Oscillation Detection and Root Cause Analysis with Machine Learning

by

Amirreza Memarian

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

in

Process Control

Department of Chemical and Materials Engineering

University of Alberta

Abstract

Abnormal events in chemical processes present significant challenges, often disrupting the performance of industrial operations. Accurate detection and diagnosis of abnormal behavior among control system elements are crucial to prevent potential process degradation and associated economic losses. This study aims to address the complex task of identifying and characterizing such anomalies, with a particular emphasis on oscillations and their two common root causes: control valve stiction and poor controller tuning. Oscillations, a recurrent issue in process industries, can have far-reaching consequences by compromising stability and undermining product quality. Control valve stiction, another major concern, introduces inefficiencies that not only hinder smooth operations but also degrade the overall quality of the end product. Furthermore, the importance of proper controller tuning cannot be overstated, as it directly impacts control loop performance and, consequently, the efficiency and stability of the entire process.

In response to these multifaceted challenges, the study aims to provide innovative solutions that go beyond traditional approaches. By introducing novel methodologies rooted in cutting-edge techniques and leveraging data-driven analyses and machine-learning approaches, the study seeks to empower industries to proactively address these issues. The research showcases the potential of shape-based pattern recognition, deep learning, and advanced data processing techniques to revolutionize the detection and diagnosis of abnormalities in chemical processes. With an unwavering commitment to enhancing industrial operations, the study strives to establish a path toward achieving heightened precision, efficiency, and promptness in the identification of abnormal events. Through a comprehensive exploration of the complexities

associated with oscillations, control valve stiction, and poor controller tuning, the objective is to provide industries with the necessary resources and understanding to ensure process stability, elevate product quality, and maximize operational efficiency.

The study delves into shape-based pattern recognition for detecting oscillations in process control loops. Two distinct methods are proposed, one utilizing nonlinear algebraic functions and the other harnessing deep convolutional neural networks (CNN). Both methods aim to identify oscillations by recognizing distinct triangle-like patterns in data plots. Evaluation across diverse industries demonstrates their effectiveness in detecting oscillatory control loops. In the realm of control valve stiction detection, a unique method is introduced, combining Markov Transition Field (MTF) and CNN. By transforming process variable data into images using MTF and training CNN with the MTF images, the proposed methodology effectively distinguishes stiction-induced oscillations from other types of oscillations. The integration of transfer learning enhances the stiction detection capability. Application to benchmark control loops confirms the robustness of this approach. The study also explores poor controller tuning detection using Gramian Angular Field (GAF) and Stack Autoencoder (SAE) techniques. These advanced tools offer real-time monitoring and alert operators about poorly tuned controllers. Case studies conducted on diverse datasets validate the accuracy and practicality of the proposed method.

To my family, whom I missed so much and supported me in this challenging path

Acknowledgments

First and foremost, I should appreciate my supervisor Professor Biao Huang who gave me this opportunity to continue my studies in his research group and follow my interests. He helped me patiently and believed in me on the way to the destination through his constructive suggestions. I really appreciate him for his supportive attitude and inspiration during my studies. It was a great pleasure for me to investigate my graduate studies in M.Sc. under his supervision.

Next, I should appreciate my friend Dr. Seshu Kumar Damarla, a postdoctoral fellow in our research group, for his indefinite help during these two years. He was beside me from the beginning of my research, and he was always there to help me and discuss various topics regarding the research. Not only had he several discussions with me about the research topics, but also he taught me how to talk and write in an academic way.

It was my honor to be a member of the Laboratory of Process data analytics and smart automation (PDASA) group, where we broadened our knowledge not only in our research but also in the industry, and not hesitating to join discussions and express our thoughts. Here, I like to express my gratitude to all my colleagues in this group, Alireza Memarian, Yousef Salehi, Vamsi Krishna Puli, Amir Mohseni.

I would like to acknowledge the Department of Chemical and Materials Engineering and the University of Alberta for giving me this opportunity to continue my studies in Masters by providing a good and pleasant environment. In addition, I would like to acknowledge the Natural Sciences and Engineering Research Council of Canada for their financial support that made it possible for me to study in this prestigious university and research group.

Last but not least, I would like to thank my family for all their support emotionally and financially. They encouraged me to pursue my dreams and studies and

believe in my abilities. I would like to thank my brother Alireza, who helped me in every step of this adventure and supported me.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Introduction

Industrial control systems are important for making chemical processes work properly. They help to make process operations efficient and productive, leading to good quality products. But, surprisingly, about 60% of these systems do not work as well as they should [2]. There are many reasons for this, such as improper controller tuning, problems with the control valve, and unexpected upstream and downstream process upsets. It is similar to a complicated puzzle that is difficult to solve. When these problems happen, they introduce financial loss. So, it is really important to come up with appropriate plans to take when things go wrong and make the whole system work properly.

There are many challenges in industrial process control. Oscillation is one of these challenges. Imagine oscillations like a rhythm of ups and downs in how things change over time. These oscillations can impact how well the process stays steady and works efficiently. They can lead to undesired consequences like the production of low-quality products, using up more energy than needed, and even making operations unsafe. It is really important to detect and diagnose these oscillations because they play an important role in how well the process will operate.

In practice, oscillation detection and diagnosis is quite challenging and requires advanced methods. Think of oscillations having repeating patterns. However, these patterns can get lost amidst other unrelated noises or when they occur at varying frequencies.

The main challenge is to correctly detect and group these oscillations based on their root causes. Oscillations can originate from different sources, like when a valve is not moving smoothly, when the controller is not tuned well, or when there are external oscillatory disturbances. Each of these sources creates a unique pattern of oscillation, which is the reason to separate them. This is key for determining the root cause of the problem and finding the best path to remove it.

In the past few years, there have been many approaches proposed to detect and diagnose oscillations. Some methods focus on individual components of the control system, while others encompass a comprehensive analysis of the entire plant. One approach is to use data shapes to recognize oscillations, like patterns that stand out from randomness. These methods can help find where the problems are but cannot always tell why they have happened.

Another approach is to use "machine learning." These techniques help processes learn from experience and improve without needing to be instructed exactly what to do. They have been effective in finding and fixing oscillations in control loops [3][4]. For example, some methods use architectures to determine if a valve functions normally. Other techniques use deep neural networks to dive into more complex problems in the control loops and identify them.

However, even with all these methods, there still exist many challenges. Some methods require a large amount of data and extensive processing which can take a long time. In some cases, they might not distinguish between different causes of oscillations [5]. For instance, they might mix up problems resulting from poor controller tuning with other issues like stiction.

1.1.1 Thesis Contributions

This thesis contributes to the domain of oscillation detection and root cause analysis, with a focus on enhancing the understanding and management of oscillatory behavior within industrial control systems by using machine-learning methods. The primary contributions include:

1. Enhanced Oscillation Detection: A novel approach employing shape-based pattern recognition methods is introduced for oscillation detection. This method advances the state-of-the-art in identifying oscillations, even in scenarios involving complex noise patterns and non-stationary data.

2. Root Cause Analysis for Oscillations: The thesis presents innovative methodologies for uncovering the root causes of oscillations. These methodologies involve advanced techniques like Markov Transition Field and Deep Convolutional Neural Networks for detecting control valve stiction, and Gramian Angular Field (GAF) and Stack Autoencoder for poor controller tuning detection.

1.1.2 Thesis outline

The thesis adheres to the paper format requirements set by the Faculty of Graduate Study and Research. The significant achievements of this study, some of which have been published in peer-reviewed journals or are in the process of being published, encompass:

1. Submitted to Industrial & engineering chemistry research as A. Memarian, SK.

Damarla, B. Huang. "Shape-based pattern recognition approaches toward oscillation detection".

2. A. Memarian, SK. Damarla, B. Huang. "Control valve stiction detection using Markov transition field and deep convolutional neural network". *The Canadian Journal of Chemical Engineering* 101 (2023): 6114-6125.

3. Submitted to Computers and Chemical Engineering as A. Memarian, SK. Damarla, A. Memarian B. Huang. "Detection of poor controller tuning with Gramian Angular Field (GAF) and StackAutoencoder (SAE)".

Chapter 2 introduces the novel methodology of automatic oscillation detection through the recognition of unique shape-based pattern. This chapter delves deep into the principles and techniques that leverage these shape-specific patterns to identify oscillations more effectively. By leveraging the distinct pattern exhibited by oscillations, this approach significantly enhances the accuracy of detecting oscillatory behavior, especially when dealing with challenging factors like noise and non-stationarity in the data. The chapter presents the development of two methods capable of automatically detecting oscillations. The methods discussed in this chapter offer a robust framework for enhancing the precision of oscillation detection in industrial systems.

Chapter 3 addresses the problem of detecting control valve stiction. Here, we introduce a novel approach that combines Markov Transition Field with Deep Convolutional Neural Networks. This powerful combination helps us detect stiction-induced oscillations. By bringing together these advanced techniques, we can better detect and understand stiction in industrial control systems. This not only enhances stiction detection but also contributes to better overall process stability and efficiency.

Chapter 4 delves into the intricacies of detecting poor controller tuning using advanced methodologies. This chapter is dedicated to unraveling the intricacies of identifying poor tuning within industrial control systems. By harnessing the power of tools such as Gramian Angular Field (GAF) and Stack Autoencoder, a better understanding is achieved regarding the origins of oscillations stemming from poorly tuned controllers. The outcome is an elevated level of process stability and optimized performance.

Chapter 5 marks the conclusion and a glimpse into the road ahead. This chapter encapsulates the essential contributions made in this thesis and paves the way for prospective research avenues. It underscores the ongoing significance of comprehend-

ing and effectively addressing oscillatory phenomena within the realm of industrial control systems.

Through these chapters, this thesis strives to deepen the understanding of oscillation detection, root cause analysis, and their practical implications within the realm of industrial control systems.

Chapter 2

Shape-based Pattern Recognition Approaches Toward Oscillation Detection[1]

[1]Submitted to Industrial & engineering chemistry research as A. Memarian, SK. Damarla, B. Huang, Z. Han, M. Marvan. "Shape-based pattern recognition approaches toward oscillation detection".

2.1 Abstract

Oscillation in control loops is a frequent problem in process industries. Oscillations directly impact product quality, leading to decreased plant profit. Also, oscillation increases energy consumption, wastes raw materials and it is a significant restriction on the performance of the operation unit. Therefore, it is essential to isolate the loops that have oscillations. In this work, two methods are proposed to detect oscillations in process control loops. Both methods aim to detect the presence of a triangle-like shape in the D vs PV (or D vs OP) plot to identify oscillations in the control loops. Method 1 accomplishes the objective in an unsupervised way by fitting a nonlinear algebraic function to the data: D and PV (OP). Method 2 uses a deep convolutional neural network for detecting the triangle-like shape. The performance of both the methods was evaluated by applying them to benchmark control loops sourced from various industries, including chemical, paper, mining, and metal, along with control loops in a local refinery unit. While both the methods have their own advantages and application scenarios, results demonstrated that both the proposed methods identified oscillatory control loops for the majority of the cases studied.

2.2 Introduction

In process industries, virtually one-third of the control loops experience oscillations, hence delivering unsatisfactory performance. Oscillations originated in one control loop can propagate to other control loops through interactions between the control loops [6]. An oscillating control loop cannot maintain process variables of interest at their desired values, leading to excessive consumption of energy and raw materials, curtailment in product quality, increased production cost, etc. [7], [8]. The oscillations are often the outcomes of poorly tuned controllers, control valve faults, sensor faults, or equipment faults [9]. The identification and localization of the source of oscillations is the first and foremost step in oscillation removal. This task can be performed by manual inspection of each control loop. However, this approach turns out to be infeasible when many control loops need to be analyzed. Therefore, automatic oscillation detection methods are needed to overcome this limitation.

Over the last few decades, many automatic oscillation detection methods have been reported in the literature. Single time series oscillation detection (STSOD) and plant-wide oscillation detection (PWOD) are the two broad categories into which the existing oscillation detection techniques may be divided. The PWOD approaches are

useful to analyze a group of control loops whereas the STSOD techniques aim to detect oscillations in individual control loops. The STSOD approaches further divide into five categories: time domain, auto-covariance function (ACF), frequency domain, continuous wavelet transform (CWT), and decomposition methods. This diversified toolkit equips researchers and practitioners with versatile strategies for unearthing oscillatory behaviors within complex control systems.

Numerous STSOD techniques identify oscillations in the time domain. These approaches can be categorized as IAE-based, algorithm-based, and other approaches. For instance, in the time domain category, the seminal work by Hägglund [10, 11] introduced the IAE-based approach. This method calculates the Integral of Absolute Error (IAE) magnitude between zero-crossings, applying a threshold and counter mechanism for confirming oscillations. In a similar way, Thornhill and Hägglund [12] refined the IAE-based approach by analyzing ratios between consecutive IAE magnitudes and applying thresholds. Additionally, Salsbury and Singhal [13] introduced an innovative online detection technique, utilizing the estimation of poles in an AutoRegressive Moving Average (ARMA) process via higher-order crossing. In this method, oscillation is identified by assessing the estimated damping ratio, particularly when its values approach zero. However, these methods encounter challenges related to parameter dependencies. Algorithm-based methods offer a systematic approach to identify oscillation characteristics. The work of Zakharov and colleagues [14, 15], introduced an algorithm that defines an oscillation period, detects maximum and minimum values, and evaluates correlations between fragments. Counter values generated from this algorithm serve as indicators of oscillation presence. Additionally, Tikkala et al. [16] introduced a novel algorithm known as Robust Zero-Crossing (RZC). This method is particularly notable for its efficacy in detecting oscillations in time series with mean-nonstationarity. The RZC algorithm starts by capturing the maximum and minimum values of the time series. It then calculates the regularity of the resulting stationary time series using robust statistical measures such as the median and mean absolute deviation (MAD). By incorporating these measures, the RZC method minimizes the impact of noise-induced spurious zero-crossings, making it adept at detecting oscillations even in the presence of underlying mean-nonstationarity.

The Auto-Covariance Function (ACF) category explores methods that leverage the inherent oscillatory pattern of the autocorrelation function. Miao and Seborg [17] proposed a technique that evaluates the decay ratio of the ACF for oscillation detection. Thornhill et al. [18] introduced an approach based on oscillation regularity

evaluation using the ACF. Naghoosi and Huang [19] innovatively clustered peaks in the ACF to identify oscillatory components.

In the Frequency Domain, Babuska et al. [20] put forth a frequency-based method, leveraging power spectral density (PSD) amplitude ratios to detect oscillations. Similarly, Zhang et al. [21] employed the amplitude within the discrete Fourier transform (DFT) to detect oscillations.

Continuous Wavelet Transform (CWT) methods, as demonstrated by Guo et al. [22] and Naghoosi and Huang [23], enable simultaneous time and frequency analysis, enhancing the detection of intermittent and multiple oscillations.

Additionally, the field has witnessed the emergence of Decomposition-based methods. Empirical Mode Decomposition (EMD) [24], Local Mean Decomposition (LMD) [25], and Variational Mode Decomposition (VMD) [26] are examples of techniques that decompose time series into distinct oscillatory components. These methods particularly cater to systems with multiple oscillations, offering a specialized approach.

In refrences [27, 28, 29, 30, 31, 32] the authors thoroughly discussed various kinds of automatic oscillation detection methods.

2.2.1 Motivation and Contributions

Most existing methods for oscillation detection are complex and require high computation time, which hinders their practical applicability. There is a pressing need for an automated detection approach that can efficiently evaluate the entire plant, saving time and resources. The proposed methods highlight the development of an empirical approach that addresses this need by simplifying and expediting the detection process.

The motivation behind this research stems from the development of a novel approach to detect oscillatory control loops. By leveraging the unique shape generated by OP and PV signals, we have devised two methods for automatically detecting this distinct pattern. By adopting these approaches, this work provides a practical and effective means of identifying oscillatory control loops. Furthermore, the use of automated detection in these approaches significantly enhances the efficiency and accuracy of the process. By automating the detection process, the method can identify oscillatory loops faster and more efficiently than traditional manual detection methods.

This work makes several key contributions to the field of oscillation detection in control loops, which can be summarized as follows:

- A distinct triangle-like shape based on PV and OP signals for oscillatory loops is introduced, which enables the clear detection of oscillatory loops.
- Two methods were developed to detect triangle-like shapes in the D vs PV (or D vs OP) plot to identify oscillations in the control loop.
- Method 1 uses unsupervised learning by fitting a nonlinear algebraic function to the data: D and PV(OP).
- Method 2 employs an advanced deep convolutional neural network (CNN) to identify patterns reminiscent of triangles. The CNN is trained and evaluated using graphs derived from the generated data, allowing it to acquire the capability to differentiate between images depicting oscillatory and non-oscillatory characteristics.
- Both Method 1 and Method 2 have been tested using the benchmark control loops found within ISDB and the refinery unit. Both methods possess their distinct strengths, which will be discussed in the chapter.

The rest of the chapter is organized as follows. Section 2 provides an in-depth explanation of the proposed methodologies. In Section 3, the analysis focuses on appraising the effectiveness of the proposed methodologies within the realm of industrial control loops. Section 4 presents the pros and cons of the methods proposed in this chapter. Finally, the chapter is concluded in Section 5.

2.3 The proposed methods

The proposed methods are based on the underlying concept that when a control loop exhibits oscillation, plotting D signal (defined in Equation. 2.1) against PV or OP signal results in a triangle-like shape, as illustrated in Figure. 2.1.

The D signal is defined as:

$$\mathbf{D} = \sqrt{(\mathbf{OP} - \bar{\mathbf{OP}})^2 + (\mathbf{PV} - \bar{\mathbf{PV}})^2} \tag{2.1}$$

where $\bar{OP}$ and $\bar{PV}$ are the average of the OP and PV signals, respectively.

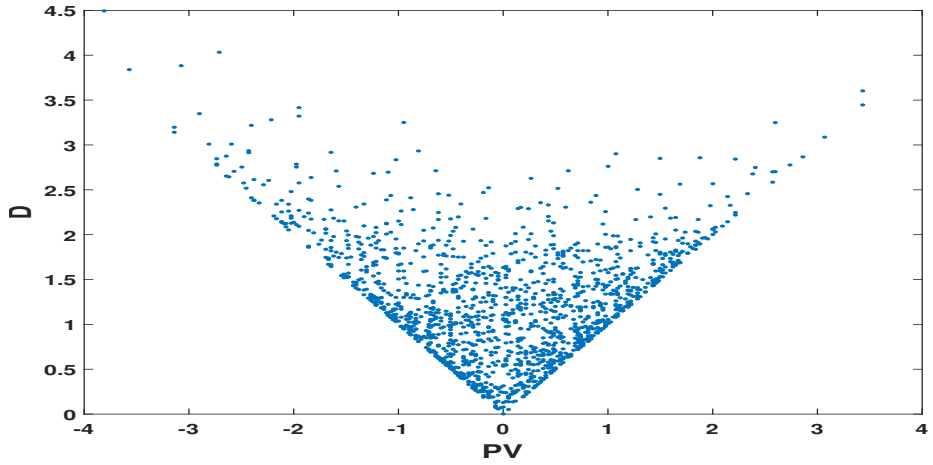
Conversely, loops without oscillation can manifest various other shapes, as de-

picted in Figure. 2.2. The presence of a triangle-like shape in the plots serves as an indication of an oscillation problem. While the visual examination of the triangle-like shape in the plots can confirm the presence of oscillation, this approach is subjective and relies on individual interpretation. To address these limitations and achieve more reliable and automated results, two methods are proposed in the present work.

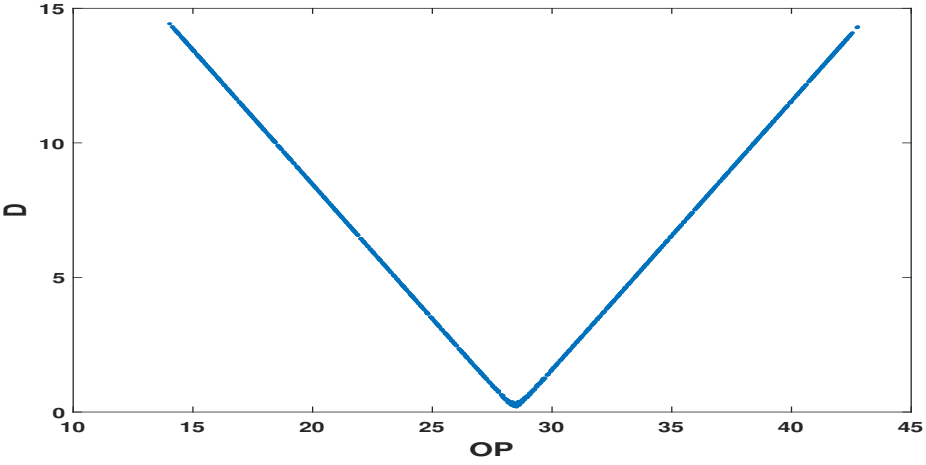
2.3.1 Method 1

In this method, to achieve more reliable results automatically, a function defined in Eq. 2.2 is developed, which produces a perfect triangle-like shape.

$$\mathbf{y} = \frac{\mathbf{x} - \mathbf{a}}{|\mathbf{x} - \mathbf{a}|} (\mathbf{m} \times \mathbf{x}) + \frac{\mathbf{a} - \mathbf{x}}{|\mathbf{a} - \mathbf{x}|} (\mathbf{m} \times \mathbf{a}) \tag{2.2}$$

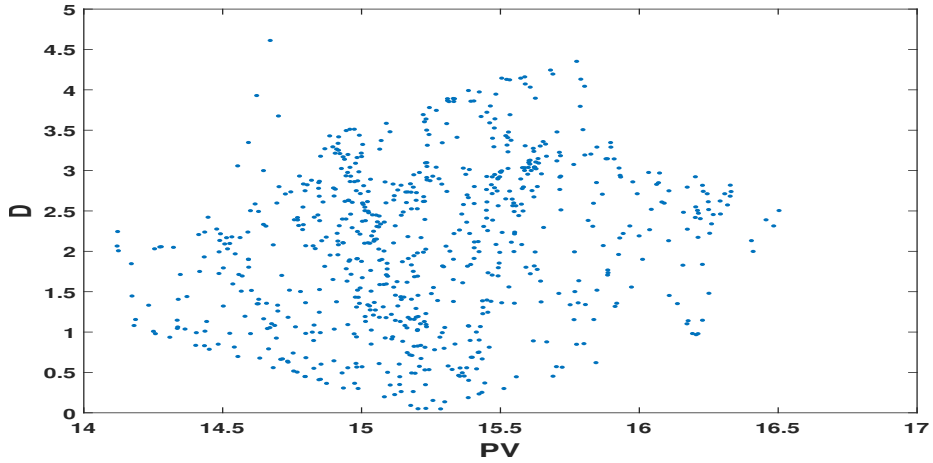


(a) Triangle-like shape in D vs PV plot

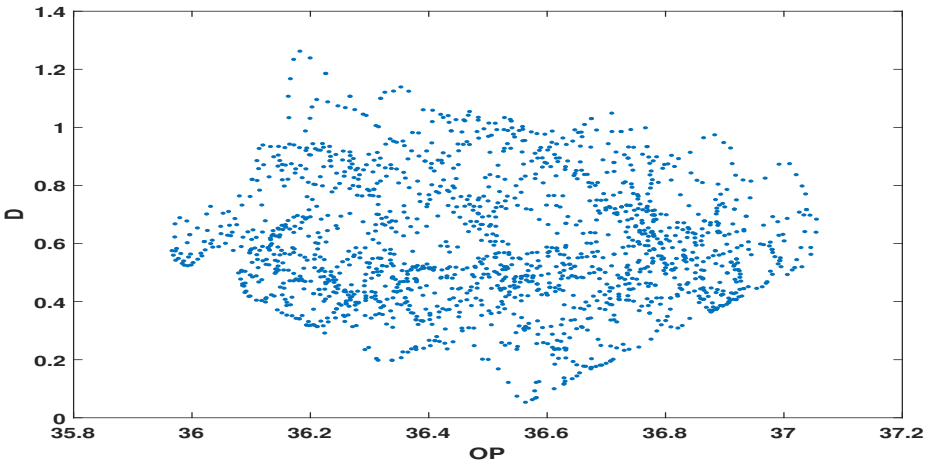


(b) Triangle-like shape in D vs OP plot

Figure 2.1: Triangle-like shape obtained from oscillatory loops.



(a) Non triangle-like shape in PV-D plot



(b) Non triangle-like shape in OP-D plot

Figure 2.2: Flowchart for Method 1.

This equation incorporates several variables, including **x**, which represents either the PV or OP depending on the specific plots where the triangle-like function is fitted. Additionally, there is the parameter **a**, which signifies the minimum value of the **D** signal, and **m**, indicating the slope of the triangle-like shape. Both **a** and **m** are unknown values that must be determined.

For a = 3, m = 6, and x $\in [-10, 16]$, the function produces the triangle-like shape as shown in Fig. 2.3.

To determine the unknown parameters, **a** and **m**, the following optimization

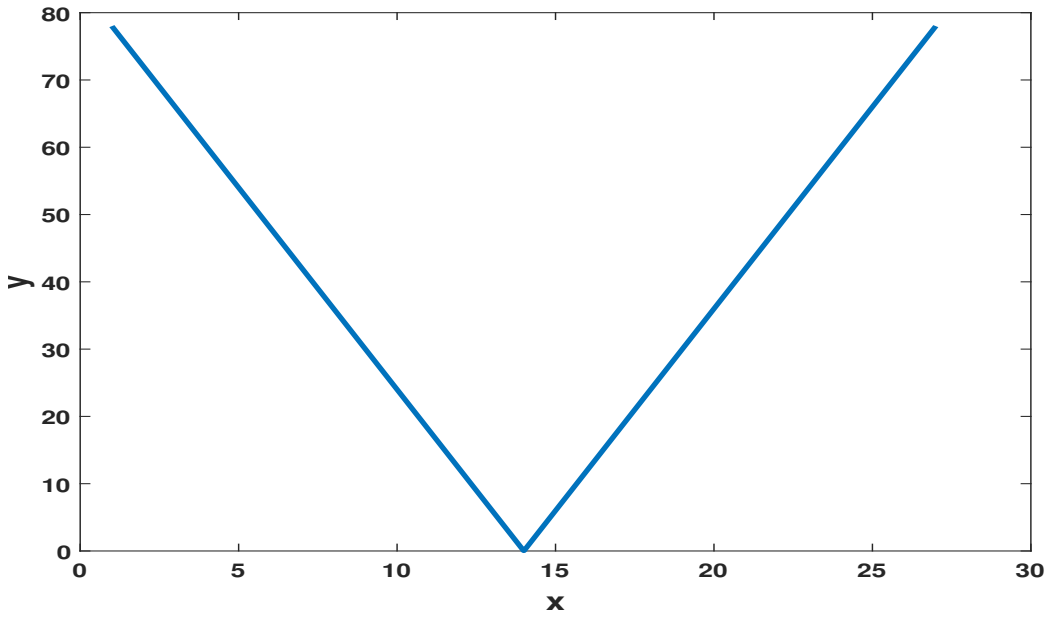


Figure 2.3: Triangle-like shape

problem (Eq. 2.3) can be solved.

$$\min_{a,m} \quad \frac{1}{N}\sum_{i=1}^{N}(y(i)-\tilde{y}(i))^2 \tag{2.3}$$

where N is the length of the PV or OP signal, $y(i) = D(i)$, and $\tilde{y}(i)$ is the output of Eq. 2.2.

Once the function is fitted to the data, the correlation coefficient defined in Eq. 2.4 can be computed.

$$R = \frac{cov(y,\tilde{y})}{\sigma_y \sigma_{\tilde{y}}} \tag{2.4}$$

where $cov(y,\tilde{y})$ represents the covariance between vectors y and $\tilde{y}$, and σ_y and $\sigma_{\tilde{y}}$ denote the standard deviations of y and $\tilde{y}$, respectively.

To determine the presence of oscillations, a predefined threshold for the correlation coefficient is established. If this threshold is surpassed by the correlation coefficient ($R \geq R_{threshold}$), it indicates the presence of oscillations in the loop. Conversely, if the threshold is not violated by the correlation coefficient ($R < R_{threshold}$), it implies that no oscillations are occurring in that specific loop. Fig. 2.4 illustrates the step-by-step process involved in developing the proposed method.

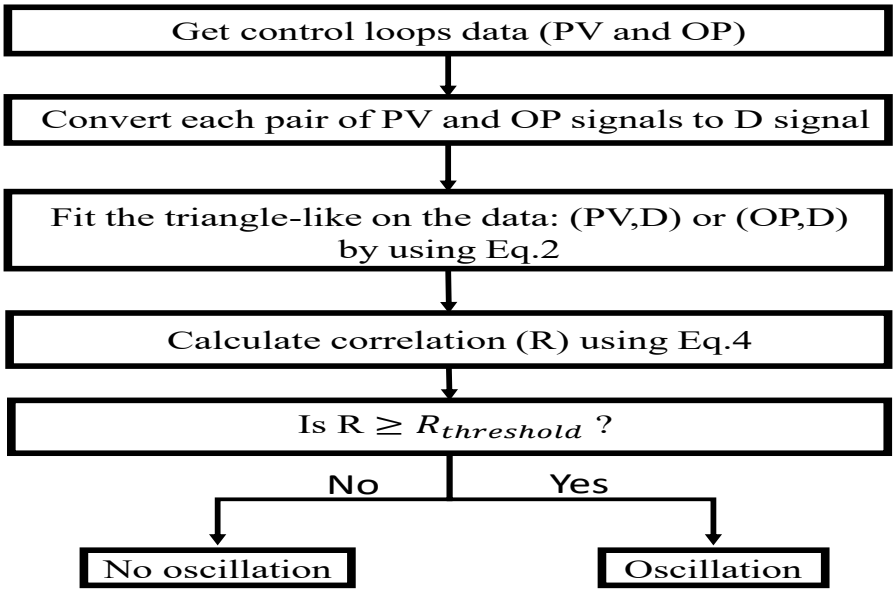


Figure 2.4: Triangle-like fitting steps

2.3.2 Method 2

Convolutional neural networks (CNNs) are a popular type of neural network that excel in the task of classifying images and videos. Unlike traditional neural networks that rely on fully connected layers to analyze inputs, CNNs employ a series of specialized layers to extract valuable information from images. What sets CNNs apart from their predecessors is their ability to automatically identify crucial features without requiring human intervention. [33]

Fig. 2.5 shows an illustration of a CNN architecture that is used in this method. The most crucial layer in a CNN is the convolutional layer since it handles the majority of calculations and is in charge of learning and extracting information from the input image. A feature map, which depicts the presence of various features at various scales, is created by this layer using a filter or kernel that glides over the input image. To learn and extract various characteristics from the input, this procedure is repeated with different filters. Let I be the input image, K be the kernel (or filter), b be the bias, and G be the output feature map. Then the convolution operation can be defined as:

$$G_{i,j} = \sigma\left(\sum_m \sum_n I_{i+m,j+n} K_{m,n} + b\right) \tag{2.5}$$

where σ is the activation function, and m and n are the indices of the kernel.

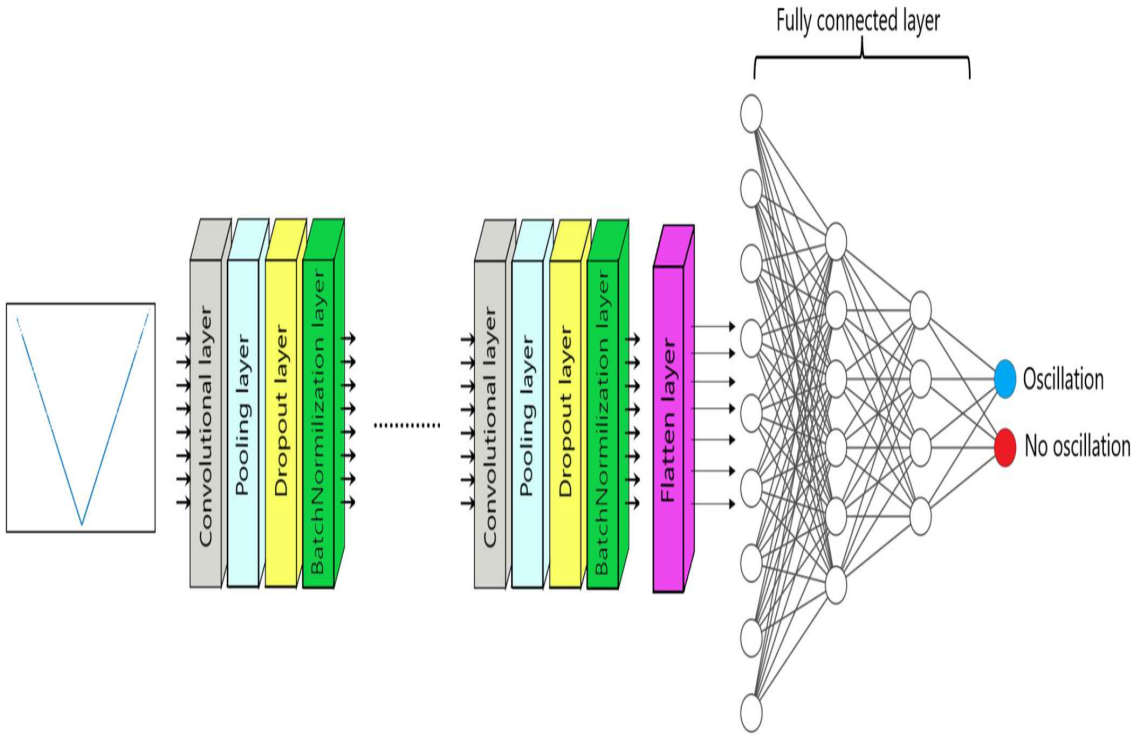


Figure 2.5: CNN architecture for oscillation detection

The pooling layer comes after the convolution layer and receives its input from the output of the preceding convolutional layer. The pooling layer reduces the size of the feature maps created in the convolutional layer. Two popular methods for this are max pooling and average pooling. Max pooling chooses the highest value possible within each pooling zone, whereas average pooling determines the average value. As a result, the computational complexity of the model is reduced and its ability to generalize to new images is enhanced. The max pooling strategy was employed in this project. Let F be the input feature map and P be the output pooled feature map. Then the max pooling operation can be defined as:

$$P_{i,j} = \max_{m,n} F_{i+m,j+n} \tag{2.6}$$

To avoid overfitting, a major issue in deep neural networks, CNNs frequently employ the dropout layer. During training, the random removal of certain neurons by this layer forces the model to acquire more robust and generalized features.

$$Y_{i,j} = \begin{cases} 0, & \text{with probability } P \\ \frac{X_{i,j}}{1-P}, & \text{otherwise} \end{cases} \tag{2.7}$$

where the probability P is typically set to a value between 0.2 and 0.5.

CNNs commonly incorporate batch normalization layers alongside other layers. These batch normalization layers are strategically positioned after the convolutional or fully connected layers to rectify the activations of the preceding layer. This is accomplished by normalizing the activations, which involves subtracting the mean and dividing by the standard deviation. By normalizing the inputs to each layer, batch normalization speeds up training, enhances generalization, and lowers overfitting. Larger learning rates may be employed during training thanks to this normalizing procedure, which may speed up convergence. Let x be the input feature map, and μ and σ be the batch mean and standard deviation, respectively. Then the batch normalization operation can be defined as:

$$\hat{x} = \frac{x - \mu}{\sqrt{\sigma^2 + \epsilon}} \tag{2.8}$$

where ϵ is a small constant to avoid division by zero. Then, the batch normalized feature map is obtained as:

$$y = \gamma\hat{x} + \beta \tag{2.9}$$

where γ and β are learned parameters that scale and shift the normalized features, respectively.

The fully connected layer, sometimes referred to as the dense layer and the final layer in a CNN, is utilized for classification. The output of the preceding layer is transformed into a one-dimensional vector and then passed into the fully connected layer. In turn, the fully connected layer generates an output vector. The output vector, consisting of the same number of neurons as the number of classes in the training data, is used to generate an estimated class label for the input image. In this project, images are divided into two categories using CNN: oscillatory and non-oscillatory. If an image obtained was oscillatory, it is labeled 0. Otherwise, it is labeled 1. For an image representing an oscillatory data set, the target vector for the CNN model is [1 0]. The target vector becomes [0 1] for the image attained from a non-oscillatory data set. There can be multiple layers of each kind in the CNN configuration. Rectified linear unit (ReLU) is used as an activation function in the convolutional and dense layers. The activation function employed in the output layer is either the sigmoid function or the softmax function. Moreover, the objective function employed in the fully connected layer is the Cross-Entropy loss function. Fig. 2.6 illustrates the step-by-step process involved in establishing the proposed method.

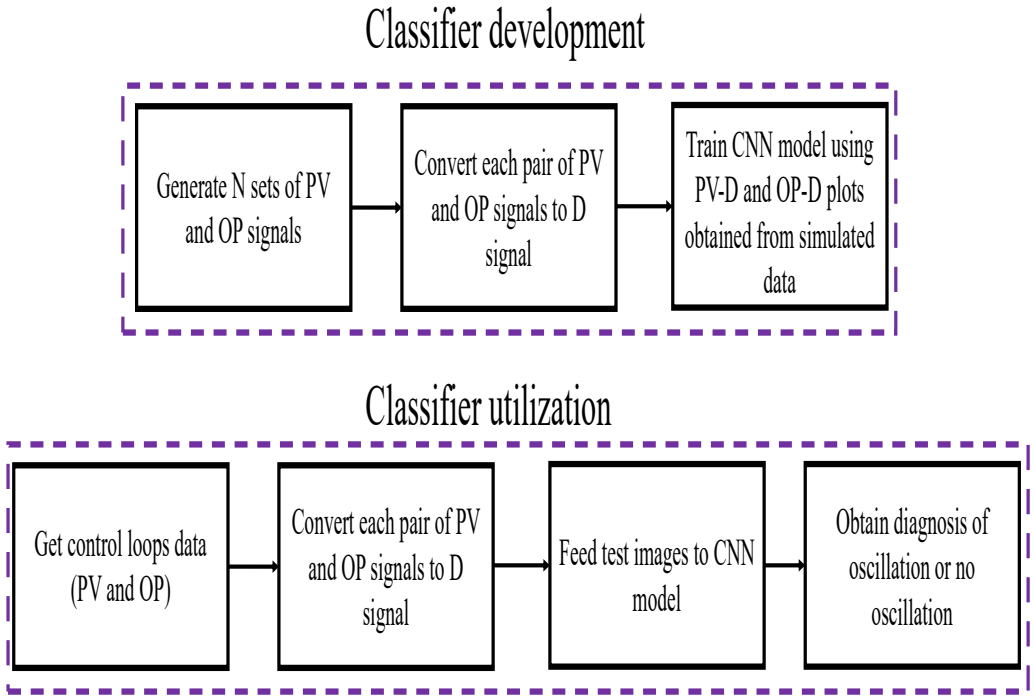


Figure 2.6: Development and utilization steps of Method 2

2.4 Results and Discussions

Within this section, an assessment of the efficacy of both techniques was conducted concerning their ability to discern oscillatory patterns. This evaluation encompassed an analysis of control loops originating from both the ISDB and the refinery unit. In addition, the proposed methods are compared with several representative existing methods from the literature.

The benchmark control loops from ISDB [34] are frequently utilized to evaluate oscillation and stiction detection techniques. The process variable (PV) and controller output (OP) data from several control loops pertaining to diverse process industries, such as buildings, chemicals, pulp and paper, power, mining, and metals, are included in the international stiction database (ISDB). The control loops take into account a variety of process variables, including temperature, pressure, level, flow rate, gauge, and concentration.

The effectiveness of the proposed methods in identifying oscillatory control loops is evaluated using several metrics: accuracy, precision, recall, specificity, and F1-

score. The accuracy metric, which evaluates the total correct categorization rate of the method, is described as follows:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{2.10}$$

where TP, TN, FP, and FN, represent the number of true positives, true negatives, false positives, and false negatives, respectively. Precision is a metric that quantifies the proportion of correctly identified positive instances among all the instances predicted as positive. It is defined as:

$$Precision = \frac{TP}{TP + FP} \tag{2.11}$$

The term recall, which is often referred to as sensitivity, quantifies the percentage of true positives among all cases that are actually positive and is defined as:

$$Recall = \frac{TP}{TP + FN} \tag{2.12}$$

The term, specificity, is the ratio of real negatives to all other actual negative cases and is determined as follows:

$$Specificity = \frac{TN}{TN + FP} \tag{2.13}$$

Last but not least, the F1 score is a harmonic mean of recall and accuracy and is given as follows:

$$F1\ score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{2.14}$$

By computing these metrics on the results of the proposed methods for the control loops taken from the ISDB and the refinery, the oscillation detection capability of each method can be evaluated, and their performance can be compared.

2.4.1 Method 1

The efficacy of the fitting triangle-like approach in identifying oscillations has been evaluated using control loops from both the ISDB and the local refinery unit. The objective was to assess the ability of the method to detect and characterize oscillatory behavior in the analyzed systems. As a first evaluation, Method 1 was applied to the time series data from the ISDB. Elaborated information regarding the control loops employed for this scrutiny can be found in Table S1, provided within the Supplementary File. This comprehensive database consists of 94 control loops, and the results of Method 1 are presented in Table 2.1 where IDC denotes "Is Diagnosis

Correct". The second evaluation focused on assessing the performance of Method 1 in a local refinery unit. Table S2 given in the Supplementary File provides the details of the control loops adopted from the refinery. For proprietary reasons, the detailed information about the local refinery is not disclosed. In the Supplementary File, the real names of the control loops have therefore been replaced with alternative identifiers to maintain proprietaries. Method 1 was applied to determine the effectiveness of oscillation detection within the refinery unit system. The results for the refinery control loops are presented in Table 2.2.

Table 2.3 presents the performance metrics computed using the results obtained from Method 1 for the ISDB and the refinery control loops. For both the ISDB and the refinery control loops, a threshold of 0.9 was chosen for the correlation coefficient. This threshold was carefully selected to strike a balance between minimizing false positives and capturing genuine oscillatory behavior. Table 2.3 shows an accuracy of 80.85% for ISDB and 81.25% for refinery control loops.

2.4.2 Method 2

In contrast to Method 1, which does not require training, convolutional neural networks (CNN) need training and have limited effectiveness when minimal data is available. As a result, the ISDB and refinery datasets cannot provide sufficient data to establish optimal parameters for Method 2. To address this limitation, an alternative approach was employed.

To generate a more extensive dataset for oscillation detection using CNN, a control loop model (depicted in Fig. 4.4) was utilized. This model allowed for the simulation of closed-loop signals, PV and OP, to create an ample amount of data and overcome the data scarcity challenge.

Table 2.1: Performance of Method 1 on ISDB

LN	AP	OM1		IDC?
BAS 1	No oscillation	0.11	0.999	No
BAS 2	No oscillation	0.75	0.68	Yes
BAS 3	Oscillation	0.6	0.999	Yes
BAS 4	Oscillation	0.01	0.999	Yes
BAS 5	Oscillation	1	—	Yes
BAS 6	Oscillation	0.08	0.993	Yes
BAS 7	Oscillation	0.006	0.999	Yes
BAS 8	No oscillation	0.999	0.42	No

Table 2.1: (continued) Performance of Method 1 on ISDB

LN	AP	OM1		IDC?
CHEM 1	Oscillation	0.93	0.054	Yes
CHEM 2	Oscillation	0.98	0.73	Yes
CHEM 3	Oscillation	0.08	0.999	Yes
CHEM 4	Oscillation	0.91	0.94	Yes
CHEM 5	Oscillation	0.99	0.44	Yes
CHEM 6	Oscillation	0.999	0.64	Yes
CHEM 7	Oscillation	0.99	0.58	Yes
CHEM 8	Oscillation	0.99	0.46	Yes
CHEM 9	Oscillation	0.999	0.96	Yes
CHEM 10	Oscillation	0.83	0.07	No
CHEM 11	Oscillation	0.96	0.17	Yes
CHEM 12	Oscillation	0.96	0.07	Yes
CHEM 13	Oscillation	0.8	0.798	No
CHEM 14	Oscillation	0.75	0.72	No
CHEM 15	Oscillation	0.71	0.78	No
CHEM 16	Oscillation	0.72	0.79	No
CHEM 17	Oscillation	0.69	0.65	No
CHEM 18	Oscillation	0.96	0.86	Yes
CHEM 19	Oscillation	0.46	0.95	Yes
CHEM 20	Oscillation	0.9	0.98	Yes
CHEM 21	Oscillation	0.05	0.99	Yes
CHEM 22	Oscillation	0.12	0.99	Yes
CHEM 23	Oscillation	0.07	0.999	Yes
CHEM 24	Oscillation	0.99	0.08	Yes
CHEM 25	Oscillation	0.96	0.97	Yes
CHEM 26	Oscillation	0.96	0.97	Yes
CHEM 27	Oscillation	0.99	0.93	Yes
CHEM 28	Oscillation	0.12	0.797	No
CHEM 29	Oscillation	0.999	0.44	Yes
CHEM 30	Oscillation	1	0.38	Yes
CHEM 31	Oscillation	1	0.06	Yes
CHEM 32	Oscillation	0.02	0.99	Yes
CHEM 33	Oscillation	0.98	0.001	Yes
CHEM 34	Oscillation	0.98	0.001	Yes
CHEM 35	Oscillation	0.95	0.62	Yes
CHEM 36	Oscillation	0.99	0.94	Yes
CHEM 37	Oscillation	0.98	0.91	Yes
CHEM 38	Oscillation	0.35	0.999	Yes
CHEM 39	Oscillation	0.45	0.999	Yes
CHEM 40	No oscillation	0.81	0.47	Yes
CHEM 41	No oscillation	0.85	0.88	Yes
CHEM 42	No oscillation	0.74	0.67	Yes

Table 2.1: (continued) Performance of Method 1 on ISDB

LN	AP	OM1		IDC?
CHEM 43	No oscillation	0.78	0.42	Yes
CHEM 44	No oscillation	0.61	0.62	Yes
CHEM 45	No oscillation	0.72	0.65	Yes
CHEM 46	No oscillation	0.92	0.94	No
CHEM 47	No oscillation	0.78	0.82	Yes
CHEM 48	No oscillation	0.51	0.84	Yes
CHEM 49	No oscillation	0.44	0.72	Yes
CHEM 50	No oscillation	0.80	0.68	Yes
CHEM 51	No oscillation	0.66	0.66	Yes
CHEM 52	No oscillation	0.78	0.63	Yes
CHEM 53	No oscillation	0.87	0.88	Yes
CHEM 54	No oscillation	0.94	0.95	No
CHEM 55	No oscillation	0.83	0.81	Yes
CHEM 56	No oscillation	0.96	0.96	No
CHEM 57	No oscillation	0.74	0.73	Yes
CHEM 58	No oscillation	0.89	0.89	Yes
CHEM 59	No oscillation	0.95	0.92	No
CHEM 60	No oscillation	0.68	0.64	Yes
CHEM 61	No oscillation	0.78	0.74	Yes
CHEM 62	No oscillation	0.85	0.89	Yes
CHEM 63	No oscillation	0.60	0.03	Yes
CHEM 64	No oscillation	0.78	0.84	Yes
PAP 1	Oscillation	0.3	0.88	No
PAP 2	Oscillation	0.70	0.27	No
PAP 3	Oscillation	0.98	0.87	Yes
PAP 4	Oscillation	0.998	0.947	Yes
PAP 5	Oscillation	0.328	0.942	Yes
PAP 6	Oscillation	0.879	0.139	No
PAP 7	Oscillation	0.692	0.794	No
PAP 8	Oscillation	0.883	0.639	No
PAP 9	Oscillation	0.999	0.969	Yes
PAP 10	No oscillation	0.176	0.825	Yes
PAP 11	Oscillation	0.999	0.047	Yes
PAP 12	Oscillation	0.924	0.248	Yes
PAP 13	Oscillation	0.41	0.994	Yes
POW 1	Oscillation	0.596	0.999	Yes
POW 2	Oscillation	0.074	0.999	Yes
POW 3	Oscillation	0.323	0.999	Yes
POW 4	Oscillation	0.087	0.998	Yes
POW 5	Oscillation	0.596	0.976	Yes
MIN 1	Oscillation	0.999	0.008	Yes
MET 2	Oscillation	0.11	0.985	Yes
MET 3	No oscillation	0.08	0.635	Yes

Table 2.2: Performance of the Method 1 on the refinery unit

LN	AP	OM1		IDC?
FC 1	Oscillation	0.99	0.995	Yes
FC 2	Oscillation	0.99	0.999	Yes
FC 3	Oscillation	0.04	0.999	Yes
FC 4	Oscillation	0.98	0.28	Yes
FC 5	Oscillation	0.06	0.999	Yes
FC 6	No oscillation	0.56	0.88	Yes
FC 7	No oscillation	0.98	0.999	No
FC 8	Oscillation	0.75	0.87	No
FC 9	Oscillation	0.30	0.997	Yes
FC 10	Oscillation	0.05	0.997	Yes
FC 11	Oscillation	0.18	0.98	Yes
TC 1	Oscillation	0.14	0.98	Yes
TC 2	Oscillation	0.95	0.45	Yes
TC 3	Oscillation	0.76	0.34	No
TC 4	No oscillation	0.78	0.57	Yes
TC 5	Oscillation	1	0.99	Yes
TC 6	Oscillation	0.04	0.97	Yes
TC 7	Oscillation	0.02	0.96	Yes
TC 8	Oscillation	0.08	0.99	Yes
LC 1	Oscillation	0.98	0.36	Yes
LC 2	Oscillation	0.999	0.999	Yes
LC 3	Oscillation	0.84	0.64	No
LC 4	Oscillation	0.998	0.999	Yes
LC 5	Oscillation	0.94	0.97	Yes
LC 6	Oscillation	0.80	0.89	No
LC 7	No oscillation	0.53	0.96	No
LC 8	Oscillation	0.48	0.99	Yes
LC 9	Oscillation	0.66	0.99	Yes
LC 10	Oscillation	0.92	0.80	Yes
LC 11	No oscillation	0.62	0.88	Yes
LC 12	Oscillation	0.73	0.94	Yes

Various factors known to create oscillations in industrial control loops, such as stiction, poor tuning, and external disturbances, were simulated to produce oscillatory data that closely resembles real-world scenarios. Both concentration and level loops, representing different control loop types, were simulated to capture diverse oscillatory behaviors. Table 2.4 furnishes the process dynamics and controller transfer functions employed to formulate the simulation dataset. The data-inferred stiction model introduced in [1] was employed to elicit oscillations arising from stiction within both

Table 2.3: Performance metrics for Method 1.

Performance metric	ISDB	Refinery unit
True positive	52	24
True negative	24	4
False positive	6	1
False negative	12	2
Precision (%)	90.00	92.31
Recall (%)	81.00	85.71
Specificity (%)	80.00	60.00
F1 score (%)	85.26	88.89
Accuracy (%)	80.85	81.25

the concentration and level control loops. Delving into specifics, Table 2.5 presents the chosen parameter values for the data-driven stiction model. Furthermore, oscillatory data were generated under two distinct scenarios: inadequately tuned settings and external oscillatory disturbances. The specific parameter values for generating oscillatory data in these scenarios can be found in Tables 2.6 and 2.7.

To generate non-oscillatory database, a specific approach was employed. The focus was on creating a dataset that can represent the behavior of the system under stable conditions, without any stiction-induced oscillations or disturbances. This was accomplished by making adjustments to the control parameters and system settings. By carefully fine-tuning these parameters, a smooth and stable behavior was achieved throughout the dataset, ensuring the absence of any undesired oscillations.

The generated oscillatory and non-oscillatory data are used to generate PV vs D and OP vs D images and they are considered a database for CNN training. The database consists of 340 control loops. To ensure a thorough evaluation of the performance of Method 2, the database was divided into a training set and a test set. The training set consisted of 80% of the total dataset, amounting to 272 images, while the remaining 20%, totaling 68 images, was allocated for testing the accuracy of Method 2. Table 3.5 lists the hyperparameters that were employed in the model and Figure. 4.7(a) and Figure. 4.7(b) show the training loss and training accuracy of the CNN model, respectively. The trained CNN model was tested on both the benchmark control loops and the refinery control loops.

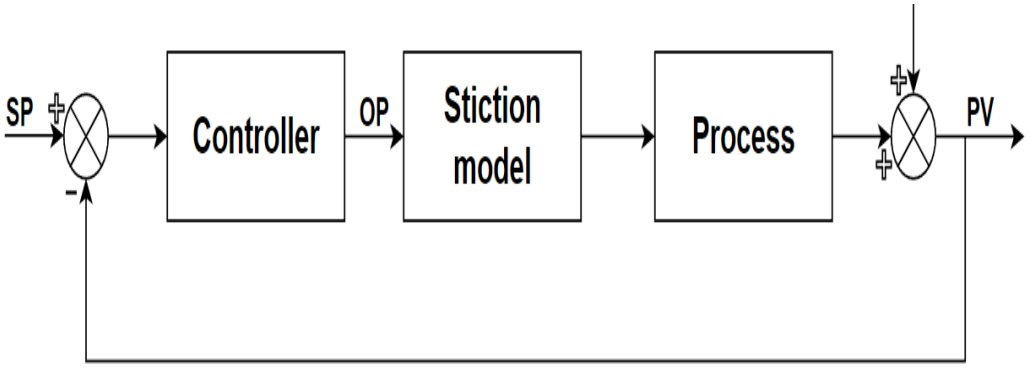
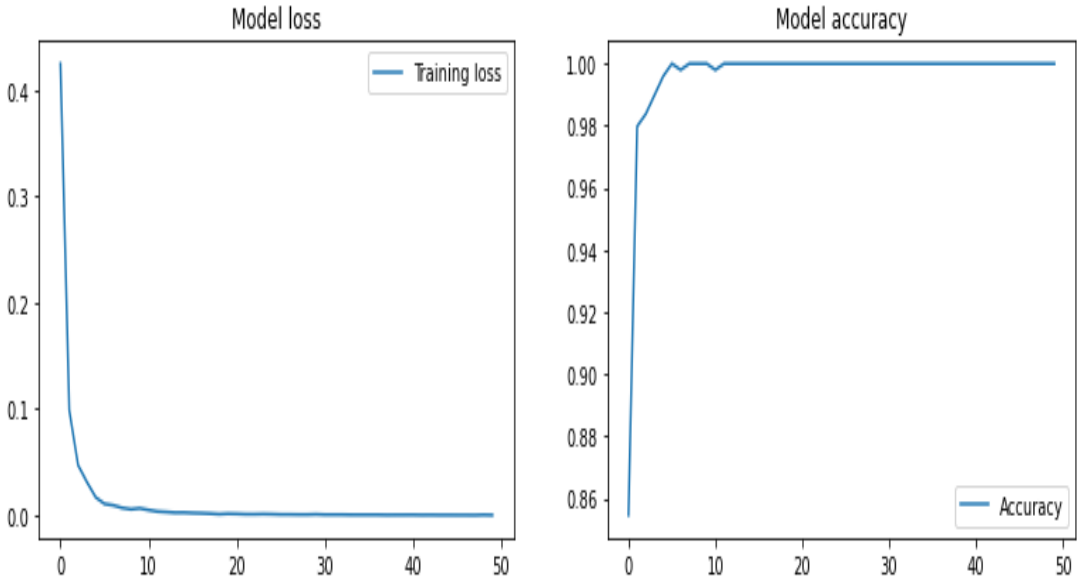


Figure 2.7: Control loop

Results obtained via Method 2 for the ISDB and refinery control loops are provided in Table 2.9 and Table 2.10. Table 2.11 provides values attained for the performance metrics for the ISDB and refinery unit dataset. Upon reviewing Tables 2.11, it becomes evident that Method 2 exhibits a performance of 91.45% for the ISDB dataset and 90.32% for the refinery unit dataset. Method 2 demonstrated superior oscillation detection performance over Method 1. According to the comparison in Table 2.12, it is evident that among the methods under consideration, Method 2 exhibited superior accuracy compared to them.



(a) Training loss of the model

(b) Training accuracy of the model

Figure 2.8: Performance of CNN model.

Table 2.4: Representation of the closed-loop simulations encompasses the process model and the controller transfer function.

Loop type	Process	Controller
Concentration	$G_p = \frac{3e^{-10s}}{10s+1}$	$G_c = 0.2(\frac{10s+1}{10s})$
Level	$G_p = \frac{1}{s}$	$G_c = 0.4(\frac{2s+1}{2s})$

Table 2.5: Variation of Parameters for Generating Oscillatory Data with Stiction-induced Controller.

Parameter	Description	Parameter range
S	Stiction band	[0.5:0.25:10]
J	Slip jump	[0.1:0.25:5]
V	Variance of white noise	0.01

Table 2.6: Variation of Parameters for Generating Oscillatory Data with tightly-tuned Controllers.

Parameter	Description	Parameter range
K_c	Controller gain	[0.1:0.01:0.3]
I	Final controller integral value	[0.01:0.01:0.27]
V	Variance of noise	0.01

Table 2.7: Variation of Parameters for Generating Oscillatory Data with external oscillatory disturbances.

Parameter	Description	Parameter range
A	Amplitude	[1, 1.5, 2, 2.5]
F	Frequency of oscillation value	[0.01:0.01:0.27]
V	Variance of noise	0.01

Table 2.8: CNN hyperparameters.

Description	Values/Functions
No. of Convolution layers	2
No. of Pooling layers	2
No. of BatchNormilization layers	2
Size of each filter (Conv. layers)	3×3
Size of each filter (Pooling layers)	2×2
Type of pooling operation used	Max pooling
Dropout rate	0.3
No. of neurons in fully connected layer	256
Optimization method	Adam
Activation function	ReLU,Sigmoid

Table 2.9: Performance of Method 2 on ISDB

LN	AP	OM2		IDC?
Bas 1	No oscillation	0.001	0.999	Yes
Bas 2	No oscillation	0	1	Yes
Bas 3	Oscillation	1	0	Yes
Bas 4	Oscillation	0.996	0.004	Yes
Bas 5	Oscillation	1	0	Yes
Bas 6	Oscillation	1	0	Yes
Bas 7	Oscillation	1	0	Yes
Bas 8	No oscillation	0.697	0.303	No
CHEM 1	Oscillation	0.999	0.001	Yes
CHEM 2	Oscillation	0.999	0.001	Yes
CHEM 3	Oscillation	0.993	0.007	Yes
CHEM 4	Oscillation	1	0	Yes
CHEM 5	Oscillation	1	0	Yes
CHEM 6	Oscillation	1	0	Yes
CHEM 7	Oscillation	1	0	Yes
CHEM 8	Oscillation	1	0	Yes
CHEM 9	Oscillation	1	0	Yes
CHEM 10	Oscillation	1	0	Yes
CHEM 11	Oscillation	1	0	Yes
CHEM 12	Oscillation	1	0	Yes
CHEM 13	Oscillation	0.86	0.14	Yes
CHEM 14	Oscillation	0.41	0.59	No
CHEM 15	Oscillation	0.377	0.623	No
CHEM 16	Oscillation	0.998	0.002	Yes
CHEM 17	Oscillation	0.89	0.11	Yes
CHEM 18	Oscillation	0.999	0.001	Yes
CHEM 19	Oscillation	0.966	0.034	Yes

Table 2.9: (continued) Performance of Method 2 on ISDB

LN	AP	OM2		IDC?
CHEM 20	Oscillation	0.998	0.002	Yes
CHEM 21	Oscillation	0.99	0.01	Yes
CHEM 22	Oscillation	0.995	0.005	Yes
CHEM 23	Oscillation	0.999	0.001	Yes
CHEM 24	Oscillation	0.999	0.001	Yes
CHEM 25	Oscillation	1	0	Yes
CHEM 26	Oscillation	0.96	0.04	Yes
CHEM 27	Oscillation	0.998	0.002	Yes
CHEM 28	Oscillation	0.982	0.018	Yes
CHEM 29	Oscillation	1	0	Yes
CHEM 30	Oscillation	0.98	0.02	Yes
CHEM 31	Oscillation	0.978	0.022	Yes
CHEM 32	Oscillation	0.99	0.01	Yes
CHEM 33	Oscillation	0.931	0.069	Yes
CHEM 34	Oscillation	0.82	0.18	Yes
CHEM 35	Oscillation	0.999	0.001	Yes
CHEM 36	Oscillation	0.999	0.001	Yes
CHEM 37	Oscillation	0.98	0.02	Yes
CHEM 38	Oscillation	1	0	Yes
CHEM 39	Oscillation	0.794	0.206	Yes
CHEM 40	No oscillation	0.001	0.999	Yes
CHEM 41	No oscillation	0.04	0.96	Yes
CHEM 42	No oscillation	0.001	0.999	Yes
CHEM 43	No oscillation	0.001	0.999	Yes
CHEM 44	No oscillation	0.042	0.958	Yes
CHEM 45	No oscillation	0.07	0.93	Yes
CHEM 46	No oscillation	0.92	0.08	No
CHEM 47	No oscillation	0.18	0.82	Yes
CHEM 48	No oscillation	0.001	0.999	Yes
CHEM 49	No oscillation	0.1	0.9	Yes
CHEM 50	No oscillation	0	1	Yes
CHEM 51	No oscillation	0	1	Yes
CHEM 52	No oscillation	0	1	Yes
CHEM 53	No oscillation	0.074	0.926	Yes
CHEM 54	No oscillation	0.05	0.95	Yes
CHEM 55	No oscillation	0.19	0.81	Yes
CHEM 56	No oscillation	0.53	0.47	No
CHEM 57	No oscillation	0.27	0.73	Yes
CHEM 58	No oscillation	0.11	0.89	Yes
CHEM 59	No oscillation	0.001	0.999	Yes
CHEM 60	No oscillation	0.001	0.999	Yes

Table 2.9: (continued) Performance of Method 2 on ISDB

LN	AP	OM2		IDC?
CHEM 61	No oscillation	0	1	Yes
CHEM 62	No oscillation	0.11	0.89	Yes
CHEM 63	No oscillation	0	1	Yes
CHEM 64	No oscillation	0	1	Yes
PAP 1	Oscillation	0.404	0.596	No
PAP 2	Oscillation	0.999	0.001	Yes
PAP 3	Oscillation	0.938	0.062	Yes
PAP 4	Oscillation	0.971	0.029	Yes
PAP 5	Oscillation	1	0	Yes
PAP 6	Oscillation	0.363	0.637	No
PAP 7	Oscillation	0.46	0.54	No
PAP 8	Oscillation	0.883	0.117	Yes
PAP 9	Oscillation	0.999	0.001	Yes
PAP 10	No oscillation	0.165	0.825	Yes
PAP 11	Oscillation	0.999	0.001	Yes
PAP 12	Oscillation	0.924	0.076	Yes
PAP 13	Oscillation	1	0	Yes
POW 1	Oscillation	0.999	0.001	Yes
POW 2	Oscillation	0.999	0.001	Yes
POW 3	Oscillation	0.999	0.001	Yes
POW 4	Oscillation	0.998	0.002	Yes
POW 5	Oscillation	0.769	0.231	Yes
MIN 1	Oscillation	0.999	0.001	Yes
MET 2	Oscillation	0.99	0.01	Yes
MET 3	No oscillation	0.2	0.8	Yes

Table 2.10: Performance of the CNN method on the refinery unit

LN	AP	OM2		IDC?
FC 1	Oscillation	0.903	0.097	Yes
FC 2	Oscillation	0.966	0.034	Yes
FC 3	Oscillation	0.923	0.999	Yes
FC 4	Oscillation	0.78	0.22	Yes
FC 5	Oscillation	0.996	0.004	Yes
FC 6	No oscillation	0.107	0.893	Yes
FC 7	No oscillation	0.009	0.991	Yes
FC 8	Oscillation	0.85	0.15	Yes
FC 9	Oscillation	0.997	0.003	Yes
FC 10	Oscillation	0.996	0.004	Yes
FC 11	Oscillation	0.996	0.004	Yes
TC 1	Oscillation	0.998	0.002	Yes

Table 2.10: (continued) Performance of the CNN method on the refinery unit

LN	AP	OM2		IDC?
TC 2	Oscillation	0.998	0.002	Yes
TC 3	Oscillation	0.998	0.002	No
TC 4	No oscillation	0.032	0.968	Yes
TC 5	Oscillation	0.991	0.009	Yes
TC 6	Oscillation	0.985	0.015	Yes
TC 7	Oscillation	0.995	0.005	Yes
TC 8	Oscillation	0.997	0.003	Yes
LC 1	Oscillation	0.98	0.02	Yes
LC 2	Oscillation	0.999	0.001	Yes
LC 3	Oscillation	0.36	0.64	No
LC 4	Oscillation	1	0	Yes
LC 5	Oscillation	0.94	0.06	Yes
LC 6	Oscillation	0.11	0.89	No
LC 7	No oscillation	0.34	0.66	No
LC 8	Oscillation	0.99	0.01	Yes
LC 9	Oscillation	0.998	0.002	Yes
LC 10	Oscillation	0.92	0.08	Yes
LC 11	No oscillation	0.12	0.88	Yes
LC 12	Oscillation	0.999	0.001	Yes

Table 2.11: Performance metrics for Method 2.

Performance metric	ISDB	Refinery unit
True positive	59	24
True negative	27	4
False positive	3	1
False negative	5	2
Precision (%)	95.16	96
Recall (%)	92.19	92.31
Specificity (%)	90.00	80
F1 score (%)	93.65	94.12
Accuracy (%)	91.49	90.32

2.5 Pros and cons of Methods 1 and 2

Table 2.13 offers a comparative analysis of the merits and demerits associated with Methods 1 and 2. In cases where the response is highlighted in *red* for any of the criteria specified in the initial column of the table, it signifies a limitation or disadvantage.

As it has been mentioned in Table 2.13, Method 1 is inaccurate for detecting shapes that deviate from a perfect triangle. Fig. 2.9 shows an oscillatory loop that deviates from the perfect triangle-like shape.

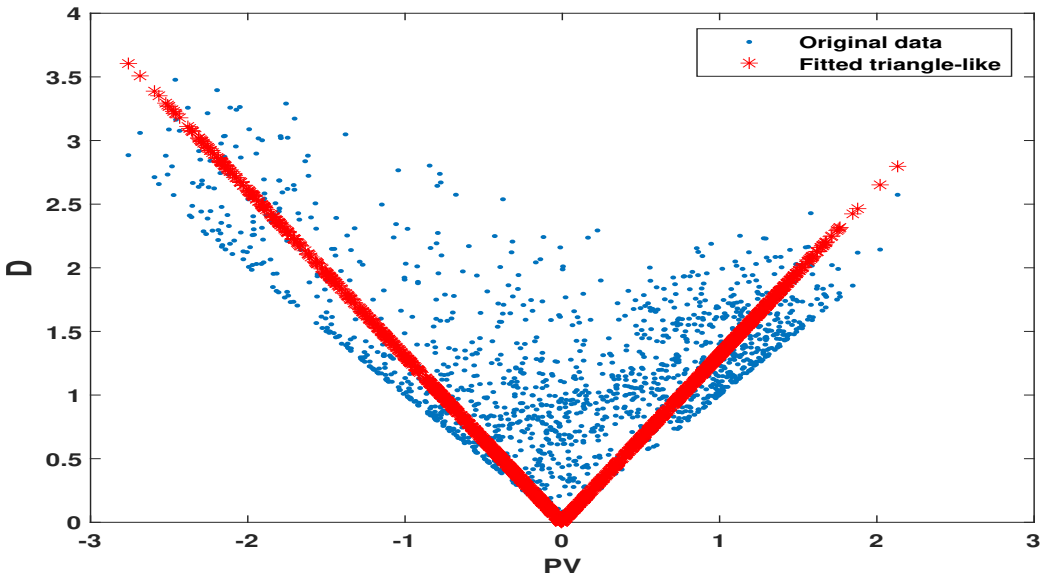


Figure 2.9: The fitted triangle-like shape on the plot deviates from a perfect triangle.

In conclusion, both methods have their own benefits and drawbacks, and the choice of the method relies on the particular application and the resources that are available.

Table 2.12: Comparison with existing methods

Oscillation Detection Method	No. of Correct Diagnoses (ISDB)	No. of Correct Diagnoses (Refinery)
Proposed Method (Method 1)	76	25
Proposed Method (Method 2)	**86**	**28**
IAE [10]	35	10
RZC [16]	82	**28**
ACF [19]	64	21
CWT [23]	78	26
ARMA [13]	49	15
Fuzzy-logic [20] System	68	24
EMD [24]	78	24

Table 2.13: Characteristics of the proposed methods.

Criterion	Method 1	Method 2
Requires routine operation data	Yes	Yes
Practically implementable	Yes	Yes
Applicable to all control loops	Yes	Yes
Valid for control loops with constant or varying setpoint	Yes	Yes
Applicable to loops with multiple oscillations	Yes	Yes
Robustness to noise in PV and OP	No	Yes
Involves extensive training	No	Yes
Entails complex computations	No	Yes
Accurate detection of shapes deviate from the perfect triangle-like shapes	No	Yes
Exhibits comparable or better performance than the existing methods	No	Yes

2.6 Conclusion

The proposed methods provide effective approaches for identifying the specific loop shape that exhibits oscillatory behavior, offering potential benefits in various industries. Method 1 and Method 2 have demonstrated respective accuracy rates of approximately 81% and 91%. These methods, while complementing to each other, can be valuable tools in practical applications, contributing to improvements in product quality, energy efficiency, reduction of raw material waste, and enhancement of safety measures in process industries. By leveraging these identification techniques, industries can optimize their operations and achieve better overall performance.

Acknowledgement

The authors gracefully acknowledge the funding from the Natural Sciences and Engineering Research Council (NSERC) of Canada.

Chapter 3

Control Valve Stiction Detection using Markov Transition Field and Deep Convolutional Neural Network[1]

[1] A. Memarian, SK. Damarla, B. Huang. "Control valve stiction detection using Markov transition field and deep convolutional neural network". *The Canadian Journal of Chemical Engineering* 101 (2023): 6114-6125

3.1 Abstract

Control valve stiction is an industrial problem and often causes oscillations in process control loops. Oscillating control loops are not capable of maintaining key process variables near or at their desired values, thus yielding low-quality products, inducing economic loss, and increasing environmental impacts. Therefore, it is of vital importance to detect stiction in industrial control valves. In this regard, the present work proposes a new method based on the Markov transition field and convolutional neural network (CNN) to identify sticky control valves in industrial control loops. The Markov transition field is employed to convert process variable (PV) and controller output (OP) into two-dimensional images, which are then utilized by CNN to learn to distinguish stiction induced oscillations from oscillations brought out by a non-stiction condition. A transfer learning strategy is adopted to improve the stiction detection capability of the proposed method. Its performance is evaluated via its application to benchmark control loops taken from the chemical, paper, mining, and metal industries. Results demonstrate that the proposed method obtains the correct verdict for the majority of the control loops studied.

3.2 Introduction

Process industries embody hundreds or even thousands of control loops maintaining key process variables near or at their desired values. Satisfactorily performing control loops aid in meeting production targets, adhering to product quality specifications, and obeying environmental regulations [35, 36]. Practical issues such as control valve faults, abnormally behaving sensors, interactions between control loops, and upstream or downstream upsets have the potential to create oscillations in the control loops, making it difficult to accomplish the above-mentioned objectives. Therefore, it is of vital importance to timely detect the oscillations and their root causes. Studies conducted on over twenty thousand PID control loops revealed that the majority of the control loops had oscillations and the oscillations in twenty to thirty percent of these loops were engendered by control valve stiction [37], [38], [39], [40]. Valve stiction is an equipment fault and impedes the free movement of the valve stem, negatively impacting the performance of a control loop. In spite of the fact that intrusive techniques can precisely detect sticky control valves, they are not popular in practice because their application interrupts running plants. On the contrary, non-invasive or automatic methods detect stiction based on closed-loop signals (OP and PV) and therefore do not disturb the plant. Owing to this reason, automatic stiction

detection methods are becoming more popular in process industries, and research has been active in this field. Non-linearity detection, limit cycles, optimization methods, multivariate statistical approaches, waveform forms, and machine learning algorithms have been used in the existing automatic stiction detection methods [34], [30].

Machine learning (ML) and artificial intelligence benefit process industries by providing systems the ability to automatically learn and improve from experience without being explicitly programmed for a specific task. These ubiquitous techniques are being utilized to improve process operations in varied fields such as process monitoring, inferential measurement, predictive maintenance, autonomous PID tuning, etc. ML finds applications in the detection and diagnosis of oscillations in industrial control loops. Zheng et al. [9] presented a novel stiction detection and quantification method based on moving window methodology and K-means clustering. Besides detecting sticky valves, this method can also identify abrupt valve closures. Damarla at el. [41] presented a practical linear regression-based method for the detection and quantification of stiction in industrial control loops. Damarla et al. [42] discussed a further developed approach for stiction detection. This approach fits Sigmoid function (or logistic function) to the processed data: the first derivative of PV and OP. If the given valve has stiction, the sigmoid function fits the data reasonably well and the correlation between OP and the output of the sigmoid function is above 0.5.

Multi-layer perceptron network-based methodologies have been proved to be effective in the identification of valve stiction [43], [44], [45], [46]. Yazdi et al. [47] developed support vector machine-based methodology as an alternative machine learning technique to valve stiction detection. This method uses Hotelling's T^2 distribution to combine PV and OP signals into a single time series. However, it has the limitation that it relies on a chosen kernel function and is not robust to noise in real data. A deep neural network (DNN) is a significant improvement in artificial intelligence that expands on regular neural networks by including additional hidden neurons and network layers. DNNs, unlike standard artificial neural networks (ANNs), have a complicated structure that enables them to automatically recognize nonlinear and complex aspects in process data. This enhanced feature allows DNNs to outperform standard ANN models in terms of generalization performance. Lately, DNNs have been proven very effective for the diagnosis of oscillating control loops. In contrast to shallow neural networks, deep neural networks such as convolutional neural network (CNN) requires many labeled data for their training. However, the available industrial data for stiction and non-stiction categories (tightly tuned controllers and

external oscillatory disturbances) are not sufficient to learn the parameters of the CNN model. To overcome this problem, Henry et al. [3] generated simulated data sets by using the valve stiction model devised by Choudhury et al. [48] and these data sets were used to partially retrain AlexNet. The retrained CNN was tested on the industrial data available in the international stiction database (ISDB) posted by Jelali and Huang [34]. Kamaruddin et al. [49] formulated a simple model-free butterfly shape-based detection (BSD) method integrated with CNN for valve stiction detection and quantification. Henry et al. [3] combined CNN with principal component analysis for stiction detection. In this hybrid methodology, CNN was first used to extract features from time series data belonging to stiction and non-stiction categories, and the extracted features were utilized to build the principal component analysis (PCA) model and determine stiction detection metrics: T^2-statistic and Q-statistic. The details of other existing methods can be found in [50] and [5].

3.2.1 Motivation and Contributions

More researchers are utilizing machine learning in their work today thanks to advancements in the field since it is more efficient, more precise, and applicable to a wider range of situations. Several publications that merged machine learning with conventional stiction detection techniques have been published, as was indicated in the preceding section. The main issue is that when the approaches were tested on industrial datasets like ISDB, the accuracy of the methods was subpar. This issue arises because a model is trained on significantly less complex simulated examples than the actual stiction cases. The model cannot perform very well on the industrial loops, so the accuracy of the model won't be very great.

Markov transition field (MTF) is a recent image encoding method introduced by Wang and Oates [51]. MTF converts a time series signal into a two-dimensional image by retaining complete time-domain information in the form of Markov transition probabilities and translating that information into the two-dimensional image. Since its inception, MTF has been employed in various fields like time series signal classification, malware classification, and equipment fault detection [52], [53], [54].

The application of MTF in the detection and diagnosis of oscillating control loops is relatively new. In the present work, a new intelligent method based on MTF and a deep neural network is proposed to detect sticky control valves in industrial control loops. The following explains the attractive features of the proposed method.

- MTF is required because it offers a means of converting time series data into an image format that a convolutional neural network (CNN) can understand. This is beneficial since CNNs are very good at processing images and may be used for a variety of tasks, including classification, anomaly detection, and prediction. Even though one-dimensional CNN can be used for time series data, it suffers from computational complexity (i.e. the model involves several layers) due to the length of the time series signals. MTF images provide a concise representation of the time series signals.

- Existing methods (discussed in the introduction section) are feature-based approaches. While each of these approaches has benefits of its own, the MTF technique circumvents their drawbacks. For instance, the feature-based approaches call for the extraction of useful features (manual feature extraction) from the time series data. These approaches are not able to fully capture all of the useful information in the time series data and also require a considerable amount of time and effort for feature extraction. MTF, on the other hand, creates a visual representation of the time series data that captures temporal relationships. CNNs can automatically extract useful features from MTF images.

- The proposed MTF-based strategy is robust to noise in the time series data. By creating a transition probability matrix that accounts for temporal relationships that exist in the time series data, MTF is able to smooth out unpredictable or noisy behavior of the time series data.

- Often, we do not get enough practical data for the detection of stiction. When there is a scarcity of sufficient practical data, simulations are performed to create data that is used for training of machine learning-based algorithms. However, these algorithms do not perform well in industrial case studies since the data used for their training do not possess characteristics that the practical data have. In such cases, MTF images of practical data can be used to generate more images by using data augmentation techniques. This is certainly an advantage over the methods relying on time series data only.

- Compared to time series data, it is easier for CNN to find patterns in MTF images. These patterns help CNN differentiate stiction induced oscillation from oscillations caused by non-stiction conditions.

The contributions of the present work are outlined below.

- Closed loop signals (OP and PV), attained by carrying out control loop simulations under various scenarios such as control valve stiction, external oscillatory disturbances, noise in PV and OP, and inappropriately tuned controllers, are converted to two-dimensional MTF images.

- CNN is trained and tested using the MTF images (obtained from the simulated data) to learn to classify the MTF images into stiction and non-stiction categories.

- Transfer learning is adopted to retrain the CNN model on data acquired from industrial control loops.

- The retrained CNN model is tested on the benchmark control loops accessible in ISDB.

The rest of the chapter is organized as follows. The research problem, control valve stiction, is reviewed in Section3.3. Section 3.4 provides a brief description of MTF and CNN. In Section 3.5, the proposed methodology is explained in detail. The performance of the proposed methodology on the industrial control loops is analyzed in Section 3.6. Finally, the chapter is concluded in Section 3.7.

3.3 Control Valve Stiction

Fig. 3.1 depicts how controller output (OP) and valve position relate to each other in the event of valve stiction. If there is no stiction, the control valve obeys the commands (OP) produced by the controller i.e. the valve position data exactly matches OP. Therefore, there exists a linear relationship between OP and the valve position data (the dashed line passing through the origin). However, this linear relationship is disturbed by the presence of valve stiction, and as a result, OP is non-linearly (parallelogram shown in Fig. 3.1) related to the valve position data.

The stiction band (sum of deadband and stickband), the slip-jump, and the moving phase together form the signature phase plot of the sticky control valve. When the valve becomes sticky and remains idle, the valve position does not change even though OP continuously varies. When a cumulative change in OP matches stiction band (S), the valve overcomes stiction, and at this time, the valve position abruptly changes giving rise to the slip-jump phase. From this point onward, the valve glides continuously either upward or downward. When OP changes the direction of

its movement, the valve may stick again.

The size of the stiction band is affected by factors such as the condition of the valve, the type of fluid flowing through the valve, and the operating conditions. When the valve becomes stuck, the size of the stiction band may change due to changes in these factors. For example, if the valve becomes more worn over time, the stiction band may widen, making it more likely for the valve to become stuck.

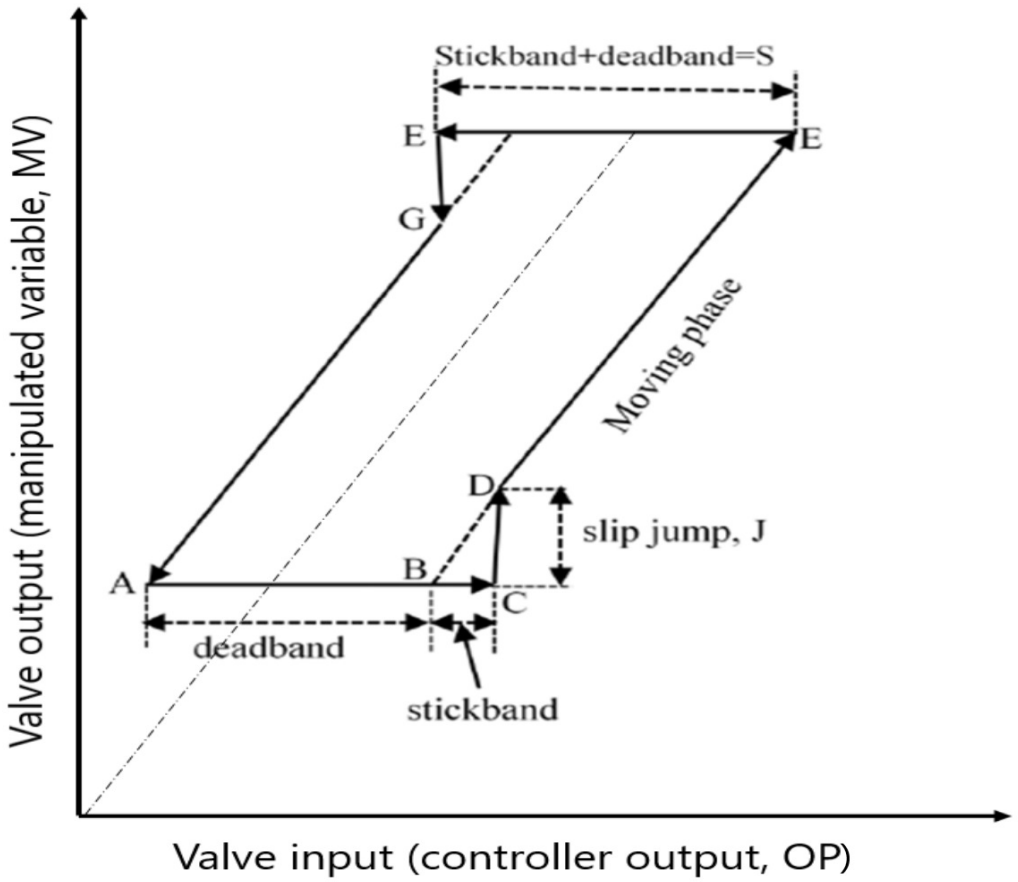


Figure 3.1: Sticky control valve behavior[1]

3.4 Preliminaries

This section provides an overview of the building blocks (MTF and CNN) of the proposed stiction detection methodology.

3.4.1 Markov Transition Field

MTF converts a time series signal into a two-dimensional image representing a field of transition probabilities for a discretized time series. This process is elucidated below.

Consider the normalized time series signal portrayed in Fig. 3.2(a). The normalized signal is divided into N non-overlapping sub-intervals, each of which is called a bin. With the help of first-order Markov chain, the Markov matrix (M) given in the following expression can be obtained.

$$M = \begin{bmatrix} m_{1,1} & m_{1,2} & \cdots & m_{1,N} \\ m_{2,1} & m_{2,2} & \cdots & m_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ m_{N,1} & m_{N,2} & \cdots & m_{N,N} \end{bmatrix} \tag{3.1}$$

where $m_{a,b} = p\{x_t \in n_a | x_{t-1} \in n_b\}$, and n_a, n_b are bin a and bin b, respectively.

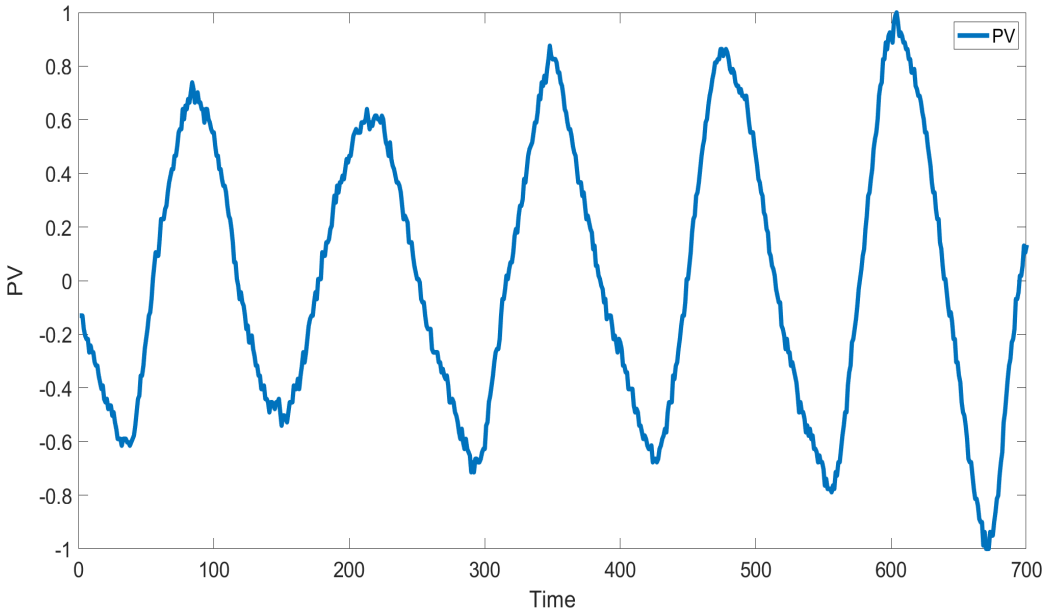
As the Markov chain has no memory, the element in the $(i,j)^{th}$ position of M is not related to the prior elements, resulting in the loss of a significant amount of information of the normalized signal. This issue can be addressed by extending the Markov matrix to the Markov transition field (shown in Eq. (3.2)) with the inclusion of a time axis.

$$\mathbf{W} = \begin{bmatrix} w_{1,1} & w_{1,2} & \cdots & w_{1,k} \\ w_{2,1} & w_{2,2} & \cdots & w_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ w_{k,1} & w_{k,2} & \cdots & w_{k,k} \end{bmatrix} \tag{3.2}$$

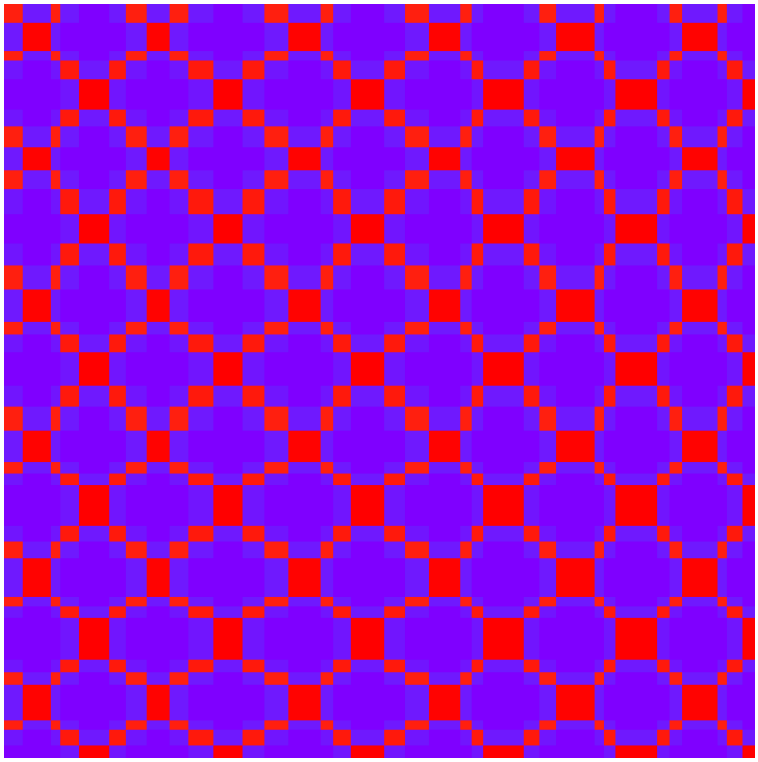
$$\mathbf{W_{a,b}} = \begin{bmatrix} m_{a,b}|x_1 \in n_a, x_1 \in n_b & \cdots & m_{a,b}|x_1 \in n_a, x_k \in n_b \\ \vdots & \ddots & \vdots \\ m_{a,b}|x_k \in n_a, x_1 \in n_b & \cdots & m_{a,b}|x_k \in n_a, x_k \in n_b \end{bmatrix} \tag{3.3}$$

where $W_{a,b}$ is the likelihood that n_a will transition to n_b.

Eq. 3.3 can be used to calculate the MTF of the signal. The multi-span transition probability of the normalized time series signal is encoded by MTF [55]. Figure 3.2(b) provides an illustration of an MTF image, which displays the probabilities of transition between various states of the PV signal. The axes in the graphic reflect time and each pixel represents a certain state transition probability. The MTF image is produced by mapping each member of the **W** matrix, which represents the likelihood of changing from one state to another, to a pixel value using a specific colormap. A particular colormap was utilized in Fig. 3.2(b), where high probability values are mapped to red, and low probability values to purple.



(a) Normalized time series signal



(b) Two-dimensional image obtained using the MTF technique

Figure 3.2: MTF conversion.

3.4.2 Convolutional Neural Networks

CNN is also known as shift invariant or space invariant neural network that is commonly employed for classification and computer vision tasks [56]. Compared with traditional neural networks, CNNs offer a more scalable strategy for image categorization. Fig. 3.3 depicts CNN architecture for image categorization. The CNN model has four types of layers: convolutional layer, pooling layer, dropout layer, and fully connected layer. The convolutional layer is the most important layer performing the majority of computations. This layer utilizes a filter (also known as kernel or feature detector) that slides along the receptive fields of the input image to yield a feature map (G). It possesses the ability to learn features at different scales using the process of spatial hierarchies of features, making them well-suited for this type of problem. Additionally, the CNN model uses weight sharing across the entire image that reduces the number of parameters required to be learned, rendering the training process more efficient and less prone to overfitting.

The process of convolution can be represented as:

$$\mathbf{G}[a,b] = (\mathbf{f} * \mathbf{h})[a,b] = \sum_{p}\sum_{q} \mathbf{h}[p,q]\mathbf{f}[a-p, b-q] \tag{3.4}$$

where a and b are the a^{th} row and the b^{th} column of matrix representing the image, $\mathbf{f}$ is the given image and $\mathbf{h}$ is the kernel.

The pooling layer's objective is to minimize the spatial size of feature maps learned by the convolutional layer. Max pooling and average pooling are two techniques that can be used to achieve this. In contrast to average pooling, which computes the average value within an area of the feature map, max pooling includes choosing the largest value there. The max pooling strategy was employed in this particular project. Let F be the input feature map and M be the output pooled feature map. Then the max pooling operation can be defined as:

$$M_{a,b} = \max_{m,n} F_{a+m,b+n} \tag{3.5}$$

The dropout layer gets its input from the previous layer (the pooling layer). Dropout is often employed in CNNs to decrease or avoid overfitting problems. A random number is generated in the range $(0, 1)$ for each neuron in the dropout layer. If the random number generated for a specific neuron is smaller than the predefined dropout rate, the output of the neuron becomes zero. The formula for the dropout layer can be represented as follows:

Let P be the predefined dropout rate, and let r be a random number generated for each neuron in the dropout layer. Then, for each neuron, the output is calculated as:

$$Y_{a,b} = \begin{cases} 0, & \text{if } r < P \\ \frac{X_{a,b}}{1-P}, & \text{otherwise} \end{cases} \tag{3.6}$$

where X is the input value from the previous layer.

Features obtained from the dropout layer are sent to the flattened layer producing a one-dimensional vector, which passes through the subsequent fully connected layers and the output layer. The output layer possesses the same number of neurons as the number of groups or classes in the given training data and produces an estimated class label for the given image. There can be multiple layers of each kind in the CNN configuration. The convolutional, pooling, and dropout layers are collectively called feature detectors. Whereas the flatten, dense, and output layers are jointly called decision blocks. Rectified linear unit (ReLU) is used as an activation function in the convolutional and dense layers. The activation function employed in the output layer is either sigmoid function or softmax function.

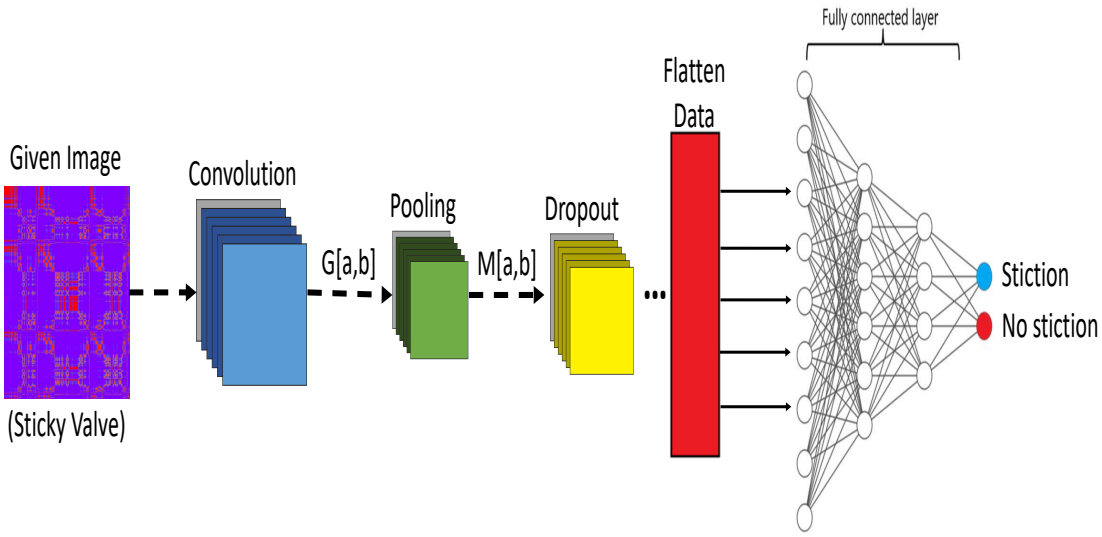


Figure 3.3: CNN architecture for image categorization

3.5 The Proposed Method

In this section, strategies employed in developing the proposed CNN-based method and metrics used to quantify the stiction detection capability of the method are discussed.

Fig. 3.4 shows the steps involved in the establishment and application of the proposed method. CNN usually needs an ample amount of labeled data (or images) to acquire the ability to distinguish stiction induced oscillations from oscillations caused by a non-stiction condition.

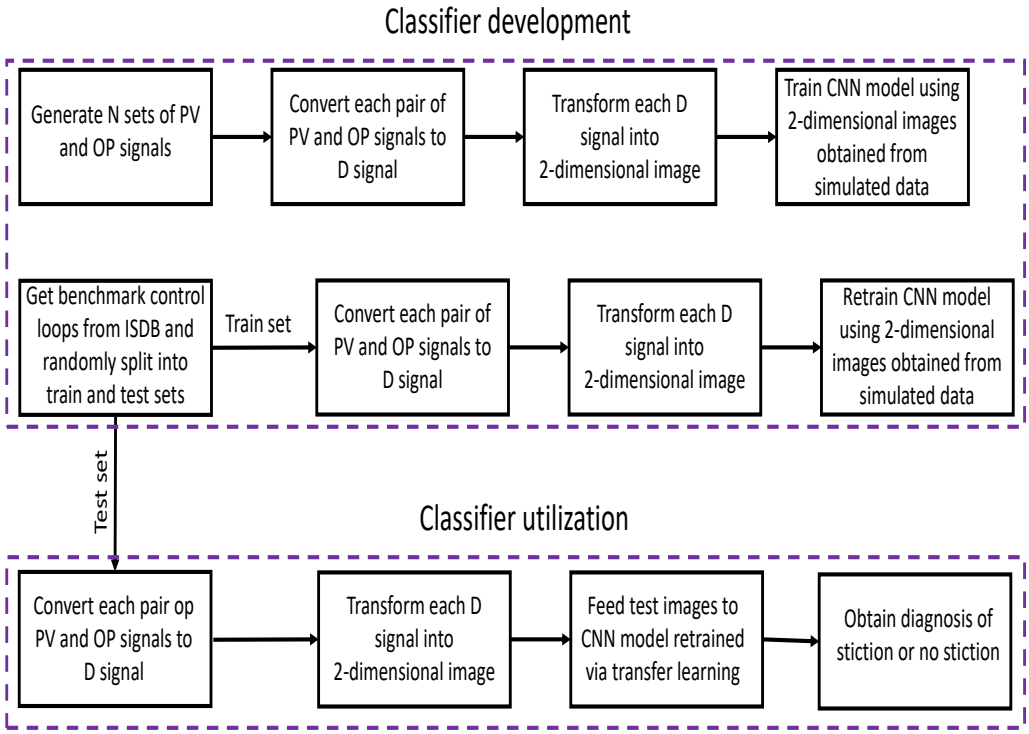


Figure 3.4: Development and utilization steps of CNN-based mehtod

3.5.1 Datasets generation

Industrial data available in the ISDB are not adequate to determine optimum values for the parameters of the CNN model. To overcome this stumbling block, enough data (closed loop signals: PV and OP) was generated by running control loop simulations. The control loop used in the simulations is displayed in Fig. 4.4. In the presence of valve stiction, oscillations in the feedback loop controlling a self-regulating process are often very different from oscillations in the control loop regulating an integrating process. Therefore, two types of control loops (concentration and level loops) were considered in the simulations. In the presence of a non-stiction condition (valve is non-sticky), these two control loops produce similar oscillations. Table 4.1 provides the process models and the controller transfer functions used in the two control loops. The data-driven stiction model proposed by Choudhury [1] was employed to produce stiction data. Values used for the parameters (S and J) of the stiction model, for both

the concentration and level control loops, are given in Table 3.2. Two cases: excessively tuned controller (Table 3.3) and external oscillatory disturbances (Table 4.4) represent non-stiction conditions that were simulated to yield non-stiction data. The simulated data (including stiction and non-stiction data) was randomly divided into training and validation sets.

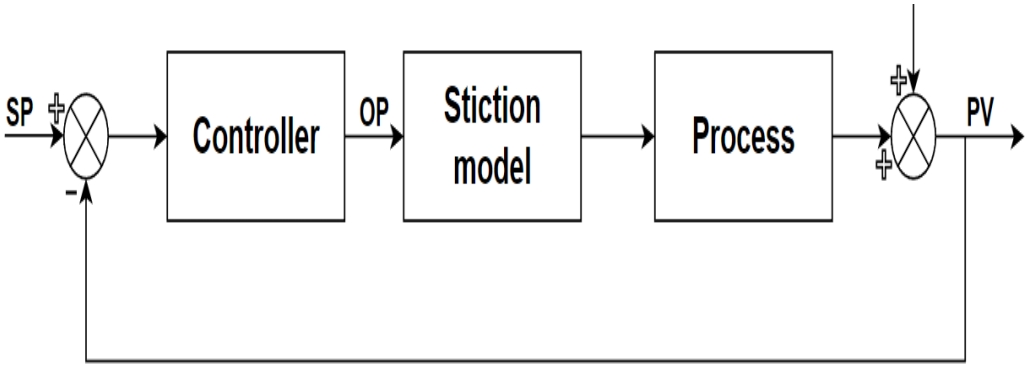


Figure 3.5: Control loop

Table 3.1: Process model and controller transfer function for closed loop simulations

Loop type	Process	Controller
Concentration	$G_p = \frac{3e^{-10s}}{10s+1}$	$G_c = 0.2(\frac{10s+1}{10s})$
Level	$G_p = \frac{1}{s}$	$G_c = 0.4(\frac{2s+1}{2s})$

Table 3.2: Parameter variation for generating stiction data

Parameter	Description	Parameter range
S	Stiction band	[0.5:0.25:10]
J	Slip jump	[0.1:0.25:5]
V	Variance of white noise	0.01

Table 3.3: Parameter variation for generating non-stiction data (tightly-tuned controllers).

Parameter	Description	Parameter range
K_c	Controller gain	[0.1:0.01:0.3]
I	Final controller integral value	[0.01:0.01:0.27]
V	Variance of noise	0.01

Table 3.4: Parameter variation for generating non-stiction data (external oscillatory disturbance).

Parameter	Description	Parameter range
A	Amplitude	[1, 1.5, 2, 2.5]
F	Frequency of oscillation value	[0.01:0.01:0.27]
V	Variance of noise	0.01

3.5.2 Markov Transition Field

The pre-processing method reported in [57] was adopted in the present work to convert each pair of PV and OP signals to the D signal which is represented in Eq. 3.7.

$$D_i = \sqrt{(PV_i - \overline{PV})^2 + (OP_i - \overline{OP})^2} \tag{3.7}$$

where $i = 1, 2, ..., Q$, Q is the length of PV or OP signal.

In the above equation, $\overline{PV}$ and $\overline{OP}$ indicate the mean of PV and OP signals, respectively. Each pair of PV and OP signals in the training and validation sets is first converted into D signal, which is then transformed into two-dimensional image using the MTF technique described in Section 3.4.1. It is important to note that the patterns and characteristics of the MTF images for the stiction and no stiction scenarios are very different from one another. The MTF images for the cases with and without stiction are shown in fig. 3.6, which amply demonstrates the variations in the patterns between the stiction, poor tuning, and disturbance examples.

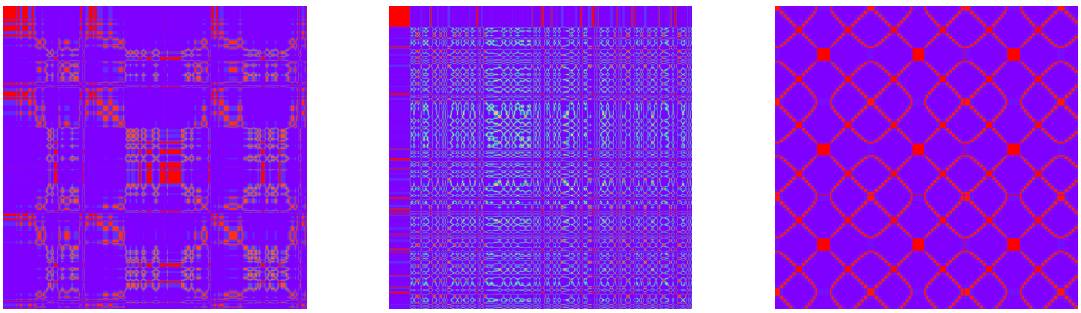


Figure 3.6: MTF image for stiction (left image), poor tuning (center image), and disturbances (right image).

3.5.3 Counvolutional Neural Network

If an image obtained from PV and OP of a control loop oscillating due to a non-stiction condition, it is labeled 0. Otherwise, it is labeled 1. For an image representing a stiction data set, the target vector for the CNN model is [1 0]. The target vector becomes [0 1] for the image attained from a non-stiction data set. The MTF images from the training set are fed to the CNN model (Fig. 3.7).

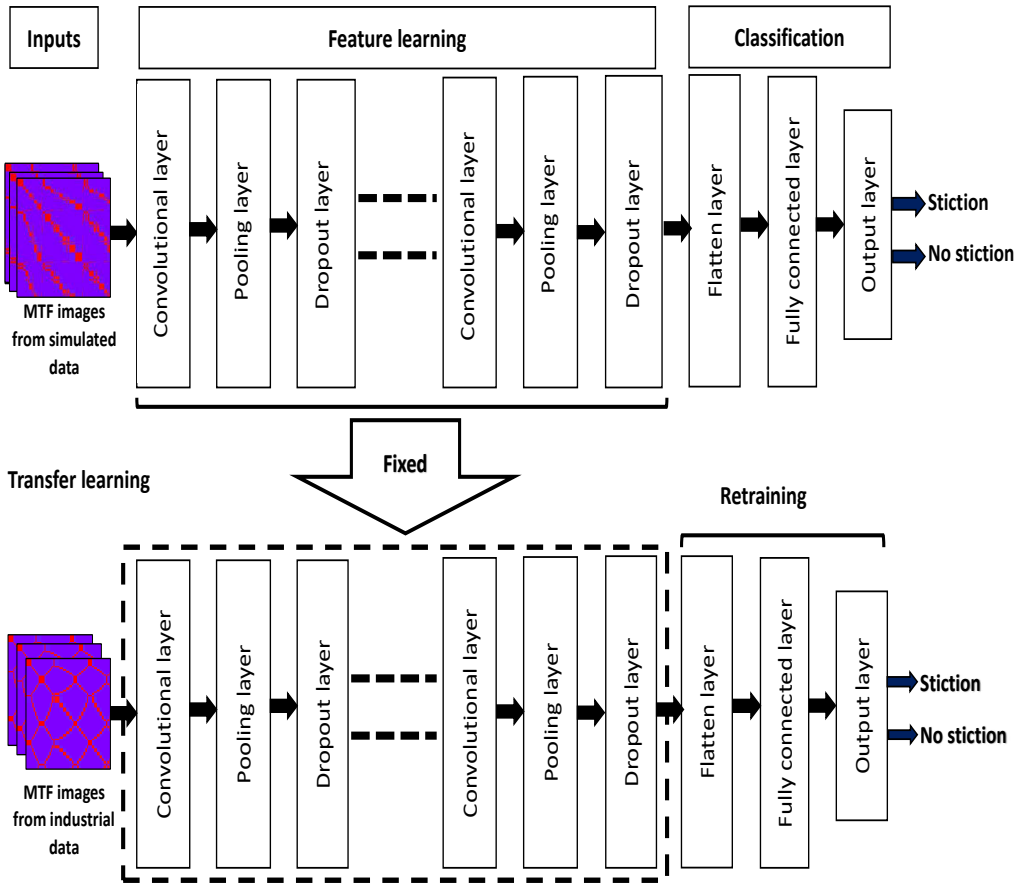


Figure 3.7: CNN model development

The feature learning block, consisting of a set of convolutional, pooling, and dropout layers, extracts features from the MTF images. These features are passed to the flattening layer, which places the features one below another thus preparing input suitable to be processed by the fully connected layer that learns to classify the features into stiction and no stiction classes. The output layer calculates the probability of an input image pertaining to each of the two classes (stiction and no stiction). The loss function (Cross entropy or maximum likelihood estimation) provided in the following

equation is minimized during training with the help of a gradient descent optimization algorithm. The validation set is used to fine tune the free parameters (weights and biases) of the CNN model.

$$L = -\frac{1}{N}\sum_{p=1}^{M}[y_p \log \tilde{y_p} + (1 - y_p) \log (1 - \tilde{y_p})] \tag{3.8}$$

where M is the number of classes, y_p is the target vector and $\tilde{y_p}$ is the CNN output.

3.5.4 Transfer learning

As we shall notice in the next section, the CNN model trained using the MTF images acquired from the simulated data fails to deliver satisfactory performance when it is utilized to detect sticky control valves in industrial control loops. In this regard, the transfer learning approach is used to achieve improved stiction detection performance. In this approach (Fig. 3.7), some of the industrial control loop data in the ISDB [34] is used to retrain the CNN model that has already been trained. The details of the industrial control loops (benchmark control loops) available in the ISDB are provided in Table A1 given in A. The entire network is not retrained but only the fully connected layer is retrained. The weights and biases (these are learned during pre-training) of the layers in the feature learning block are fixed and are not relearned. Optimum values obtained, during pre-training, for the parameters of the fully connected layer are considered as initial values. MTF images for the industrial data are first obtained and these images are fed to the pre-trained CNN model. Features obtained from the feature extraction block are treated as inputs to the subsequent layers. The loss function given in the above equation is used to find new optimum values for the parameters of the fully connected layer.

3.5.5 Model performance calculation

The stiction detection capability of the proposed method is assessed using the following metrics.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{3.9}$$

$$Precision = \frac{TP}{TP + FP} \tag{3.10}$$

$$Recall = \frac{TP}{TP + FN} \tag{3.11}$$

$$Specificity = \frac{TN}{TN + FP} \tag{3.12}$$

$$F_1 score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{3.13}$$

Here, TP is true positive (control valve is sticky and the proposed method detects stiction), TN is true negative (the control valve is not sticky and the proposed method does not detect stiction), FP is false positive (the control valve is not sticky but the proposed method detects stiction) and FN is false negative (the control valve is sticky but the proposed method does not detect stiction).

The CNN based method developed in this section is formulated so as to satisfy the requirement of being able to diagnose an oscillating control loop without needing additional information (other than PV and OP signals) like process type, valve position measurements, etc. This is accomplished so that it may be applicable across a wide variety of processes and loops.

3.6 Results and Discussions

In this section, the performance of the proposed method is analyzed via its application to industrial control loops.

By carrying out control loops simulations under various scenarios explained in the previous section, simulated database consisting of 2830 data sets (comprising stiction and non-stiction data) was created. The data sets were randomly split into training and validation sets. PV and OP signals in each data set of the training and validation sets were converted to the corresponding D signal. The resulting D signals were then transformed into two-dimensional images using the MTF technique. The training set containing 2320 images was used to learn the parameters of the CNN model while the validation set possessing 510 images was utilized to get better estimations for the parameters. Table 3.5 provides values chosen for the hyperparameters of the CNN model and selected training algorithm.

The learning of the CNN model is shown in Fig. 3.8. The industrial data sets available in the ISDB were randomly divided into a new training set (contains 30 data sets) and a test set (contains 37 data sets). The CNN model trained on the MTF images obtained from the simulated data (which is termed as the pre-trained model) was applied to the test set and the obtained results are provided in Table 3.7. In Table 3.7, LN means "Loop Name", AM indicates "Actual Malfunction", OCM signifies "Output of the CNN Model" and IDC denotes "Is Diagnosis Correct".

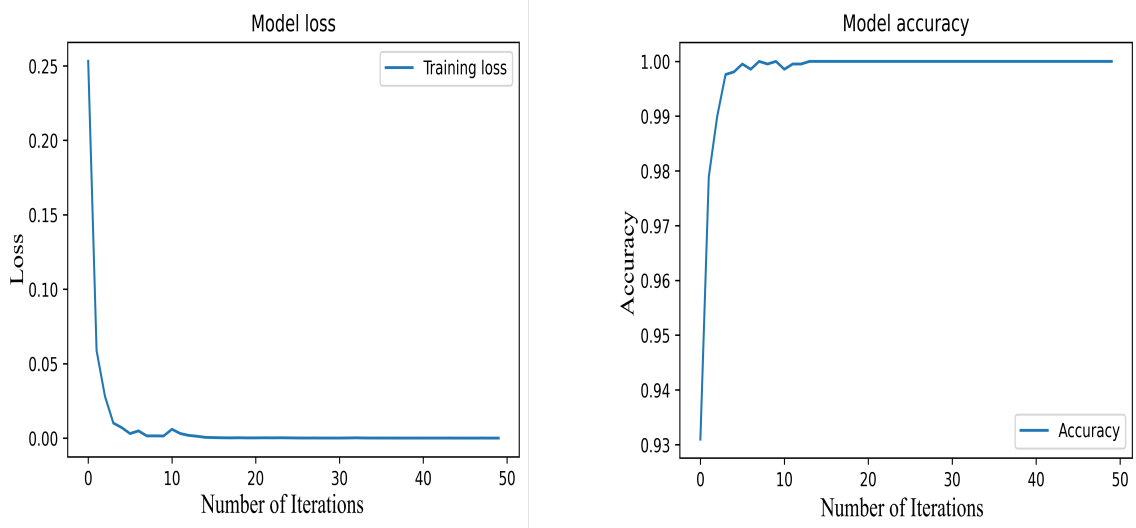


Figure 3.8: Performance of CNN model during pre-training

Table 3.5: Hyperparameters and training details of CNN model

Description	Values/Functions/Method
No. of Conv. layers	2
No. of pooling layers	2
No. of dropout layers	4
No. of filters in each conv. layer	32
Size of each filter (conv. layers)	3×3
Size of each filter (pooling layers)	2×2
Type of pooling operation used	Max pooling
Dropout rate	0.2
No. of neurons in fully connected layer	512
Optimization method	Adam
Image size	[64*64*3]
Batch-size	64
Initialization	Normal

Table 4.8 provides values attained for the performance metrics. The pre-trained CNN model provided correct diagnosis for 23 control loops and incorrect verdict for the remaining control loops. Therefore, the transfer learning strategy (explained in the preceding section) was applied to the pre-trained CNN model to further fine tune the parameters of the fully connected layer. The performance of the CNN model retrained using the new training set is depicted in Fig. 3.9. It can be noticed from Tables 3.7 and 3.8 that, after the application of the transfer learning strategy, the

performance of the CNN model significantly improved from 62.16% to 91.89%.

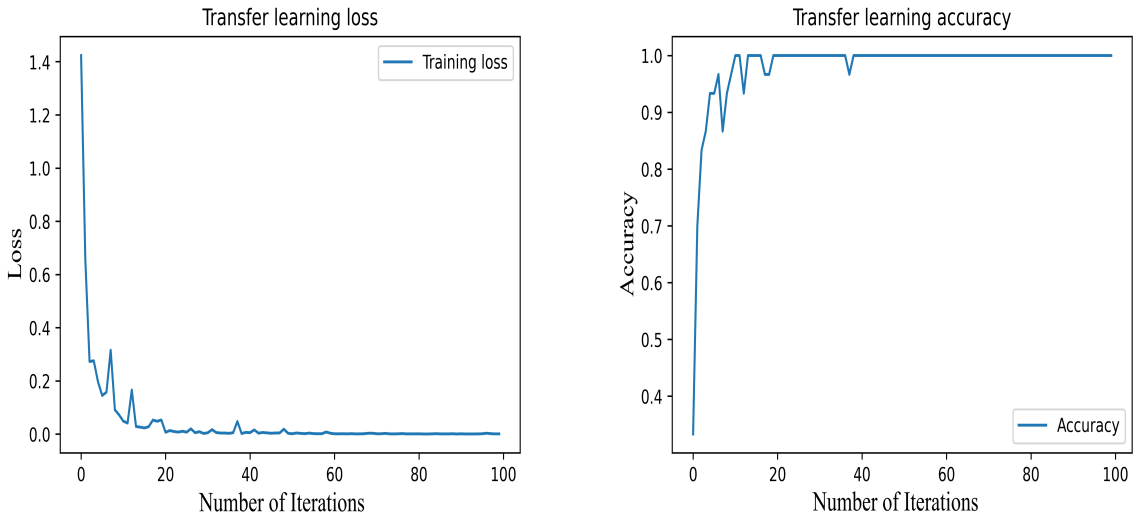


Figure 3.9: Learning curve of CNN model retrained via transfer learning approach

3.7 Conclusions

The proposed method presented a promising approach for the detection of sticky control valves in industrial control loops. The combination of CNN and MTF provides a powerful tool for automated fault detection and diagnosis, potentially leading to significant cost savings for industrial processes. The results of the study demonstrated the effectiveness of the proposed method in detecting sticky control valves with a high success rate of 91%. The use of transfer learning, which enables pre-trained neural network models to be retrained with industrial data, is a critical innovation in the

Table 3.6: Performance metrics for proposed method

Performance metric	Pre-trained CNN	Retrained CNN
True positive	18	26
True negative	5	8
False positive	3	0
False negative	11	3
Precision (%)	85.71	100
Recall (%)	62.07	89.66
Specificity (%)	62.50	100
F1 score	72.0	94.55
Accuracy	62.16	91.89

Table 3.7: Application of pre-trained CNN model to industrial data

LN	AP	OCM		IDC?
CHEM 1	Stiction	0.994	0.006	Yes
CHEM 2	Stiction	0.996	0.004	Yes
CHEM 3	No stiction	0.866	0.134	No
CHEM 5	Stiction	0.96	0.04	Yes
CHEM 6	Stiction	0.975	0.025	Yes
CHEM 9	Stiction	0.99	0.01	Yes
CHEM 10	Stiction	1	0	Yes
CHEM 11	Stiction	0.1	0.9	No
CHEM 12	Stiction	0.07	0.93	No
CHEM 18	Stiction	0.994	0.006	Yes
CHEM 19	Stiction	0.85	0.15	Yes
CHEM 20	Stiction	0.791	0.109	Yes
CHEM 21	No stiction	0.015	0.985	Yes
CHEM 22	Stiction	0.91	0.09	Yes
CHEM 23	Stiction	0.005	0.995	No
CHEM 24	Stiction	0.938	0.062	Yes
CHEM 26	Stiction	0.998	0.002	Yes
CHEM 28	Stiction	0.027	0.973	No
CHEM 29	Stiction	0.98	0.02	Yes
CHEM 30	Stiction	0	1	No
CHEM 32	Stiction	0.14	0.86	No
CHEM 33	No stiction	0.167	0.833	Yes
CHEM 34	No stiction	0.12	0.98	Yes
CHEM 35	Stiction	0.98	0.02	Yes
CHEM 39	No stiction	0.9	0.1	No
PAP 1	Stiction	0.9	0.1	Yes
PAP 2	Stiction	0.966	0.034	Yes
PAP 4	No stiction	0.124	0.876	Yes
PAP 5	Stiction	0.05	0.95	No
PAP 7	Stiction	0.004	0.996	No
PAP 12	Stiction	0.23	0.87	No
PAP 13	Stiction	0.994	0.006	Yes
POW 1	Stiction	0.002	0.998	No
POW 2	Stiction	0.103	0.897	No
MIN 1	Stiction	0.98	0.02	Yes
MET 1	No stiction	0.016	0.984	Yes
MET 2	No stiction	0.752	0.248	No

Table 3.8: Application of retrained CNN model to industrial data

LN	AP	OCM		IDC?
CHEM 1	Stiction	0.999	0.001	Yes
CHEM 2	Stiction	0.999	0.001	Yes
CHEM 3	No stiction	0.015	0.985	Yes
CHEM 5	Stiction	0.96	0.04	Yes
CHEM 6	Stiction	0.977	0.023	Yes
CHEM 9	Stiction	0.991	0.009	Yes
CHEM 10	Stiction	1	0	Yes
CHEM 11	Stiction	1	0	Yes
CHEM 12	Stiction	0.988	0.012	Yes
CHEM 18	Stiction	0.994	0.006	Yes
CHEM 19	Stiction	0.985	0.015	Yes
CHEM 20	Stiction	0.791	0.109	Yes
CHEM 21	No stiction	0.015	0.985	Yes
CHEM 22	Stiction	0.99	0.01	Yes
CHEM 23	Stiction	0.995	0.005	Yes
CHEM 24	Stiction	0.988	0.012	Yes
CHEM 26	Stiction	0.998	0.002	Yes
CHEM 28	Stiction	0.976	0.024	Yes
CHEM 29	Stiction	0.978	0.022	Yes
CHEM 30	Stiction	1	0	Yes
CHEM 32	Stiction	0.63	0.37	Yes
CHEM 33	No stiction	0.167	0.833	Yes
CHEM 34	No stiction	0.12	0.98	Yes
CHEM 35	Stiction	0.989	0.011	Yes
CHEM 39	No stiction	0.089	0.911	Yes
PAP 1	Stiction	0.9	0.1	Yes
PAP 2	Stiction	0.966	0.034	Yes
PAP 4	No stiction	0.006	0.994	Yes
PAP 5	Stiction	0.996	0.004	Yes
PAP 7	Stiction	0.004	0.996	No
PAP 12	Stiction	0.347	0.653	No
PAP 13	Stiction	0.994	0.006	Yes
POW 1	Stiction	0.002	0.998	No
POW 2	Stiction	0.992	0.008	Yes
MIN 1	Stiction	0.98	0.02	Yes
MET 1	No stiction	0.023	0.977	Yes
MET 2	No stiction	0.002	0.998	Yes

proposed method. By leveraging pre-existing neural network architectures and training them on industrial data, the proposed method could overcome the challenges of limited data availability that have traditionally hindered fault detection methodologies. The ability to retrain neural network models using industrial data provides a valuable tool for industrial practitioners seeking to implement automated fault detection and diagnosis in their processes. However, there is still room for improvement. The dataset used in this study was relatively small, and more extensive datasets could be used to train and validate the proposed method. Additionally, the proposed method only detects sticky control valves, and further work is needed to diagnose other types of valve problems. In our future work, we intend to go deeper into this subject and consider several concepts for calculating the stiction band from PV and OP. We'll take into account strategies like combining statistical techniques with machine learning algorithms. We anticipate that this will enhance our comprehension of the stiction band and our capacity to identify issues with control valves in industrial settings.

Chapter 4

Detection of poor controller tuning with Gramian Angular Field (GAF) and StackAutoencoder (SAE)[1]

[1]Submited to Computers and Chemical Engineering as A. Memarian, SK. Damarla, A. Memarian B. Huang. "Detection of poor controller tuning with Gramian Angular Field (GAF) and StackAutoencoder (SAE)"

4.1 Abstract

Efficient control loop performance is pivotal in process industries to ensure optimal production, maintain product quality, and adhere to regulatory standards. Poorly tuned controllers can disrupt these objectives, necessitating accurate detection methods. This chapter introduces a novel approach for detecting poor controller tuning through advanced techniques: the Gramian Angular Field (GAF) and Stack Auto-Encoder (SAE). Unlike manual methods, this automated system promptly identifies poorly tuned controllers, offering real-time monitoring and timely alerts to operators. The proposed methodology is substantiated through two case studies: the ISDB dataset and the pulp and paper dataset. The outcomes illustrate that the proposed approach correctly determines the appropriate outcome for the majority of the analyzed control loops across diverse industries.

4.2 Introduction

In process industries, the proper functioning of control loops is vital for maintaining key process variables near or at their desired values, ensuring optimal production, adhering to product quality specifications, and complying with environmental regulations [6]. Efficiently tuned controllers play a crucial role in achieving these objectives by effectively regulating process variables and responding to disturbances. However, the presence of poorly tuned controllers can lead to significant challenges and disruptions in the process [58].

Detecting poor controller tuning is of utmost importance as it directly impacts process performance and stability. When controllers are inadequately tuned, the control system response to changes in setpoints or disturbances becomes sluggish or even oscillatory [59]. These oscillations can cause adverse effects on production rates, product quality, energy consumption, and may even compromise the safety of the entire process. Thus, identifying and addressing poor controller tuning promptly is essential for maintaining smooth and efficient process operations [6].

Various methods for tuning controllers exist, such as Ziegler-Nichols, Cohen-Coon, and gain scheduling, among others. While these methods have proven effective in achieving proper tuning, the sheer number of control loops present in modern process industries poses a significant challenge. With hundreds or even thousands of control loops in operation, manual tuning for each controller becomes impractical and time-consuming.

Moreover, even with sophisticated tuning methods, it is challenging to ensure that each controller is optimally tuned at all times. Factors such as changes in process dynamics, interactions between control loops, and uncertainties in process parameters can lead to performance degradation over time, requiring periodic re-tuning. Manual re-tuning on a regular basis is labor-intensive and may not be cost-effective.

In light of these challenges, the development of automated and efficient methods for detecting poor controller tuning has become a critical area of research. Previous works have explored various techniques, including higher-order statistical (HOS) techniques, to diagnose the causes of poor control-loop performance [60]. These HOS techniques analyze closed-loop data to detect non-Gaussianity and nonlinearity in regulated systems, offering a potential avenue for detecting poor tuning in controllers [60]. Hägglund [59] introduced the "Idle index," a metric that quantifies control loop sluggishness by analyzing the correlation between control and measurement signal increments. This approach provides an effective means to identify poorly performing control loops due to improper tuning. Additionally, Naghoosi et al. [58] developed a method based on the autocorrelation function (ACF) for detecting poor controller tuning. However, existing methods may have limitations and might not address the specific issue of improper controller tuning.

One limitation of the existing methods is that some of them require extensive data preprocessing and feature engineering, making them computationally expensive and time-consuming. In real-time industrial settings where quick response and decision-making are critical, these methods might not be practical for detecting poor controller tuning promptly.

Additionally, certain methods may have limited accuracy in distinguishing poor tuning from other causes of oscillations, such as disturbances or equipment faults like stiction in control valves. Misdiagnosing the source of oscillations can lead to inappropriate control loop adjustments, exacerbating the problem and negatively impacting process performance.

4.2.1 Motivation and Contributions

To overcome these limitations and contribute to the field of poor controller tuning detection, a novel methodology is introduced in this chapter. This methodology leverages the capabilities of Gramian Angular Field (GAF) and Stack Auto-Encoder. Here, GAF plays a pivotal role by transforming time series data into images. This

transformation not only simplifies the visualization of data but also ensures that temporal relationships are preserved, giving a robust visual representation of the control loop behavior. The GAF images serve as input to the Stack Autoencoder, which is a powerful unsupervised learning technique capable of automatically learning and extracting intricate features from the encoded images.

The proposed methodology offers several distinguishing contributions:

- The introduction of GAF to transform time series data for poor controller tuning detection marks a unique step in the field. By employing this technique, temporal relationships within the data are effectively preserved, yielding a distinctive visual representation of control loop behavior. Such a representation is particularly tailored for subsequent automated analysis, showcasing the adaptability and innovation of employing GAF.

- The integration of GAF with the Stack Auto-Encoder is a notable feature of the proposed methodology. This combination facilitates the automated extraction of intricate features from GAF images, entirely bypassing the need for manual feature engineering. This streamlined feature extraction process stands in contrast to traditional methodologies, which typically hinge on labor-intensive manual pattern recognition, emphasizing the advanced nature of this approach.

- The efficiency of the proposed methodology is evident in its capability for real-time monitoring. By converting control loop data into GAF images and leveraging the Stack Auto-Encoder for feature extraction, the system can swiftly pinpoint poorly tuned controllers. More than just detection, this methodology ensures that timely alerts are dispatched to operators and engineers the moment poor tuning is identified, underscoring the proactive nature of this system.

4.3 Preliminaries

This section provides an overview of the building blocks (GAF and Stack Auto-Encoder) of the proposed poor controller tuning detection methodology.

4.3.1 Gramian Angular Field (GAF)

Gramian Angular Field (GAF) is a sophisticated method used to analyze time series data and extract essential temporal patterns. GAF transforms the sequential data into a two-dimensional visual representation that retains the inherent temporal

relationships between data points. This unique feature allows analysts to observe complex patterns and trends that may not be evident in the original time series [61].

The fundamental concept behind GAF is to encode temporal relationships as angles in a circular domain. To create the GAF image, the time series is first normalized to a specific range, either [-1,1] or [0, 1]. Suppose a time series $X = \{x_1, x_2, ..., x_i, ..., x_N\}$, contains N observations. The normalization process can be expressed as follows:

$$\tilde{x}^i_{-1} = \frac{(x_i - max(X)) + (x_i - min(X))}{max(X) - min(X)} \tag{4.1}$$

$$\tilde{x}^i_0 = \frac{(x_i - min(X))}{max(X) - min(X)} \tag{4.2}$$

Afterward, every element in the rescaled time series $\tilde{X}$ undergoes a conversion to polar coordinates, where its value is expressed as an angular cosine, and the corresponding time stamp serves as the radius, as depicted in Equation 4.3.

$$D_{i,t} = \begin{cases} \phi_i = arccos(\tilde{x}_i) & -1 \leq \tilde{x}_i \leq 1, \tilde{x}_i \in \tilde{X}_i \\ r_i = \frac{t_i}{N} & t_i \in N \end{cases} \tag{4.3}$$

where t_i represents the time stamp, and N is a constant factor utilized to standardize the range of the polar coordinate system.

Once the rescaled time series X is transformed, we proceed to construct the GAF. GAF utilizes trigonometric functions to capture temporal correlations between sampling points, represented as angles. There are two variations: the Gramian Angular Summation Field (GASF) and the Gramian Angular Difference Field (GADF). Figure 4.1, in which GASF and GADF are presented, offers valuable insights into the temporal relationships within the rescaled time series X. GASF and GADF are defined as Equation 4.4 and Equation 4.5, respectively:

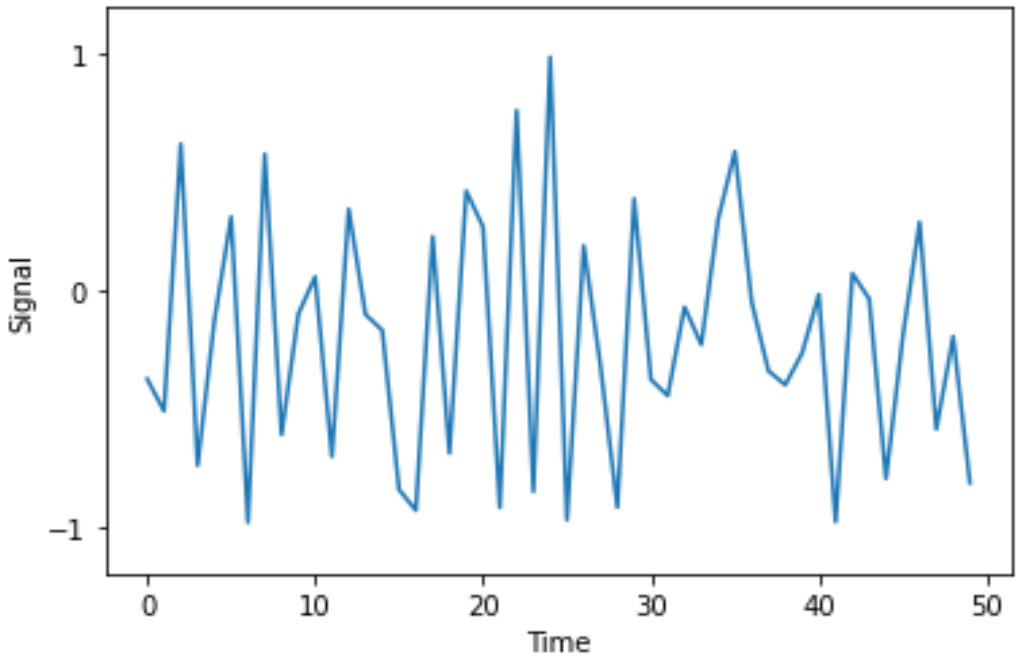
$$GASF = [cos(\phi_i + \phi_j)] \tag{4.4}$$

$$GADF = [sin(\phi_i - \phi_j)] \tag{4.5}$$

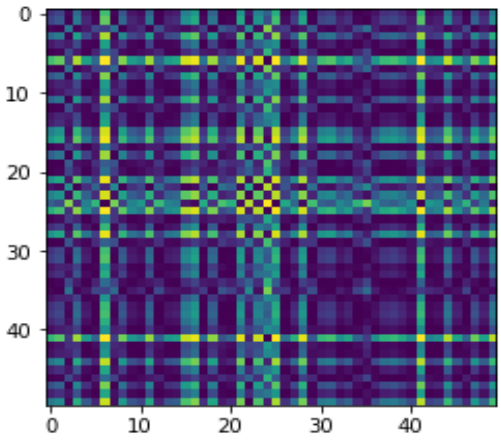
The conversion maintains the temporal relationship between values while yielding temporal correlations through the overlapping alignment within the time interval [62].

The application of the GASF and GADF is as follows:

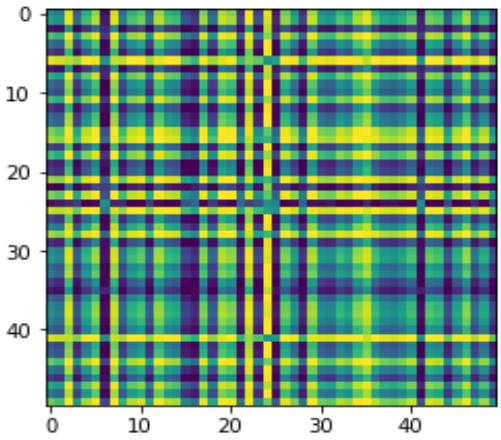
- GASF



(a) Normalized time series signal



(b) GASF



(c) GADF

Figure 4.1: Timeseries converted to images by using GASF and GADF.

  - Combines angle cosine values to depict the overall time series pattern.
  - Detects periodicity, symmetries, and cyclic patterns.
  - Useful for analyzing repetitive or oscillatory data.

- GADF

- Measures the absolute sine difference for insights into the time series.
- Detects local changes, transitions, and irregularities.
- Emphasizes the magnitude of shifts, making it suitable for abrupt or non-periodic data.

4.3.2 Stack Auto-Encoder (SAE)

A SAE is a type of neural network used for unsupervised learning and dimensionality reduction tasks. It is composed of multiple layers of Auto-Encoders (AE), where each AE layer attempts to reconstruct its input data at the output layer. Figure 4.2 shows the structure of SAE. AE consists of an encoder and a decoder: the encoder compresses the input data into a lower-dimensional representation, known as the bottleneck or encoding layer, while the decoder aims to reconstruct the original data from this compressed representation. The objective of the AE is to minimize the reconstruction error, promoting the learning of meaningful features and patterns within the data. The SAE architecture takes this concept further by stacking multiple AE layers on top of each other. The data flows through each layer, with each subsequent layer learning more abstract and higher-level representations of the original input data. This hierarchical learning approach allows the network to capture complex and intricate patterns, leveraging the features learned in lower layers to build more meaningful representations in higher layers [63].

To train a SAE, layer-wise pre-training is commonly employed. In this process, each autoencoder layer undergoes independent and sequential pretraining. First, the bottom-most layer is trained using the input data, resulting in a compressed representation of the data. Then, the compressed representation learned in the previous layer is used as input to train the next layer. This process is repeated until all layers are pretrained. Subsequently, the entire network is fine-tuned using backpropagation, enabling the weights of all layers to be collectively fine-tuned and further refining the learned representations [64].

SAEs have demonstrated remarkable capabilities in various applications. By learning meaningful representations of high-dimensional data, they effectively reduce the dimensionality and extract important features. This makes them highly valuable for feature learning, denoising, and anomaly detection tasks. SAEs have been successfully utilized in domains such as image recognition, natural language processing, recommendation systems, and more.

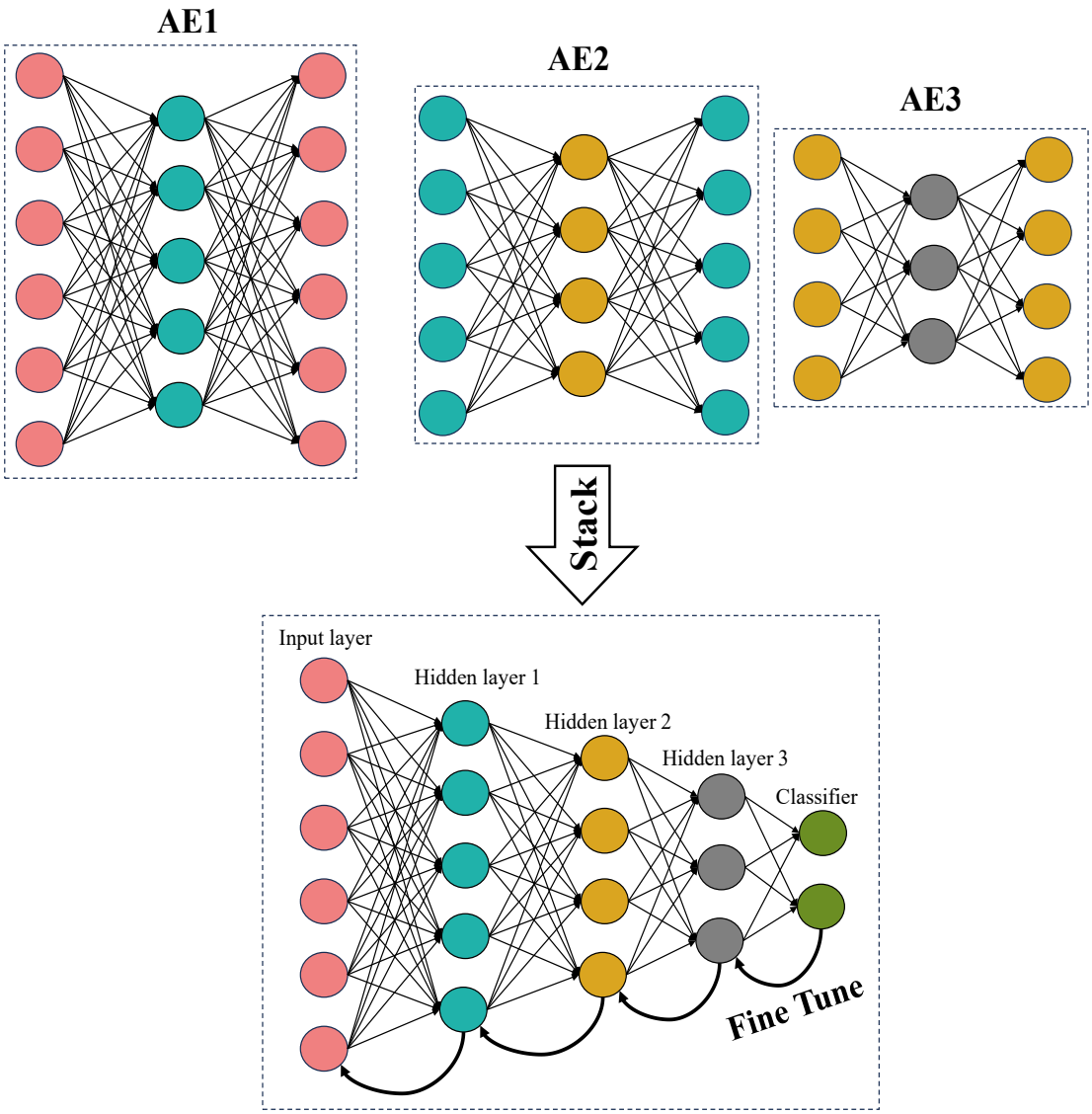


Figure 4.2: Stack Auto-Encoder structure with Greedy Layer-wise Pre-training.

4.4 The proposed method

This section provides an overview of the methodologies adopted in the proposed SAE-based approach and the metrics chosen to evaluate its ability to discern poor controller tuning. As depicted in Figure 4.3, the steps for the development and application of the method are laid out. It is essential to note that oscillations observed in control loops can stem from three primary causes: stiction, disturbance, and poor controller tuning. The primary objective of the SAE is to differentiate between control loops with poor tuning and those with no poor tuning, the latter possibly arising from causes like stiction or disturbance. For this differentiation to be effective, the SAE

demands a considerable volume of labeled data or images.

4.4.1 Dataset Generation

The International Stiction Database (ISDB) does not have sufficient industrial data to determine the optimal parameters for the SAE model. To address this limitation, data is generated through control loop simulations (depicted in Figure 4.4), specifically focusing on closed-loop signals (Process Variable (PV) and Controller Output (OP)). The decision to use simple models, specifically first-order processes, in the simulations, is based on their prevalence in basic process control systems. While these models may be perceived as simplistic, they serve as representatives for numerous real-world industrial processes. An assumption underpinning this approach is the exclusive focus on first-order processes. However, without loss of generality, it is understood that higher-order processes could be trained similarly, broadening the applicability of insights derived. Two types of control loops are incorporated in the simulations: concentration and level loops. Table 4.1 outlines the process models and controller transfer functions used. Parameter values (Kc, I, and V) for the excessively tuned controller in both the concentration and level control loops are detailed in Table 4.2. Data indicative of non-poor controller tuning is sourced from simulations that represent two conditions: one using the stiction model (Table 4.3) and another denoting external oscillatory disturbances (Table 4.4). Once generated, the data, representing both poor and non-poor controller tuning instances, is randomly segmented into training and validation sets. In subsequent steps, the neural networks trained on this data will undergo refinement via transfer learning to compensate limitation of selecting specific process order model parameters in this stage, ensuring model diversity and broadened applicability across various scenarios and systems.

4.4.2 Gramian Angular Field (GAF)

In the preprocessing phase, individual PV and OP signals are combined into a composite signal, represented as $\frac{PV}{OP}$. While this transformation facilitates certain analyses, it presents challenges when further converting these signals into Gramian Angular

Table 4.1: Process model and controller transfer function for closed loop simulations.

Loop type	Process	Controller
Concentration	$G_p = \frac{3e^{-10s}}{10s+1}$	$G_c = 0.2(\frac{10s+1}{10s})$
Level	$G_p = \frac{1}{s}$	$G_c = 0.4(\frac{2s+1}{2s})$

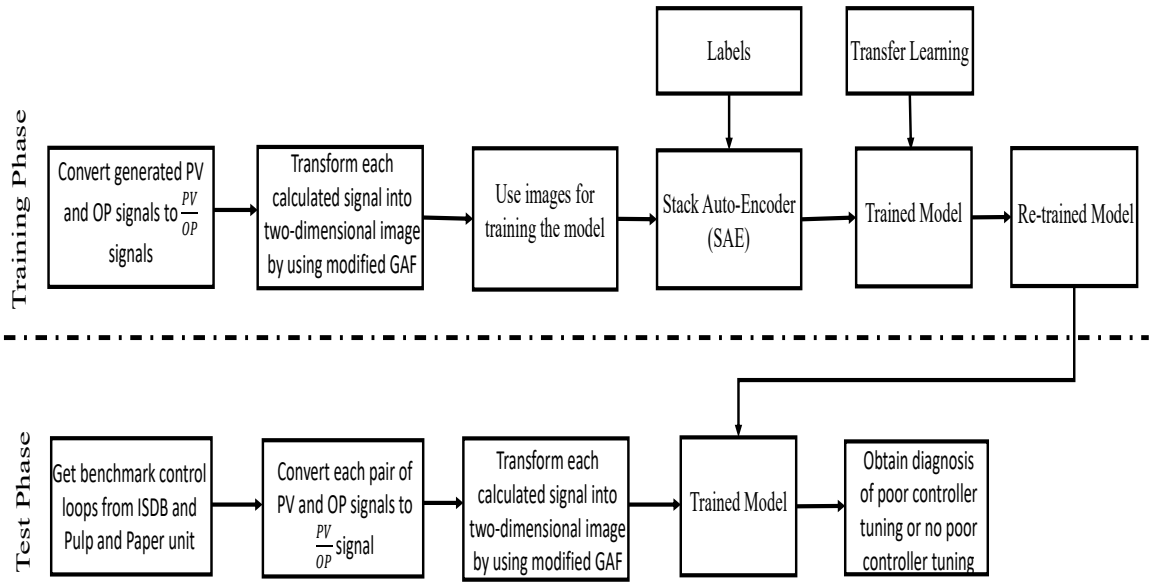


Figure 4.3: Development and utilization steps of SAE-based method.

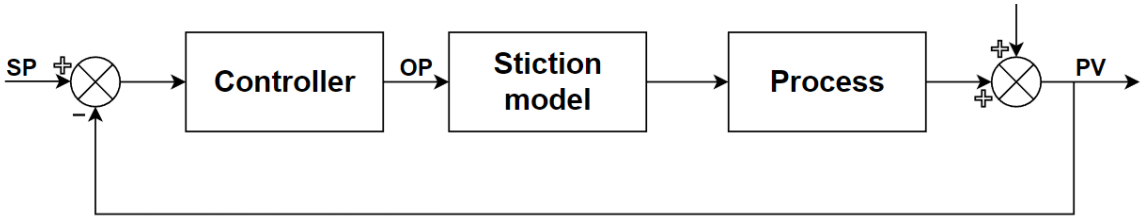


Figure 4.4: Control loop.

Field (GAF) images. Specifically, the traditional GAF transformation, using the cosine function, suffers from a lack of uniqueness.

The GAF technique typically employs the cosine or sine function, as delineated in Equation 4.4 and Equation 4.5, to generate the GAF images. A critical observation is that when the cosine function is utilized, the transformation does not result in unique representations. For any θ in the range $[0, 2\pi]$, there exists an equivalent value in the same range. This is due to the symmetry of the cosine function around $\theta = 2\pi$, meaning that $cos(\theta)$ is identical to $cos(2\pi - \theta)$. Consequently, an inverted time series,

Table 4.2: Parameter variation for generating tightly-tuned controllers.

Parameter	Description	Parameter range
K_c	Controller gain	[0.1:0.01:0.3]
I	Final controller integral value	[0.01:0.01:0.27]
V	Variance of noise	0.01

Table 4.3: Parameter variation for generating non-poor tuning data (sticky control valve).

Parameter	Description	Parameter range
S	Stiction band	[0.5:0.25:10]
J	Slip jump	[0.1:0.25:5]
V	Variance of white noise	0.01

Table 4.4: Parameter variation for generating non-poor tuning data (external oscillatory disturbance).

Parameter	Description	Parameter range
A	Amplitude	[1, 1.5, 2, 2.5]
F	Frequency of oscillation value	[0.01:0.01:0.27]
V	Variance of noise	0.01

denoted as $-X = \{-x_1, -x_2, \ldots, -x_n\}$ can produce an identical GAF matrix as its original counterpart, X.

For Original Time series:

$$GASF = cos(arccos(x_i) + arccos(x_j)) \tag{4.6}$$

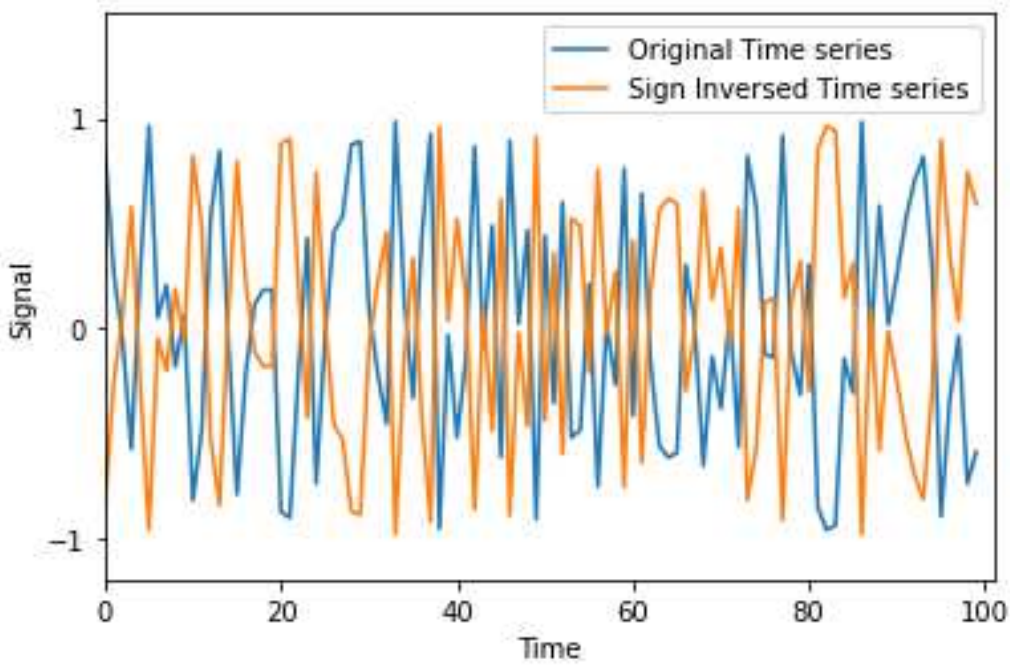
For sign-inversed Time series:

$$\begin{aligned} & cos(arccos(-x_i) + arccos(-x_j)) \\ =& cos([\pi - arccos(x_i)] + [\pi - arccos(x_j)]) \\ =& cos(2\pi - [arccos(x_i) + arccos(x_j)]) \\ =& cos(arccos(x_i) + arccos(x_j)) = GASF \end{aligned} \tag{4.7}$$

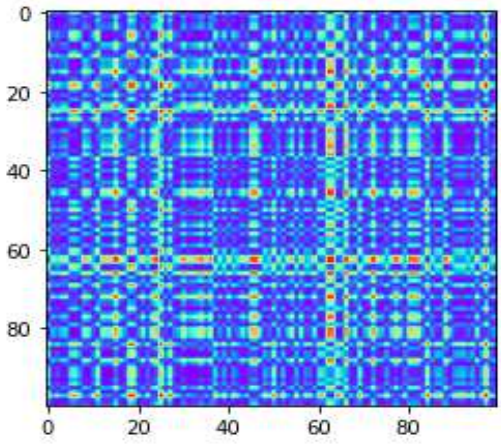
Equation 4.6 and Equation 4.7 underscore the fact that the GAF images for both the original and its sign-inversed version are indistinguishable. This is further visualized in Figure 4.5, which displays the GASF images for two sign-inversed time series.

In the realm of the Gramian Angular Field (GAF) transformations, the strengths of the cosine function in detecting periodicity, symmetries, and cyclic patterns are well acknowledged. However, its inherent limitation of non-uniqueness poses a significant challenge. While the sine function might offer a unique transformation, the depth of analysis provided by the cosine function, especially in capturing oscillatory data nuances, cannot be ignored.

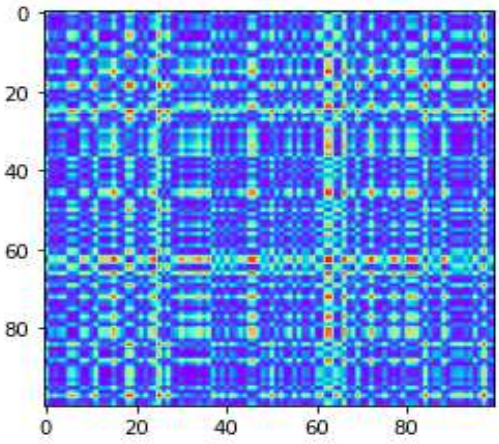
Addressing this gap, an innovative approach is introduced that seeks to harness the full potential of the cosine function while mitigating its drawbacks. Specifically,



(a) Original and sign inversed time series



(b) GASF of original time series



(c) GASF of sign-inversed time series

Figure 4.5: Similar GASF generated for original and sign-inversed time series.

an alternative solution to the prevailing non-uniqueness problem associated with GAF transformation using the cosine function is proposed.

The idea stems from the need for a more comprehensive representation. A three-dimensional matrix is generated by integrating both the sine and cosine functions as the first and second dimensions. The third dimension is populated with a matrix derived from the equation $\frac{1}{2} + sin(\phi_i + \phi_j) \times cos(\phi_i + \phi_j)$, ensuring precise computation

of angular values. This approach not only nullifies the non-uniqueness associated with the traditional GAF but also capitalizes on the merits of the cosine function.

Following this, each signal undergoes a transformation into two-dimensional images via the enhanced GAF method, detailed in section 4.3.1.

4.4.3 Stack Auto-Encoder (SAE)

The proposed methodology revolves around classifying control loops according to their tuning quality. Specifically, control loops demonstrating poor tuning are designated as 1, while those showcasing no poor tuning receive a label of 0. For an image representing a poor controller tuning data set, the target vector for the SAE model is [1 0]. The target vector becomes [0 1] for the image attained from a non poorly tuned data set.

The two-dimensional images generated in section 4.4.2, reflecting control loop behaviors across diverse tuning scenarios, are subsequently divided into distinct training and test subsets. Within the training phase, the focus centers on training a SAE. As training progresses, the SAE hones its internal representations, capturing essential features from the GAF images. This process unfolds through layer-wise pre-training, sequentially acquiring localized features, and culminates in fine-tuning achieved through backpropagation. The aim is to harmonize the learned representations with the intrinsic data structure. The SAE, now equipped as a potent feature extractor, distills pivotal patterns and attributes from the GAF images. These extracted features subsequently drive the classification of control loops based on their tuning quality. The output layer of the SAE computes probabilities, assigning an image to either the poor tuned or non poorly tuned class based on these probabilities.

4.4.4 Transfer Learning

Transfer learning is a strategy that allows the re-application of acquired knowledge from one domain, termed as the source domain, to enhance learning in another related, target domain. This approach is particularly promising for augmenting the adaptability of control system models. In this methodology, the SAE was initially honed on a dataset from the source domain, wherein only the controller parameters were subject to variation. A graphical representation of the applied Transfer Learning structure for the retraining of the SAE model can be viewed in Figure 4.6.

Control systems inherently display variations in not just controller parameters

but also in model parameters, further complicated by the intricate nature of real-world industrial data. Given this, the methodology introduces a target domain, involving datasets where model parameters were altered and melded with specific industrial datasets.

With a foundational understanding of the SAE's architecture and training process, the integration of transfer learning necessitates meticulous adaptations. The encoder of the SAE, along with the bottleneck and decoder components, needs careful recalibration to acclimate to the new data characteristics introduced by the target domain. The proficiency of encoder in distilling data is enhanced by fine-tuning its weights to the specifics of the target domain, maintaining its efficacy in discerning pivotal features across both domains.

The bottleneck undergoes subtle transformations, preserving initial representations while adapting to new patterns and variances from altered model parameters and industry-specific data. The decoder, essential for reconstructing input from compressed representations, also requires refinements to its weights to accurately decipher data patterns from diverse origins.

Post-comprehensive training on the source domain, the SAE enters a fine-tuning phase on the target domain, characterized by training on varied model parameters and specific industrial control loops. This phase, executed with a tempered learning rate, refines the foundational knowledge without overwriting it.

The resultant SAE is robust and insightful, adept at extracting features from GAF images under varied model parameters and synthesizing traditional SAE methodology with transfer learning. This synergy enables the model to be more accurate and attuned to real-world variations, establishing it as a powerful asset in control loop tuning classification.

4.5 Results and Discussions

To assess the capability of the proposed method to distinguish poor controller tuning, control loops from the ISDB and those from the pulp and paper industries were meticulously examined. Simulating control loop behaviors utilizing the process models and controller transfer functions detailed in Table 4.1—yielded an extensive artificial database with 2830 data entries. This simulation process considered two primary types of control loops: concentration and level loops.

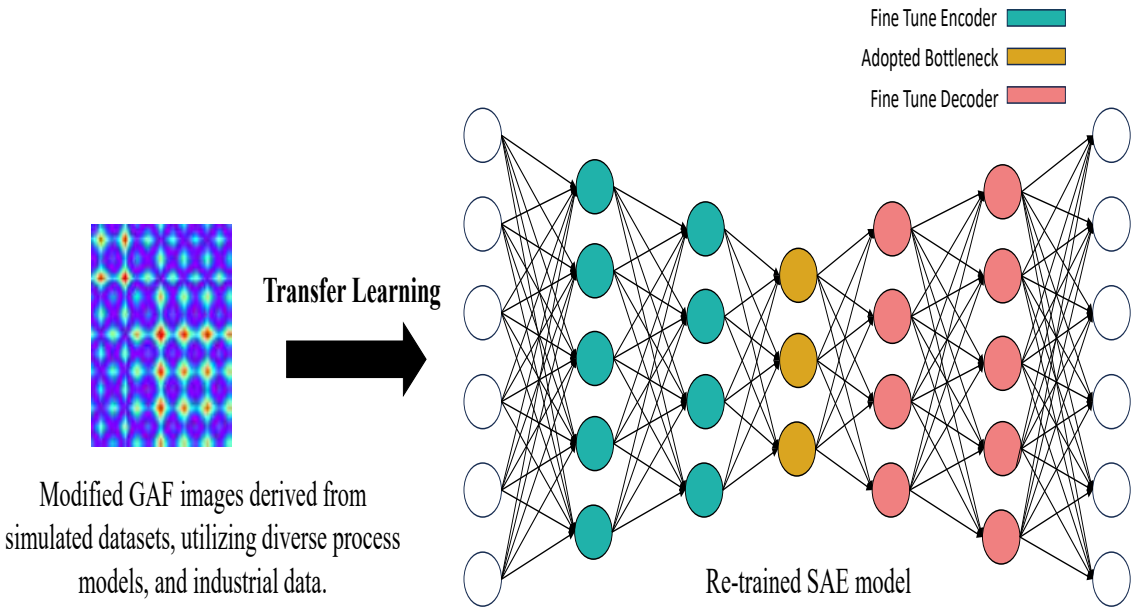
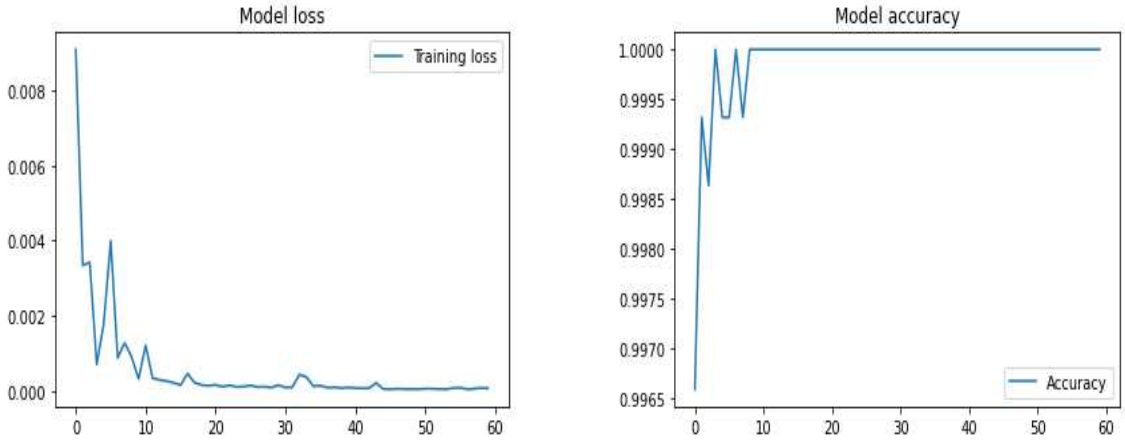


Figure 4.6: Transfer Learning structure for re-training the SAE model.

The simulated data, generated using parameters from Tables 4.2, 4.3, and 4.4, comprised both instances of poor tuning and controllers that did not exhibit such tuning but might be affected by issues like stiction or external oscillatory disturbances. These datasets underwent a transformation where the PV and OP signals were converted into their corresponding $\frac{PV}{OP}$ signal representation. This ratio was chosen not only to capture particular characteristics of the control loop behavior but also because it emphasizes the dynamic relationship between PV and OP. More precisely, the ratio encapsulates the response of the system to controller actions, offering profound insight into the influence of the controller on the system. This characteristic renders the ratio particularly useful when differentiating between non poorly tuned and poorly tuned controllers. Post this transformation, the signals were then morphed into two-dimensional images utilizing the modified GAF approach.

Having generated a plethora of data samples, these were randomly allocated into training and test sets. The training set, with its 2320 images, was pivotal for parameter learning of the SAE model. On the other hand, the test set, comprising 510 images, was instrumental in refining the parameter estimation of the model. Comprehensive specifics about the hyperparameters of the SAE model and the employed training algorithm can be consulted in Table 4.5. The learning progression of the SAE model is visually demonstrated in Figure 4.7.

To harness the power of transfer learning, data samples, generated using models



(a) Training loss of the model

(b) Training accuracy of the model

Figure 4.7: Performance of SAE model.

Table 4.5: Hyperparameters and training details of SAE model

Description	Values/Functions/Method
Number of layers	3
Neurons per layer	[512, 256, 128]
Activation function	relu
Output activation	sigmoid
Learning rate	0.001
Batch size	32
Epochs	50
Loss function	binary-crossentropy
optimizer	adam
Dropout rate	0.02

with different time constants, time delays, and noise levels detailed in Table 4.6, were integrated with select images from the industry. These data points formed the target domain, aimed at fine-tuning the pre-trained SAE model. The primary training set, consisting of 2320 images, was essential for the foundational training of the SAE on the source domain. For transfer learning, the model was then fine-tuned on a blend of the new dataset from Table 4.6 and industry images. The test set, comprising 510 images, was employed to gauge and refine the capability of model post-transfer learning. The loss and accuracy of the trained SAE, after implementing transfer learning, are illustrated in Figure 4.7.

Table 4.6: Parameter variation for process model.

Parameter	Description	Parameter range
τ	Time constant	[0.1:0.4:50]
θ	Time delay	[1:5:100]
ξ	noise	[0.01:0.05:0.3]

Effectiveness measurement for the proposed approaches, concerning the detection of poorly tuned controllers, employs a set of key metrics. These encompass accuracy, precision, recall, specificity, and the F1-score which are formulated as follows:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{4.8}$$

$$Precision = \frac{TP}{TP + FP} \tag{4.9}$$

$$Recall = \frac{TP}{TP + FN} \tag{4.10}$$

$$Specificity = \frac{TN}{TN + FP} \tag{4.11}$$

$$F_1 score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{4.12}$$

Where TP is true positive (control valve has poor tuning problem and the proposed method detects poor controller tuning), TN is true negative (the control valve has no poor tuning problem and the proposed method does not detect poor controller tuning), FP is false positive (the control valve has no poor tuning but the proposed method detects poor controller tuning problem) and FN is false negative (the control valve has poor tuning problem but the proposed method does not detect poor controller tuning).

By applying these metrics to the results obtained from the proposed methodology, encompassing ISDB-derived control loops as well as those sourced from the pulp and paper industry, a comprehensive evaluation of the method capability in detecting poor controller tuning can be conducted.

4.5.1 Case study 1

The ISDB, characterized by its benchmark control loops[34], serves as a widely adopted reference for the appraisal of methods dedicated to the detection of oscillations, stiction, and poor controller tuning. The database encompasses an array of PV and OP data from diverse industries, encompassing domains like buildings, chemicals, pulp and paper, power generation, mining, and metallurgy. Notably, an assortment of process variables is accommodated, spanning parameters such as temperature, pressure, level, flow rate, gauge readings, and concentrations.

A collection of 30 control loops has been drawn from the industrial datasets accessible within ISDB, these loops being imbued with issues that can generate oscillation. This assemblage serves as the designated test dataset. Subsequently, the modified SAE model with transfer learning, trained using the modified GAF images originating from the simulated data, was deployed onto the test set. The ensuing outcomes are laid out in Table 4.7. The columns of the table contain designations such as "Loop Name" (LN), "Actual Malfunction" (AM), "Output of the SAE Model" (OSM), and "Is Diagnosis Correct" (IDC). The achieved values for the performance metrics are documented in Table 4.8. The threshold was set at 0.5, a decision stemming from comprehensive empirical evaluations and insights gleaned from the nuances of the simulated data. This threshold was not case-dependent, further highlighting the robustness and adaptability of the model. After closely examining the information presented in Tables 4.8, it is clear that the proposed method showcases an achievement of 93% in accuracy when applied to the ISDB dataset.

Table 4.7: Application of pre-trained SAE model to ISDB

LN	AM	OSM		IDC?
CHEM 1	No poor tuning	0.13	0.87	Yes
CHEM 2	No poor tuning	0.02	0.98	Yes
CHEM 4	Poor tuned	0.86	0.14	Yes
CHEM 5	No poor tuning	0.83	0.17	No
CHEM 6	No poor tuning	0.09	0.91	Yes
CHEM 9	No poor tuning	0.08	0.92	Yes
CHEM 10	No poor tuning	0.11	0.89	Yes
CHEM 11	No poor tuning	0.22	0.78	Yes
CHEM 12	No poor tuning	0.93	0.007	No
CHEM 18	No poor tuning	0.05	0.95	Yes
CHEM 19	No poor tuning	0.01	0.99	Yes
CHEM 20	No poor tuning	0.23	0.77	Yes
CHEM 22	No poor tuning	0.06	0.94	Yes

Table 4.7: (continued) Application of pre-trained SAE model to ISDB

LN	AM	OSM		IDC?
CHEM 23	No poor tuning	0.19	0.81	Yes
CHEM 24	No poor tuning	0.06	0.94	Yes
CHEM 26	No poor tuning	0.10	0.90	Yes
CHEM 28	No poor tuning	0.03	0.97	Yes
CHEM 29	No poor tuning	0.2	0.80	Yes
CHEM 30	No poor tuning	0	1	Yes
CHEM 32	No poor tuning	0.07	0.93	Yes
CHEM 35	No poor tuning	0.25	0.75	Yes
PAP 1	No poor tuning	0.01	0.99	Yes
PAP 2	No poor tuning	0.24	0.76	Yes
PAP 4	Poor tuned	0.24	0.76	Yes
PAP 5	No poor tuning	0.05	0.95	Yes
PAP 7	No poor tuning	0.01	0.99	Yes
PAP 12	No poor tuning	0.09	0.91	Yes
PAP 13	No poor tuning	0.01	0.99	Yes
POW 1	No poor tuning	0.17	0.83	Yes
POW 2	No poor tuning	0.14	0.86	Yes

Table 4.8: Performance of the proposed method on ISDB

Performance metric	Trained SAE
True positive	2
True negative	26
False positive	2
False negative	0
Precision (%)	50
Recall (%)	100
Specificity (%)	92.80
F1 score (%)	66.67
Accuracy (%)	93.33

4.5.2 Case study 2

To demonstrate the real-time performance of the proposed algorithm, it is tested on real industrial data. 20 control loops, including 10 flow control loops, 5 temperature control loops, and 5 level control loops, are collected from the pulp and paper industry. The list of these control loops is provided in Table C1 with their diagnosed AM. The trained model with the simulated data is used for testing the control loops. The performance metrics of the proposed algorithm are represented in Table 4.10.

Results demonstrated in Table C1 verify the applicability of the proposed algorithm in detecting poor control tuning as a source of oscillation. It can be noticed from Table C1 that, the accuracy of the SAE model in detecting poor controller tuning in pulp and paper industry is 90%

The robustness of the model developed predominantly using simulated data, attests to the precision and representativeness of the simulations. These simulations were carefully constructed to encompass a myriad of real-world conditions, ensuring that the model remained adaptable and responsive to genuine system behaviors. Moreover, the $\frac{PV}{OP}$ ratio, a pivotal component of the method, emphasizes the dynamic interplay between PV and OP. This feature allowed the SAE model to penetrate the intricate patterns of system behaviors. By utilizing this ratio, the model demonstrated an enhanced sensitivity to subtle deviations, aligning it closely with the complexities of real-world industrial operations. However, it is worth noting that every model has its constraints. A limitation of this model is its adaptability to new images it hasn't been trained on, emphasizing the ongoing need for updates with emerging data.

Table 4.9: Application of pre-trained SAE model to Pulp and Paper

LN	AM	OSM		IDC?
FC 1	Poor tuned	0.76	0.24	Yes
FC 2	No poor tuning	0.07	0.93	Yes
FC 3	Poor tuned	0.86	0.14	Yes
FC 4	No poor tuning	0.83	0.17	No
FC 5	No poor tuning	0.29	0.71	Yes
FC 6	No poor tuning	0.17	0.83	Yes
FC 7	No poor tuning	0.23	0.77	Yes
FC 8	Poor tuned	0.89	0.11	Yes
FC 9	Poor tuned	0.95	0.05	Yes
FC 10	No poor tuning	0.29	0.71	Yes
LC 1	No poor tuning	0.11	0.89	Yes
LC 2	No poor tuning	0.35	0.65	Yes
LC 3	No poor tuning	0.38	0.62	Yes
LC 4	Poor tuned	0.28	0.72	No
LC 5	Poor tuned	0.88	0.12	Yes
TC 1	No poor tuning	0.07	0.93	Yes
TC 2	No poor tuning	0.31	0.69	Yes
TC 3	Poor tuned	0.79	0.21	Yes
TC 4	No poor tuning	0.28	0.72	Yes
TC 5	No poor tuning	0.23	0.77	Yes

Table 4.10: Performance of the proposed method on the pulp and paper industry

Performance metric	Trained SAE
True positive	6
True negative	12
False positive	1
False negative	1
Precision (%)	85.71
Recall (%)	85.71
Specificity (%)	92
F1 score (%)	85.71
Accuracy (%)	90

4.6 Conclusion

In conclusion, this chapter has presented a distinctive methodology for analyzing time series data by leveraging a modified Gramian Angular Field (GAF) transformation coupled with a Stacked Autoencoder (SAE). This involved creating a specialized dataset transformed into two-dimensional imagery through an altered GAF technique, serving as input for the SAE model. The model is renowned for its adeptness at recognizing intricate patterns and representations in deep learning contexts.

The applicability and efficacy of the proposed methodology have been corroborated through meticulous examinations on two diverse case studies: the International Stiction Database (ISDB) and a dataset from the pulp and paper industry. Attaining accuracy rates of 93.33% and 90% for the ISDB and pulp and paper industry respectively, reaffirms the method's adaptability and resilience across varying industrial spectra. Such elevated levels of accuracy elucidate the potential of the integrated GAF and SAE technique in precise time series analysis and prognostication.

However, it is crucial to acknowledge the potential limitations inherent in the current methodology. For instance, the methodology may face challenges when applied to higher-order processes. In cases where external oscillatory disturbances are nonlinear, or oscillations are induced by valve nonlinearity that cannot be depicted accurately by the adopted stiction models, discrepancies may arise. Additionally, this approach might encounter difficulties when dealing with processes having parameters that significantly deviate from those of the training models without sufficient target-domain data, potentially compromising the reliability and accuracy of predictions and analyses derived therefrom.

These acknowledged limitations underscore the avenues available for future research, focusing not only on refining the SAE's architecture and investigating alternate transformation techniques for time series data but also on exploring the adaptability of this approach to datasets with complex, nonlinear attributes. Consequently, the advancements discussed in this chapter provide insightful perspectives and avenues for enhancing the precision and efficacy of time series analysis, fostering progress in predictive modeling and decision-making processes across diverse practical applications. The contributions herein are steps toward resolving the intricate challenges associated with time series analysis in varying real-world contexts.

Chapter 5

Conclusions

In this chapter, summaries of the thesis are provided in section 5.1, and some possible future research is discussed in section 5.2.

5.1 Summary

This thesis presents novel methodologies for addressing three common challenges in the process control industry: oscillatory behavior in process control loops, stiction in control valves, and poor controller tuning. Each of these challenges can lead to suboptimal process performance, resulting in increased energy consumption, lower product quality, and even unsafe operating conditions. Addressing these challenges requires robust detection and diagnosis techniques capable of identifying the underlying issues and facilitating corrective action.

The thesis provides a comprehensive overview of the methodologies developed in this thesis, each tailored to tackle one of the aforementioned challenges. The methods proposed in these chapters have been rigorously tested and validated using both synthetic and real-world datasets, demonstrating their effectiveness in accurately detecting and diagnosing the targeted abnormalities. By implementing these methodologies, industrial practitioners can expect improved process control, optimized performance, and enhanced safety in their operations.

Chapter 1 provided the background and identified the challenges associated with detecting and diagnosing abnormalities in chemical processes. The aim was to develop reliable, accurate, and efficient methodologies for identifying oscillatory behavior, control valve stiction, and poor controller tuning. This chapter presented

the objectives and the structure of the thesis.

Chapter 2 proposed two shape-based pattern recognition approaches for detecting oscillations in process control loops. The first method utilized nonlinear algebraic functions, while the second employed deep convolutional neural networks (CNN). Both methods were designed to identify triangle-like shape in data plots, which are indicative of oscillatory behavior. The effectiveness of these approaches was demonstrated through their application across multiple industries. In particular, Method 1 and Method 2 achieved accuracy rates of approximately 81% and 91%, respectively.

Chapter 3 introduced a novel methodology for control valve stiction detection, combining Markov Transition Field (MTF) and CNN. The approach aimed to differentiate stiction-induced oscillations from other types of oscillations by transforming process variable data into images and training a CNN model. Transfer learning was employed to enhance stiction detection capabilities further. Benchmark control loops were used for testing, confirming the robustness of this method. The proposed approach achieved a high success rate of 91% in detecting sticky control valves.

Chapter 4 presented an innovative approach for poor controller tuning detection using Gramian Angular Field (GAF) and Stack Autoencoder (SAE) techniques. The methodology involved transforming time series data into 2D images using the modified GAF technique and training a SAE model on the transformed data. This approach was designed for real-time monitoring, alerting operators to poorly tuned controllers. Testing on two distinct case studies, the ISDB dataset and the pulp and paper dataset, demonstrated the robustness and generalizability of the proposed methodology. The accuracy rates achieved were 93.33% for the ISDB case study and 90% for the pulp and paper case study.

5.2 Future Work

For Chapter 2, future research could explore alternative pattern recognition techniques or more sophisticated algorithms for extracting shape-based features. Different types of oscillations and oscillation patterns could also be investigated to enhance the versatility of these methods. Moreover, an area of future research could focus on constructing a causal map based on the detected oscillations could help in understanding the interconnectedness of variables in the system and their influence on oscillatory behavior. Such a causal map would be instrumental in identifying the root causes of oscillations and designing strategies for their mitigation.

In Chapter 3, future work should explore deeper into the subject of stiction band calculation from PV and OP. Combining statistical techniques with machine learning algorithms could provide a more accurate and reliable approach for estimating the stiction band. This comprehensive strategy would likely enhance the understanding of the stiction band and improve the ability to identify issues with control valves in industrial settings.

In Chapter 4, there is an opportunity to further explore the development of a method that can distinguish between aggressive and sluggish controllers by identifying unique shapes in their respective time-series responses. By employing techniques such as shape-based pattern recognition, it could be possible to uncover these unique shapes and use them as a basis for differentiation. Machine learning algorithms, especially those capable of pattern recognition, could be employed to learn and recognize these specific shapes within the time-series data. Such a method would enable a more nuanced understanding of controller behavior and provide valuable insights for optimizing controller settings. This, in turn, could lead to enhanced process control, increased stability, and ultimately improved operational efficiency in industrial processes.

Bibliography

[1] MAA Shoukat Choudhury, Nina F Thornhill, and Sirish L Shah. Modelling valve stiction. *Control engineering practice*, 13(5):641–658, 2005.

[2] Thomas J Harris, CT Seppala, and LD Desborough. A review of performance monitoring and assessment techniques for univariate and multivariate control systems. *Journal of process control*, 9(1):1–17, 1999.

[3] YYS Henry, C Aldrich, and H Zabiri. Detection and severity identification of control valve stiction in industrial loops using integrated partially retrained cnn-pca frameworks. *Chemometrics and Intelligent Laboratory Systems*, 206:104143, 2020.

[4] Yaser Arbabi Yazdi, Heydar Toossian Shandiz, and Hosein Gholizadeh Narm. Stiction detection in control valves using a support vector machine with a generalized statistical variable. *ISA transactions*, 126:407–414, 2022.

[5] Wahiba Bounoua, Muhammad Faisal Aftab, and Christian Walter Peter Omlin. Controller performance monitoring: A survey of problems and a review of approaches from a data-driven perspective with a focus on oscillations detection and diagnosis. *Industrial & Engineering Chemistry Research*, 2022.

[6] Nina F Thornhill, Sirish L Shah, Biao Huang, and Anand Vishnubhotla. Spectral principal component analysis of dynamic process data. *Control Engineering Practice*, 10(8):833–846, 2002.

[7] Nina F Thornhill and Alexander Horch. Advances and new directions in plant-wide disturbance detection and diagnosis. *Control Engineering Practice*, 15(10):1196–1206, 2007.

[8] Amirreza Memarian, Seshu Kumar Damarla, and Biao Huang. Control valve stiction detection using markov transition field and deep convolutional neural network. *Canadian Journal of Chemical Engineering*, 101(11):6114–6125, 2023. (Accepted for publication).

[9] Da Zheng, Xi Sun, Seshu K Damarla, Ashish Shah, Joseph Amalraj, and Biao Huang. Valve stiction detection and quantification using a k-means clustering based moving window approach. *Industrial & Engineering Chemistry Research*, 60(6):2563–2577, 2021.

[10] Tore Hägglund. A control-loop performance monitor. *Control Engineering Prac-*

tice, 3(11):1543–1551, 1995.

[11] Tore Hägglund. Automatic supervision of control valves. *IFAC Proceedings Volumes*, 27(5):411–416, 1994.

[12] Nina F Thornhill and Tore Hägglund. Detection and diagnosis of oscillation in control loops. *Control Engineering Practice*, 5(10):1343–1354, 1997.

[13] Timothy I Salsbury and Ashish Singhal. A new approach for arma pole estimation using higher-order crossings. In *Proceedings of the 2005, American Control Conference, 2005.*, pages 4458–4463. IEEE, 2005.

[14] Alexey Zakharov, Elena Zattoni, Lei Xie, Octavio Pozo Garcia, and Sirkka-Liisa Jämsä-Jounela. An autonomous valve stiction detection system based on data characterization. *Control Engineering Practice*, 21(11):1507–1518, 2013.

[15] Alexey Zakharov, Alexey, and Sirkka-Liisa Jämsä-Jounela. Robust oscillation detection index and characterization of oscillating signals for valve stiction detection. *Industrial & Engineering Chemistry Research*, 53(14):5973–5981, 2014.

[16] Vesa-Matti Tikkala, Alexey Zakharov, and Sirkka-Liisa Jämsä-Jounela. A method for detecting non-stationary oscillations in process plants. *Control Engineering Practice*, 32:1–8, 2014.

[17] Tina Miao and Dale E Seborg. Automatic detection of excessively oscillatory feedback control loops. In *Proceedings of the 1999 IEEE International Conference on Control Applications (Cat. No. 99CH36328)*, volume 1, pages 359–364. IEEE, 1999.

[18] Nina F Thornhill, Biao Huang, and H Zhang. Detection of multiple oscillations in control loops. *Journal of Process control*, 13(1):91–100, 2003.

[19] Elham Naghoosi and Biao Huang. Automatic detection and frequency estimation of oscillatory variables in the presence of multiple oscillations. *Industrial & Engineering Chemistry Research*, 53(22):9427–9438, 2014.

[20] Robert Babuska, Jelmer van Ast, and Samir Mesic. A fuzzy-logic system for detecting oscillations in control loops. In *2006 IEEE International Conference on Fuzzy Systems*, pages 1310–1315. IEEE, 2006.

[21] Kangkang Zhang, Biao Huang, and Guoli Ji. Multiple oscillations detection in control loops by using the dft and raleigh distribution. *IFAC-PapersOnLine*, 48(21):529–534, 2015.

[22] Zixu Guo, Jiajun Shen, Lei Xie, Xiaoqiang Chen, and Hongye Su. Automatic detection of multiple oscillations by wavelet analysis. *Computers & Electrical Engineering*, 40(7):2167–2177, 2014.

[23] Elham Naghoosi and Biao Huang. Wavelet transform based methodology for detection and characterization of multiple oscillations in nonstationary variables. *Industrial & Engineering Chemistry Research*, 56(8):2083–2093, 2017.

[24] Norden E Huang, Zheng Shen, Steven R Long, Manli C Wu, Hsing H Shih, Qua-

nan Zheng, Nai-Chyuan Yen, Chi Chao Tung, and Henry H Liu. The empirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis. *Proceedings of the Royal Society of London. Series A: mathematical, physical and engineering sciences*, 454(1971):903–995, 1998.

[25] Xun Lang, Lei Xie, Yating Sun, and Hongye Su. Automatic oscillation detection based on improved local mean decomposition. In *2016 35th Chinese Control Conference (CCC)*, pages 6766–6771. IEEE, 2016.

[26] Lei Xie, Xun Lang, Alexander Horch, and Yuxi Yang. Online oscillation detection in the presence of signal intermittency. *Control Engineering Practice*, 55:91–100, 2016.

[27] Alexander Horch. *Condition monitoring of control loops*. PhD thesis, Signaler, sensorer och system, 2000.

[28] Alexander Horch. Benchmarking control loops with oscillations and stiction. In *Process control performance assessment*, pages 227–257. Springer, 2007.

[29] Srinivas Karra, Mohieddine Jelali, M Nazmul Karim, and Alexander Horch. Detection of oscillating control loops. *Detection and diagnosis of stiction in control loops*, pages 61–100, 2010.

[30] Riccardo Bacci di Capaci and Claudio Scali. Review and comparison of techniques of analysis of valve stiction: From modeling to smart diagnosis. *Chemical Engineering Research and Design*, 130:230–265, 2018.

[31] Mohieddine Jelali. An overview of control performance assessment technology and industrial applications. *Control engineering practice*, 14(5):441–466, 2006.

[32] Xinqing Gao, Fan Yang, Chao Shang, and Dexian Huang. A review of control loop monitoring and diagnosis: Prospects of controller maintenance in big data era. *Chinese Journal of Chemical Engineering*, 24(8):952–962, 2016.

[33] Laith Alzubaidi, Jinglan Zhang, Amjad J Humaidi, Ayad Al-Dujaili, Ye Duan, Omran Al-Shamma, José Santamaría, Mohammed A Fadhel, Muthana Al-Amidie, and Laith Farhan. Review of deep learning: Concepts, cnn architectures, challenges, applications, future directions. *Journal of big Data*, 8(1):1–74, 2021.

[34] Mohieddine Jelali, Biao Huang, et al. *Detection and diagnosis of stiction in control loops: state of the art and advanced methods*. Springer, 2010.

[35] Alireza Memarian, Santhosh Kumar Varanasi, and Biao Huang. Data-driven self-optimization of processes in the presence of the model-plant mismatch. *IFAC-PapersOnLine*, 55(7):532–537, 2022.

[36] Alireza Memarian, Santhosh Kumar Varanasi, Biao Huang, and Graham Slot. Smart optimization with ppcr modeling in the presence of missing data, time delay and model-plant mismatch. *Chemometrics and Intelligent Laboratory Systems*, 237:104812, 2023.

[37] Michael A Paulonis and John W Cox. A practical approach for large-scale controller performance assessment, diagnosis, and improvement. *Journal of Process*

Control, 13(2):155–168, 2003.

[38] Lane Desborough and Randy Miller. Increasing customer value of industrial control performance monitoring-honeywell's experience. In *AIChE symposium series*, number 326, pages 169–189. New York; American Institute of Chemical Engineers; 1998, 2002.

[39] WL Bialkowski. Dreams versus reality: a view from both sides of the gap: manufacturing excellence with come only through engineering excellence. *Pulp & Paper Canada*, 94(11):19–27, 1993.

[40] David B Ender. Process control performance: Not as good as you think. *Control Engineering*, 40(10):180–190, 1993.

[41] Seshu K Damarla, Xi Sun, Fangwei Xu, Ashish Shah, Joseph Amalraj, and Biao Huang. Practical linear regression-based method for detection and quantification of stiction in control valves. *Industrial & Engineering Chemistry Research*, 61(1):502–514, 2021.

[42] Seshu K Damarla, Xi Sun, Fangwei Xu, Ashish Shah, and Biao Huang. A sigmoid function based method for detection of stiction in control valves. In *2022 IEEE International Symposium on Advanced Control of Industrial Processes (AdCONIP)*, pages 210–215. IEEE, 2022.

[43] Raghunathan Rengaswamy, Tore Hägglund, and Venkat Venkatasubramanian. A qualitative shape analysis formalism for monitoring control loop performance. *Engineering Applications of Artificial Intelligence*, 14(1):23–33, 2001.

[44] Weng Kean Teh, Haslinda Zabiri, Yudi Samyudia, Sean S Jeremiah, Bashariah Kamaruddin, Azhari AA Mohd Amiruddin, and Nasser M. Ramli. An improved diagnostic tool for control valve stiction based on nonlinear principle component analysis. *Industrial & Engineering Chemistry Research*, 57(33):11350–11365, 2018.

[45] Jonathan WV Dambros, Jorge O Trierweiler, and Marcelo Farenzena. Oscillation detection in process industries–part i: Review of the detection methods. *Journal of Process Control*, 78:108–123, 2019.

[46] Ahmad Azharuddin Azhari Mohd Amiruddin, Haslinda Zabiri, Sean Suraj Jeremiah, Weng Kean Teh, and Bashariah Kamaruddin. Valve stiction detection through improved pattern recognition using neural networks. *Control Engineering Practice*, 90:63–84, 2019.

[47] Yaser Arbabi Yazdi, Heydar Toossian Shandiz, and Hossein Gholizade Narm. Automatic oscillations detection and classification of control loop using generalized machine learning algorithms. *Transactions of the Institute of Measurement and Control*, page 01423312221118129, 2022.

[48] MAA Shoukat Choudhury, Mridul Jain, and Sirish L Shah. Stiction-definition, modelling, detection and quantification. *Journal of Process control*, 18(3-4):232–243, 2008.

[49] B Kamaruddin, H Zabiri, AAA Mohd Amiruddin, WK Teh, M Ramasamy, and SS Jeremiah. A simple model-free butterfly shape-based detection (bsd) method

integrated with deep learning cnn for valve stiction detection and quantification. *Journal of Process Control*, 87:1–16, 2020.

[50] Vijoy Akavalappil and TK Radhakrishnan. Comparison of current state of control valve stiction detection and quantification techniques. *Transactions of the Institute of Measurement and Control*, 44(3):562–579, 2022.

[51] Zhiguang Wang and Tim Oates. Imaging time-series to improve classification and imputation. *https://arxiv.org/abs/1506.00327,last accessed on 01/06/2015*.

[52] Rongjie Li, Yao Wu, Qun Wu, Nilanjan Dey, Rubén González Crespo, and Fuqian Shi. Emotion stimuli-based surface electromyography signal classification employing markov transition field and deep neural networks. *Measurement*, 189:110470, 2022.

[53] Bin Han, Hui Zhang, Ming Sun, and Fengtong Wu. A new bearing fault diagnosis method based on capsule network and markov transition field/gramian angular field. *Sensors*, 21(22):7762, 2021.

[54] Wei Sun, Jie Zhou, Bintao Sun, Yuqing Zhou, and Yongying Jiang. Markov transition field enhanced deep domain adaptation network for milling tool condition monitoring. *Micromachines*, 13(6):873, 2022.

[55] Jialin Yan, Jiangming Kan, and Haifeng Luo. Rolling bearing fault diagnosis based on markov transition field and residual network. *Sensors*, 22(10):3936, 2022.

[56] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, United States of America, 2016.

[57] Allan RS Venceslau, Luiz Affonso Guedes, and Diego RC Silva. Artificial neural network approach for detection and diagnosis of valve stiction. In *Proceedings of 2012 IEEE 17th International Conference on Emerging Technologies & Factory Automation (ETFA 2012)*, pages 1–4. IEEE, 2012.

[58] Elham Naghoosi and Biao Huang. Diagnosis of oscillations between controller tuning and harmonic external disturbances. *IEEE Transactions on Control Systems Technology*, 23(4):1283–1293, 2014.

[59] Tore Hägglund. Automatic detection of sluggish control loops. *Control Engineering Practice*, 7(12):1505–1511, 1999.

[60] MAA Shoukat Choudhury, Sirish L Shah, and Nina F Thornhill. Diagnosis of poor control-loop performance using higher-order statistics. *Automatica*, 40(10):1719–1728, 2004.

[61] Gong Zhang, Yujuan Si, Di Wang, Weiyi Yang, and Yongjian Sun. Automated detection of myocardial infarction using a gramian angular field and principal component analysis network. *IEEE Access*, 7:171570–171583, 2019.

[62] K Palani Thanaraj, B Parvathavarthini, U John Tanik, V Rajinikanth, Seifedine Kadry, and Krishnamurthy Kamalanand. Implementation of deep neural net-

works to classify eeg signals using gramian angular summation field for epilepsy diagnosis. *arXiv preprint arXiv:2003.04534*, 2020.

[63] Jaime Zabalza, Jinchang Ren, Jiangbin Zheng, Huimin Zhao, Chunmei Qing, Zhijing Yang, Peijun Du, and Stephen Marshall. Novel segmented stacked autoencoder for effective dimensionality reduction and feature extraction in hyperspectral imaging. *Neurocomputing*, 185:1–10, 2016.

[64] Peicheng Zhou, Junwei Han, Gong Cheng, and Baochang Zhang. Learning compact and discriminative stacked autoencoder for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 57(7):4823–4833, 2019.

Appendices

Appendix A

Details of Industrial Control Loops (International stiction database (ISDB) loops)

Table A1: International stiction database (ISDB) loops

Loop name	Industrial field	Control loop type	Actual condition
Bas 1	Buildings	Temperature	No oscillation
Bas 2	Buildings	Temperature	No oscillation
Bas 3	Buildings	Temperature	Intermittent oscillation
Bas 4	Buildings	Pressure	Intermittent oscillation
Bas 5	Buildings	Pressure	OP not available
Bas 6	Buildings	Temperature	Stiction and tight tunning
Bas 7	Buildings	Temperature	Stiction
Bas 8	Buildings	Temperature	No oscillation
CHEM 1	Chemicals	Flow	Stiction
CHEM 2	Chemicals	Flow	Stiction
CHEM 3	Chemicals	Temperature	Quantisation
CHEM 4	Chemicals	Level	Tuning problem
CHEM 5	Chemicals	Flow	Stiction
CHEM 6	Chemicals	Flow	Stiction
CHEM 7	Chemicals	Pressure	Stiction
CHEM 8	Chemicals	Pressure	Stiction
CHEM 9	Chemicals	Pressure	Stiction
CHEM 10	Chemicals	Pressure	Stiction
CHEM 11	Chemicals	Flow	Stiction
CHEM 12	Chemicals	Flow	Stiction
CHEM 13	Chemicals	Analyser	Faulty stem sensor
CHEM 14	Chemicals	Flow	Faulty stem sensor
CHEM 15	Chemicals	Pressure	Interaction
CHEM 16	Chemicals	Pressure	Interaction
CHEM 17	Chemicals	Temperature	Faulty stem sensor
CHEM 18	Chemicals	Flow	Stiction
CHEM 19	Chemicals	Flow	Stiction
CHEM 20	Chemicals	Flow	Stiction
CHEM 21	Chemicals	Flow	Disturbance
CHEM 22	Chemicals	Flow	Stiction
CHEM 23	Chemicals	Flow	Stiction
CHEM 24	Chemicals	Flow	Stiction
CHEM 25	Chemicals	Pressure	Possible margin Stability
CHEM 26	Chemicals	Level	Stiction
CHEM 27	Chemicals	Level	Disturbance
CHEM 28	Chemicals	Temperature	Stiction
CHEM 29	Chemicals	Flow	Stiction
CHEM 30	Chemicals	Flow	Stiction
CHEM 31	Chemicals	Flow	Stiction
CHEM 32	Chemicals	Flow	Stiction
CHEM 33	Chemicals	Flow	Disturbance

Table A1: (continued) International stiction database (ISDB) loops

Loop name	Industrial field	Control loop type	Actual condition
CHEM 34	Chemicals	Flow	Disturbance
CHEM 35	Chemicals	Flow	Stiction
CHEM 36	Chemicals	Level	Disturbance
CHEM 37	Chemicals	Level	Disturbance
CHEM 38	Chemicals	Pressure	Disturbance
CHEM 39	Chemicals	Pressure	Disturbance
CHEM 40	Chemicals	Temperature	No oscillation
CHEM 41	Chemicals	Temperature	No oscillation
CHEM 42	Chemicals	Temperature	No oscillation
CHEM 43	Chemicals	Temperature	No oscillation
CHEM 44	Chemicals	Temperature	No oscillation
CHEM 45	Chemicals	Pressure	No oscillation
CHEM 46	Chemicals	Pressure	No oscillation
CHEM 47	Chemicals	Pressure	No oscillation
CHEM 48	Chemicals	Pressure	No oscillation
CHEM 49	Chemicals	Pressure	No oscillation
CHEM 50	Chemicals	Level	No oscillation
CHEM 51	Chemicals	Level	No oscillation
CHEM 52	Chemicals	Level	No oscillation
CHEM 53	Chemicals	Level	No oscillation
CHEM 54	Chemicals	Level	No oscillation
CHEM 55	Chemicals	Level	No oscillation
CHEM 56	Chemicals	Flow	No oscillation
CHEM 57	Chemicals	Flow	No oscillation
CHEM 58	Chemicals	Flow	No oscillation
CHEM 59	Chemicals	Flow	No oscillation
CHEM 60	Chemicals	Flow	No oscillation
CHEM 61	Chemicals	Flow	No oscillation
CHEM 62	Chemicals	Flow	No oscillation
CHEM 63	Chemicals	Flow	No oscillation
CHEM 64	Chemicals	Flow	No oscillation
PAP 1	Pulp and Papers	Flow	Stiction
PAP 2	Pulp and Papers	Flow	Stiction
PAP 3	Pulp and Papers	Level	Stiction
PAP 4	Pulp and Papers	Concentration	Tight tunning
PAP 5	Pulp and Papers	Concentration	Stiction
PAP 6	Pulp and Papers	Level	No stiction
PAP 7	Pulp and Papers	Flow	External disturbance
PAP 8	Pulp and Papers	Level	No stiction
PAP 9	Pulp and Papers	Temperature	No stiction
PAP 10	Pulp and Papers	Level	No oscillation

Table A1: (continued) International stiction database (ISDB) loops

Loop name	Industrial field	Control loop type	Actual condition
PAP 11	Pulp and Papers	Level	Stiction
PAP 12	Pulp and Papers	Level	Stiction
PAP 13	Pulp and Papers	Level	Stiction
POW 1	Power Plants	Level	Stiction
POW 2	Power Plants	Level	Stiction
POW 3	Power Plants	Level	No stiction
POW 4	Power Plants	Level	Stiction
POW 5	Power Plants	Level	No stiction
MIN 1	Mining	Temperature	Stiction
MET 1	Metals	Gauge	External disturbance
MET 2	Metals	Gauge	External disturbance
MET 3	Metals	Gauge	No oscillation

Appendix B

Details of Industrial Control Loops (Refinary loops)

Table B1: Refinary loops

Loop name	Industrial field	Control loop type	Actual condition
FC 1	Chemicals	Temperature	No oscillation
FC 2	Chemicals	Temperature	No oscillation
FC 3	Chemicals	Temperature	Oscillation
FC 4	Chemicals	Pressure	Oscillation
FC 5	Chemicals	Pressure	Oscillation
FC 6	Chemicals	Temperature	Oscillation
FC 7	Chemicals	Temperature	Oscillation
FC 8	Chemicals	Temperature	No oscillation
FC 9	Chemicals	Temperature	No oscillation
FC 10	Chemicals	Temperature	No oscillation
FC 11	Chemicals	Temperature	No oscillation
TC 1	Chemicals	Flow	Oscillation
TC 2	Chemicals	Flow	Oscillation
TC 3	Chemicals	Temperature	Oscillation
TC 4	Chemicals	Level	Oscillation
TC 5	Chemicals	Flow	Oscillation
TC 6	Chemicals	Flow	Oscillation
TC 7	Chemicals	Pressure	Oscillation
TC 8	Chemicals	Pressure	Oscillation
TC 9	Chemicals	Pressure	Oscillation
TC 10	Chemicals	Pressure	Oscillation
TC 11	Chemicals	Flow	Oscillation
LC 1	Chemicals	Flow	Oscillation
LC 2	Chemicals	Analyser	Oscillation
LC 3	Chemicals	Flow	Oscillation
LC 4	Chemicals	Pressure	Oscillation
LC 5	Chemicals	Pressure	Oscillation
LC 6	Chemicals	Temperature	Oscillation
LC 7	Chemicals	Flow	Oscillation
LC 8	Chemicals	Flow	Oscillation
LC 9	Chemicals	Flow	Oscillation
LC 10	Chemicals	Flow	Oscillation
LC 11	Chemicals	Flow	Oscillation
LC 12	Chemicals	Flow	Oscillation

Appendix C

Details of Industrial Control Loops (Pulp and Paper loops)

Table C1: Pulp and Paper loops

Loop name	Industrial field	Control loop type	Actual condition
FC 1	Pulp and Paper	Flow	Poor tuned
FC 2	Pulp and Paper	Flow	Well tuned
FC 3	Pulp and Paper	Flow	Poor tuned
FC 4	Pulp and Paper	Flow	Well tuned
FC 5	Pulp and Paper	Flow	Well tuned
FC 6	Pulp and Paper	Flow	Well tuned
FC 7	Pulp and Paper	Flow	Well tuned
FC 8	Pulp and Paper	Flow	Poor tuned
FC 9	Pulp and Paper	Flow	Poor tuned
FC 10	Pulp and Paper	Flow	Well tuned
LC 1	Pulp and Paper	Level	Well tuned
LC 2	Pulp and Paper	Level	Well tuned
LC 3	Pulp and Paper	Level	Well tuned
LC 4	Pulp and Paper	Level	Poor tuned
LC 5	Pulp and Paper	Level	Poor tuned
TC 1	Pulp and Paper	Temperature	Well tuned
TC 2	Pulp and Paper	Temperature	Well tuned
TC 3	Pulp and Paper	Temperature	Poor tuned
TC 4	Pulp and Paper	Temperature	Well tuned
TC 5	Pulp and Paper	Temperature	Well tuned