSVD-based chart is subsequently applied in order to illustrate how this method can be used for image-based monitoring in practice. Figure 3-10 shows the estimated mode-1 singular vectors (left singular vectors) for (a) faultless image, (b) an image with a single transverse defect, and (c) an image with double transverse defects.

As it is obvious in Figure 3-10, the singular vectors for the defective billets show different variation pattern in comparison to that of the faultless billet, and this dissimilarity forms the basis for the decomposition-based monitoring schemes. For the rolling-bar dataset, $I_3 = m = 30$ faultless images containing 10% contamination i.e. $P = 0.1$ (three outliers as shown in Figure 3-3) are used to compute the baseline mode-1 and mode-2 singular vectors and the associated high-order features $hf_l; l = 1,2, \dots, L,$ as in Eq. (3-26). The scale parameters for mode-1 and mode-2 quadratic operators ($\mathbf{C}_1$ and $\mathbf{C}_2$) are set to $\eta = \nu = 10$.



(a) faultless image (b) single transverse defect (c) double transverse defect

Figure 3-10: Mode-1 singular vectors for different type of images

The explained variability for the first component $L = 1$ is 90%; as a result, only one component suffices for constructing the control chart. The robust minimum absolute deviation estimator (mad) is used to estimate the standard deviation of hf_1 i.e. $\hat{\sigma}_{hf_1} = 1.4826\, mad_{hf_1} = 65.66$. A simple Shewhart control chart is considered for monitoring with $LCL = -196.6$ and $UCL = 196.6$. After ensuring that all $I_3 = 30$ points are statistically in control (no defective image), we used the established control chart for monitoring a set of images with two defective units shown in Figure 3-10 (b) and Figure 3-10 (c). Figure 3-11 plots the simple Shewhart statistics (x values) calculated using hf_1 values. Point 31 on the chart belongs to the single transverse defective billet shown in Figure 3-10 (b), and the 32nd point represents Figure 3-10 (c).

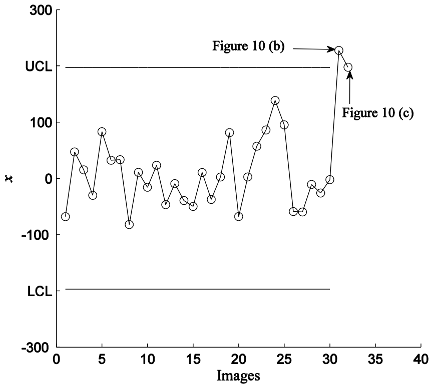


Figure 3-11: The RGHOSVD-based control chart applied to the rolling-bar dataset

Appendix 3-A: Proof of Theorem 3-1

To minimize the objective function in Eq. (3-8), we begin with adding the constraint to the objective function using the Lagrange multiplier λ. This gives us

Minimize

$$\frac{1}{m}\sum_{i=1}^{m} f_1\left(\frac{\left\|\mathbf{x}_i-u_i\mathbf{v}^T\right\|_C^2}{\tau}\right)+\lambda(\mathbf{v}^{\boldsymbol{T}}\mathbf{R}\mathbf{v}-1) \tag{3-A-1}$$

We can further extend the objective function in Eq. (3-A-1) to have

$$\frac{1}{m}\sum_{i=1}^{m} f_1\left(\frac{trace(\mathbf{x}_i\mathbf{C}\mathbf{x}_i{}^T)-2trace(\mathbf{x}_i\mathbf{C}\mathbf{v}u_i)+u_i^2}{\tau}\right)+\lambda(\mathbf{v}^T\mathbf{C}\mathbf{v}-1) \tag{3-A-2}$$

Taking derivative from (3-A-2) with respect to **v** and setting to zero gives

$$\frac{-1}{m\tau}\sum_{i=1}^{m} u_i\mathbf{x}_i\mathbf{C}\omega_i+2\lambda\mathbf{C}\mathbf{v}^T=0 \tag{3-A-3}$$

where $\omega(x)=\frac{df_1(x)}{dx}$ is called the weight function. Based on Karush-Kuhn-Tucker condition, If $\lambda>0$, the right singular vector is computed as

$$\mathbf{v}^*=\frac{\frac{1}{m\tau}\sum_{i=1}^{m} u_i\mathbf{x}_i^T\omega_i}{2\lambda} \tag{3-A-4}$$

If λ is selected to satisfy $\mathbf{v}^T\mathbf{C}\mathbf{v}=1$, we have $\frac{1}{(m\tau 2\lambda)^2}\left[\sum_{i=1}^{m} u_i\mathbf{x}_i^T\omega_i\right]^T\mathbf{C}\sum_{i=1}^{m} u_i\mathbf{x}_i^T\omega_i=1$,

and $\lambda=\frac{1}{2m\tau}\sqrt{\left[\sum_{i=1}^{m} u_i\mathbf{x}_i^T\omega_i\right]^T\mathbf{C}\sum_{i=1}^{m} u_i\mathbf{x}_i^T\omega_i}$.

Finally, we can plug in λ in (3-A-4), and calculate $\mathbf{v}^*$ as

$$\mathbf{v}^* = \frac{\sum_{i=1}^{m} u_i \mathbf{x}_i^T \omega_i}{\sqrt{[\sum_{i=1}^{m} u_i \mathbf{x}_i \omega_i]\mathbf{C}[\sum_{i=1}^{m} u_i \mathbf{x}_i^T \omega_i]}} \tag{3-A-5}$$

Similarly, the left singular vector is calculated by setting the derivative of Eq. (3-A-2) with respect to u_i to zero. The vector format of the left singular vector is calculated as

$$\mathbf{u} = \mathbf{XCv} \tag{3-A-6}$$

The normalized left singular vector is finally given by $\mathbf{u}^* = \frac{\mathbf{XCv}}{\sqrt{trace((\mathbf{XCv})^T\mathbf{XCv})}}$.

Appendix 3-B: Proof of Lemma 3-1

$\mathbf{x}_i - u_i\mathbf{v}^T$ is an affine function of u_i when $\mathbf{v}^T$ is fixed. Furthermore, the quadratic function $\mathbf{ACA}^T$ is convex since $\mathbf{C}$ is a positive semidefinite matrix. It follows that $\|\mathrm{A}\|_{\mathrm{C}}$ is a convex function, because Frobenius norm is a convex function. Hence, $\|\mathbf{x}_i - u_i\mathbf{v}^T\|_{\mathrm{C}}^2$ is convex in u_i when $\mathbf{v}^T$ is fixed. This is also true when u_i is fixed. Since $f(r)$ is a concave and non-decreasing function, the problem in Eq. (3-8) is not convex. Using the same logic, it is straightforward to show that the objective function in Eq. (3-14) is also nonconvex.

Appendix 3-C: Proof of Theorem 3-2

To prove Theorem 3-2, first, we must show that

$$\left\|\mathcal{X}_{:,:,\dots,:i_N} - \mathbf{v}^{(1)} \times_2 \mathbf{v}^{(2)} \times_3 \dots \times_N v_{i_N}^{(N)}\right\|_{\mathbf{C}_1,\mathbf{C}_2,\dots,\mathbf{I}}^2 = \left\|\mathcal{X}_{:,:,\dots,:i_N} \times_1 \tilde{\mathbf{C}}_1 \times_2 \tilde{\mathbf{C}}_2 \times_3 \dots \times_{N-1} \tilde{\mathbf{C}}_{N-1}\right\|^2 +$$

$$v_{i_N}^{(N)^2} - 2v_{i_N}^{(N)}\mathcal{X}_{:,:,\dots,:i_N} \times_1 \mathbf{C}_1\mathbf{v}^{(1)} \times_2 \mathbf{C}_2\mathbf{v}^{(2)} \times_3 \dots \times_{N-1} \mathbf{C}_{N-1} \tag{3-C-1}$$

To begin with, we need to show that $\mathbf{C}_1, \mathbf{C}_2, \dots, \mathbf{C}_N$-norm of an Nth-order tensor $\mathcal{B}$ can be obtained as

$$\|\mathcal{B}\|_{\mathbf{C}_1,\mathbf{C}_2,\dots,\mathbf{C}_N} = \left\|\mathcal{B} \times_1 \tilde{\mathbf{C}}_1 \times_2 \tilde{\mathbf{C}}_2 \times_3 \dots \times_N \tilde{\mathbf{C}}_N\right\| =$$

$$\sqrt{Tr\left\{(\mathbf{C}_N \otimes \dots \otimes \mathbf{C}_{n-1} \otimes \mathbf{C}_{n+1} \otimes \dots \otimes \mathbf{C}_1)\mathbf{B}_{(n)}^T\mathbf{C}_n\mathbf{B}_{(n)}\right\}} \tag{3-C-2}$$

where $\tilde{\mathbf{C}}_n; n = 1,2,\dots,N$, is any matrix square root, such as Cholesky decomposition, and $\mathbf{C}_n = \tilde{\mathbf{C}}_n^T\tilde{\mathbf{C}}_n$. In terms of matricized tensors, we can write

$$\boldsymbol{\mathcal{D}} = \mathcal{B} \times_1 \tilde{\mathbf{C}}_1 \times_2 \tilde{\mathbf{C}}_2 \times_3 \dots \times_N \tilde{\mathbf{C}}_N \Leftrightarrow \mathbf{D}_{(\boldsymbol{n})} = \tilde{\mathbf{C}}_n\ \mathbf{B}_{(n)}(\tilde{\mathbf{C}}_N \otimes \dots \otimes \tilde{\mathbf{C}}_{n-1} \otimes \tilde{\mathbf{C}}_{n+1} \otimes \dots \otimes \tilde{\mathbf{C}}_1)^T$$

$$\|\boldsymbol{\mathcal{D}}\| = \left\|\mathbf{D}_{(\boldsymbol{n})}\right\| = \left\|\tilde{\mathbf{C}}_n\mathbf{B}_{(n)}(\tilde{\mathbf{C}}_N \otimes \dots \otimes \tilde{\mathbf{C}}_{n-1} \otimes \tilde{\mathbf{C}}_{n+1} \otimes \dots \otimes \tilde{\mathbf{C}}_1)^T\right\| =$$

$$\sqrt{Tr\left\{\left(\tilde{\mathbf{C}}_n\mathbf{B}_{(n)}(\tilde{\mathbf{C}}_N \otimes \dots \otimes \tilde{\mathbf{C}}_{n-1} \otimes \tilde{\mathbf{C}}_{n+1} \otimes \dots \otimes \tilde{\mathbf{C}}_1)^T\right)^T\tilde{\mathbf{C}}_n\mathbf{B}_{(n)}(\tilde{\mathbf{C}}_N \otimes \dots \otimes \tilde{\mathbf{C}}_{n-1} \otimes \tilde{\mathbf{C}}_{n+1} \otimes \dots \otimes \tilde{\mathbf{C}}_1)^T\right\}} =$$

$$\sqrt{Tr\left\{(\tilde{\mathbf{C}}_N \otimes \dots \otimes \tilde{\mathbf{C}}_{n-1} \otimes \tilde{\mathbf{C}}_{n+1} \otimes \dots \otimes \tilde{\mathbf{C}}_1)\mathbf{B}_{(n)}^T{\tilde{\mathbf{C}}_n}^T\tilde{\mathbf{C}}_n\mathbf{B}_{(n)}\left({\tilde{\mathbf{C}}_N}^T \otimes \dots \otimes {\tilde{\mathbf{C}}_{n-1}}^T \otimes {\tilde{\mathbf{C}}_{n+1}}^T \otimes \dots \otimes {\tilde{\mathbf{C}}_1}^T\right)\right\}} =$$

$$\sqrt{Tr\left\{\left({\tilde{\mathbf{C}}_N}^T \otimes \dots \otimes {\tilde{\mathbf{C}}_{n-1}}^T \otimes {\tilde{\mathbf{C}}_{n+1}}^T \otimes \dots \otimes {\tilde{\mathbf{C}}_1}^T\right)(\tilde{\mathbf{C}}_N \otimes \dots \otimes \tilde{\mathbf{C}}_{n-1} \otimes \tilde{\mathbf{C}}_{n+1} \otimes \dots \otimes \tilde{\mathbf{C}}_1)\mathbf{B}_{(n)}^T{\tilde{\mathbf{C}}_n}^T\tilde{\mathbf{C}}_n\mathbf{B}_{(n)}\right\}} =$$

Based on the mixed-product property of Kronecker product, we have

$$= \sqrt{Tr\left\{\left({\tilde{\mathbf{C}}_N}^T\tilde{\mathbf{C}}_N \otimes \dots \otimes {\tilde{\mathbf{C}}_{n-1}}^T\tilde{\mathbf{C}}_{n-1} \otimes {\tilde{\mathbf{C}}_{n+1}}^T\tilde{\mathbf{C}}_{n+1} \otimes \dots \otimes {\tilde{\mathbf{C}}_1}^T\tilde{\mathbf{C}}_1\right)\mathbf{B}_{(n)}^T{\tilde{\mathbf{C}}_n}^T\tilde{\mathbf{C}}_n\mathbf{B}_{(n)}\right\}}$$

$$= \sqrt{Tr\{(\mathbf{C}_N \otimes \dots \otimes \mathbf{C}_{n-1} \otimes \mathbf{C}_{n+1} \otimes \dots \otimes \mathbf{C}_1)\mathbf{B}_{(n)}^T\mathbf{C}_n\mathbf{B}_{(n)}\}} = \|\mathcal{B}\|_{\mathbf{C}_1,\mathbf{C}_2,\dots,\mathbf{C}_N}$$

Using the results of Eq. (3-C-2), we can extend (3-C-1) as

$$\left\|\mathcal{X}_{:,:,\dots,:,i_N} - \mathbf{v}^{(1)} \circ \mathbf{v}^{(2)} \circ \dots \circ \mathbf{v}^{(N-1)}v_{i_N}^{(N)}\right\|_{\mathbf{C}_1,\mathbf{C}_2,\dots,\mathbf{I}}^2$$

$$= \left\|\left(\mathcal{X}_{:,:,\dots,:,i_N} - \mathbf{v}^{(1)} \circ \mathbf{v}^{(2)} \circ \dots \circ \mathbf{v}^{(N-1)}v_{i_N}^{(N)}\right) \times_1 \tilde{\mathbf{C}}_1 \times_2 \tilde{\mathbf{C}}_2 \times_3 \dots \times_{N-1} \tilde{\mathbf{C}}_{N-1}\right\|^2$$

$$= \left\|\mathcal{X}_{:,:,\dots,:,i_N} \times_1 \tilde{\mathbf{C}}_1 \times_2 \tilde{\mathbf{C}}_2 \times_3 \dots \times_{N-1} \tilde{\mathbf{C}}_{N-1}\right\|^2 + \left\|\tilde{\mathbf{C}}_1\mathbf{v}^{(1)} \circ \tilde{\mathbf{C}}_2\mathbf{v}^{(2)} \circ \dots \circ \tilde{\mathbf{C}}_{N-1}\mathbf{v}^{(N-1)}v_{i_N}^{(N)}\right\|^2$$

$$-2\langle\mathcal{X}_{:,:,\dots,:,i_N} \times_1 \tilde{\mathbf{C}}_1 \times_2 \tilde{\mathbf{C}}_2 \times_3 \dots \times_{N-1} \tilde{\mathbf{C}}_{N-1}, \tilde{\mathbf{C}}_1\mathbf{v}^{(1)} \circ \tilde{\mathbf{C}}_2\mathbf{v}^{(2)} \circ \dots \circ \tilde{\mathbf{C}}_{N-1}\mathbf{v}^{(N-1)}v_{i_N}^{(N)}\rangle$$

Since we know that ${\mathbf{v}^{(n)}}^T\mathbf{C}_n\mathbf{v}^{(n)} = 1$ for $n = 1, 2, \dots, N-1$, we have

$$\left\|\tilde{\mathbf{C}}_1\mathbf{v}^{(1)} \circ \tilde{\mathbf{C}}_2\mathbf{v}^{(2)} \circ \dots \circ \tilde{\mathbf{C}}_{N-1}\mathbf{v}^{(N-1)}v_{i_N}^{(N)}\right\|^2 =$$

$$Tr\left\{\left(\tilde{\mathbf{C}}_1\mathbf{v}^{(1)} \circ \tilde{\mathbf{C}}_2\mathbf{v}^{(2)} \circ \dots \circ \tilde{\mathbf{C}}_{N-1}\mathbf{v}^{(N-1)}v_{i_N}^{(N)}\right)^T\left(\tilde{\mathbf{C}}_1\mathbf{v}^{(1)} \circ \tilde{\mathbf{C}}_2\mathbf{v}^{(2)} \circ \dots \circ \tilde{\mathbf{C}}_{N-1}\mathbf{v}^{(N-1)}v_{i_N}^{(N)}\right)\right\} =$$

$$v_{i_N}^{(N)^2} Tr\left\{\left(\mathbf{v}^{(1)^T}\mathbf{C}_1\mathbf{v}^{(1)} \circ \mathbf{v}^{(2)^T}\mathbf{C}_2\mathbf{v}^{(2)} \circ \ldots \circ \mathbf{v}^{(N-1)^T}\mathbf{C}_{N-1}\mathbf{v}^{(N-1)}\right)\right\} = v_{i_N}^{(N)^2}$$

So (3-C-1) can be computed as

$$\left\|\mathcal{X}_{::\cdots:i_N} - \mathbf{v}^{(1)} \circ \mathbf{v}^{(2)} \circ \ldots \circ \mathbf{v}^{(N-1)} v_{i_N}^{(N)}\right\|_{\mathbf{C}_1,\mathbf{C}_2,\ldots,\mathbf{I}}^2 = \left\|\mathcal{X}_{::\cdots:i_N} \times_1 \tilde{\mathbf{C}}_1 \times_2 \tilde{\mathbf{C}}_2 \times_3 \ldots \times_{N-1} \tilde{\mathbf{C}}_{N-1}\right\|^2 +$$

$$v_{i_N}^{(N)^2} - 2v_{i_N}^{(N)} \mathcal{X}_{::\cdots:i_N} \times_1 \mathbf{C}_1\mathbf{v}^{(1)} \times_2 \mathbf{C}_2\mathbf{v}^{(2)} \times_3 \ldots \times_{N-1} \mathbf{C}_{N-1}$$

Based on Eq. (3-C-1) and Eq. (3-C-2), we can reformulate the RGHOSVD problem as

$$\text{Minimize } \sum_{i_N=1}^{I_N} f\ \left(\frac{A}{\xi}\right)$$

S.t:

$$\mathbf{v}^{(n)^T}\mathbf{C}_n\mathbf{v}^{(n)} = 1 \qquad n = 1,2,\ldots,N-1, \tag{3-C-3}$$

$$\text{where } A = \left\|\mathcal{X}_{::\cdots:i_N} \times_1 \tilde{\mathbf{C}}_1 \times_2 \tilde{\mathbf{C}}_2 \times_3 \ldots \times_{N-1} \tilde{\mathbf{C}}_{N-1}\right\|^2 +$$

$$v_{i_N}^{(N)^2} - 2v_{i_N}^{(N)} \mathcal{X}_{::\cdots:i_N} \times_1 \mathbf{C}_1\mathbf{v}^{(1)} \times_2 \mathbf{C}_2\mathbf{v}^{(2)} \times_3 \ldots \times_{N-1} \mathbf{C}_{N-1}$$

We add the constraints in Eq. (3-C-3) to the objective function using the Lagrange multipliers λ_n; $n = 1,2,\ldots,N-1$, as a result, we have

Minimize

$$\sum_{i_N=1}^{I_N} f\ \left(\frac{A}{\xi}\right) + \sum_{n=1}^{N-1}\lambda_n\left(\mathbf{v}^{(n)^T}\mathbf{C}_n\mathbf{v}^{(n)} - 1\right) \tag{3-C-4}$$

Take the derivative of Eq. (3-C-4) with regard to $\mathbf{v}^{(n)}$and set it to zero

$$-2\omega_{i_N} v_{i_N}^{(N)} \frac{\left[\mathcal{X}_{::\cdots:i_N} \times_1 \mathbf{C}_1\mathbf{v}^{(1)} \times_2 \ldots \times_{n-1} \mathbf{C}_{n-1}\mathbf{V}_{i_{n-1}}^{(n-1)} \times_n \mathbf{C}_n \times_{n+1} \mathbf{C}_{n+1}\mathbf{V}_{i_{n+1}}^{(n+1)} \ldots \times_{N-1} \mathbf{C}_{N-1}\mathbf{V}_{i_{N-1}}^{(N-1)}\right]}{\xi} + 2\lambda_n\mathbf{v}^{(n)}\mathbf{C}_n = 0$$

$$\mathbf{v}^{(n)} = \frac{\left[\mathcal{X}_{::\cdots:i_N} \times_1 \mathbf{C}_1\mathbf{v}^{(1)} \times_2 \ldots \times_{n-1} \mathbf{C}_{n-1}\mathbf{v}_{i_{n-1}}^{(n-1)} \times_{n+1} \mathbf{C}_{n+1}\mathbf{v}_{i_{n+1}}^{(n+1)} \ldots \times_{N-1} \mathbf{C}_{N-1}\mathbf{v}_{i_{N-1}}^{(N-1)} \times_N \omega_{i_N} v_{i_N}^{(N)}\right]}{\xi\lambda_n} \tag{3-C-5}$$

To obtain the Lagrange multiplier λ_n, we define

$$\mathbf{b}$$

$$= \mathcal{X}_{::\cdots:i_N} \times_1 \mathbf{C}_1\mathbf{v}^{(1)} \times_2 \ldots \times_{n-1} \mathbf{C}_{n-1}\mathbf{v}_{i_{n-1}}^{(n-1)} \times_{n+1} \mathbf{C}_{n+1}\mathbf{v}_{i_{n+1}}^{(n+1)} \ldots \times_{N-1} \mathbf{C}_{N-1}\mathbf{v}_{i_{N-1}}^{(N-1)} \times_N \omega_{i_N} v_{i_N}^{(N)}$$

Based on the constraints in Eq. (3-C-3), we have

$$\frac{\mathbf{b}^T\mathbf{C}_n\mathbf{b}}{(\xi\lambda_n)^2} = 1$$

$$\lambda_n = \frac{\sqrt{\mathbf{b}^T\mathbf{C}_n\mathbf{b}}}{\xi}$$

Finally, we compute the singular vectors as

$$\mathbf{v}^{(n)} = \frac{\mathbf{b}}{\sqrt{\mathbf{b}^T\mathbf{C}_n\mathbf{b}}} \tag{3-C-6}$$

Taking derivative with respect to $v_{i_N}^{(N)}$; $n = 1,2 \ldots N-1$, and setting to zero gives

$$\frac{\omega_{i_N}}{\xi} 2v_{i_N}^{(N)} - 2\frac{\mathcal{X}_{::\cdots:i_N}\times_1\mathbf{C}_1\mathbf{v}^{(1)}\times_2\ldots\times_{n-1}\mathbf{C}_{n-1}\mathbf{V}_{i_{n-1}}^{(n-1)}\times_n\mathbf{C}_n\times_{n+1}\mathbf{C}_{n+1}\mathbf{V}_{i_{n+1}}^{(n+1)}\ldots\times_{N-1}\mathbf{C}_{N-1}\mathbf{V}_{i_{N-1}}^{(N-1)}\times_N\omega_{i_N}}{\xi} = 0$$

$$v_{i_N}^{(N)} = \mathcal{X}_{::\cdots:i_N} \times_1 \mathbf{C}_1\mathbf{v}^{(1)} \times_2 \ldots \times_{n-1} \mathbf{C}_{n-1}\mathbf{V}_{i_{n-1}}^{(n-1)} \times_n \mathbf{C}_n \times_{n+1} \mathbf{C}_{n+1}\mathbf{V}_{i_{n+1}}^{(n+1)} \ldots \times_{N-1} \mathbf{C}_{N-1}\mathbf{V}_{i_{N-1}}^{(N-1)}$$

$$\mathbf{v}^{(N)} = \frac{\mathcal{X}\times_1\mathbf{C}_1\mathbf{v}^{(1)}\times_2\ldots\times_{n-1}\mathbf{C}_{n-1}\mathbf{V}_{i_{n-1}}^{(n-1)}\times_n\mathbf{C}_n\times_{n+1}\mathbf{C}_{n+1}\mathbf{V}_{i_{n+1}}^{(n+1)}\ldots\times_{N-1}\mathbf{C}_{N-1}\mathbf{V}_{i_{N-1}}^{(N-1)}}{\left\|\mathcal{X}\times_1\mathbf{C}_1\mathbf{v}^{(1)}\times_2\ldots\times_{n-1}\mathbf{C}_{n-1}\mathbf{V}_{i_{n-1}}^{(n-1)}\times_n\mathbf{C}_n\times_{n+1}\mathbf{C}_{n+1}\mathbf{V}_{i_{n+1}}^{(n+1)}\ldots\times_{N-1}\mathbf{C}_{N-1}\mathbf{V}_{i_{N-1}}^{(N-1)}\right\|} \tag{3-C-7}$$

Appendix 3- D: Computational Complexity

Table 3-D-1 gives the number of flops in each operation for every step of the algorithm in the kth iteration in the blockwise tensor power algorithm with K iterations. Let's denote the ith algebraic operation in Step j by s_i^j. The operations needed in each step are given as

Step 1: Calculating the residuals

$$r_{i_3}^k = \left\|\mathcal{X}_{::i_3} - v_{i_3}^{(3)}\mathbf{v}^{(1)} \circ \mathbf{v}^{(2)}\right\|_{\mathrm{C_1,C_2,I}}^2;\ i_3 = 1,2 \ldots m$$

$$s_1^1 = \mathbf{v}^{(1)} \circ \mathbf{v}^{(2)}$$

$$s_2^1 = v_{i_3}^{(3)}\mathbf{v}^{(1)} \circ \mathbf{v}^{(2)}$$

$$s_3^1 = \mathcal{X}_{::i_3} - v_{i_3}^{(3)} \mathbf{v}^{(1)} \circ \mathbf{v}^{(2)}$$

$$s_4^1 = \left(\mathcal{X}_{::i_3} - v_{i_N}^{(3)} \mathbf{v}^{(1)} \circ \mathbf{v}^{(2)}\right) \times_1 \tilde{\mathbf{C}}_1$$

$$s_5^1 = \left(\mathcal{X}_{::i_3} - v_{i_3}^{(3)} \mathbf{v}^{(1)} \circ \mathbf{v}^{(2)}\right) \times_1 \tilde{\mathbf{C}}_1 \times_2 \tilde{\mathbf{C}}_2$$

$$s_6^1 = \|s_5^1\|^2$$

Step 2: Obtaining ξ^k as the solution of $\frac{1}{m}\sum_{i_3=1}^{m} f\left(\frac{r_{i_3}^k}{\xi^k}\right) = \delta$ where f is given in Eq. (3-7).

$$s_1^2 = f\left(\frac{r_{i_3}^k}{\xi^k}\right) = min\left\{1,1 - \left(1 - \frac{r_{i_3}^k}{\xi^k}\right)^3\right\}$$

$$s_2^2 = \text{solve}\, \frac{1}{m}\sum_{i_3=1}^{m} f\left(\frac{\mathbf{r}_{i_3}^k}{\xi^k}\right) = \delta \text{ for } \xi^k$$

Step 3: Obtaining the weights $\omega_{i_3}^{k}{}'s; i_3 = 1,2 \ldots m$

$$s_1^3 = \omega_{i_3}^k = \frac{1}{\xi^k}\left(1 - \frac{r_{i_3}^k}{\xi^k}\right)^2$$

Step 4: Updating the solutions.

$$s_1^4 = \mathbf{v}_k^{(1)} = \frac{\sum_{i_3=1}^{m} \mathcal{X}_{::i_3} \times_2 \mathbf{C}_2 \mathbf{v}_k^{(2)} \times_3 \omega_{i_3}^k v_{ki_3}^{(3)}}{\sqrt{\left(\sum_{i_3=1}^{m} \mathcal{X}_{::i_3} \times_2 \mathbf{C}_2 \mathbf{v}_k^{(2)} \times_3 \omega_{i_3}^k v_{ki_3}^{(3)}\right)^T \mathbf{C}_1 \left(\sum_{i_3=1}^{m} \mathcal{X}_{::i_3} \times_2 \mathbf{C}_2 \mathbf{v}_k^{(2)} \times_3 \omega_{i_3}^k v_{ki_3}^{(3)}\right)}}$$

$$s_2^4 = \mathbf{v}_k^{(2)} = \frac{\sum_{i_3=1}^{m} \mathcal{X}_{::i_3} \times_1 \mathbf{C}_1 \mathbf{v}_k^{(1)} \times_3 \omega_{i_3}^k v_{ki_3}^{(3)}}{\sqrt{\left(\sum_{i_3=1}^{m} \mathcal{X}_{::i_3} \times_1 \mathbf{C}_1 \mathbf{v}_k^{(1)} \times_3 \omega_{i_3}^k v_{ki_3}^{(3)}\right)^T \mathbf{C}_2 \left(\sum_{i_3=1}^{m} \mathcal{X}_{::i_3} \times_1 \mathbf{C}_1 \mathbf{v}_k^{(1)} \times_3 \omega_{i_3}^k v_{ki_3}^{(3)}\right)}}$$

$$s_3^4 = v_{ki_3}^{(3)} = \frac{\mathcal{X}_{::i_3} \times_1 \mathbf{C}_1 \mathbf{v}_k^{(1)} \times_2 \mathbf{C}_2 \mathbf{v}_k^{(2)}}{\sqrt{\sum_{i_3=1}^{m}\left[\mathcal{X}_{::i_3} \times_1 \mathbf{C}_1 \mathbf{v}_k^{(1)} \times_2 \mathbf{C}_2 \mathbf{v}_k^{(2)}\right]^2}}; i_3 = 1,2 \ldots m$$

Step 5: Compute $e^k = |\xi^k - \xi^{k-1}|$, $e_n^k = \left\|\mathbf{v}_k^{(n)} - \mathbf{v}_{k-1}^{(n)}\right\|_2; n = 1,2,3$

Table 3-D-1: Computational complexity with regard to each operation

Operation	Computational complexity
s_1^1	np
s_2^1	np
s_3^1	np
s_4^1	$2n^2p - np + O\left(\frac{n^3}{K}\right)$
s_5^1	$2p^2n - np + O\left(\frac{p^3}{K}\right)$
s_6^1	$2np - 1$
s_1^2	$7m + 1$
s_2^2	$O\big(F(q)\big)^*$
s_1^3	$4m$
s_1^4	$2p^2m + 2nmp + 2m - mp + 2n^2 + 2n - 1$
s_2^4	$2n^2m + 2nmp + 2m - mn + 2p^2 + 2p - 1$
s_3^4	$2p^2m + 2n^2m + 2nmp - mn + 2m$
s_1^5	$2nmp + np$
Total	$O(n^2pm + p^2nm)$

* $F(q)$ is the cost of computing $\frac{f\left(\frac{r_{i_3}^k}{\xi^k}\right)}{\acute{f}\left(\frac{r_{i_3}^k}{\xi^k}\right)}$, and $q = 5$ is the number of digits in the root ξ^k

The computational complexity of the blockwise power algorithm can be calculated similarly.

References:

1. G. I. Allen, (2012), “Regularized Tensor Factorizations and Higher-Order Principal Components Analysis,” *Rice University Technical Report No. TR2012-01*, arXiv:1202.2476.

2. G. I. Allen, L. Grosenick, and J. Taylor, (2014), “A Generalized Least Squares Matrix Decomposition,” *Journal of the american statistical association*, *Theory & methods*, vol. 109, no. 505, pp. 145-159.

3. C. Croux, G. Haesbroeck, (2000), “Principal Component Analysis Based on Robust Estimators of the Covariance or Correlation Matrix: Influence Functions and Efficiencies,” *Biometrika*, vol. 87, no. 3, pp. 603-618.

4. C. Croux, A. Ruiz-Gazen, (2005), “High breakdown estimators for principal components: the projection-pursuit approach revisited,” *Journal of multivariate analysis*, vol. 95, no. 1, pp. 206-226.

5. I. Daubechies, (1990), “The wavelet transform, time-frequency localization and signal analysis,” *IEEE transactions on information theory*, vol. 36, no. 5, pp. 961-1005.

6. F. De la Torre, M. J. Black, (2001), “Robust principal components analysis for computer vision,” *Proceedings of the International Conference on Computer Vision.*

7. G. H. Golub, C. F. Van Loan, (1996), “Matrix computations,” Johns Hopkins Studies in Mathematical Sciences.

8. A. K. Gupta, D. K. Nagar (1999), “Matrix variate distributions,” Volume 104 of Monographs and Surveys in Pure and Applied Mathematics. Boca Raton, FL: Chapman & Hall, CRC Press.

9. M. Hubert, P. J. Rousseeuw, and K. B. Vanden, (2005), “ROBPCA: a new approach to robust principal component analysis,” *Technometrics*, vol. 47, no. 1, pp. 64-79.

10. P. J. Huber, (1973), “Robust Regression: Asymptotics, Conjectures and Monte Carlo,” *The annals of statistics*, vol. 1, no. 5, pp. 799-821.

11. P. J. Huber, (1981), “Robust Statistics,” John Wiley & Sons, New York.

12. Q. Ke, T. Kanade, (2005), “Robust L1 Norm Factorization in the Presence of Outliers and Missing Data by Alternative Convex Programming,” *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (CVPR'05).

13. T. Kolda, B. Bader, (2009), “Tensor decompositions and applications,” *SIAM Rev,* vol. 51, no. 3, pp. 455-500.

14. L. Liu, D. M. Hawkins, S. Ghosh, and S. S. Young, (2003), “Robust singular value decomposition analysis of microarray data,” *Proceedings of the National Academy of Sciences*, vol. 100, no. 23, pp. 13167-13172.

15. H. –D. Lin, (2007), “Computer-aided visual inspection of surface defects in ceramic capacitor chips,” *Journal of materials processing technology*, vol. 189, no. 1-3, pp. 19-25.

16. H. –D. Lin, C. –Y. Chung, and W. –T. Lin, (2008), “Principal component analysis based on wavelet characteristics applied to automated surface defect inspection,” *WSEAS Transactions on Computer Research*, vol. 3, no. 4, pp. 193-202.

17. J. J. Liu, J. F. MacGregor, (2006), “Estimation and monitoring of product aesthetics: application to manufacturing of “engineered stone” countertops,“ Machin vision and applications, vol. 16, no. 6, pp. 374-383.

18. J.J. Liu, J.F. MacGregor, C. Duchesne, and G. Bartolacci, (2005), “Flotation froth monitoring using multiresolutional multivariate image analysis,” *Minerals engineering,* vol. 18, no. pp. 65-76.

19. C. –J. Lu, D. –M. Tsai, (2005), “Automatic defect inspection for LCDs using singular value decomposition,” *The international journal of advanced manufacturing technology*, vol. 25, no. 1, pp. 53-61.

20. R. A. Marrona, (2005), “Principal components and orthogonal regression based on robust scales,” *Technometrics*, vol. 47, no. 3¸ pp. 264-273.

21. R. A. Marrona, V. J. Yohai, (2008), “Robust lower-rank approximation of data matrices with elementwise contamination,” *Technometrics*, vol. 50, no. 3, pp. 295-304.

22. F. M. Megahed, W. H. Woodall, and J. A. Camelio, (2011) “A review and perspective on control charting with image data,” *Journal of quality technology*, vol. 43, no. 2, pp. 83-98.

23. R. Naga, G. Antille, (1990), “Stability of Robust and Non-Robust Principal Component Analysis,” *Computational statistics & data analysis*, vol. 10, no. 2, pp. 169-174.

24. H. B. Nembhard, N. J. Ferrier, T. A. Osswald, and J. R. Sanz-Uribe, (2003), “An integrated model for statistical and vision monitoring in manufacturing transitions,” *Quality and reliability engineering international*, vol. 19, no. 6, pp. 461-476.

25. C. E. Rasmussen, C. K. I. Williams, (2005), “Gaussian Processes for Machine Learning,” the MIT Press, ISBN 026218253X.

26. W. Rey, (2007), “Total singular value decomposition. Robust SVD, regression and location-scale,” http://arxiv.org/abs/0706.0096.

27. P. Rousseeuw, V. Yohai, (1984), “Robust Regression by Means of S-Estimators,” *Lecture notes in statistics*, 10.1007/978-1-4615-7821-5_15.

28. A. F. Siegel, (1982), “Robust regression using repeated medians,” *Biometrika*, vol. 69, no. 1**,** pp. 242-244.

29. J. Tan, Z. Chang, and F. Hsieh, (1996), “Implementation of an automated real-time statistical process controller,” *Journal of Food Process Engineering*, vol. 19, no. 1, pp. 49-61.

30. H. Yan, K. Paynabar, and J. Shi, (2015), “Anomaly Detection in Images with Smooth Background **v**ia Smooth-Sparse Decomposition,” *Technometrics*, DOI:10.1080/00401706.2015.1102764.

31. V. J. Yohai, (1987), “High breakdown-point and high-efficiency estimates for Regression,” *The annals of statistics*, vol. 15, no. 2, pp. 642-656.

CHAPTER 4

FUNCTIONAL LINEAR REGRESSION WITH TENSORIAL PREDICTOR

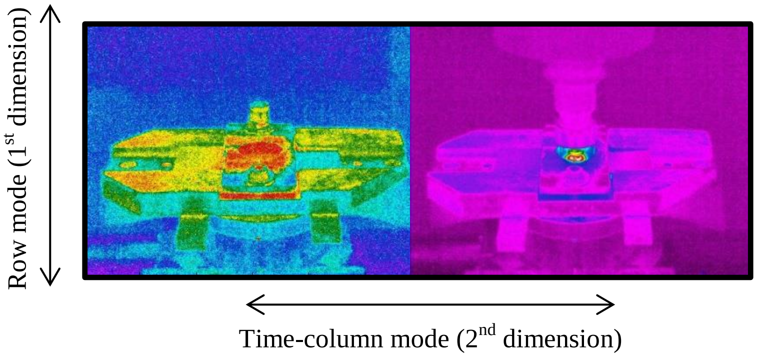
4.1 Introduction

In most manufacturing processes, finding a relationship between a response variable of product quality and process variables as predictors is an essential task in process quality control. There are several reasons bolstering the importance of constructing this relationship. For example, for some newly established processes, the underlying physical model ruling the process variables is not completely understood. A mathematical model explaining the role of these process variables can be tremendously helpful. This can help systematically optimize or control to ensure product quality.

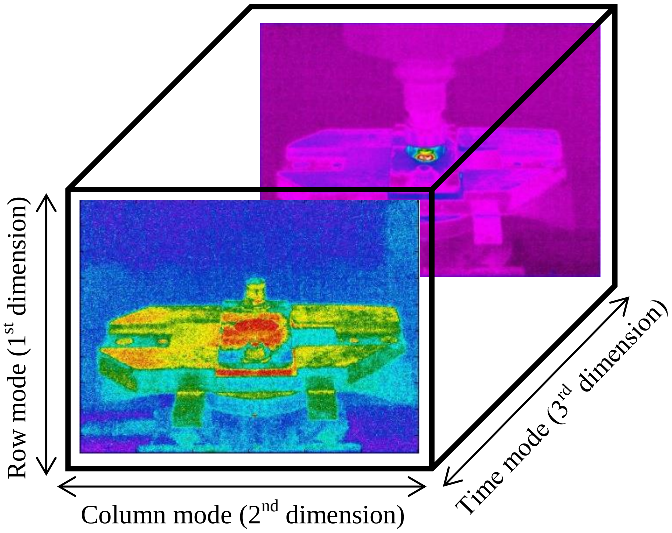
Conventionally, there is a scaler process quality output (response variable), and the purpose is to predict this response using some process variables that are scaler variables. The first step toward building the relationship is to gather the data regarding the process quality response and the process variables. Advanced sensing and data acquisition techniques offer the opportunity of collecting immense amount of information during operation. Therefore, advanced data analytics methods are needed to avoid losing information when representing and analyzing these variables having complex and unconventional structures.

In this chapter, a functional linear regression method will be studied to handle the complex predictors. The complexity of predictors is characterized by two aspects, namely, multi-stream structure and functional regression parameters to be estimated. A multi-stream dataset two dimensions of interest that must be considered in data analysis. Tensors are high-order arrays generalizing the concept of low-order arrays, such as vectors and matrices, and they have the ability of taking into account more than two dimensions (modes) that a predictor might have. This property enables tensors to represent the data with more than two dimensions and without manipulating them to be represented using low-order arrays. Some authors suggest using a tensor representation when the data are better represented by a multi-stream or high-order array (Zerehsaz et al., 2016; Hadi et al., 2016).

A useful example for explaining the necessity of using a tensor is the infrared images taken in multiple critical operation steps in a process. Each image typically can be represented by a matrix or even a vector. In addition to the regular spatial row and column dimensions, there might be a third dimension that is the time when the image is captured. For instance, the information carried by an image taken at the initial step of a process can be substantially different from that offered in an image taken in the middle of the process. Figure 4-1 shows the differences between using a matrix and a tensor to represent the predictor. The images show the temperature of the environment captured by infrared camera (IR) in two different time points from a friction stir blind riveting (FSBR) process. Each step of the FSBR process contains critical information about the process operating condition. That is, the information provided by the images will be distinctive in each step and the temporal dimension must be considered when representing the data. If a matrix is used for representation, then the temporal dimension will be ignored (merged with column mode). To avoid this problem, a tensor consisting three dimensions is employed to represent the set of IR images as shown in Figure 4-1 (b).



(a) Low-order (matrix) representation (stack up the time dimension)



(b) High-order representation

Figure 4-1: High-order and low-order representations of IR images

Another aspect of the data complexity reveals itself in the form of predictors that are not scalar in nature. Sometimes the collected observations are in the form of a function or a curve. This means that the data are functions of time and/or space, and a functional relationship exists between the variable of interest and time as an index (Ramsay and Silverman, 2002). In some cases, treating a functional variable as a scalar variable leads to misleading estimation of parameters. In an FSBR process, an example of a functional predictor is the penetration force which is used to drive a rivet into two dissimilar materials in order to join them. As it can be noticed in Figure 4-2, force signal has a functional relationship with time (distance), and it would be misleading to treat it as a scalar variable.

Although the consequences of not taking into account these critical aspects utterly depend on the application and the purpose of the study, both types of the aforementioned predictors must be considered when constructing a model to predict the process quality response variable. The regular linear regression methods including ordinary and functional regression approaches cannot be directly utilized to find the parameters of the functional linear regression model with tensorial predictor (FLRTP). A new model with the flexibility to consider mixed types of predictors is needed. A possible formulization to the FLRTP model is given as

$$y_i = \mathbf{z}_i^T \boldsymbol{\gamma} + \int X_i(t)\alpha(t)dt + \langle \mathcal{B}, \mathcal{W}_i \rangle + \varepsilon_i; i = 1,2, \dots, n, \tag{4-1}$$

where y_i is the scalar response variable that needs to be predicted, $\mathbf{z}_i \in \mathbb{R}^l$ is the vector of scalar predictors with coefficients vector $\boldsymbol{\gamma} \in \mathbb{R}^l$, the functional predictor and its coefficient function are denoted by $X_i(t)$ and $\alpha(t)$, respectively, where t is an index of time, distance, location, etc., and it is normalized so that $0 \leq t \leq 1$. The tensorial predictor is given by an Mth-order tensor $\mathcal{W}_i \in \mathbb{R}^{K_1 \times K_2 \times \dots \times K_M}$ with the tensor of coefficients represented by $\mathcal{B} \in \mathbb{R}^{K_1 \times K_2 \times \dots \times K_M}$, $\langle \mathcal{U}, \mathcal{V} \rangle = \sum_{k_1}^{K_1} \sum_{k_2}^{K_2} \dots \sum_{k_N}^{K_N} u_{k_1 k_2 \dots k_N} v_{k_1 k_2 \dots k_N}$ is the inner product of tensors $\mathcal{U}$ and $\mathcal{V}$, and $\varepsilon_i's$ are independent random error terms.

The classic approach to obtain the regression parameters in Eq. (4-1) is to minimize the residual sum of squares. As a result, the regression coefficients can be obtained as $\arg\min_{\boldsymbol{\gamma}, \alpha(t), \mathcal{B}} \sum_{i=1}^{n} (y_i - \mathbf{z}_i^T \boldsymbol{\gamma} - \int X_i(t)\alpha(t)dt - \langle \mathcal{B}, \mathcal{W}_i \rangle)^2$. However, there are several concerns that must be considered when estimating the FLRTP model. Firstly, the coefficients to be estimated significantly outnumber the samples. This is mainly due to the large number of tensorial coefficients. Hence, the number of regression coefficients must be shrunk, and an appropriate method handling this issue should be selected. Secondly, the tensorial coefficients may contain a significant number of irrelevant variables that must be discarded. One possible way is to impose sparsity on the tensorial coefficients by penalizing them in the objective function. Thirdly, solving the regression problem with functional and tensorial predictors requires different approaches appropriate for each type of predictors, and there is no unique method that can be employed to obtain the solutions when both predictors exist at the model. These issues will be discussed in details in Subsection 4.2.2.

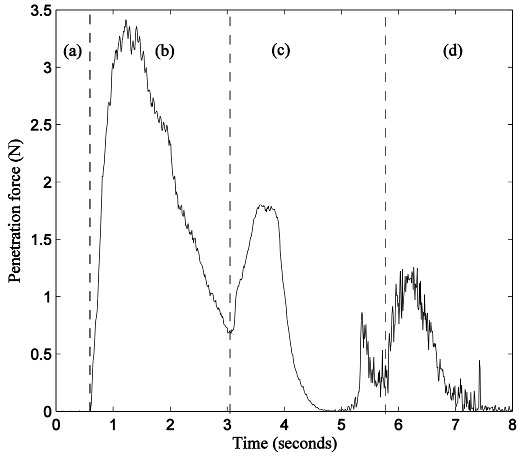


Figure 4-2: Force signal as a function of time (distance)

4.1.1 Motivational Example

In this subsection, a real example of the FSBR process is used to provide a better illustration regarding the FLRTP model. FSBR is a new mechanical fastening process used to join dissimilar materials. This new emerging joining technology combines friction stir riveting with blind riveting. In blind riveting, a rivet is driven into a predrilled hole whose diameter is slightly larger than the external diameter of the rivet. The main disadvantage of blind riveting is the need for a predrilled hole followed by some operations performed to increase the joint's fatigue life. This significantly slows down the joining process. In FSBR, on the other hand, a blind rivet is brought into contact with the upper workpiece. This blind rivet consists of two components. The first part is called mandrel which is a long rod with a head at one end and a weakened, breakable section near the head. The second component (shank) is a hollow tube with enough room for the mandrel (Figure 4-3).

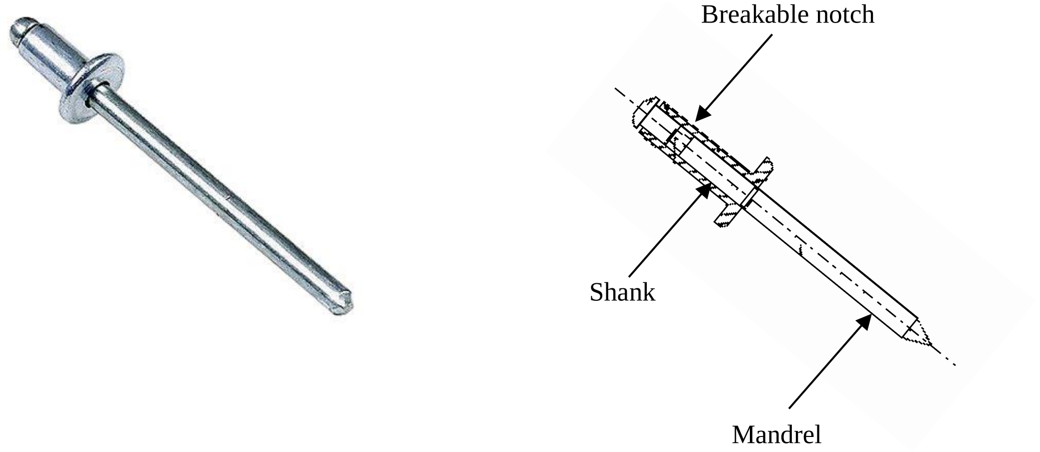
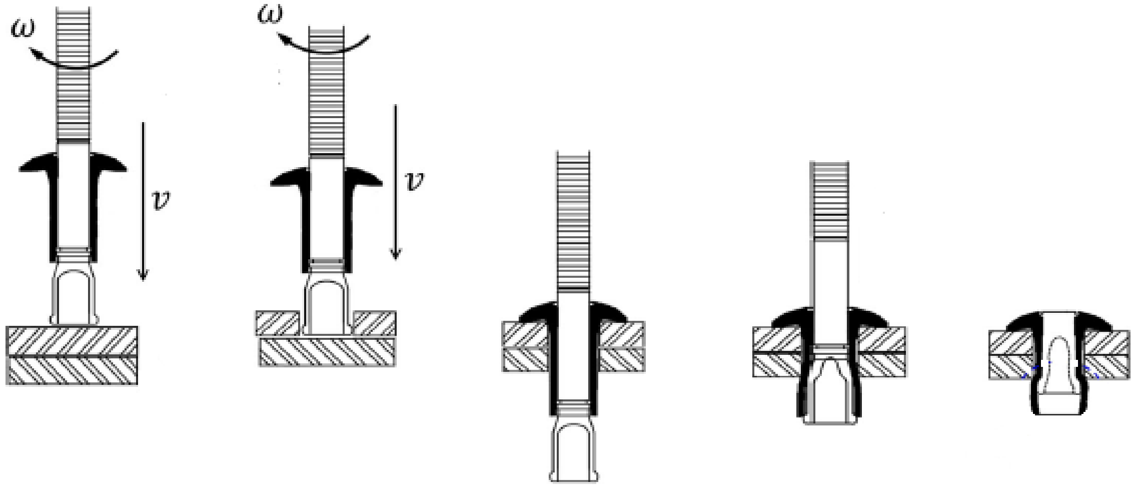


Figure 4-3: Blind rivet components

Figure 4-4 schematically demonstrates all the FSBR process steps. Typically, the rivet is rotating with spindle speed ranging from 1,000 to 20,000 rpm and feed rate spanning between 120 to 780 mm/min. These properties cause the rivet to penetrate the workpieces without requiring predrilling. The high rotational speed of the rivet leads to the frictional heat between the rivet and material, softening the material, and thus avoiding the need for high penetration force.

To further fasten the workpieces, the mandrel is pulled up and broken at the weakened notch. This widens the shank and fastens the workpieces together (Gao et al., 2009; Min et al., 2015). Potentially, the feed rate, spindle speed and configuration are three predictors that might affect the quality of the joint. Hence, it is critical to know whether and how these variables affect the joints quality. There is a set of other variables that are not controllable but measurable during operations. These are called process sensing variables. Examples of these process sensing variables are the penetration force (Figure 4-2), the environment's temperature obtained using the IR images (Figure 4-1), etc.



(a) the blind rivet is put on the workpieces

(b) the rivet is driven into the workpieces with spindle speed *ω* and feed rate *υ*

(c) the rivet is completely inserted in the workpieces with shank's head touching the top workpiece

(d) the mandrel is pulled up

(e) the mandrel is broken and the workpieces are fastened

Figure 4-4: Friction stir blind riveting process (Min et al. 2015)

The penetration force is recorded using load sensor connected to the bottom of the fixture holding the workpieces, and it measures the amount of load as the blind rivet is stirred into the workpieces. Consequently, penetration force is a function of the rivet's penetration distance or time. Figure 4-2 shows the penetration force recorded for AL-CFRP configuration from the moment the blind rivet is put in contact with the top workpiece (stage (a)) until the mandrel is pulled up (stage (d) at the 8th second). As the blind rivet is driven inside AL, the force increases until it completely goes inside the top workpiece (stage (b)). A decreasing trend is observed from Seconds 1.5 to 3.05 when the blind rivet moves in the empty space between the two workpieces. The force starts increasing again in stage (c) as the blind rivet contacts CFRP and the shoulders reach the top material. At stage (d), the mandrel is pulled up; as a result, an increasing then decreasing trend is observed. While Figure 4-2 shows an expected pattern for the penetration force which is in full compliance with the FSBR major steps, sometimes, due to process faults, the penetration function might show a different pattern, like dramatic fluctuations or flat peaks over a long region. These different patterns can be a sign of a damaged or poor-quality join because the quality of joints is sensitive to penetration force.

The IR images, as discussed, are examples of high-order arrays which are the tensorial predictors in the FLRTP model. The temperature of the environment changes both in time and space while the process is running. The pattern of temperature's change is not as clear as that of the penetration force; however, the key point is that a major change in this temperature pattern can indicate a fault in the process, which in turn, leads to low-quality joints. As a result, such process sensing variables can provide useful information about the process operation conditions.

After completing the FSBR process, one way to evaluate the joints quality of the process is to perform the tensile/lap shear test on the joined workpieces. The tensile test usually performed by a testing machine indicates the resistance of materials to stretching or pulling forces. The maximum load that a joint can resist before a failure happens is an informative measure that can be used as a quality metric (Min et al., 2015). Developing a statistical model to predict the maximum tensile load can be tremendously helpful, for it quantifies the importance of process variables in predicting the tensile load or the strength of the joints. This would help developing a statistical model based on some process variables which will be discussed later.

4.1.2 Literature Review and Related Work

There are a limited number of papers discussing the effects of process variables on the tensile test results in the FSBR process. Min et al. (2015) studied the effects of spindle speed and feed rate on the tensile test results, and concluded that feed rate and spindle speed have no effects on the tensile test for the carbon-fiber-reinforced polymer (CFRP) over CFRP or aluminum (AL) configurations. Gao et al. (2009) ran some experiments to compare the strength of the joints under different settings of four variables including feed rate, spindle speed, penetration angle which ideally must be perpendicular, and the holding time for an FSBR process used to join aluminum alloys. The results showed that the joint's strength is significantly affected by neither of the variables. It is noteworthy that none of these papers developed a statistical model to investigate the relationship between the joint strength and the process variables. The results are made based on the load-elongation curves with the aid of visual comparisons among the curves under different settings of these variables.

For the tensorial predictors, there are a few papers focusing on modeling the relationship between a continuous or binary response and a tensorial predictor (Guo et al. 2012). The main idea of Guo et al. (2012) is to reduce the dimensionality of the regression parameters space by applying a decomposition method, like canonical parallel-factor (CP) decomposition, and solving a linear regression problem. However, their algorithm is limited to only one tensorial predictor in the model without other predictors.

Zhou et al. (2013) developed a generalized tensor regression model with an iterative algorithm used to obtain the regression parameters. Similar to Guo et al. (2012), they applied the CP decomposition on the parameter space, leading to a low-dimensional problem. Li et al. (2013) replaced the CP decomposition with a Tucker decomposition in order to attain more flexibility in choosing different numbers of components in each mode. The algorithms in these papers did not explicitly account for other types of predictors in the model (other than tensorial variables). Furthermore, as it will be discussed later, a sub-algorithm is added to the main algorithm in order to obtain the optimal tensorial coefficients at each iteration of the main algorithm. This will increase the accuracy of the main algorithm in obtaining all the coefficients.

A conventional method for solving the FLRTP model is to vectorize the tensorial predictor. At this point, it is possible to look at this vectorized predictor in two different ways. One is to consider the tensorial predictor as a functional variable and use a functional regression approach. Several papers have been written on this topic, which consider both continuous and categorical response variables in relation to functional predictors (James et al., 2009; James, 2002; Ramsay and Silverman, 2002; Muller and Stadtmuller, 2005). This approach will be called stack-up vectorized functional linear regression (VFLR) method.

Another naïve way to model the tensorial predictor is to vectorize the tensorial predictor, and consider it as a scalar variable. The method will be called vectorized linear regression (VLR) throughout the chapter. In this chapter, a new generic model is developed, which can handle both tensorial and functional predictors without manipulating the structure of the predictors.

The remainder of this chapter is laid out as follows: Section 4.2 briefly elaborates on some important multilinear algebraic concepts, and it proceeds with proposing the FLRTP model. An iterative algorithm is, moreover, suggested to solve the problem and obtain the regression parameters. In Section 4.3, a simulation study is conducted to evaluate both the prediction and estimation accuracies of the proposed FLRTP model and the estimation algorithm. The method is also compared to the existing VFLR and VLR approaches. Section 4.4 applies all three methods to a real dataset in FSBR process to further assess the performance of the FLRTP method and show how this model can be employed in practice.

4.2 Basic Multilinear Algebra and FLRTP Problem

Subsection 4.2.1 introduces the CP decomposition with the associated main multilinear algebraic operations used in this chapter. The FLRTP problem is formulated and thoroughly discussed in Subsection 4.2.2. An algorithm is suggested to obtain the solutions for the FLRTP model in Subsection 4.2.3.

4.2.1 High-order Algebraic Operations

The outer product of M vectors $\mathbf{u}^{(m)} \in \mathbb{R}^{K_m}; m = 1,2,\dots,M,$ is defined as $\mathcal{U} = \mathbf{u}^{(1)} \circ \mathbf{u}^{(2)} \circ \dots \circ \mathbf{u}^{(M)}$, where $\mathcal{U} \in \mathbb{R}^{K_1 \times K_2 \times \dots \times K_M}$ is a rank-one tensor with M modes, and "$\circ$" denotes the outer product. Note that if $M = 2$, $\mathcal{U}$ is a two-dimensional matrix or a 2nd-order tensor. It is possible to assume that each Mth-order tensor $\mathcal{W}$ can be approximated by the sum of R rank-one tensors as

$$\widehat{\mathcal{B}} = \sum_{r=1}^{R} \mathbf{u}_r^{(1)} \circ \mathbf{u}_r^{(2)} \circ \dots \circ \mathbf{u}_r^{(M)} \tag{4-2}$$

where $\widehat{\mathcal{B}}$ is an approximation of the original tensor $\boldsymbol{\mathcal{B}}$, R is the selected number of components for the approximated tensor (tensor rank) and $\mathbf{u}_r^{(m)} \in \mathbb{R}^{K_m}; m = 1,2,\dots,M; r = 1,2,\dots,R$. Another helpful way to express Eq. (4-2) is to use the *matricized* version instead of the tensorial form. Matricization of a tensor means to transform a tensor to a matrix. Since a tensor contains M modes, it is possible to have M forms of matricizations. The mode-m matricization of the Mth-order tensor $\mathcal{B}$ is to map the $K_1 K_2 \dots K_M$th element in the tensor to element (k_m, j) of matrix $\mathbf{B}^{(m)}$, where $j = 1 + \sum_{\substack{i=1 \\ i \neq m}}^{M} (k_i - 1) J_i$ and $J_i = \sum_{\substack{l=1 \\ l \neq m}}^{i-1} k_l$. The matricized format of Eq. (4-2) over mode m is defined as

$$\widehat{\mathbf{B}}^{(m)} = \mathbf{U}^{(m)} (\mathbf{U}^{(M)} \odot \mathbf{U}^{(M-1)} \odot \dots \odot \mathbf{U}^{(m-1)} \odot \mathbf{U}^{(m+1)} \odot \dots \odot \mathbf{U}^{(1)})^T \tag{4-3}$$

where $\widehat{\mathbf{B}}^{(m)} \in \mathbb{R}^{k_m \times \prod_{\substack{i=1 \\ i \neq m}}^{M} k_i}$ is the mode-m matricization of tensor $\widehat{\mathcal{B}}$, $\mathbf{U}^{(m)} \in \mathbb{R}^{K_m \times R}$ is called the mode-m factor matrix whose columns are the vectors $\mathbf{u}_r^{(m)} \in \mathbb{R}^{K_m}; r = 1,2,\dots,R$. In addition, if $\mathbf{A} \in \mathbb{R}^{a \times b}$ and $\mathbf{B} \in \mathbb{R}^{c \times b}$, then $\mathbf{C} \in \mathbb{R}^{ac \times b} = \mathbf{A} \odot \mathbf{B}$ is the Khatri-Rao product of matrices $\mathbf{A}$ and $\mathbf{B}$. For the formal proof of Eq. (4-3), please refer to Kolda and Bader (2009).

Similar to a matrix, it is possible to decompose a tensor using one of the common decomposition methods, such as the CP decomposition which is a classical high-order decomposition approach. The CP decomposition problem aims at minimizing the Frobenious norm of the residuals' tensor computed as $\left\|\mathcal{B} - \hat{\mathcal{B}}\right\|$ with $\hat{\mathcal{W}}$ given in Eq. (4-3). The problem can be stated in matrix format as

$$min \left\|\mathbf{B}^{(m)} - \hat{\mathbf{B}}^{(m)}\right\|$$

were $\hat{\mathbf{B}}^{(m)} = \mathbf{U}^{(m)}(\mathbf{U}^{(M)} \odot \mathbf{U}^{(M-1)} \odot \dots \odot \mathbf{U}^{(m-1)} \odot \mathbf{U}^{(m+1)} \odot \dots \odot \mathbf{U}^{(1)})^T$ (4-4)

Knowing the fact that the problem in Eq. (4-4) is similar to a least-squares problem, we can obtain the solution as $\mathbf{U}^{(m)} = \mathbf{W}^{(m)} \left\{ \left(\mathbf{U}^{(M)} \odot \mathbf{U}^{(M-1)} \odot \dots \odot \mathbf{U}^{(m-1)} \odot \mathbf{U}^{(m+1)} \odot \dots \odot \mathbf{U}^{(1)} \right)^T \right\}^+$ where $\mathbf{A}^+$ is the generalized inverse of $\mathbf{A}$. Starting with an initial value for $\mathbf{U}^{(i)}(i = 1,2, \dots, M$ with $i \neq m)$ the problem can be solved for $\mathbf{U}^{(m)}; m = 1,2, \dots, M$. This is the basis for the alternating least squares algorithm that is employed to perform the CP decomposition. The CP decomposition will be used later on to decompose the parameters of the tensorial predictor; however, the decomposition is used here merely to reduce the number of parameters to be estimated and for obtaining the coefficients.

4.2.2 The Functional Linear Regression with Tensorial Predictor Problem

In this subsection, the FLRTP model in Eq. (4-1) is further expanded to be applicable. Subsection 4.2.2.1 adjusts the model to consider the tensorial predictor, and the functional predictor is accounted for in Subsection 4.2.2.2.

4.2.2.1 Tensorial Predictor

The number of regression parameters to be estimated for barely one tensorial predictor is $\prod_{m=1}^{M} K_m$, that can be very large. Hence, an appropriate approach for reducing the number of parameters is to decompose the tensor of parameters using CP decomposition (Guo et al., 2012;

Zhou et al., 2013). From Eq. (4-2) and Eq. (4-3), the tensor of parameters can be decomposed as $\mathcal{B} = \sum_{r=1}^{R} \mathbf{u}_r^{(1)} \circ \mathbf{u}_r^{(2)} \circ \dots \circ \mathbf{u}_r^{(M)}$, Thus yielding

$$\langle \mathcal{B}, \mathcal{W}_i \rangle = \left\langle \mathbf{U}^{(m)} \left(\mathbf{U}^{(M)} \odot \mathbf{U}^{(M-1)} \odot \dots \odot \mathbf{U}^{(m-1)} \odot \mathbf{U}^{(m+1)} \odot \dots \odot \mathbf{U}^{(1)} \right)^T, \mathbf{W}_i^{(m)} \right\rangle =$$
$$\left\langle \mathbf{U}^{(m)}, \mathbf{W}_i^{(m)} \left(\mathrm{U}^{(M)} \odot \mathrm{U}^{(M-1)} \odot \dots \odot \mathrm{U}^{(m-1)} \odot \mathrm{U}^{(m+1)} \odot \dots \odot \mathrm{U}^{(1)} \right) \right\rangle \quad (4\text{-}5)$$

where $\mathbf{U}^{(m)} = \left(\mathbf{u}_1^{(m)}, \mathbf{u}_2^{(m)}, \dots, \mathbf{u}_R^{(m)} \right)$ are the factor matrices for $m = 1,2, \dots, M$, and the last equality in Eq. (4-5) is true because of the linearity property of the inner product. Using Eq. (4-5) and assuming that $\mathbf{U}^{(j)} (j \neq m = 1,2, \dots, M)$, are known, the FLRTP model can be reformulated as

$$y_i = \mathbf{z}_i^T \boldsymbol{\gamma} + \int X_i(t) \alpha(t) dt + \mathrm{vec}\left(\mathbf{U}^{(m)}\right)^T \mathrm{vec}(\mathbf{V}_i) + \varepsilon_i;$$
$$i = 1,2, \dots, n; m = 1,2, \dots, M \quad (4\text{-}6)$$

where $\mathrm{vec}\left(\mathbf{U}^{(m)}\right) \in \mathbb{R}^{RK_m}$ is the column vector of mode-m coefficients for the tensorial predictor, $\mathbf{V}_i = \mathbf{W}_i^{(m)} \left(\mathbf{U}^{(M)} \odot \mathbf{U}^{(M-1)} \odot \dots \odot \mathbf{U}^{(m-1)} \odot \mathbf{U}^{(m+1)} \odot \dots \odot \mathbf{U}^{(1)} \right)$, and $\mathrm{vec}(\mathbf{V}_i) \in \mathbb{R}^{RK_m}$. The advantage of the model in Eq. (4-6) over that of Eq. (4-1) is that the number of tensorial parameters to be estimated is $R \sum_{m=1}^{M} K_m$ for the FLRTP model in Eq. (4-6), which is much less than the number of parameters to be estimated for the model in Eq. (4-1) when the number of elements in each mode increases. As an example, each IR image in the FSBR process is represented by a 3rd-order tensor; hence, the tensorial predictor containing the 3 temperature matrices recorded in three major FSBR operating steps is represented by a 3rd-order tensor $\mathcal{B} \in \mathbb{R}^{133 \times 200 \times 3}$. Using the FLRTP model based on the CP decomposition with $R = 1$ reduces the order of estimation from 79800 $(133 \times 200 \times 3 = 79800)$ to 336 $(133 + 200 + 3 = 336)$ parameters.

4.2.2.2 Functional Predictor

To solve the problem for the functional variables, the common practice is to assume that the coefficient function can be written in terms of a q-dimensional basis function as $\alpha(t) = \boldsymbol{\psi}^T(t)\boldsymbol{\vartheta} + e(t)$ where $\boldsymbol{\psi}(t) \in \mathbb{R}^q$ is a basis function, like spline, Fourier, etc., $\boldsymbol{\vartheta} \in \mathbb{R}^q$ is the vector of coefficients, and $e(t)$ is deviation from the true $\alpha(t)$ (James et al., 2009). In this chapter, the functional linear regression that is interpretable (FLiRTI) method proposed by James et al. (2009) is employed in order to acquire more interpretable estimations of the coefficient function. The FLRTP model in Eq. (4-6) can be rewritten as

$$y_i = \mathbf{z}_i^T\boldsymbol{\gamma} + \mathbf{x}_i^T\boldsymbol{\vartheta} + \mathrm{vec}\left(\mathbf{U}^{(m)}\right)^T \mathrm{vec}(\mathbf{V}_i) + \varepsilon_i + \tilde{\mathbf{e}}_i\vec{\mathbf{1}} \qquad (4\text{-}7)$$

where $\mathbf{x}_i = (\int X_i(t)\boldsymbol{\psi}(t)dt)^T$and $\mathbf{x}_i \in \mathbb{R}^q$, $\tilde{\mathbf{e}}_i = \int X_i(t)e(t)dt$, and $\vec{\mathbf{1}}$ is a q-dimensional vector of ones. The word "interpretable" in the FLiRTI method refers to the fact that it is possible to have more meaningful estimates for the coefficient function $\alpha(t)$ by imposing a specific structure, like sparsity, constancy, linearity etc. on the coefficients (James et al., 2009). This can be achieved by setting the appropriate derivatives of the coefficient function to zero i.e. $\alpha^{(d)}(t) = 0;\ d = 0,1,2, \ldots$. For instance, if $\alpha(t)$ is linear over a specific region, then the second derivative of the coefficient function must be set to zero i.e. $\alpha^{(2)}(t) = 0$ in that region. This can be achieved by penalizing these derivatives in the objective function and obtaining the solution. The difference finite operator obtained for the basis function is proposed by James et al. (2009) to approximate the derivatives. In particular, the dth derivative of the coefficient function is denoted by $D^d\boldsymbol{\psi}(t_r) \in \mathbb{R}^q$, and it is approximated as $D^d\boldsymbol{\psi}(t_r) = q^d \sum_{i=0}^{d}(-1)^i \binom{d}{i} \boldsymbol{\psi}(t_{r-i})$, where $t_r, r = 1,2, \ldots, q,$ are q evenly-spaced points in t.

Let’s define $\mathbf{C} = \left[D^d\boldsymbol{\psi}(t_1), D^d\boldsymbol{\psi}(t_2), \dots, D^d\boldsymbol{\psi}(t_q)\right]^T$ as the matrix of derivatives of all the coefficient functions, the FLRTP model in Eq. (4-7) can be reformulated as

$$y_i = \mathbf{z}_i^T\boldsymbol{\gamma} + \mathbf{s}_i^T\boldsymbol{\varphi} + \mathrm{vec}\left(\mathbf{U}^{(m)}\right)^T\mathrm{vec}(\mathbf{V}_i) + \tilde{\mathbf{e}}_i\vec{\mathbf{1}} + \varepsilon_i \tag{4-8}$$

where $\mathbf{s}_i = \mathbf{C}^{-1}\mathbf{x}_i$ and $\boldsymbol{\varphi} = \mathbf{C}^T\boldsymbol{\vartheta}$. Imposing the sparsity on the vector of coefficients $\boldsymbol{\varphi}$ in Eq. (4-8) is similar to setting the dth derivative of the coefficient function to zero, thus obtaining more interpretable estimates. This can be done using the Elastic Net (EN) method to both obtain sparse estimates and avoid the complications when $q \gg n$, which is the case in the FSBR process (Zou and Hastie, 2005).

In order to have a parsimonious FLRTP model, imposing the sparsity on the functional coefficient vector $\boldsymbol{\varphi}$ and the tensorial coefficients $\mathrm{vec}\left(\mathbf{U}^{(m)}\right)$ seems to be a reasonable solution. For this purpose, the EN method (Zou and Hastie, 2005) is selected over other sparse approaches, like Lasso (Tibshirani, 1996) due to the capability of the EN in generating sparse coefficients and selecting the germane variables even when the number of predictors is highly larger than the number of samples. The objective function of the FLRTP problem can be defined as

$$Minimize \sum_{i=1}^{n}\left(y_i - \mathbf{z}_i^T\boldsymbol{\gamma} - \mathbf{s}_i^T\boldsymbol{\varphi} - \mathrm{vec}\left(\mathbf{U}^{(m)}\right)^T\mathrm{vec}(\mathbf{V}_i)\right)^2 + \lambda_1\|\boldsymbol{\varphi}\|_1 + \lambda_2\|\boldsymbol{\varphi}\|^2 +$$

$$+\lambda_1\left\|\mathrm{vec}\left(\mathbf{U}^{(m)}\right)\right\|_1 + \lambda_2\left\|\mathrm{vec}\left(\mathbf{U}^{(m)}\right)\right\|^2 \tag{4-9}$$

where λ_1 is a nonnegative fixed term penalizing the L_1-norm of the functional (tensorial) coefficients, and λ_2 penalizes the L_2-norm of the functional (tensorial) coefficients. To solve the problem in Eq. (4-9), we divide the main problem in Eq. (4-9) into three sub-problems as

$$\mathrm{S}_1 \coloneqq \sum_{i=1}^{n}(\tilde{y}_{i1} - \mathbf{z}_i^T\boldsymbol{\gamma})^2 + c_1$$

$$\mathrm{S}_2 \coloneqq \sum_{i=1}^{n}(\tilde{y}_{i2} - \mathbf{s}_i^T\boldsymbol{\varphi})^2 + \lambda_1\|\boldsymbol{\varphi}\|_1 + \lambda_2\|\boldsymbol{\varphi}\|^2 + c_2$$

$$\mathrm{S}_3 \coloneqq \sum_{i=1}^{n}\left(\tilde{y}_{i3} - \mathrm{vec}\big(\mathbf{U}^{(m)}\big)^T \mathrm{vec}(\mathbf{V}_i)\right)^2 + \lambda_1 \left\|\mathrm{vec}\big(\mathbf{U}^{(m)}\big)\right\|_1 + \lambda_2 \left\|\mathrm{vec}\big(\mathbf{U}^{(m)}\big)\right\|^2 + c_3 \quad (4\text{-}10)$$

where $\tilde{y}_{i1} = y_i - \mathrm{vec}\big(\mathbf{U}^{(m)}\big)^T \mathrm{vec}(\mathbf{V}_i) - \mathbf{s}_i^T \boldsymbol{\varphi}$, $\tilde{y}_{i2} = y_i - \mathrm{vec}\big(\mathbf{U}^{(m)}\big)^T \mathrm{vec}(\mathbf{V}_i) - \mathbf{z}_i^T \boldsymbol{\gamma}$, and $\tilde{y}_{i3} = y_i - \mathbf{z}_i^T \boldsymbol{\gamma} - \mathbf{s}_i^T \boldsymbol{\varphi}$ are the new response variables for sub-problems S_1, S_2 and S_3, respectively. Each response variable is adjusted based on the solutions for the other two sub-problems. Also, $c_1 = \lambda_1 \left\|\mathrm{vec}\big(\mathbf{U}^{(m)}\big)\right\|_1 + \lambda_2 \left\|\mathrm{vec}\big(\mathbf{U}^{(m)}\big)\right\|^2 + \lambda_1 \|\boldsymbol{\varphi}\|_1 + \lambda_2 \|\boldsymbol{\varphi}\|^2$, $c_2 = \lambda_1 \left\|\mathrm{vec}\big(\mathbf{U}^{(m)}\big)\right\|_1 + \lambda_2 \left\|\mathrm{vec}\big(\mathbf{U}^{(m)}\big)\right\|^2 + \mathbf{z}_i^T \boldsymbol{\gamma}$, and $c_3 = \lambda_1 \|\boldsymbol{\varphi}\|_1 + \lambda_2 \|\boldsymbol{\varphi}\|^2 + \mathbf{z}_i^T \boldsymbol{\gamma}$ are some constants that can be obtained based on the solutions of the sub-problems. This approach apparently requires an iterative algorithm to attain the solutions. In the next subsection, a detailed algorithm is proposed to obtain the solution of the FLRTP problem.

4.2.3 An Iterative Algorithm for Solving FLRTP Problem

This subsection provides an algorithm that can be employed for computing the regression parameters of the FLRTP model. The basic principle in every iteration is to solve the sub-problems S_1, S_2 and S_3 sequentially. The algorithm can start with S_1 sub-problem. Ordinary least squares (OLS) method is used to calculate the coefficients for the scalar predictors. The residuals are calculated and the EN method is applied to the residuals to compute the coefficient functions. To get the solutions for sub-problem S_3, unlike the common approach typically used in the tensorial regression literature, we suggest computing the optimal values of $\mathrm{vec}\big(\mathbf{U}^{(m)}\big); m = 1,2,\dots,M,$ using an iterative sub-algorithm similar to the ALS algorithm proposed by Carroll and Chang (1970) with the modification that the algorithm is alternating between EN problems instead of the least-square estimation problems. The algorithm is called alternating elastic net (AEN), and it is shown in Figure 4-5. This will lead to more precise estimates for both the functional and tensorial variables. It is noteworthy that the EN penalty terms and the optimal degree of the derivatives (d) can be obtained based on cross-validation techniques. The main computational algorithm is presented in Figure 4-6. It is noteworthy that the logic of the algorithm in moving along different sub-problems is similar to what is known as block-wise power algorithm to find the eigenvalues in a singular value decomposition problem. The similarity of this algorithm and the power algorithm with deflation is that in each sub-problem, say S_1, $\boldsymbol{\gamma}$ plays the role of the singular vector in the first component, and the rest of coefficients ($\boldsymbol{\varphi}$ and $\mathbf{U}^{(\boldsymbol{m})}$) are similar to the singular vectors in the second component.

```
Initialization step
At each iteration $it$ of the main algorithm:
    1. Compute $\tilde{y}_{i3} = y_i - \mathbf{z}_i^T \hat{\boldsymbol{\gamma}}^{it-1} - \mathbf{s}_i^T \hat{\boldsymbol{\varphi}}^{it-1}; i = 1,2,\dots,n$. When $it = 1$, $\hat{\boldsymbol{\gamma}}^0 = \vec{\mathbf{0}}$, $\hat{\boldsymbol{\varphi}}^0 = \vec{\mathbf{0}}$.
    2. Set $\hat{\mathbf{U}}_{it}^{(j)} (j \neq m = 1,2,\dots,M)$ to random matrices when $it = 1$; otherwise, $\hat{\mathbf{U}}_{it}^{(j)} = \hat{\mathbf{U}}_{it-1}^{(j)}$. Calculate
    $\mathbf{V}_i^{it} = \mathbf{W}_i^{(m)} \left( \hat{\mathbf{U}}_{it}^{(M)} \odot \hat{\mathbf{U}}_{it}^{(M-1)} \odot \dots \odot \hat{\mathbf{U}}_{it}^{(m-1)} \odot \hat{\mathbf{U}}_{it}^{(m+1)} \odot \dots \odot \hat{\mathbf{U}}_{it}^{(1)} \right)^{\mathrm{T}}; i = 1,2,\dots,n$.

Alternating EN step (AEN)
At each iteration $it_s = 1,2,\dots,$ of the sub-algorithm:
    1. For $m = 1,2,\dots,M$, compute $\hat{\mathbf{U}}_{it}^{(j)}$ using EN method with inputs $\tilde{y}_{i3}$ and $\mathbf{V}_i$ for $i = 1,2,\dots,n$.
    2. Calculate $\hat{\mathcal{B}}^{it_s} = \sum_{r=1}^{R} \hat{\mathbf{u}}_r^{(1)} \circ \hat{\mathbf{u}}_r^{(2)} \circ \dots \circ \hat{\mathbf{u}}_r^{(M)}$ with $\hat{\mathbf{u}}_r^{(m)}$ as the $r$th column of $\hat{\mathbf{U}}_{it}^{(m)}$.
Stopping criterion
At each iteration $it_s$ of the sub-algorithm:
    1. Compute the relative estimation difference for two consecutive iterations as $d^{it_s} = \frac{\|\hat{\boldsymbol{\mathcal{B}}}^{(it_s)} - \hat{\boldsymbol{\mathcal{B}}}^{(it_s-1)}\|}{\|\hat{\boldsymbol{\mathcal{B}}}^{(it_s-1)}\|}$
    2. Stop the sub-algorithm and return $\hat{\mathbf{U}}_{it}^{(j)} (j = 1,2,\dots,M)$, and $\mathbf{V}_i^{it} (i = 1,2,\dots,n)$, if $d^{it_s} < \epsilon_s$, where $\epsilon_s$ is
    a small, predefined value; otherwise, go to the AEN step.
```

Figure 4-5: AEN algorithm to compute the tensorial coefficients

```
Main computational step
At each iteration $it = 1,2,\dots,$ of the algorithm:
For $i = 1,2,\dots,n$,
    1. Compute $\tilde{\mathrm{y}}_{\mathrm{i}1}$ as $\tilde{\mathrm{y}}_{\mathrm{i}1} = \mathrm{y}_{\mathrm{i}} - \mathrm{vec}\left(\hat{\mathbf{U}}_{it-1}^{(m)}\right)^{\mathrm{T}} \mathrm{vec}\left(\mathbf{V}_i^{it-1}\right) - \mathrm{s}_{\mathrm{i}}^{\mathrm{T}} \hat{\boldsymbol{\varphi}}^{it-1}$ and update $\hat{\boldsymbol{\gamma}}^{it}$ using OLS applied to the pairs
    $(\tilde{y}_{i1}, \mathbf{z}_i)$.
    2. Calculate $\tilde{\mathrm{y}}_{\mathrm{i}2} = y_i - \mathrm{vec}\left(\hat{\mathbf{U}}_{it}^{(m)}\right)^{\mathrm{T}} \mathrm{vec}\left(\mathbf{V}_i^{it}\right) - \mathbf{z}_i^T \hat{\boldsymbol{\gamma}}^{it}$ and apply the EN method to obtain $\hat{\boldsymbol{\varphi}}^{it}$ using the
    pairs $(\tilde{y}_{i2}, \mathbf{s}_i)$.
    3. Calculate $\tilde{y}_{i3} = y_i - \mathbf{z}_i^T \hat{\boldsymbol{\gamma}}^{it-1} - \mathbf{s}_i^T \hat{\boldsymbol{\varphi}}^{it-1}$ and obtain $\hat{\mathbf{U}}_{it}^{(j)} (j = 1,2,\dots,M)$, and $\mathbf{V}_i^{it}$ using AEN sub-algorithm in
    Figure 4-5.
Stopping criterion
    1. Calculate the residuals sum of squares as $e^{(it)} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^{\mathbf{2}}$ with $\hat{y}_i = \mathrm{vec}\left(\hat{\mathbf{U}}_{it}^{(m)}\right)^T \mathrm{vec}\left(\mathbf{V}_i^{it}\right) + \mathbf{z}_i^T \hat{\boldsymbol{\gamma}}^{it} +$
    $\mathbf{s}_i^T \hat{\boldsymbol{\varphi}}^{it}$.
    2. Obtain the difference between two consecutive $e^{(it)}$'s as $er^{(it)} = \left|e^{(it)} - e^{(it-1)}\right|$.
    3. For a small value of $\epsilon$, stop the algorithm in case $er^{(it)} < \epsilon$, and return $\hat{\mathbf{U}}_{it}^{(m)}; (m = 1,2,\dots,M)$, $\hat{\boldsymbol{\gamma}}^{it}$ and $\hat{\boldsymbol{\varphi}}^{it}$.
```

Figure 4-6: The main computational algorithm for FLRTP problem

4.3 A Simulation Study

In this section, the performance of the proposed FLRTP approach is evaluated using a comprehensive simulation study. The performance of the FLRTP method is reflected by

the prediction accuracy of the FLRTP method is evaluated based on the root-mean-square errors (RMSE).

Three scalar predictors included in the FLRTP model are spindle speed, feed rate and configuration are used. The values scalar predictors are set based on the potential feed rate and spindle speed levels that are possibly used in practice. That is, for every sample, the feed rate takes one of the two possible values of 120 or 420 $\mathrm{mm/min}$, and spindle speed is either 3000 or 5000 rpm. Configuration is a binary variable showing that which material is on the top. The vector of coefficients is set to $\boldsymbol{\gamma} = (-2,0.4,5)^T$ by considering the scale of each of the predictors. To generate the tensorial predictor, a surrogated multilinear model is used as

$$\mathcal{W}_i = \mathcal{M} + \mathcal{E}_i; i = 1,2,\dots,n, n = 500, \tag{4-12}$$

where $\mathcal{W}_i$ is the ith simulated tensor, $\mathcal{M}$ is a tensor of the temperature obtained using the IR image shown in Figure 4-1. Also, $\mathcal{E}_i$ is a tenor-variate normal random error with mean zero tensor and mode-m covariance matrix of $\sigma_f^2 \mathbf{I}^{(m)} \in \mathbb{R}^{K_m \times K_m}; m = 1,2,3$ with $\mathbf{I}^{(m)}$ being the identity matrix for mode m. In this simulation study, σ_f^2 is specifically set to 0.5. Furthermore, the number of elements in each mode is defined as $K_1 = 20$ (number of rows) , $K_2 = 10$ (number of columns) and $K_3 = 3$ (number of major FSBR operating steps). The tensor of coefficients is simulated using the model in Eq. (4-2) with $R = 1$ in two different scenarios. Mode-1's tensorial coefficients representing the row signal are defined as $\mathrm{b}_1^{(1)} = \cos(\pi \mathrm{x}_{11})$ with $\mathbf{x}_{11} \in (1,2)$ and $\mathrm{b}_2^{(1)} = \begin{cases} 0 & |\mathbf{x}_{12}| > 0.5 \\ 0.5 & |\mathbf{x}_{12}| = 0.5 \\ 1 & |\mathbf{x}_{12}| < 0.5 \end{cases}$. For Mode 2, the column signals are defined as $\mathrm{b}_1^{(2)} = \sin(\frac{\pi \mathrm{x}_{21}}{2})$ with

$\mathbf{x}_{21} \in (-4,4)$ and $\mathrm{b}_2^{(2)} = \begin{cases} 0 & |\mathbf{x}_{22}| \geq 1 \\ 1 - |\mathbf{x}_{22}| & |\mathbf{x}_{22}| < 1 \end{cases}$. Mode-3 coefficients are the effects of temperature changes in the three major FSBR operating steps on the product quality response variable. These parameters are set to $\mathbf{b}^{(3)} = (150,220,40)^T$ showing the fact that the effect of temperature's changes on the quality response variable when the rivet is penetrating inside the workpieces is higher than the effect when the FSBR process is close to the final stage where the mandrel is pulled up. The coefficient tensors are computed as $\mathcal{B}_1 \in \mathbb{R}^{20\times10\times5} = \mathrm{b}_1^{(1)} \circ \mathrm{b}_2^{(2)} \circ \mathrm{b}^{(3)}$ and $\mathcal{B}_2 \in \mathbb{R}^{20\times10\times5} = \mathrm{b}_2^{(1)} \circ \mathrm{b}_1^{(2)} \circ \mathrm{b}^{(3)}$.

For the functional predictor, two different functions are considered for both the predictors and coefficient functions. To have a more realistic simulation study, the first functional predictor is generated using real penetration force data. Second-degree B-spline curve is fitted to the original penetration force. The functional penetration force is then simulated using the fitted B-spline curve as

$$X_{i1}(t) = \widetilde{\boldsymbol{\psi}}\widetilde{\boldsymbol{\vartheta}} + \xi_i; i = 1,2,\dots,n, \tag{4-13}$$

where $\widetilde{\boldsymbol{\psi}} \in \mathbb{R}^{q\times q}$ gives the basis function values, $\widetilde{\boldsymbol{\vartheta}} \in \mathbb{R}^q (q = 200)$ is the vector of B-spline coefficients, and $\xi_i \sim N(0,1)$ is a random error term. The second functional predictor is given as $X_{i2}(t) = asin(2\pi t) + bsin(4\pi t)$ with a and $b \sim N(0,1)$ and $0 \leq t \leq 1$. The coefficients functions $\alpha_1(t)$ and $\alpha_2(t)$ are generated using the polynomial functions defined as

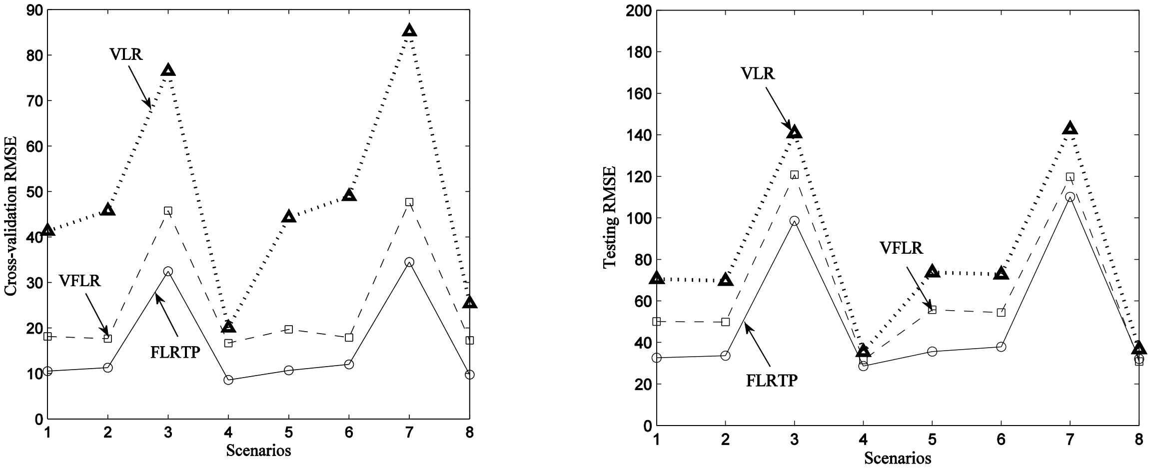
$$\alpha_1(t) = \begin{cases} (t - 0.5)^2 - 0.025 & 0 \leq t < 0.34 \\ 0 & 0.3 \leq t \leq 0.64 \\ -(t - 0.5)^2 + 0.025 & 0.64 < t \end{cases}$$

$$\alpha_2(t) = t^3 - t^2 \qquad 0 \leq t \leq 1$$

In total, eight different simulation scenarios (two different sets of tensorial coefficients and four scenarios for the functional variable) are considered in this study to evaluate the performance of the model. After generating all the predictors and their corresponding coefficients, the maximum tensile load (scalar quality response variable) is simulated using Eq. (4-1) and $\varepsilon_i \sim N(0, 0.5^2); (i = 1,2, \dots, n),$ with $n = 500$ training samples are generated and used to estimate the model parameters by using the main algorithm given in Figures 4-5 and 4-6. The performance of the FLRTP model is also compared to the stack-up VFLR and VLR methods. For comparison purposes, the Mode-1 to Mode-3 coefficient vectors (for the tensorial predictor) estimated using FLRTP approach are used to calculate the vector of coefficients $\hat{\mathbf{b}} \in \mathbb{R}^{K_1 K_2 K_3}$ similar to that of the vectorization-based regression methods i.e., $\hat{\mathbf{b}} = \hat{\mathbf{b}}^{(1)} \odot \hat{\mathbf{b}}^{(2)} \odot \hat{\mathbf{b}}^{(3)}$.

Ten-fold cross validation was performed to obtain appropriate values for the penalization parameters λ_1, λ_2 and the order of the coefficient function's derivative (d) by minimizing RMSE. As pointed out, the prediction accuracy of the methods is evaluated using the RMSE criterion based on the testing dataset. The testing RMSE is computed after training the model, performing cross-validation, simulating 1000 new testing datasets of predictors, and calculating testing RMSEs using the computed parameters. Figure 4-7 gives the best cross-validation and testing RMSEs. As it is notable, the FLRTP method uniformly yields the lowest RMSEs in all 8 scenarios. As expected, the VFLR method seems to be more effective than VLR in all scenarios.

Based on the results of the simulation study, it can be concluded that in the cases where the predictors contain functional and tensorial variables, the FLRTP method performs better than the VFLR and VLR methods in more precisely predicting the response variable of interest.



(a) Cross-validation RMSEs

(a) Testing RMSEs

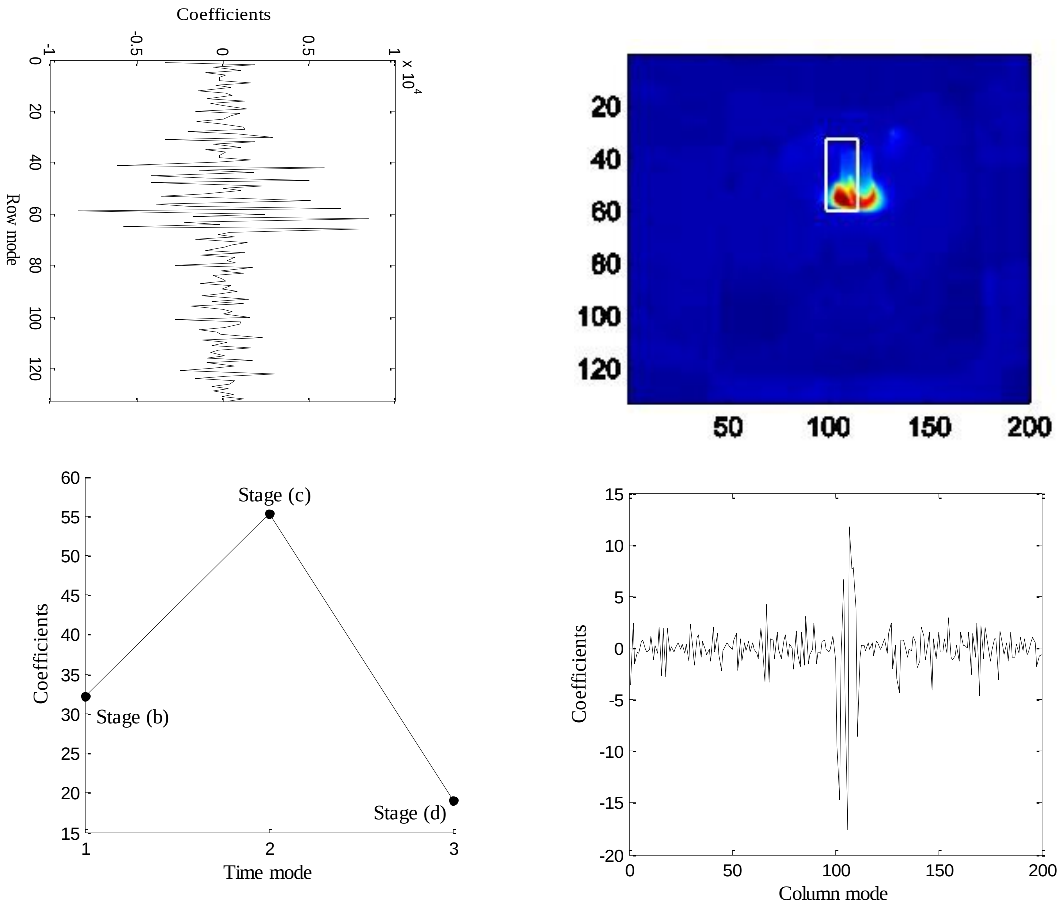
Figure 4-7: Cross-validation and testing RMSEs computed for FLRTP, VFLR and VLR methods

4.4 Case Study

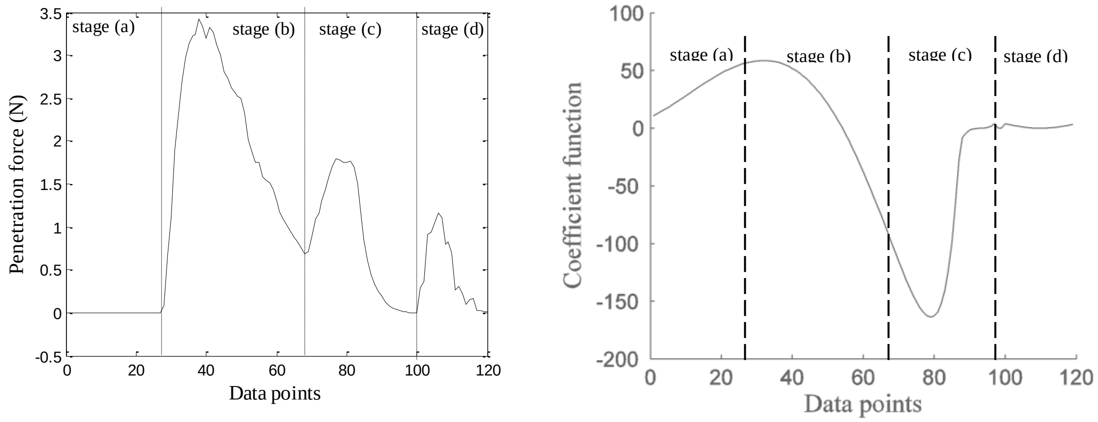
This section applies the FLRTP method on a real dataset from the FSBR process to show how the method can be utilized for practical applications. The data regarding the FRBR process variables of the AL/CFRP combination are used to validate the FLRTP model. The dataset contains 28 samples with the maximum tensile load as the quality response variable, 28 sets of penetration force as the functional predictor with $q = 100$ data points representing stage (b), the first part of stage (c) belonging to the movement of the rivet inside CFRP, and stage (d). All 28 samples are normalized to have the same time (location) index as $0 \leq t \leq 1$. For the tensorial predictor, 3 images are selected during the period when the rivet is penetrating and the mandrel is pulled up. These images are selected to reflect the temperature of the environment in the essential steps of the FSBR process. Three scalar predictors are the feed rate with two levels of 120 and 420 mm/min, spindle speed at 3000 and 5000 rpm, and configuration as an indicator variable showing whether AL is on top or not (the indicator is one in this case).

The FLRTP and VFLR approaches are applied to the data and the regression parameters are computed. Out of the 28 samples, 23 samples are selected to perform the ten-fold cross-validation in order to obtain the optimal penalization parameters λ_1 to λ_4 and the derivatives' order d. These parameters are determined by minimizing the cross-validation RMSE's which are computed based on 100 replications of the whole cross-validation procedure. The final testing RMSEs are computed employing the rest of the 5 samples. The whole testing procedure is repeated 50 times and the average of the testing RMSE's is calculated for the comparison purposes. For this dataset, the optimal parameters are attained as $\lambda_1 = 4$, $\lambda_2 = 0.1$, and $d = 3$. Figure 4-8 (a) shows the estimated Mode 1 to Mode 3 tensorial coefficients and the coefficient function for the penetration force using the FLRTP method. As it is noticeable in Figure 4-8 (a), Mode 1 (row) and Mode 2 (column) coefficients have several zero components, showing that the majority of the rows and columns in the IR images are not important for predicting the maximum tensile load. As it is highlighted on the IR image in Figure 4-8 (a), the temperature changes in the neighborhood of the rivet when it penetrates into the workpieces seem to have more effect on the maximum tensile load. Mode 3 (time mode) coefficients show that while the rivet is penetrating (second point in the time mode), the tensile load is more sensitively affected than when the mandrel is pulled up (third point in the time mode) as shown in Figure 4-8 (a). Figure 4-8 (b) gives the coefficient function with one sample of functional predictor. The amount of coefficients increase as the rivet goes through the top workpiece and it decreases as the rivet moves the space between the two workpieces. When the blind rivet gets to the second workpiece, the coefficients start decreasing until the rivet passes through the second workpiece (Stage (b) in Figure 4-4). The coefficient function moves to zero for the rest of process (Stage (d) in Figure 4-4). The vector of coefficients for the feed rate, spindle speed and configuration is computed as $\hat{\boldsymbol{\gamma}} = (6.11, 0.93, 442.11)^T$ with the standard error of 0.12, 0.01 and 25.68, respectively.

For the FLRTP model, the RMSE is computed as 800.05 with the standard error 24.36. The RMSE of the VFLR method is calculated as 1274.36 with the standard error 69.85. Since the difference between the RMSEs is significant, the FLRTP method is expected to be a better approach for predicting the maximum tensile load. The range of the response variable is about 2200, showing that the error is roughly 36% of the range. It is noteworthy that although $R = 1$ is used for these results, different values for the number of components $R = 3,5$ are also studied, and the testing RMSEs do not show significant improvements over the case of $R = 1$.



(a) Tensorial coefficients



(b) Functional predictor

Figure 4-8: Estimated coefficients for the tensorial and functional predictors

References:

1. Carroll, J. D., Chang, J. J., (1970), “Analysis of individual differences in multidimensional scaling via an N-way generalization of “Eckart-Young” decomposition,” *Psychometrika*, 35 (1970), pp. 283–319.

2. Gao, D., Ersoy, U. , Stevenson, R. , and Wang, P.-C. , (2009), “A New One-Sided Joining Process for Aluminum Alloys: Friction Stir Blind Riveting,” *ASME J. Manuf. Sci. Eng*., 12(131), p. 061002.

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

4. James, G. M., (2002), “Generalized linear models with functional predictors,” *J. R. Stat. Soc. Ser. BStat. Methodol*. 64 411–432. MR1924298.

5. James, G., Wang, J., and Zhu, J. (2009), “Functional Linear Regression That’s Interpretable,” *The Annals of Statistics*, 37, 2083–2108.

6. Kolda, T., and Bader, B., (2009), “Tensor decompositions and applications,” *SIAM Rev*, 51(3), pp.455-500.

7. Li X., Zhou H., Li L., (2013), “Tucker tensor regression and neuroimaging analysis,” Available at arXiv:1304.5637.

8. Min, J. , Li, J. , Li, Y. , Carlson, B. E. , Linc, J. , and Wang, W.-M. , (2015), “Friction Stir Blind Riveting for Aluminum Alloy Sheets,” *J. Mater. Process. Technol*., 215, pp. 20–29.

9. Muller, H. G., and Stadmuller, U., (2005), “Generalized functional linear models,” *Ann. Statist*.33, 774–805. MR2163159.

10. Ramsay, J. O. and Silverman, B. W., (2002), "Applied Functional Data Analysis," Springer, New York. MR1910407.

11. Tibshirani, R. (1996), "Regression Shrinkage and Selection via the Lasso," *Journal of the Royal Statistical Society*, Series B, 58, 267–288.

12. Zerehsaz, Y., Shao, C., Jin, J., (2016), "Tool Wear Monitoring in Ultrasonic Welding Using High-order Decomposition," *Journal of Intelligent Manufacturing*, DOI: 10.1007/s10845-016-1272-4.

13. 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.

14. Zou, H., and Hastie, T. (2005), "Regularization and Variable Selection via the Elastic Net," *Journal of the Royal Statistical Society*, Series B, 67, 301–320.

CHAPTER 5

CONCLUSIONS AND FUTURE RESEARCH

5.1 Conclusions and Contributions

This dissertation aims to develop new data analytics methods for effectively utilizing multi-stream sensing signals or image data for improving manufacturing process control and product quality. While there are many challenging issues in analyzing such complex data, this dissertation particularly focuses on the following three critical issues. The first issue is how to effectively analyze sensor data with a multi-stream structure for process or tooling condition monitoring. For this purpose, a high-order-decomposition-based SVD (HSVD) method is developed, which is used to effectively extract a low dimensional monitoring features from a high dimensional multi-stream data of tool wear measurements. The second issue is to deal with image data, which may contain some outliers and correlated noise components. In this aspect, in addition to proposing a new robust decomposition method for a traditional data representation using a low order stack-up approach, this dissertation has further extended the proposed method to be applicable to a high-order data representation. Finally, the third issue is to develop a quality response model using a functional linear regression, which considers mix-types of predictors including unitary, functional, and tensorial process variable. A summary of the major results and new contributions with regard to each chapter is provided below:

(1) *A new high-order-decomposition-based control chart for monitoring tool wear data*: A high-order-decomposition-based control chart is proposed in Chapter 3 for monitoring the tool wear in ultrasonic metal welding process. The proposed HOSVD method for extracting

monitoring features not only can help reduce the data dimensionality in the multivariate control charts but also can provide a better diagnostic inference by keeping the original data's cross-correlation structure. The developed method is applied to tool wear monitoring in the ultrasonic welding process, which shows a superior benefit in detecting slight wear. In addition, the HOSVD method can help automatically discover the unusual knurls' wear patterns among different rows of knurls in an anvil. Studying knurls' wear patterns on the anvil can further help identify and remove potential root causes of excessive or unusual wear of the anvil. The simulation studies have also conducted to show the better monitoring and diagnostic performance of the proposed HSVD method compared to the existing stack-up SVD method.

(2) *New robust GSVD decomposition methods with outliners and correlated noise*: low-order and high-order robust SVD decomposition methods are developed to be robust to potential outliers that might exist in an image dataset. In addition to handling outliners, the proposed method can also be applicable to the datasets with spatiotemporally correlated noise components. In contrast, the existing decomposition methods fail to work properly when the dataset contains outliner observations and structured noises simultaneously. Furthermore, the proposed high-order robust generalized decomposition method (RGHOSVD) can more effectively factorize the tensorial structure of data when the number of variables in each dimension (mode) is larger than the number of samples, and meanwhile there is a high correlation that must be preserved among the elements of each mode. The simulation studies in Chapter 4 show that the proposed methods (RGHOSVD and RGSVD) not only outperform other existing decomposition methods in accurately estimating the true singular vectors, but also achieve a higher monitoring performance compared to other commonly used methods in image-based monitoring. The proposed methods

have also been effectively applied to image data monitoring for products' surface defects detection in the rolling process.

(3) *A new functional linear regression model with mix-type of scalar, functional and tensorial predictors*: a functional linear regression model is developed to regress a scalar quality response variable on a set of mix-type predictors including scalars, functional and tensorial variables. The current models merely focus on solving the problem with either a tensorial or a functional predictor. As a result, there is a need for a generic model that can consider both types of predictors. A new estimation algorithm is proposed in Chapter 4 and the simulations have been conducted under different scenarios, which show that the proposed method (FLRTP) outperforms the current stack-up vectorization-based methods (VLR and VFLR) in the both aspects of accurately estimating the model parameters and yielding higher prediction accuracy. The FLRTP method is also applied to a FSBR joining process for modeling the relationship between joints quality and process variables which include various process set-up variables and process sensing signals.

5.2 Future Research

Data analytics for process modeling and variation monitoring and fault diagnosis is highly demanded in manufacturing industry for improving process control and product quality. Developing effective data analysis methods with the ability of handling complex data characteristics, such as multi-stream structure, outliers, correlated noise, etc. is one of the research topics studied in this dissertation. Having generic predictive models with the ability of incorporating mix-type of predictors is another critical issue investigated in this dissertation. Although this dissertation has performed some initial work in these areas, there are still many other topics that need to be further scrutinized in this direction. A few examples of these topics are enumerated as follows:

- The robust decomposition methods that are proposed in Chapter 3 can work with row-wise outliers. This means, the methods can deal with datasets that contain samples of outliers. In the image data, an outlier sample is an abnormal image with unclear visibility, and the decomposition method discards these outliers. On the other hand, sometimes the images may have unclear visibility in a local region of the image and the rest of the image is clear. This type of outliers is called elementwise outliers, and it would be better to remove these outlier elements instead of discarding the whole sample of image. An advanced robust decomposition methods will be needed to account for correlated noise and handle such elementwise outliers.
- In functional linear regression literature, there are some situations where the response variable has a functional or a tensorial structure instead of a scalar response variable. This problem is worth more considerations since there is no such a model and estimation algorithm in the literature that can handle functional or tensorial response variables with mixed types of predictors.
- In the proposed FLRTP model, the assumption is that the random error terms are independently distributed. In some situations, this independence assumption does not hold. An example of this situation is when the response variable is measured in small time interval, which leads to an autocorrelation problem. A generic model is needed to consider autocorrelated random error terms.