GANTRY HEALTH MONITORING AND FAULT DETECTION BASED ON PROCESS STATUS SEQUENCE

By

RUI SONG

A thesis submitted to the

School of Graduate Studies

Rutgers, The State University of New Jersey

In partial fulfillment of the requirements

For the degree of

Master of Science

Graduate Program in Industrial and Systems Engineering

Written under the direction of

Weihong Guo and Myong K. Jeong

And approved by

_____

_____

_____

New Brunswick, New Jersey

January, 2019

ABSTRACT OF THE THESIS

Gantry Health Monitoring and Fault Detection Based on Process Status Sequence

by RUI SONG

Thesis Director:

Weihong Guo and Myong K. Jeong

Gantry refers to the system that moves the hoist by the machinery house along tracks on the floor level and transfers the material. As a critical asset, gantry has wide applications in many fields such as medical image area, infrastructure, and heavy industry. Mostly, gantry is reliable, however, the loss led by the gantry lockout is inestimable enormous. Moreover, there are limited previous gantry studies concentrate on the statistical quality control to detect the fault not to mention the research that focuses on the algorithms applied to the process status sequence to detect the fault. The categorical process status sequence is hard to obtain the features when dealing with fault identification. This thesis provides a novel method applying texture extraction in image processing to obtain the features of gantry process status sequence. Texture extraction techniques such as the histogram of oriented gradients (HOG) and local binary pattern are applied to the process status sequence. To demonstrate the effectiveness of image-based feature extraction, *k*-nearest neighbors, support vector

machine, linear discriminant analysis, and quadratic discriminant analysis are applied to the time-series gantry process status sequences provided by a leading automobile manufacturer. Result demonstrates that the sequence after the transformation of both texture extraction techniques have improved the accuracy. The process status sequence after HOG transformation has the best performance. Besides, the HOG technique also dramatically reduces the dimension of the process status sequence. This result can help the on-site expert prognosis the fault as well as prepare the corresponding troubleshooting guide to save time and resources.

Table of Contents

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

Introduction

1.1 Overview

Gantry refers to the system that moves the hoist by the machinery house along tracks on the floor level and transfers the material by loading, moving, rotating, and unloading. The whole gantry system consists of mechanical parts, structural parts, and electrical parts. In the structural scope, gantry crane is mainly composed of the mast structure, truck, carts operating agencies, electrical equipment, cabs and other components. In the electrical parts, modern gantry crane is combined with various modules so that different needs may meet. For illustration, gantry can be equipped with articulated arm robot module for high-speed automation. Such combination is called gantry robot system that is developed for storage and retrieval, machine loading, parts handling and automated assembly. Besides, gantry can be equipped with high-end computer operating system for the workers to improve the positioning performance of high-dynamic motion systems. This system increases the throughput by greatly reduces the effects of frame motion on the servo system. Figure 1.1 shows the adapted flow chart of one common gantry process from setting speed, move, rotate, load, to unload (Khan *et al.* 2011). When an instruction arrives at the gantry control system, the embedded logic control section determines under what condition the gantry should change its current speed, move to a new position, rotate to designated angle, load or unload the robotic arm at designated position. If the condition is true, the mechanical, structural, and the electrical part would function together to convey the specified requirement.

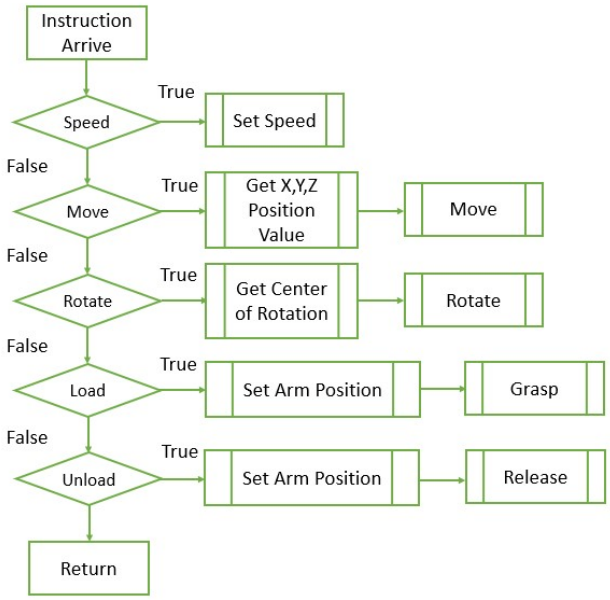

Figure 1.1 Flow Chart of Gantry Working Process Adapted from (Khan *et al.* 2011)

As a critical asset, the gantry is widely used in many fields such as heavy industry and medical image area. In the field of heavy industry, the gantry crane is widely used for transferring bulk material from one place to another. The applications of gantry greatly relief the manual work and help the precision operations. Guo *et al.* (2017) developed a diagnostic system by extending the basic attributes control charts and helped saving unnecessary waiting time in the field. In the medical image area, the gantry is a frame facility includes radiation, collimators, and detectors in a CT machine and the corresponding mounted diagnostic can be applied through taking 3D radiographs, real-time tumor tracking, thus predict disease (Berbeco *et al.* 2004, Oppelt 2011, Si *et al.* 2013).

Gantry is a critical asset along the single line for the upstream and downstream workstation, and the status of gantry dramatically influences the factory-wise operations. If the gantry is down, all processes relied on gantry will be delayed or

postponed. Not only labor intensity of loading and unloading workers will be increased. Moreover, the line throughput will be greatly decreased. From Figure 1.2, if Process E-1 is down, the working line can still function well by transfer the material to Process E-2. While if gantry is down, the whole working line would be dramatically affected.

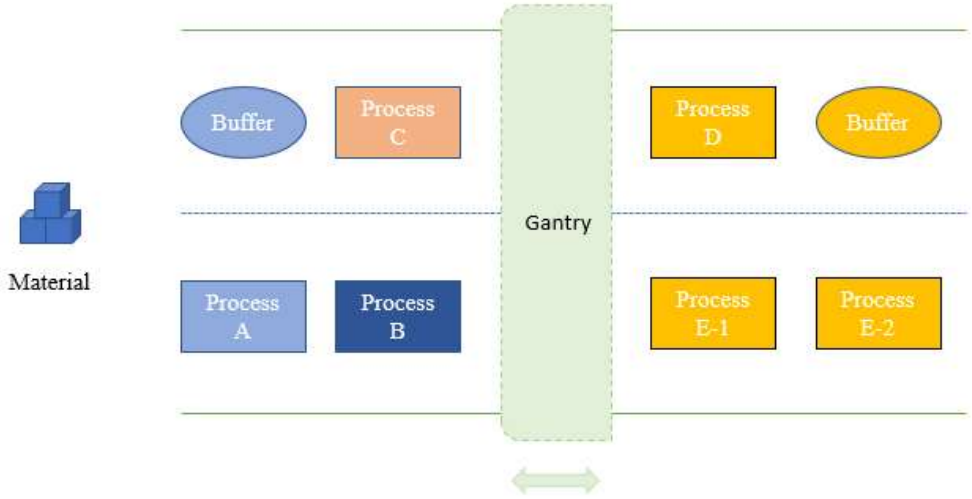

Figure 1.2 Example of the Gantry Movement Track

In production, the time-series gantry process status sequence is usually in categorical type. For the categorical sequence, “blocked”, “cycling”, “down”, and “starved” are the most common statuses. Blockage occurs when the events must end because there is no location for placing the item just completed; machine starvation occurs when the activities must stop because there is no work. Machine blockage and starvation not only degrade the process flow, but also cause the production loss, money and energy waste.

For better understanding, Figure 1.3 illustrates one case of the inefficient manufacturing planning on a single line led by starvation and blockage. Figure 1.4 shows the percentage of cycling in 10 minutes. The needed cycling time for machine 1, machine

2, and machine 3 are 7 minutes, 10 minutes, and 6 minutes, respectively. Suppose each machine can only deal with one process at a time. If Machine 1 finishes its progress and passes the product to the machine 2, the machine 2 starts the work. In the first 8 minutes of the beginning, machine 2 cannot receive the second product of machine 1. At the same time, machine 3 has no work because the product hasn't been passed to it. The bottleneck of machine 2 blocks the machine 1 and starves machine 3. To solve this situation, buffers can be added between activities to reduce the occurrence of starvation and blockage (Sople 2016).

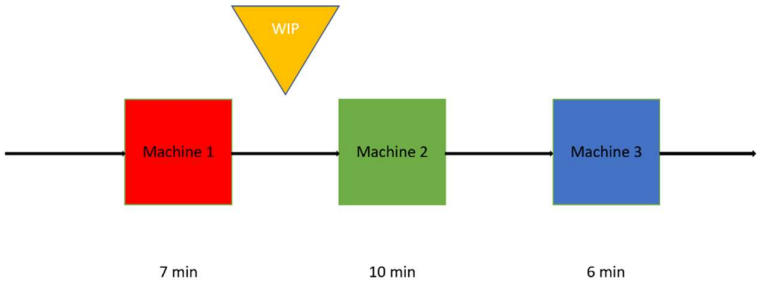

Figure 1.3 Single Line of Starved and Blocked Status

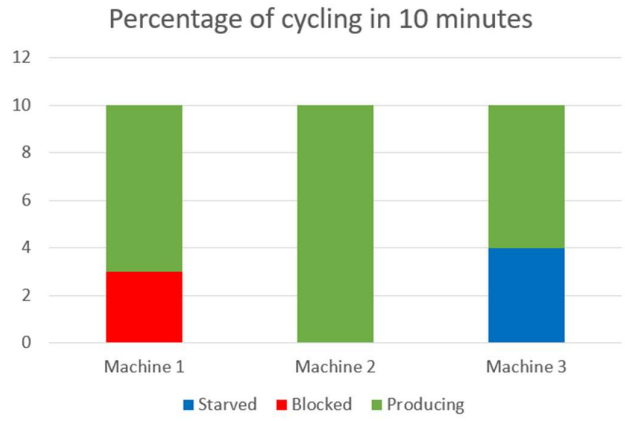

Figure 1.4 Percentage of Cycling in 10 Minutes

Fault detection consists of data with sharp changes, and the methods solve this kind of problem vary from statistical quality control to algorithms (Jeong *et al.* 2018). Traditional gantry status monitoring focus on the statistical process control (SPC). The SPC is embedded in the supervisory control and data acquisition system based on Windows Operating System. Figure 1.3 shows the overview of the SCADA system. This logic control system receives all the information via bus connection remotely. Process status can be monitored by the signal change in switchgear and external electrical components. Though SPC can detect product quality change at its early stage, the SPC is sensitive to the time, and it only rejects the product out of pre-setting quality line but not telling the defective degree. Besides, the traditional measurements need the physical sensor information, which may cost a lot in purchasing sensors and waste time in estimating parameters for some SPC methods. Furthermore, conventional way can only alarm the fault and warning situation by the time it happens.

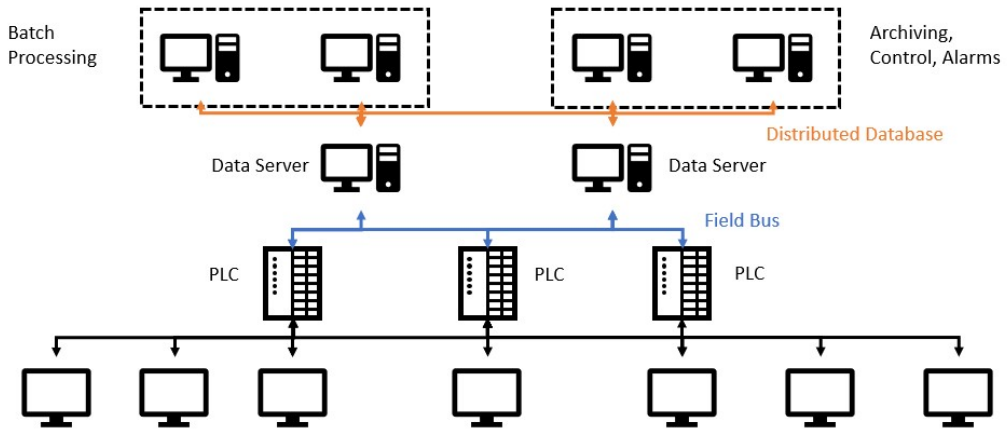

Figure 1.5 SCADA System

A vector could illustrate the general observed gantry process status sequence showed below.

$$\mathbf{x} = \left[x_1, x_2, x_3, ..., x_j, ..., x_p \right]$$

Where, vector $\mathbf{x}$ consists the gantry process statuses in the p time dimension. This time dimension could be in second, minute, hour, day or other units according to the study. For example, $x_j = \mathrm{S}$ means that the gantry is starved at the time j .

1.2 Problem Statement and Motivations

Modern industrial plants are gradually transforming according to the concept of Industry 4.0 that consists of the cyber-physical system, internet of things, cloud computing, and cognitive computing and are digitizing and intelligentizing the supply, manufacturer, sales to achieve rapid, effective, and personalized product supply (Jasperneite 2012, Kagermann *et al.* 2013, Lasi *et al.* 2014, Hermann *et al.* 2016). For the big data acquisition and monitoring, manufacturers are developing their own data architecture by open-source applications such as Apache Hadoop and MapReduce. The data architecture is composed of algorithms, models, and rules that determine the storage, acquisition, integration, and use of the data in the database.

To achieve the goal of smart manufacturing in Industrial 4.0, “5Ps” is proposed in a set of core intelligence: predictability, producibility, productivity, pollution prevention, and performance, which means the fault can be predicted before its occurrence in the future. In this way, cloud technologies and big data are combined to assure the reliable manufacturing and lead the insight into algorithms analytics. Since the collection of gantry status data includes both the historical and the current, it’s easy to find and update the patterns hide behind the abnormal and normal status sequence. After the fault detection which contains a combination of pattern recognition and classification, the

remote monitor screen displays the alarm data image indicating a problem class with the corresponding troubleshooting guide (Markle and Weaver 2003).

Though there are numerous applications of the gantry, the studies on gantry health monitoring and fault detection are limited. There are several challenges in proposing a new model in dealing with the gantry health monitoring and fault detection. First, there's no existed model deals with the categorical process status sequence. Second, while collecting the sequences from the system database and building a new model, one may find there's information overlap and difficult to study the incident timestamps. Third, some algorithms in fault classification may over stack flow due to the large computation.

Machine learning usually uses feature extraction to reduce the computation complexity. Most of the researches of the fault detection feature extraction are limited to the numeric matrix. Principal component analysis (PCA) is widely used for the numeric matrix feature extraction while multiple correspondence analysis (MCA) is usually used in categorical matrix feature extraction to reduce the dimension and therefore save the time for the computation. However, PCA is not applicable to all scenarios. For example, in the semiconductor manufacturing, applying PCA would lead to the nonlinearity in most batch processes, multimodal batch trajectories caused by the product mix, and steps combined with process variable durations (He and Wang 2007). As for the MCA, the problematic aspects would be low percentages of variation on principal axes and the normalization chosen for purposes of the graphical display (Greenacre 1990).

To solve the challenges presented above, we are motivated to develop a method that

deals with the process status sequence in gantry health monitoring and fault detection. We propose to apply the texture extraction analysis in the image processing field aiming to find new solutions to reduce the dimension, which provide new idea in processing categorical sequence and extend the scope of the study. By setting certain combination of pattern recognition and classification model, we could monitor the gantry and prognostic specific fault. Allocating workers and material kit prior the real problem can be achieved in the smart manufacturing. In our study, classification algorithms *k* Nearest Neighborhood (*k*-NN), Support Vector Machine (SVM), linear discriminant analysis (LDA), and quadratic discriminant analysis (QDA) are applied to classify status sequence according to its features and predict if there will be any abnormal event.

1.3 Thesis Organization

This thesis is organized as follows. Chapter 2 is the introduction towards classification and feature extraction based on texture extraction. Classification methods such as *k*-NN, SVM, LDA, and QDA are reviewed in this chapter for both fault detection as well as evaluating the performance of the texture extraction. The rest of Chapter 2 reviews texture extraction techniques such as the histogram of oriented gradients and local binary pattern. Chapter 3 proposes a novel matrix presentation for the categorical process status sequence, which makes taking use of the texture extraction in the image feature extraction area as feature extraction for categorical matrix into possible. Chapter 4 provides a case study of a real gantry system in automobile manufacturing to validate the effectiveness of the proposed approach. The case study introduces data description, variable and fault selection, data pre-processing, and the result. Finally, Chapter 5 concludes this work and outlines future work.

CHAPTER 2

Classification and Feature Extraction Based on Texture Extraction

2.1 Literature of the Classification in Fault Detection

Classification is the process that classifies particular class according to some common characteristics. The classification algorithm can determine which specific fault is presenting when applied in fault detection field.

According to the type of the inputs, fault detection can be divided into three categories: waveform-based, vector-based, image-based. Waveforms like current, voltage and other circuit characteristic indicators are widely used in the traditional methods. The signal will be used directly or after necessary techniques such as spectral analysis, Fourier transform, Hilbert transform. The relevant application literature are presented in subsections 2.1.1, 2.1.2, and 2.1.3. Image-based fault detection is a method analyzing the information from the image and classify the fault according to the image features. Standard approaches for the image-based fault detection are focusing on the geometrical, statistical, and specific model (Tuceryan and Jain 1993). Nath *et al.* (2014) performed a survey of the techniques applied in image classification including different sensors information, the nature of training sample, parameter, the quality of pixel information, the number of outputs generated for each spatial data element, the quality of spatial information, and special classification techniques. To study the characteristic in the image, image texture analysis is widely used.

Vector-based fault detection not only relies on the mathematical and statistical

approaches but also takes advantage of the algorithms that recognize the fault pattern (Liu 2012). Relevant studies based on different classification method are presented in subsections 2.1.1, 2.1.2, and 2.1.3. In the following subsections, we will present three most commonly used classification algorithms.

2.1.1 *k*-Nearest Neighborhood

Casimir *et al.* (2003) proposed the application of *k*-nearest neighborhood rule using the pattern vector extracted by the frequency dependent parameters to detect broken bars and stator unbalance in an induction motor. He and Wang (2007) developed a fault detection method taking advantages of the *k*-NN rule that deals with nonlinearity data, this rule can preprocess the data automatically which on the other hand proves the ability of online fault detection. Cheng *et al.* (2011) performed a supervised locally linear embedding-based method combined with the *k*-NN overcoming the correlation and high dimensionality in the time series multi-sensor signals for fault diagnosis. Fezari *et al.* (2014) conducted analysis detecting a specific fault of rotating machines by applying the Euclidean distance and *k*-NN method to the real-time machine status, and the average accuracy result reaches 94%. Tang and Xu (2016) proposed a novel multi-class classification method that combines the conventional kernel density estimation and *k*-NN techniques to identify multiple parametric faults in the analog circuit, and the result shows that the process has excellent performance in both accuracy and speed. Majd *et al.* (2017) introduced a new methodology based on the *k*-NN algorithm by finding the distance between each sample and its fifth nearest neighbor in a pre-default window to detect and classify the faulty phases and occurrence time in the power transmission system.

K-nearest neighborhood (*k*-NN) is a rule that object is classified by the majority vote of its nearest *k* neighbors' class. Different distance measures for continuous variables and discrete variables are derived from this rule. Suppose in the space of $(x_1, y_1), (x_2, y_2), ...(x_n, y_n)$, $\vec{y}$ takes value from the class set. If randomly choosing a point x_0 and computing some norm $||\bullet||$ on $\mathbb{R}^d$ with its neighbors, then its class is determined by the class of its nearest neighbor. The number of neighbors *k* is usually taken value from 1 to 5. Common used distance measures are listed in the Table 2.1. Among these distance measures, we can transform Minkowski distance to the City Block distance, the Euclidean distance, and the Chebyshev distance by changing $p = 1,\ p = 2,$ and $p \to \infty$.

Figure 2.1 shows examples of 1-NN, 3-NN, 5-NN for the class prediction. For 1-NN, the green dot calculates the distance with its nearest one observation and thus determines its class. For the 3-NN, the green dot calculates the distances with its nearest three observations. The class for the green dot is belong to the yellow group because there're two yellow group observations in its majority voting pool. In the same way, the class label for the green dot in 5-NN is the yellow group.

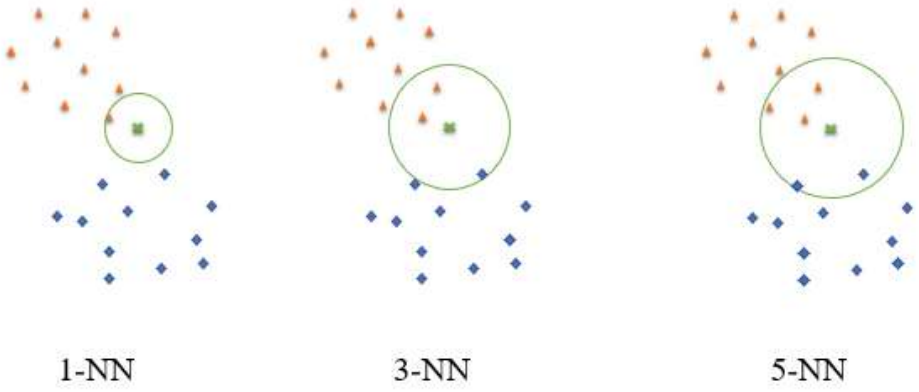

Figure 2.1 Examples of 1-NN, 3-NN, and 5-NN

Table 2.1 Common *k*-NN Distance Measures

Distance Name	Equation
City Block (Manhattan) Distance	$d = \sum_{i=1}^{n} \lvert x_i - y_i \rvert$
Chebyshev Distance	$d = \max_{i,j}(\lvert x_i - x_j \rvert, \lvert y_i - y_j \rvert)$
Correlation	$\rho_{x,y} = corr(x,y) = \frac{\operatorname{cov}(x,y)}{\sigma_x \sigma_y} = \frac{E[(x-\mu_x)(y-\mu_y)]}{\sigma_x \sigma_y}$
Cosine	$d = \cos(\theta) = \frac{\mathbf{x} \cdot \mathbf{y}}{\lvert\lvert\mathbf{x}\rvert\rvert_2 \lvert\lvert\mathbf{y}\rvert\rvert_2} = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2 \sum_{i=1}^{n} y_i^2}}$
Euclidean Distance	$d = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$
Hamming Distance	The number of difference in the same position of two same length vectors
Jaccard	$d = \frac{\lvert x \cap y \rvert}{\lvert x \cup y \rvert}$
Minkowski	$d = \left(\sum_{i=1}^{n} \lvert x_i - y_i \rvert^p \right)^{1/p}$

2.1.2 Support Vector Machine

Shin *et al*. (2005) present one-class support vector machines for machine fault detection and classification in electro-mechanical machinery and evaluate the performance by comparing it with that of a multilayer perceptron. Shahid *et al.* (2012) propose one-class quarter-sphere support vector machine to increase accuracy and decrease computational complexity for fault detection in unsupervised online power

transmission lines. Ray and Mishra (2016) investigate the distance estimation scheme by using post-fault single cycle current waveform and wavelet packet transform and the fault type by using support vector machine whose parameters are tuned by particle swarm optimization in a long transmission line. Al-Obaidy *et al.* (2017) propose the integrated circuit fault detection based on multilayer perceptron, support vector machine, and adaptive neuro-fuzzy inference system using the thermal image that captured by the finite element method of a printed circuit board model.

Support Vector Machine (SVM) is derived from the perceptron concept and aims to maximize the margin between the hyperplane between the different groups by different kernels. SVM can deal with both liner case and non-linear case which SVM projects the data into high dimensional space though the kernel functions. The most commonly used four kernel functions are the linear kernel, polynomial kernel, sigmoid kernel, and the radial-based kernel, which are listed in Table 2.2.

Table 2.2 Common Kernels of SVM

Kernel Type	Function
Linear	$K(x_i, x_j) = \mathbf{x}_i \cdot \mathbf{x}_j$
Polynomial	$K(x_i, x_j) = (\gamma \cdot \langle \mathbf{x}_i \cdot \mathbf{x}_j \rangle + C)^d$
Sigmoid	$K(x_i, x_j) = \tanh(\langle \mathbf{x}_i, \mathbf{x}_j \rangle + C)$
Gaussian Radial Basis	$K(x_i, x_j) = \exp(-\gamma \cdot \lvert \mathbf{x}_i - \mathbf{x}_j \rvert^2)$

2.1.3 Linear and Quadratic Discriminant Analysis

Linear Discriminant Analysis (LDA) and Quadratic Discriminant Analysis (QDA) are

two conventional classification algorithms. These two classifiers have a linear and quadratic decision boundary respectively as their names suggest. Yoshida *et al.* (2008) used a combination of linear discriminant analysis (LDA) and boundary-based discriminative subspace identification method to address the identification problem of the causal variables for the system anomaly. da Silva Soares and Galvão (2010) proposed successive projections algorithm employed with LDA classifiers to discriminate between normal operating conditions and faults. Haddad and Strangas (2016) applied LDA as a classification method with respect to the accuracy to detect the status of permanent magnet synchronous machines whether it is healthy or faulted as well as determine the type of that fault.

Both LDA and QDA work by specifying the prior probability given by the training set, the class mean and the covariance matrices. Bayes' theorem is then used to calculate posterior probability as Eq. (2.1).

$$P(y=k \mid X)=\frac{P(X \mid y=k)}{P(X)}=\frac{P(X \mid y=k)P(y=k)}{\sum_{t} P(X \mid y=l)\cdot P(y=l)} \tag{2.1}$$

2.2 Texture Extraction

Texture extraction refers to the pattern matrices extracted from the image (Forsyth and Ponce 2011). The minimum unit of the image is called the cell or pixel. Each cell can store numeric number and reflect the information in the pixel. An image can be stored in binary values as raster dataset, or integer values as the heat map. The obtained matrix sets can provide the information of color, density, regularity, smoothness and so on. In the thesis, only the histogram of oriented gradients and local binary pattern are

presented as the techniques in the feature extraction.

2.2.1 Histogram of Oriented Gradients (HOG)

This algorithm quantifies the object appeared on the image as the intensity gradients or edge directions distribution and yield good performance even under severe circumstances (Alorf and Abbott 2017). Zhu *et al.* (2006) performed the HOG and the cascade-of-rejectors approach, the result shows the similar accuracy level to the existing methods. Kopaczka *et al.* (2016) used LBP, HOG for feature extraction, then applied random forest and SVM to test the classification performance of the feature descriptors in the strip defects in the circularly knitted fabric. Tsai *et al.* (2017) performed three cases in implementing the HOG to the convolutional neuron network and achieved similar accuracy compared to existing approaches but more effective.

Histogram of oriented gradients (HOG) records the gradient orientation occurrences in some portion of the image. This technique is widely used to detect an object in the computer vision and image processing field. To obtain the HOG features for an image, following steps are given:

(1) Transform the image into grayscale;

(2) Normalize the gray-scaled image by gamma compression showed in Eq. (2.2);

$$G(x,y) = F(x,y)^{\frac{1}{\gamma}} \tag{2.2}$$

(3) Obtain the horizontal and vertical gradient for each cell in the block by Eq. (2.3);

$$\begin{cases} G_x(x,y) = G(x+1,y) - G(x-1,y) \\ G_y(x,y) = G(x,y+1) - G(x,y-1) \end{cases} \tag{2.3}$$

(4) Calculate the magnitude and the gradient direction for each cell in Eq. (2.4);

$$\begin{cases} \nabla G(x,y) = \sqrt{G_x(x,y)^2 + G_y(x,y)^2} \\ \theta(x,y) = \tan^{-1} \dfrac{G_y(x,y)}{G_x(x,y)} \end{cases} \tag{2.4}$$

(5) Use L2-norm expressed in Eq. (2.5) to normalize the contrast. Where v is considered as the RGB vector, and ε is the noise;

$$\frac{v}{\sqrt{\| v \|_2^2 + \varepsilon^2}} \tag{2.5}$$

(6) Combine the HOG features.

The size of the moving window is usually multiple of the blocks. The window captures the equal-sized blocks given the requirement of the starting position and moving direction as shown in Figure 2.2.

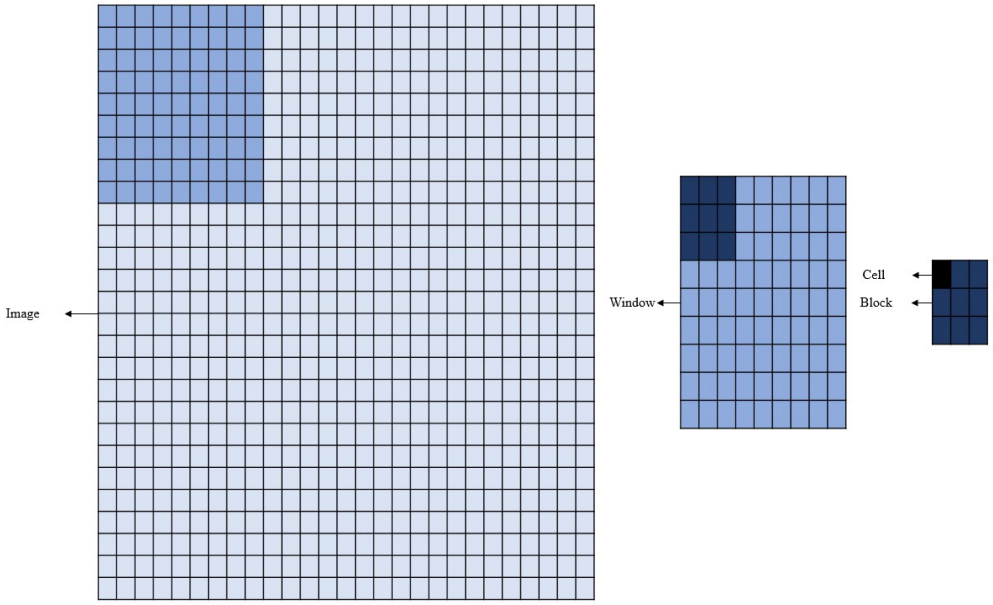

Figure 2.2 Illustration of Image, Windows, Block, and Cell for HOG Method

The HOG features for one image are concatenated by all the HOG features extracted by all the blocks. The number of blocks can be obtained by below equation.

$$(\frac{window\_width-block-width}{block\_stride}+1)*(\frac{window\_width-block-width}{block\_stride}+1)$$

Table 2.3 shows an example of the number of the HOG features with block-stride = 8. If we assume a cell of size 8×8 comprises 9 HOG features, then the block of size 16×16 it located in is then comprises 36 features. If we extend this concept to the image of size 64×64, we can conclude that this image has 1764 features.

Table 2.3 Example of HOG Feature Extraction with Block-stride = 8

	Size	Number of HOG Features
Cell	8×8	9
Block	16×16	$4\times9=36$
Image	64×64	$7\times7\times36=1764$

2.2.2 Local Binary Pattern (LBP)

Local binary pattern (LBP) extracts local binary pattern features from the grayscale image. When LBP is combined with the HOG technique, the detection performance will improve considerably on some datasets (Wang *et al.* 2009). Li *et al.* (2015) implemented LBP to reduce the computation complexity and obtain the fault features which later on are applied to the learning vector quantization (LQV) neural network in the fault problem in analog circuit field. Omidi *et al.* (2017) formed the feature vector by the LBP and applied it to the conventional neural networks; the result showed that approximately 91% of the F-measure has higher performance in defecting the faulty situation.

LBP is a technique that focuses on the binary image pattern. It has the advantages of rotation invariance and gray-scale invariance. This technique is applicable to categorical matrices but has limitation to numerical matrices because useful information would have a higher chance to be overlooked. Dividing the image into cells and compare the central pixel gray level among its adjacent cells to obtain the LBP information. An example showing the matrix after LBP transformation is presented below in Figure 2.3.

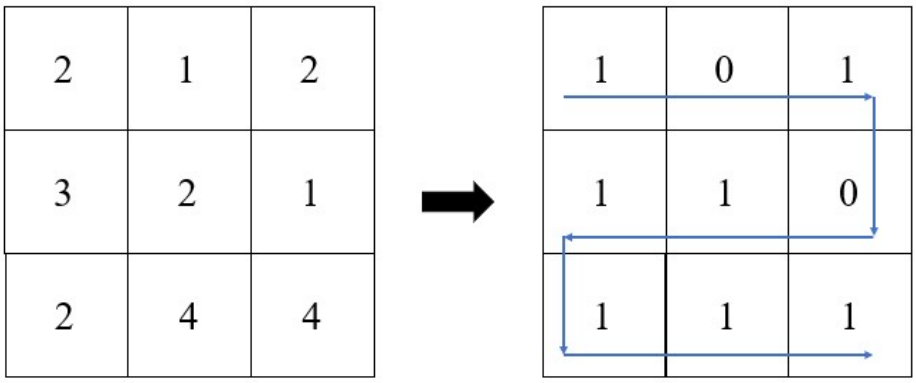

Figure 2.3 Example of Matrix after LBP Transformation

After the transformation from the regular matrix into the binary matrix, the binary numbers can be extracted and converted into decimal numeral following the pre-set direction. In the case showed in Figure 2.3, the center number is 2 and we need to compare the size within its range by marking 1 as greater or equal to and 0 as less than. By pre-defining the direction as zig-zag, a set of binary number 10101111 can be obtained and transformed into decimal number 175. This 175 then can be taken as the feature in this 3×3 matrix. After collecting all the features in the moving window, it then will be concatenated as the LBP features for this image.

CHAPTER 3

Matrix Representation for Categorical Process Status Sequence

In this chapter, we will propose our method of matrix representation. This matrix representation helps to transform the time series gantry process status sequence into binary matrix without losing any information. Besides than that, this matrix representation makes feature extraction from categorical matrix possible. Other than that, we bring about a new idea of texture extraction in image processing area to the binary matrix feature extraction, which extend the study scope of feature extraction.

To represent the time series gantry categorical process status sequence into the binary matrix, we first need to acquire the data. The data acquisition process defers from the scope of studies. Though the SCADA system we discussed in chapter 1 stores numerous signal and information, there's no advanced embedded module that just extracts and monitors the gantry process status sequence. Therefore, we propose the below scheme in gantry process status sequence acquisition:

(1) Locate the starting and ending time of one certain fault;

(2) Extract the process status sequence within same time length before the fault start time and the same length sequence after the fault end time.

While extracting the sequence followed by step 2, one may encounter the information overlap. The information overlap is led by the paralleled raw incident sequences between the statuses and the faults. It's difficult to obtain the time intervals for each process status sequence before and after the fault observation if there is overlapping

among the status time interval with the fault. Figure 3.1 displays four overlapping scenarios.

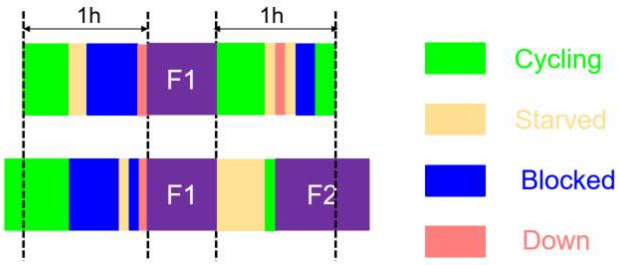

Figure 3.1 Gantry Process Status Sequence Overlapping

After proper gantry process status sequence extraction for each fault, we will get the sequences of categorical process status. Such categorical sequence is hard to extract features when the data becomes huge. The size of the massive data is one of the biggest obstacles in machine computation. Other than that, common feature extraction cannot be deployed to the categorical sequence. To deploy HOG and LBP as stated in the Chapter 1 as a novel idea in binary process matrix, we provide the following sequence transformation to binary matrix that considered as gray-scaled matrix in the image processing area. This binary process matrix is different from the atomic unit of data in computer system, but the philosophy of embedding information with binary numbers is pretty much same.

Consider a sequence consists of “blocked”, “cycling”, “down”, “starved” in t time interval showed below.

$$[BBBC,...,CCSS,...CDDD]$$

Where “B”, “C”, “D”, “S” stands for “blocked”, “cycling”, “down”, and “starved, respectively. The sequence then can be transformed to a matrix of size 4-by-t based on the process status appearance in the minimum unit. This matrix can present a better view of the process status change as Table 3.1.

Table 3.1 Example of Matrix after Transformation

t	1	2	3	4	…	66	67	68	69	…	t-3	t-2	t-1	t
B	1	1	1	0	…	0	0	0	0	…	0	0	0	0
C	0	0	0	1	…	1	1	0	0	…	1	0	0	0
D	0	0	0	0	…	0	0	0	0	…	0	1	1	1
S	0	0	0	0	…	0	0	1	1	…	0	0	0	0

As stated earlier in sections 2.2.1 and 2.2.2, we now can apply the HOG and LBP accordingly. We will use a case study of a real gantry system from automobile manufacturing to demonstrate the effectiveness of the proposed method.

CHAPTER 4

Case Study: Gantry System in Automotive Manufacturing

4.1 Data Description

The case study investigates the process status log of the gantry status sequence, which is a sample dataset constructed based on automobile manufacturing in real world. The dataset consists of the gantry timestamp sequence which refers to the specific time records of the system's status which may include process status, fault signals, and warnings. Raw file information such as date, number of entries, size are summarized in Table 4.1. Each entry represents one incident and its corresponding records are also included.

Table 4.1 File Information

Date	Number of Entries	File Size (MB)
2014-04-07 to 2016-04-05	4985651	900
2016-04-06 to 2016-04-29	84407	14.4
2016-05-01 to 2016-05-31	145852	25
2016-06-01 to 2016-08-06	320508	55

The data is extracted from assets datasets during April 4th 2014 to August 6th 2016 of size 5536418×13 , which includes diverse information such as "Asset Name", "Description", "Start Time", "End Time", "Duration", "Machine Faults" (133), "Warnings" (9), and "Machine States Incidents" (18). "Machine Faults" refer to the faults that lead to machine down saved in the database for management. "Warnings"

indicate the possible danger of specific situations. Tables 4.2 shows the sample dataset with the most typical information.

Table 4.2 Sample Dataset

Category	Description	Start Time	End Time	Duration
Machine Faults	Undefined Machine Fault	2014/4/7 6:24:53	2014/4/7 6:32:17	444
Warnings	Tool Life Nearly Expired	2014/4/17 16:04:04	2014/4/17 19:32:14	12490
Machine States Incidents	Cycling	2014/5/2 5:07:50	2014/5/2 5:07:56	6

“Description” shows the detail of the “Machine Faults”, “Warnings”, and “Machine States Incidents”. The common types of gantry statuses are: setup, cycling, starved, loading, blocked, unloading, shut down, manual emergency stop, waiting for attention, down, repair. “Start Time” records the start point of each time a fault or warning signal shows and the start point before the duration of asset operation status. “End Time” records the end of each time a fault or warning signal ends and the endpoint of the duration of asset operating status. “Duration” is the time difference between one event start time and end time.

4.2 Variable and Fault Selection

Since we only focus on the gantry property, the row dimension can be reduced to 3405064 by selecting gantry in the “Asset Name” section. By choosing necessary

information, the variables can be shorted to “Category”, “Description”, “Start Time”, and “End Time”. Selecting a subset of the data will speed up the processing and increase accuracy (Wilcox *et al.* 2014). In other words, the dimension of the new data set becomes 3405064×4 and hence reduces the study scope and dimension. Gantry transfers bulk from one place to another, and during this process, statuses and fault information are recorded by the system. This case study will focus on the most informative gantry status: blocked, cycling, down, and starved.

As for the fault information, Table 4.3 shows the occurrences and accumulated duration of each fault. Different types of faults are coded as “F1”, “F2”, etc.

Table 4.3 Fault Occurrence and Duration

Fault Type	Fault Occurrence	Fault Duration	Fault Type	Fault Occurrence	Fault Duration	Fault Type	Fault Occurrence	Fault Duration
F1	302	26.8386	**F13**	14	88.7525	**F25**	2	0.02806
F2	109	8.93528	**F14**	14	7.36528	**F26**	2	0.02361
F3	25	3.86667	**F15**	10	0.26611	**F27**	1	0.01972
F4	23	3.11194	**F16**	9	5.37472	**F28**	1	0.27222
F5	23	1.99139	**F17**	8	0.93139	**F29**	1	0.00306
F6	23	2.0075	**F18**	7	0.21583	**F30**	1	2.53028
F7	18	2.76972	**F19**	5	0.33472	**F31**	1	69.9178
F8	16	1.12083	**F20**	4	26.9322	**F32**	1	1.11472
F9	16	1.41639	**F21**	3	0.35556	**F33**	1	34.1647
F10	15	0.90667	**F22**	3	1.93	**F34**	1	0.04611
F11	14	0.85194	**F23**	3	0.02472	**F35**	1	10.9172
F12	14	7.49833	**F24**	3	0.02778	**F36**	1	0.00167

In this table, fault duration time includes the total time from waiting for attention, repair duration, setup, restart to function well. To start study the fault pattern, we choose the

most frequent fault F1 as the first step. The study scheme in F1 can be extended to the rest faults.

4.3 Data Pre-processing

The fault pattern recognition algorithm requires the abnormal status sequence before a certain fault and the compare set of normal status sequence. In this study, we start by selecting all the sequences 1 hour prior to the F1 as the abnormal status sequence. The normal status sequences are chosen as the status sequences after the F1 within 1-hour time range without any interventions with other faults, and the independence of the incident status hence guaranteed. This section is divided as follow: matrix build-up based on abnormal and normal status sequence, matrix representation, HOG and LBP feature extraction.

4.3.1 Matrix Build-up

As stated earlier, the normal and abnormal status dataset are formed separately. The matrix can be expressed by the Eq. (4.1), where i stands for the observation and j stands for the $\left(p-j+1\right)^{th}$ point prior to a certain incident.

$$X_{i,j} \in \{B, C, D, S\} \tag{4.1}$$

Table 4.4 shows the gantry status matrix $X_{302\times3600}$ 1 hour before the F1 and Table 4.5 shows the gantry status matrix $X_{254\times3600}$ 1 hour after the F1.

Table 4.4 Status Matrix of Gantry in the One Hour before F1

T	1	2	3	4	…	3597	3598	3599	3600	Y
1	S	S	S	S	…	S	S	S	S	1
2	C	B	B	B	…	B	B	B	B	1
3	S	S	S	S	…	S	S	S	S	1
…	…	…	…	…	…	…	…	…	…	1
301	C	C	C	C	…	C	C	C	C	1
302	S	S	S	S	…	S	S	S	S	1

Table 4.5 Status Matrix of Gantry in the One Hour after F1

T	1	2	3	4	…	3597	3598	3599	3600	Y
1	D	D	D	D	…	D	D	D	D	0
2	D	D	D	D	…	D	D	D	D	0
3	D	D	D	D	…	D	D	D	C	0
…	…	…	…	…	…	…	…	…	…	0
254	D	D	D	D	…	D	D	C	C	0

4.3.2 Matrix Representation

According to vector-based matrix formulation, the 4 types of status of one observation can be re-written into the matrix showed below $X_{1,}, X_{2,}, X_{3,} X_{4,}$ are corresponding to the status “B”, “C”, “D”, “S”. p represents the time length which in this case is 3600.

$$X_{\text{new}} = \begin{bmatrix} X_{1,1} & X_{1,2} & \dots & X_{1,j} & \dots & X_{1,p-1} & X_{1,p} \\ X_{2,1} & X_{2,2} & \dots & X_{2,j} & \dots & X_{2,p-1} & X_{2,p} \\ X_{3,1} & X_{3,2} & \dots & X_{3,j} & \dots & X_{3,p-1} & X_{3,p} \\ X_{4,1} & X_{3,2} & \dots & X_{4,j} & \dots & X_{4,p-1} & X_{4,p} \end{bmatrix}$$

For all observations, we transfer each observation of size 1×3600 into 4×3600 because of the number of the statuses. Examples of the transformed status matrix of observation 1 of each case are shown at Table 4.6 and Table 4.7.

Table 4.6 Temporal Status Matrix of Observation 1 in X1 in the One Hour before F1

T	1	2	3	4	…	3597	3598	3599	3600	Y
B	0	0	0	0	…	0	0	0	0	1
C	0	0	0	0	…	0	0	0	0	1
D	0	0	0	0	…	0	0	0	0	1
S	1	1	1	1	…	1	1	1	1	1

Table 4.7 Temporal Status Matrix of Observation 1 in X1 in the One Hour after F1

T	1	2	3	4	…	3597	3598	3599	3600	Y
B	0	0	0	0	…	0	0	0	0	0
C	0	0	0	0	…	0	0	0	0	0
D	1	1	1	1	…	1	1	1	1	0
S	0	0	0	0	…	0	0	0	0	0

4.4 Results

In order to testify the effectiveness of the image-based feature extraction, we use *k*-NN,

SVM, LDA, and QDA as the fault detection and classification tools. In this section, the non-transformed matrix, the matrix after the HOG transformation, and the matrix after the LBP transformation are all evaluated by accuracy.

To get rid of the influence of the sample size, we partition the data by increasing the training set and decreasing the testing set gradually. Figure 4.1, 4.2, and 4.3 show the relationship between the test set size and the accuracy. On average, the accuracy would decrease when decreasing the training set and increasing the testing set.

The results for the original data show SVM with radial basis function kernel with 0.4 proportion test size has the highest accuracy of 0.7. The performance of 5-NN, quadratic discriminant analysis and the SVM with radial basis function kernel are not sensitive towards the training and testing set size.

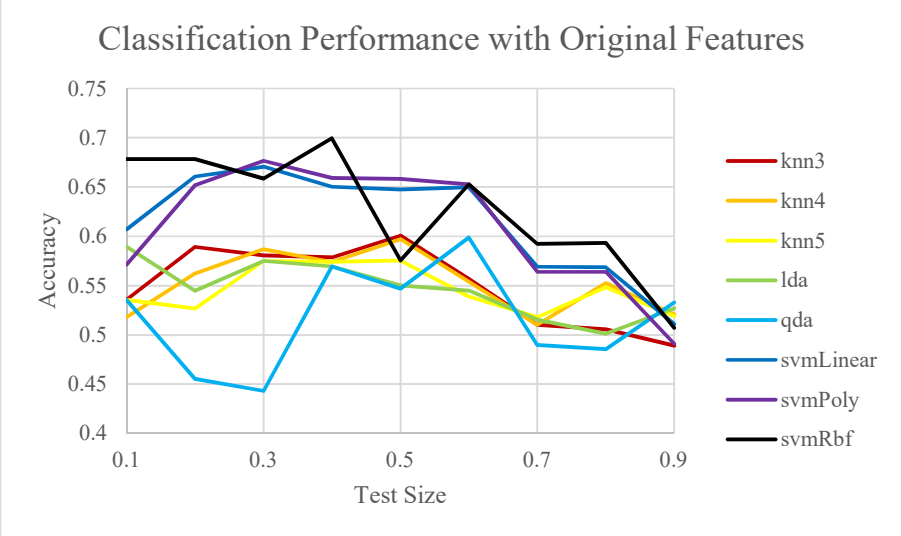

Figure 4.1 Classification Accuracy versus Test Size for Original Features

The results for data after HOG texture extraction show that both linear discriminant

analysis and the quadratic discriminant analysis have a better performance over the rest algorithms. But the linear discriminant analysis has the best overall performance over the quadratic discriminant analysis, the previous one has the accuracy of 0.91 when the test size is of 0.1 proportion. The accuracy for quadratic discriminant analysis drops drastically when changing the test size of 0.8 proportion or higher. The rest algorithms almost have the same performance and the trend change.

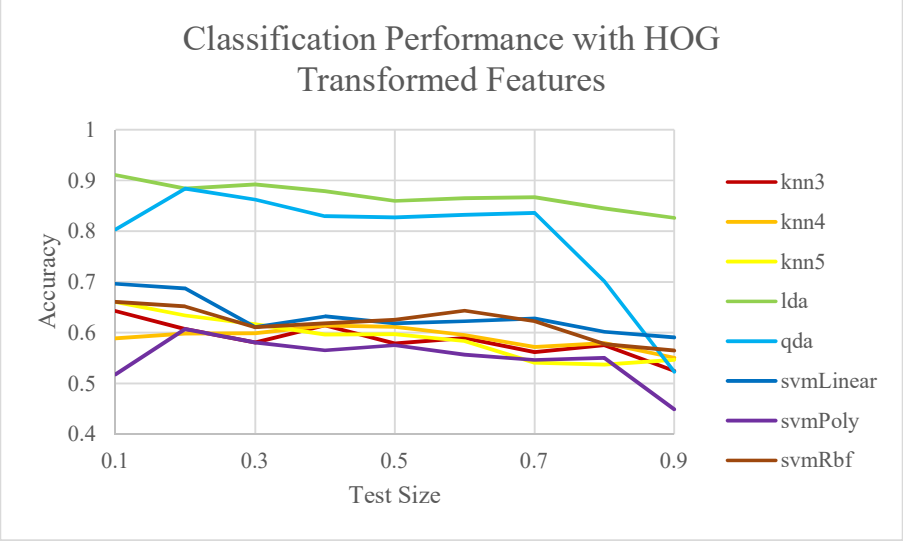

Figure 4.2 Classification Accuracy versus Test Size for HOG Transformed Features

The results for data after LBP texture extraction show that the performance of linear discriminant analysis, SVM with linear kernel, and the SVM with polynomial kernel are better than the rest algorithms. However, the linear discriminant analysis has a better performance of 0.89 over the other two when the test size is of 0.1 proportion. The rest algorithms are of the same performance as well as the trend change according to the test size. The quadratic discriminant analysis and the 3-NN are not sensitive to the test size.

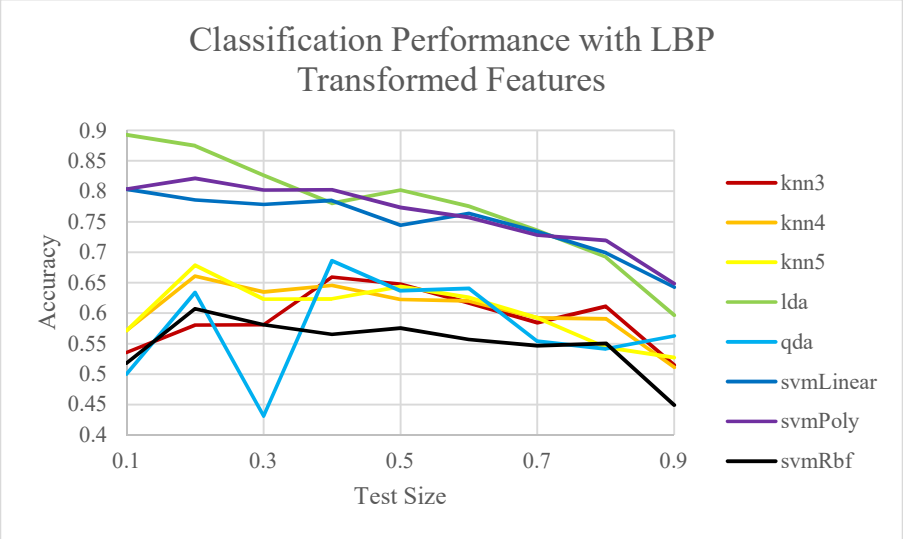

Figure 4.3 Classification Accuracy versus Test Size for LBP Transformed Features

In this case study of the gantry system from automobile manufacturing, matrix after texture extraction has a better performance than the original data in accuracy. Overall, test size with 0.1 portion of the original dataset has a better performance over the others. From Table 4.8, we can see that LDA after HOG transformation has the best performance of 0.9107. These results show that it is feasible to monitor and detect fault based on gantry time series categorical process status sequence. By selecting the most representative statuses and time length, we get rid of the information overlap situation. Our proposed technique in matrix representation makes categorical matrix feature extraction possible. We also illustrate that texture extraction in the image field can also be used in the transformed categorical matrix. The classification results of the transformed features have a better performance over the original features.

On the other hand, the case study results reveal that critical asset time series categorical process statuses can be used in health monitoring and fault detection. We can predict certain fault by the error pattern, which is a big step in the Industry 4.0.

Table 4.8 Classification Accuracy Comparison at Test Size = 0.1

	k-NN			LDA	QDA	SVM-Linear	SVM-Poly	SVM-Rbf
	k=3	*k*=4	*k*=5			*C*=0.1, 50, 0.1	*C*=1, *p*=2	
Original Features	0.5357	0.5179	0.5357	0.5893	0.5357	0.6071	0.5714	0.6786 (*C*=50, γ=0.001)
HOG Features	0.6428	0.5893	0.6607	0.9107	0.8036	0.6964	0.5179	0.6607 (*C*=200, γ=.001)
LBP Features	0.5357	0.5714	0.5714	0.8929	0.5	0.8036	0.8036	0.5179 (*C*=0.1, γ=0.001)

CHAPTER 5

Conclusion and Future Work

5.1 Conclusion

This thesis aims to study the health monitoring and fault detection of one of the most critical assets in the industrial field – Gantry. Gantry is prevalently used as the bulk transfer machine in heavy industry, healthcare, and port field. Though gantry is usually considered as a reliable asset, the loss caused by the gantry function failure is hard to value. In this way, we are motivated to study and monitor the time series gantry process sequence. In the study scope, we are also promoted to prognosis the fault by analyzing the process sequence data in order to achieve the "5P" proposed by the "Industrial 4.0". Vector of categorical statuses are commonly used to describe the time series categorical gantry process sequence. Process statuses such as cycling, blocked, starved, and down are the most identical statuses to present the gantry operation. Such categorical sequences are hard to extract features when dealing with a great amount of the sequences. We hence propose to represent the process vector into matrix and then extract the features. While representing the categorical vector into binary matrix, we are inspired to take advantage of the texture extraction techniques such as HOG and LBP in the image area rather than using the common feature extraction techniques PCA.

In Chapter 2, we present some researches based on three classification algorithms and two texture extraction techniques. The classification algorithms help to identify the fault based on the gantry process status sequence. The three classification algorithms cover *k*-NN,

SVM, and discriminant analysis. We present literatures that use these classification algorithms as the fault detection method in different area. Principle introduction and application literatures are all presented for the two texture extraction techniques HOG and LBP.

In Chapter 3, we propose our new technique in matrix representation. We give the scheme of the vector to matrix transformation as well as examples. On one hand, the transformed matrix provides a new idea when dealing with the categorical. On the other hand, this matrix representation helps to transfer the categorical vector into binary matrix, which is beneficial to both texture extraction and the field computation.

In Chapter 4, we utilize the time series gantry process status sequence from automobile manufacturing for the case study. We extract the necessary data from the massive dataset, and then sort the fault by its frequency. The most frequent fault is chosen as an example to build the fault detection model. We extract same time length's process status sequences before and after the most frequent fault. To compare the effectiveness of our proposed matrix representation and application of texture extraction. We first divide data into training and testing dataset of different proportion. Matrix with categorical vector, vector after matrix and HOG transformation, vector after matrix and LBP transformation are compared under the same evaluate criteria. From the result, training and testing size of proportion 0.9 and 0.1 overall has the best performance because we assign more data into model building. Gantry process status sequence after matrix and HOG as well as LBP transformation all have a better performance over the raw vector. LDA after HOG

transformation has the best performance. The result in the case study successfully show that we can take advantage of the texture extraction techniques over our proposed matrix representation of the time series categorical process sequences. By applying the classification algorithms, we can predict certain fault before its occurrence. This would greatly affect the resource allocation when a certain fault arises. Preparing the required manpower and repair material would expedite the repair process and make the manufacturing more efficient.

5.2 Future Work

The proposed matrix representation has some limitations and further study is needed. First, we fail to test over more classification algorithms and texture extraction techniques, which makes this thesis less robust to other solutions. Second, we could add more process statuses into consideration, which could add more information toward the gantry health monitoring. Third, we haven't tested this pattern over other faults. In this way, we dare not to say this gantry health monitoring and fault detection pattern is compatible with other faults.

REFERENCE

Aguilera, C., Orduna, E. and Ratta, G., 2006. Fault detection, classification and faulted phase selection approach based on high-frequency voltage signals applied to a series-compensated line. IEE Proceedings-Generation, Transmission and Distribution, 153(4), pp.469-475.

Ahmed, R., Gadsden, S.A., El Sayed, M., Habibi, S.R. and Tjong, J., 2012, June. A signal-based fault detection and classification strategy with application to an internal combustion engine. In Transportation Electrification Conference and Expo (ITEC), 2012 IEEE (pp. 1-7). IEEE.

Alorf, A. and Abbott, A.L., 2017, October. In defense of low-level structural features and SVMs for facial attribute classification: Application to detection of eye state, mouth state, and eyeglasses in the wild. In Biometrics (IJCB), 2017 IEEE International Joint Conference on (pp. 599-607). IEEE.

Berbeco, R.I., Jiang, S.B., Sharp, G.C., Chen, G.T., Mostafavi, H. and Shirato, H., 2004. Integrated radiotherapy imaging system (IRIS): design considerations of tumour tracking with linac gantry-mounted diagnostic x-ray systems with flat-panel detectors. Physics in Medicine & Biology, 49(2), p.243.

Chen, K., Hu, J. and He, J., 2018. Detection and classification of transmission line faults based on unsupervised feature learning and convolutional sparse autoencoder. IEEE Transactions on Smart Grid, 9(3), pp.1748-1758.

da Silva Soares, A. and Galvão, R.K.H., 2010, March. Fault detection using linear discriminant analysis with selection of process variables and time lags. In Industrial Technology (ICIT), 2010 IEEE International Conference on (pp. 217-222). IEEE.

Forsyth, D.A. and Ponce, J., 2003. A modern approach. Computer vision: a modern approach, pp.88-101.

Greenacre, M.J., 1990. Some limitations of multiple correspondence analysis. Computational Statistics Quarterly, 3, pp.249-256.

Guo, W., Guo, S., Wang, H., Yu, X., Januszczak, A. and Suriano, S., 2017. A data-driven diagnostic system utilizing manufacturing data mining and analytics. SAE International Journal of Materials and Manufacturing, 10(2017-01-0233), pp.282-292.

Haddad, R.Z. and Strangas, E.G., 2016. On the accuracy of fault detection and separation in permanent magnet synchronous machines using MCSA/MVSA and LDA. IEEE Transactions on Energy Conversion, 31(3), pp.924-934.

He, Q.P. and Wang, J., 2007. Fault detection using the k-nearest neighbor rule for semiconductor manufacturing processes. IEEE Transactions on Semiconductor Manufacturing, 20(4), pp.345-354.

Hermann, M., Pentek, T. and Otto, B., 2016, January. Design principles for industrie 4.0 scenarios. In System Sciences (HICSS), 2016 49th Hawaii International Conference on (pp. 3928-3937). IEEE.

Jana, S. and De, A., 2017, December. Transmission line fault pattern recognition using decision tree based smart fault classifier in a large power network. In Calcutta Conference (CALCON), 2017 IEEE (pp. 387-391). IEEE.

Jasperneite, J., 2012. Was hinter Begriffen wie Industrie 4.0 steckt. Computer & Automation, 19.

Jeong, Y.S., Jeong, M.K., Lu, J.C., Yuan, M. and Jin, J., 2018. Statistical process control

procedures for functional data with systematic local variations. IISE Transactions, 50(5), pp.448-462.
Kagermann, H., Helbig, J., Hellinger, A. and Wahlster, W., 2013. Recommendations for implementing the strategic initiative INDUSTRIE 4.0: Securing the future of German manufacturing industry; final report of the Industrie 4.0 Working Group. Forschungsunion.
Khan, W.A., Raouf, A. and Cheng, K., 2011. Virtual manufacturing. Springer Science & Business Media.
Kopaczka, M., Ham, H., Simonis, K., Kolk, R. and Merhof, D., 2016, September. Automated enhancement and detection of stripe defects in large circular weft knitted fabrics. In Emerging Technologies and Factory Automation (ETFA), 2016 IEEE 21st International Conference on (pp. 1-4). IEEE.
Lasi, H., Fettke, P., Kemper, H.G., Feld, T. and Hoffmann, M., 2014. Industry 4.0. Business & Information Systems Engineering, 6(4), pp.239-242.
Li, P., Zhang, S., Luo, D. and Luo, H., 2015, May. Fault diagnosis of analog circuit using spectrogram and LVQ neural network. In Control and Decision Conference (CCDC), 2015 27th Chinese (pp. 2673-2678). IEEE.
Li, X., Ding, P. and Shi, X., 2017, October. Research on bearing fault detection based on convolution neural network. In Chinese Automation Congress (CAC), 2017 (pp. 5130-5134). IEEE.
Liu, J., 2012. Shannon wavelet spectrum analysis on truncated vibration signals for machine incipient fault detection. Measurement Science and Technology, 23(5), p.055604.
Liu, Z., Wang, J., Duan, L., Shi, T. and Fu, Q., 2017, August. Infrared image combined with CNN based fault diagnosis for rotating machinery. In Sensing, Diagnostics, Prognostics, and Control (SDPC), 2017 International Conference on (pp. 137-142). IEEE.
Mahfouz, M.M. and El-Sayed, M.A., 2016. Smart grid fault detection and classification with multi-distributed generation based on current signals approach. IET Generation, Transmission & Distribution, 10(16), pp.4040-4047.
Markle, R.J. and Weaver, E., Advanced Micro Devices Inc, 2003. Troubleshooting method involving image-based fault detection and classification (FDC) and troubleshooting guide (TSG), and systems embodying the method. U.S. Patent 6,621,412.
Nath, S.S., Mishra, G., Kar, J., Chakraborty, S. and Dey, N., 2014, July. A survey of image classification methods and techniques. In Control, Instrumentation, Communication and Computational Technologies (ICCICCT), 2014 International Conference on (pp. 554-557). IEEE.
Ning, L., Tai, N., Zheng, X., Huang, W. and Nadeem, M.H., 2017, July. Detection and classification of MMC-HVDC transmission line faults based on one-terminal transient current signal. In Power & Energy Society General Meeting, 2017 IEEE (pp. 1-5). IEEE.
Omidi, H., SadeghHelfroush, M., Danyali, H., Tashk, A. and Kazemi, K., 2017, August. A novel method for classification of power quality disturbances based on a new one dimensional local binary pattern approach. In Control and System Graduate Research Colloquium (ICSGRC), 2017 IEEE 8th (pp. 225-229). IEEE.
Oppelt, A. ed., 2011. Imaging systems for medical diagnostics: fundamentals, technical solutions and applications for systems applying ionizing radiation, nuclear magnetic resonance and ultrasound. John Wiley & Sons.
SI, J.B., YANG, F., GUO, W.Y. and YAO, Y., 2013. Two-stage disease prediction model based on BP neural network. Journal of Jilin University (Engineering and Technology

Edition), p.S1.
Sople, V.V., 2016. Business process outsourcing a supply chain of expertises. PHI Learning Pvt. Ltd..
Tsai, W.Y., Choi, J., Parija, T., Gomatam, P., Das, C., Sampson, J. and Narayanan, V., 2017, June. Co-training of feature extraction and classification using partitioned convolutional neural networks. In Proceedings of the 54th Annual Design Automation Conference 2017 (p. 58). ACM.
Tuceryan, M. and Jain, A.K., 1993. Texture analysis. In handbook of pattern recognition and computer vision (pp. 235-276).
Wang, J.D. and Hwang, M.C., 2017. A novel approach to extract significant patterns of travel time intervals of vehicles from freeway gantry timestamp sequences. Applied Sciences, 7(9), p.878.
Wang, X., Han, T.X. and Yan, S., 2009, September. An HOG-LBP human detector with partial occlusion handling. In Computer Vision, 2009 IEEE 12th International Conference on (pp. 32-39). IEEE.
Wilcox, P., Horton, T.M., Youn, E., Jeong, M.K., Tate, D., Herrman, T. and Nansen, C., 2014. Evolutionary refinement approaches for band selection of hyperspectral images with applications to automatic monitoring of animal feed quality. Intelligent Data Analysis, 18(1), pp.25-42.
Yoshida, K., Inui, M., Yairi, T., Machida, K., Shioya, M. and Masukawa, Y., 2008, December. Identification of causal variables for building energy fault detection by semi-supervised LDA and decision boundary analysis. In Data Mining Workshops, 2008. ICDMW'08. IEEE International Conference on (pp. 164-173). IEEE.
Zhou, Z. ed., 2010. Manufacturing intelligence for industrial engineering: Methods for system self-organization, learning, and adaptation. IGI Global.
Zhu, Q., Yeh, M.C., Cheng, K.T. and Avidan, S., 2006. Fast human detection using a cascade of histograms of oriented gradients. In Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on (Vol. 2, pp. 1491-1498). IEEE.