QATAR UNIVERSITY

COLLEGE OF ENGINEERING

FAULT DIAGNOSIS OF SENSOR AND ACTUATOR FAULTS IN MULTI-ZONE HVAC SYSTEMS

BY

MARIAM AHMED ELNOUR

A Thesis Submitted to

the Faculty of the College of Engineering

in Partial Fulfillment of the Requirements for the Degree of

Masters of Science in Electrical Engineering

June 2019

COMMITTEE PAGE

The members of the Committee approve the Thesis of Mariam Elnour

defended on 16/04/2019.

Dr. Nader Meskin
Thesis Supervisor

Dr. Mohammed Al-Naemi
Thesis Co-supervisor

Dr. Marios M. Polycarpou
Committee Member

Dr. Mustafa S. Kiranyaz
Committee Member

Approved:

Abdel Magid Hamouda, Dean, College of Engineering

ABSTRACT

ELNOUR, MARIAM, A., Masters :

June : 2019, Master of Science in Electrical Engineering

Title: Fault Diagnosis of Sensor and Actuator Faults in Multi-Zone HVAC Systems

Supervisors of Thesis: Dr. Nader, M, Meskin and Dr. Mohammed, I, Al-Naemi.

Globally, the buildings sector accounts for 30% of the energy consumption and more than 55% of the electricity demand. Specifically, the Heating, Ventilation, and Air Conditioning (HVAC) system is the most extensively operated component and it is responsible alone for 40% of the final building energy usage. HVAC systems are used to provide healthy and comfortable indoor conditions, and their main objective is to maintain the thermal comfort of occupants with minimum energy usage.

HVAC systems include a considerable number of sensors, controlled actuators, and other components. They are at risk of malfunctioning or failure resulting in reduced efficiency, potential interference with the execution of supervision schemes, and equipment deterioration. Hence, Fault Diagnosis (FD) of HVAC systems is essential to improve their reliability, efficiency, and performance, and to provide preventive maintenance.

In this thesis work, two neural network-based methods are proposed for sensor and actuator faults in a 3-zone HVAC system. For sensor faults, an online semi-supervised sensor data validation and fault diagnosis method using an Auto-Associative Neural Network (AANN) is developed. The method is based on the implementation of Non-linear Principal Component Analysis (NPCA) using a Back-Propagation Neural Network (BPNN) and it demonstrates notable capability in sensor fault and inaccuracy correction, measurement noise reduction, missing sensor data replacement, and in both single and multiple sensor faults diagnosis. In addition, a novel on-line supervised

multi-model approach for actuator fault diagnosis using Convolutional Neural Networks (CNNs) is developed for single actuator faults. It is based a data transformation in which the 1-dimensional data are configured into a 2-dimensional representation without the use of advanced signal processing techniques. The CNN-based actuator fault diagnosis approach demonstrates improved performance capability compared with the commonly used Machine Learning-based algorithms (i.e., Support Vector Machine and standard Neural Networks).

The presented schemes are compared with other commonly used HVAC fault diagnosis methods for benchmarking and they are proven to be superior, effective, accurate, and reliable. The proposed approaches can be applied to large-scale buildings with additional zones.

DEDICATION

This study is wholeheartedly dedicated to my beloved parents,

Sawsan and Ahmed.

and to Professor Mohieddine Benammer (May Allah bless his soul)

who inspired me the passion for knowledge and research.

ACKNOWLEDGMENT

This work could not have been completed without the guidance and supervision of Dr. Nader Meskin to whom I would like to express my sincere gratitude and appreciation for his support, patience, and motivation. I really appreciate his contributions of time and ideas towards finishing my thesis. It has been an honor to be one of his students and I could never have imagined having a better advisor than Dr. Nader Meskin.

Secondly, I would like to thank my family and friends for their continuous support and encouragement, especially my parents. I would like to thank all my professors who contributed in expanding and enriching my knowledge throughout my Bachelor and Master's studies, that I am ultimately able to present this work.

TABLE OF CONTENTS

List of Tables

List of Figures

LIST OF ABBREVIATIONS

AANN	Auto-Associative Neural Network
ACC	Accuracy
Adam	Adaptive Moment Estimation
AHU	Air Handling Unit
ANN	Artificial Neural Network
ARMAX	Autoregressive Moving Average
ARX	Autoregressive Exogenous
BBN	Bayesian Belief Network
BMS	Building Management and Control Systems
BN	Batch Normalization
BPNN	Back-Propagation Neural Network
BWR	Boiling Water Reactor
CAV	Constant Air Volume
CHMM	Coupled Hidden Markov Model
CNN	Convolutional Neural Network
COP	Coefficient of Performance
CUSUM	Cumulative Sum Control Chart
CWTS	Continuous Wavelet Transform Scalogram
DAT	Discharge Air Temperature
DBN	Deep Belief Network
DFT	Discrete Fourier Transform
DT	Decision Tree

DWT	Discrete Wavelet Transform
EA	Exhaust Air
EKF	Extended Kalman Filter
EWMA	Exponentially Weighted Moving Average
FC	Fully Connected (or Fully Closed)
FO	Fully Open
FD	Fault Diagnosis
FN	False Negatives
FNR	False Negatives Rate
FP	False Positives
FPR	False Positives Rate
GA	Genetic Algorithm
GD	Gradient Decedent
GLRT	Generalized Likelihood Ratio Test
GP	Gaussian Process
HMM	Hidden Markov Model
HVAC	Heating, Ventilation, and Air Conditioning
ICA	Independent Component Analysis
JAA	Joint Angle Analysis
KF	Kalman Filters
KPCA	Kernel Principal Component Analysis
LDA	Linear Discriminant Analysis
LSFD	Local Sensor Fault Diagnosis
LSPF	Local Search Particle Filter

LSSVM	Least Square Support Vector Machine
MCC	Matthews Correlation Coefficient
MMC	Modular Multilevel Converter
MOC	Multi-Objective Clustering
MOC-RCE	Multi-Objective Clustering Rapid Centroid Estimation
NLPCA	Nonlinear Principle Component Analysis
NN	Neural Network
NPCA	Nonlinear Principle Component Analysis
OA	Outside Air
OLAD	Discrete-Time On-Line Approximator
OPTICS	Ordering Points To Identify The Clustering Structure
PCA	Principal Component Analysis
PF	Particle Filter
PID	Proportional Integral Derivative
PIs	Performance Indices
PLS	Partial Least Squares
PO	Partially Open
PPCA	Probabilistic Principal Component Analysis
PPV	Positive Predictive Value
RA	Return Air
RBF	Radius Basis Function
RCA	Refrigerant Charge Amount
RCE	Rapid Centroid Optimization
REC-ACC	Reconstruction Accuracy

ReLu	Rectified Linear Unit
RNL	Reduced Noise Level
SEMs	Spectral Energy Maps
SFD	Sensor Fault Diagnosis
SG	Satizky-Golay
SGDM	Stochastic Gradient Descent with Momentum
SPC	Statistical Process Control
SPE	Squared Prediction Error
SVM	Support Vector Machine
Tanh	Hyperbolic Tangent
TFDK	Tree-Structured Fault Dependence Kernel
TN	True Negatives
TNR	True Negatives Rate
TP	True Positives
TPR	True Positives Rate
TRNSYS	Transient System Analysis
TV	Total Variation
UKF	Unscented Kalman Filter
UPF	Unscented Particle Filter
VAV	Variable Air Volume
VAVC	Variable Air Volume Conditioning
VRF	Variable Refrigerant Flow
WD	Wavelet De-noising

Chapter 1: INTRODUCTION

Globally, the buildings sector accounts for about 40% of the energy consumption and more than 55% of the electricity demand [4]. Specifically, the Heating, Ventilation, and Air Conditioning (HVAC) system is one of the major and extensively operated components of buildings for providing healthy and comfortable indoor conditions and it is responsible alone for 40% of the final building's energy usage [5]. Its main objective is to maintain the thermal comfort of occupants with minimum energy usage. This energy demand continues to rise, driven by the improved access to energy, the greater use of energy-consuming devices, and the rapid growth in the global building sector.

1.1 HVAC System Description

A typical HVAC system in buildings is composed of an air ventilation system, an Air Handling Unit (AHU), and a coil fluid chiller/heater system connected with ducts, air chambers, and pipes as illustrated in Fig.1.1. It is equipped with temperature, humidity, pressure, and flow sensors for monitoring and control purposes as well as fans, pumps, valves, dampers, and filters.

It is concerned with three main processes which are, heating and cooling related to controlling the thermal energy within the control space, humidifying and dehumidifying concerning the control of the amount of moisture in the air, and finally ventilation, filtration, and circulation of air [6].

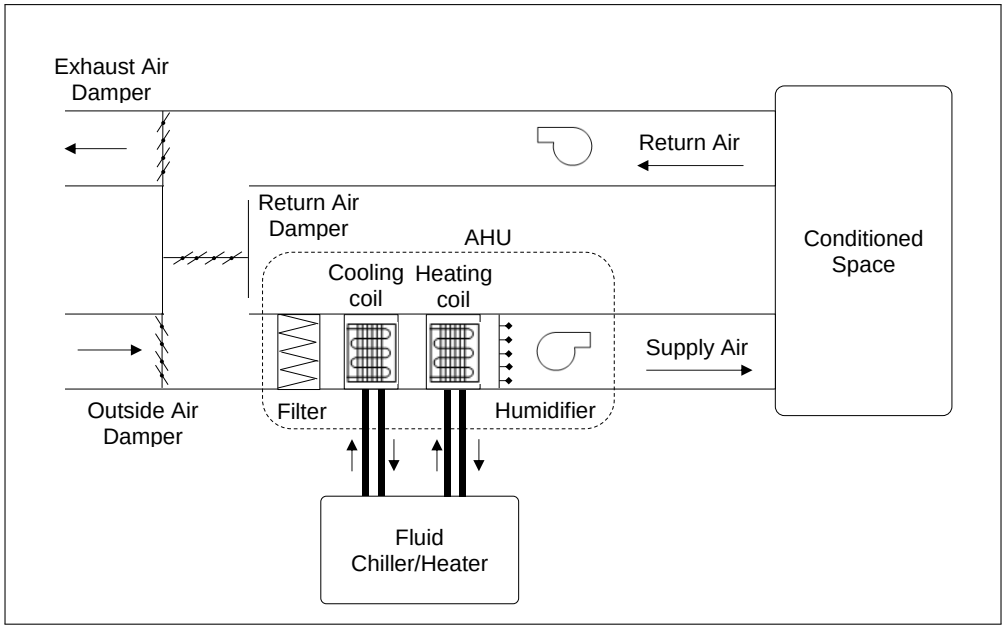


Figure 1.1: Diagram of a typical HVAC system.

1.1.1 Air Ventilation System

The air ventilation system is responsible for the air exchange between the conditioned space and the outdoor environment. Mainly, the ventilation system aims to dilute the gaseous contaminants in the air and to maintain an acceptable indoor air quality in terms of freshness. However, based on the ventilation rate, in some cases it is used for maintaining the conditioned space temperature and humidity. For example, exchanging hot indoor air with cooler air, and exhausting moist air for drier air from the outside environment. Typically, in HVAC systems in commercial and institutional buildings the ratio of ventilation air to indoor air varies from 15% to 25% of the outside air [6].

1.1.2 Air-Handling Unit

The air-handling unit is responsible for bringing the space air of the conditioned space of interest to desired setpoints. Generally, its main components are the cooling coil for the cooling and dehumidification of air, the heating coil for heating, the humidi-

fier, and the air filter. All or some of those components may exist in AHUs based on the building's requirements and the type of application. AHUs can be classified into two types in terms of the air conditioning method as constant air volume (CAV) and variable air volume (VAV) as illustrated in Figures 1.2 and 1.3. In CAV, the air is supplied to the conditioned space at a constant flow rate and variable temperature according to conditioning requirement. However, an AHU with VAV supplies air of a constant temperature that is sufficient to meet the maximum thermal load in the conditioned space and a variable air flow rate controlled to meet the desired setpoints.

The air-handling unit is one of the most extensively operated equipment in HVAC systems and hence it contributes to a significant portion of the total energy consumption in buildings. It requires follow-up maintenance closely and it is subjected to malfunction and failure due to multiple factors such as poor system integration, equipment failure, etc.

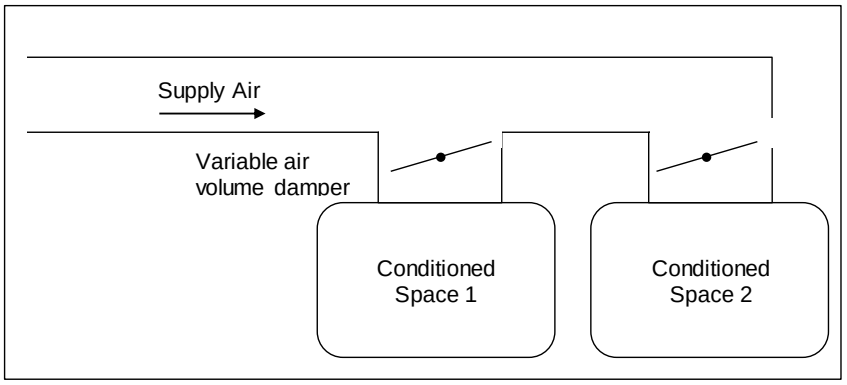


Figure 1.2: A VAV system for a two-zone HVAC system.

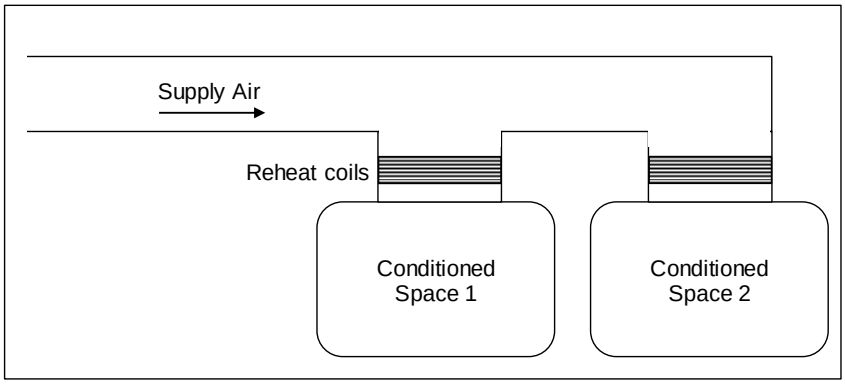


Figure 1.3: A CAV system for a two-zone HVAC system.

1.1.3 Fluid Chiller/Heater System

The fluid chiller/heater system is dedicated to adjusting the fluid temperature before supplying it to the AHU coils. Chiller plants account for a considerable portion of the energy consumption of HVAC systems. That is, under faulty operation -due to performance degradation or malfunction-, a significant amount of energy is wasted [7].

1.2 Objective of HVAC Systems

The HVAC system is desired to provide an acceptable indoor environment with optimum cost and energy usage solutions. The objective of HVAC systems can be described in terms of the following factors: air quality, thermal comfort, safety, and system's reliability and efficiency [6].

1.2.1 Air Quality

The air quality inside buildings is an important factor affecting the health of the occupants. One of the primary objectives of the HVAC system is maintaining the minimum essential air quality measures by supplying clean, odor-free outdoor air to the indoor spaces in the quantities necessary for breathing and dilution. In addition, smoke,

excessive moisture, gases, and other airborne pollutants generated must be removed from the indoor air.

1.2.2 Thermal Comfort

The thermal comfort can be defined as the measure of satisfaction of occupants with the indoor environment. It is directly affecting their health and productivity. Thermal comfort can be considered subjective due to some influencing factors such as the activity level, clothing, and individual preferences of occupants. However, there are main factors directly related to the management of the HVAC system operation which are the supply air temperature and speed, the radiant heat, and the humidity level. It is required that the indoor air temperature and motion are closely and simultaneously controlled as to maintain the desired and acceptable temperature overly and without causing draft discomfort or a stuffy environment.

1.2.3 Safety

The assurance of occupants' safety is essential. The HVAC system's equipment may be potential and serious hazard sources in case of failure or malfunction. For example, a heating equipment can cause a potential fire hazard. Improper air distribution in case of fire occurrence can result in spreading the fire in the building [8]. Moreover, a malfunction in the HVAC system components and sensors can interfere in safety and evacuation procedures, Hence, precautions and proper management of HVAC systems are necessary to prevent such incidents and to employ the system in assisting safety assurance operation.

1.2.4 Reliability and Efficiency

The reliability and efficiency of the HVAC system operation are important characteristics. The HVAC equipment is desired to operate healthily and normally on the long run with minimal maintenance requirement as to lower the repair costs. Moreover, the HVAC systems' energy consumption must be optimum, and this is achieved by proper HVAC system design based on the building's type, size, etc., and by proper control and monitoring of the system's operation.

1.3 Thesis Motivation

As mentioned previously, HVAC systems are major components of buildings for providing healthy, acceptable quality, and comfortable indoor conditions for occupants. Their main aim is to provide a safe and comfortable indoor environment for occupants while maintaining an efficient and reliable performance. However, they are subjected to failure that would affect their functionality and performance. For example, faults in HVAC systems can result in providing lower air quality, which would jeopardize the safety and health of occupants.

In addition, they would reduce energy efficiency by the faulty and unnecessary increase in energy usage. Moreover, faults can interfere in the execution of safety supervision scheme (e.g. building evacuation) and affect their effectiveness and correctness by resulting in executing crucial tasks based on faulty decisions [9]. Nevertheless, faults in the HVAC system cause components' wear which results in shortening their lifetime and increasing the maintenance cost.

Hence, Fault Diagnosis (FD) in HVAC systems becomes very essential in order to achieve the best system performance with minimum costs. It is important to develop effective and robust FD techniques for HVAC systems in order to identify the fault occurrence and source.

1.4 Fault Diagnosis Methods of HVAC Systems

Faults are defined as undesired deviations of the characteristic properties of the system from the standard conditions. They can be classified as sensor faults, actuator faults, and process faults (components or parameter faults). Faults may result in interruption of control action of the process controller, substantial measurement error, changing the dynamic properties of the system leading ultimately to degradation in the performance of the system or its breakdown and damage [10]. Fault diagnosis methodologies can be generally classified as: model-based methods, signal-based methods, knowledge-based methods, and hybrid methods as categorized in [10].

1.4.1 Model-based Methods

In model-based methods, the knowledge of the process models is required, which can be obtained using the physical principles or system's parameters identification approaches. Fault diagnosis algorithms are developed based on those models to monitor the consistency between the measured outputs of the real systems and the model-predicted outputs. They can be defined as follows:

1) The observer-based approach, in which the actual data of the system's states of interest is compared with the estimated ones by the observer. The generated error residuals are used to diagnose faults according to the detection thresholds. For deterministic

systems, Luenberger observers can be used while Kalman Filters (KFs) are employed when the system is subjected to noise that can affect the fault diagnosis procedure [11].

2) Parity relation approach where the residuals are generated by checking the parity between the estimated and the actual system measurements of the concerned outputs.

3) Parameter estimation using system identification methods (e.g., least square error, etc.) such that it is assumed that the faults are reflected on the parameters of the system. The fault diagnosis depends on the online estimation of the parameters which are then compared with the nominal ones under fault-free system operation.

1.4.2 Knowledge-based Methods

They require a considerable amount of historical data of the concerned system on which artificial intelligence techniques are applied to extract knowledge -quantitatively or qualitatively- about the system. The behavior of the actual system is compared against the knowledge-based model to produce the fault diagnosis decision [12]. They are defined as the following:

1) Expert system-based methods: They are rule-based approaches that are built based on human expertise. The evaluation is performed on the online monitored data based on the deduced rules. Even though they are easy to develop and use once the rules are at available, they are difficult to generalize and expand to variants of the system, e.g. in terms of type, size, etc.

2) Data-driven methods, which can be classified based on their analysis approach as statistical or non-statistical. The common statistical analysis used are Principal Component Analysis (PCA), Partial Least Squares (PLS), Independent Component Analysis (ICA), Support Vector Machine (SVM), and clustering analysis. On the other hand, the

non-statistical analysis-based data-driven fault diagnosis algorithms are Neural Networks (NNs) and fuzzy logic.

The data-driven FD approaches can be developed in supervised-learning, and unsupervised-learning manners. In the supervised-learning fault diagnosis methods, the knowledge of both the faulty and fault-free system data is required to build the model while in the unsupervised-learning, the knowledge is extracted from the unlabeled historical data of the system.

1.4.3 Signal-based Methods

Signal-based fault diagnosis methods work under the assumption that the faults can be linked to the measured signals features and patterns. Signals analysis such as frequency and spectral analysis are employed provided that the faults are reflected on the spectral features of the signals. In addition, signals measurements such as energy, current, etc. can be used as certain faults indicators.

The difference between the signal-based methods and data-driven methods is that the latter is based on analyzing an amount of historical data to deduce useful information about the behavior of the system of interest. However, the signals-based approaches do not require large amount of data but rather utilize the real-time signal measurements and employ signal processing techniques.

1.4.4 Hybrid Methods

In the hybrid method, a combination of their previously discussed methods is developed to make use of the distinctive advantages and strengths and to compensate for their limitations. Examples of hybrid methods are the use of parameter estimation with SVM, frequency analysis with PCA, Kalman filter with PCA, etc.

1.4.5 Characteristics of Fault Diagnosis Methods

There are a number of important features that are desired in the fault diagnosis methods which are the response time, efficiency, reliability, isolability, robustness, and low computation and modeling requirements [10]. Quick fault diagnosis is desired to prevent escalating the impact of the fault on the system components and to maintain the efficiency and reliability of the system. Moreover, the diagnostic method should be able to identify the location of the faults as well as to be robust to measurement noise and modeling uncertainty to avoid false fault alarms. The modeling and computational requirements of the fault diagnosis strategy are required to be minimal for high feasibility and ease of implementation of the method on real-life systems. In addition, high scalability of the fault diagnosis method is preferred in order to easily adapt it for variants of the system.

1.5 Literature Review on Fault Diagnosis in HVAC Systems

Fault diagnosis of HVAC systems is essential to provide preventive maintenance and to maintain the system's reliability and efficiency. Given the literature, most of the FD methods in HVAC systems are studied at the sub-system level (i.e., AHU, chiller system, etc.). The state-of-the-art diagnosis approaches used for the HVAC systems found in the literature are summarized in Fig.1.4.

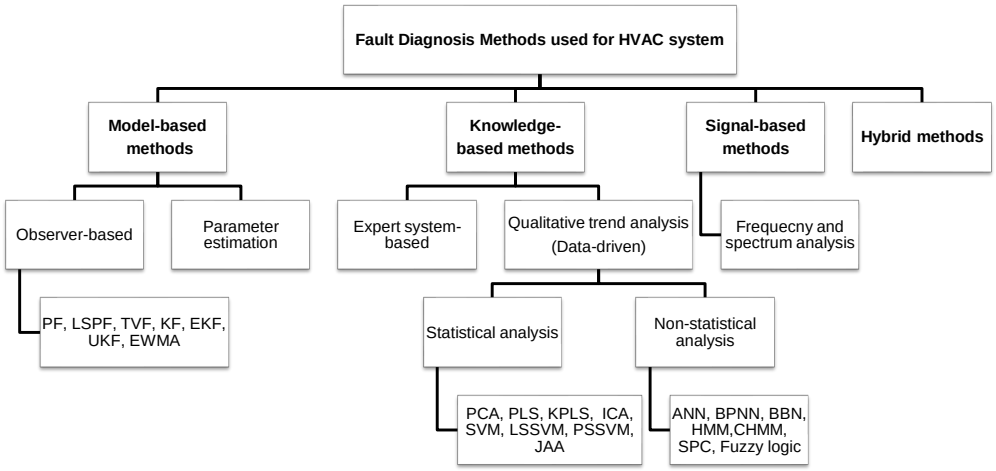


Figure 1.4: Summary of the fault diagnosis methods used for HVAC systems.

1.5.1 Model-based Methods

Model-based methods are used in developing fault diagnosis algorithms for HVAC system components, such as AHUs, chiller plants, etc. as summarized in Table 1.1. Authors in [13] and [14] present an unknown residual generation observer-based fault diagnosis algorithm for stuck dampers faults in VAV boxes in AHUs. The FD method is validated using MATLAB/Simulink on a 4-zone HVAC system. In [15], a nonlinear joint state-parameter observer-based method for damper position estimation in VAV boxes is proposed. It is implemented for stuck damper fault diagnosis such that the fault is reflected on the ratio of the air flow to each zone provided that a constant nominal flow rate supplied by the supply air fan. The method is tested using MATLAB on a simple two-zone HVAC system.

In [16], a model-based FD approach is proposed employing two state observers to perform fault detection and fault isolation. The first observer is comprised of discrete-time on-line approximator (OLAD) and a robust term while the other one includes the models of faults. The method is tested using simulation and the type of faults considered are degradation in cooling coil performance represented by an increase in the supply air

temperature, and leakage from conditioned space (or insulation degradation) emulated by the change in the zone air temperature.

In [17] and [18], a model-based fault diagnostic approach is presented for valve actuator faults in discharge air temperature (DAT) system using a combination of state and parameter estimation. In [18], a dual Extended Kalman Filter (EKF) is used while in [17], a joint Unscented Kalman Filter (UKF) is used for actuator faults and both are compatible with the nonlinearity in the HVAC system model. The actuator fault is modeled as a multiplicative fault and the FD methods are validated using MATLAB/Simulink.

In [19], a novel model-based on-line robust fault diagnostic approach is proposed based on Unscented Kalman Filtering and back-smoothing algorithm for typical faults in the chiller plants. The types of faults considered are process faults represented by reduced efficiency of components such as compressor which might result in excessive energy consumption and degrades its performance, and valve actuator faults resulting in reduced air flow rate. The FD strategy is tested and validated using simulation tools. In [20] and [21], an interactive multi-model augmented FD approach based on unscented Kalman Filter is developed for valve actuator faults and the fault modes generated by the possible perturbations in HVAC systems.

A model-based fault diagnostic algorithm is developed in [22] and [9] for sensor faults and in [23] for actuator faults in complex multi-zone HVAC systems on a distributed framework. Each subsystem is equipped with a dedicated local sensor as a fault diagnosis (LSFD) agent. The distributed sensor fault detection is conducted using robust analytical redundancy relations of estimation-based residuals and adaptive thresholds. Both sensor and actuator faults are examined for constant bias or offset

faults. The method is validated using simulation for faults in the temperature sensors and water valves.

The automated strategy proposed in [24] and [25] utilizes a Total Variation (TV) filter with an enhanced Particle Filter (PF) for faults detection in heat exchangers. The TV filter is used for transient occurrence detection and the Local Search Particle Filter (LSPF) is used for degradation tracking. The types of faults evaluated in the paper are condenser fouling, refrigerant leakage, non-condensable, and reduced water flow rate. The validation of the algorithm is carried out using simulation tools and experimental chiller faults data for different severity level of the four types of the fault.

Table 1.1: Model-based FD methods developed for HVAC systems.

Ref.	**Fault type**	**Approach**
[22], [9]	Sensor - Overall system	Observer
[23]	Sensor and actuator - Overall system	Observer
[16]	Process - AHU	Observer
[20], [21]	Actuator - DAT system	UKF
[17]	Actuator - DAT system	Joint UKF
[18]	Actuator - DAT system	Dual EKF
[14], [13]	Actuator - VAV boxes	Observer
[15]	Actuator - VAV boxes	Parameter estimation
[24], [25]	Process and actuator - Heat exchangers	TVF, LSPF, and KF
[19]	Process and actuator - Chiller system	UKF and BSA

1.5.2 Signal-based Methods

There are a few signal-based fault diagnosis methods for the HVAC system as summarized in Table 1.2. In [26], an electrical measurement-based approach is presented for fault detection of airflow blockage faults in AHUs and estimation of insufficient airflow employing induction machine diagnostic work. While in [27], a signal-based FD method is proposed for airflow blockage and unbalanced load of fan faults in the AHU based on frequency spectrum analysis. In addition, a fault detection strategy is presented in [28] for valves in AHUs employing frequency and spectral analysis as well.

Table 1.2: Signal-based FD methods developed for HVAC systems.

Ref.	**Fault type**	**Approach**
[27]	Sensor and process - AHU	Freq. spectrum analysis
[28]	Actuator - AHU	Freq. and spectral analysis
[26]	Sensor and process - AHU	Signal processing and analysis

1.5.3 Data-driven Methods

The majority of the fault diagnosis approaches proposed for the HVAC systems are data-driven given the high complexity of the system as well as the fact that modern buildings are equipped with building management and control systems (BMS) where records of the HVAC system data can be obtained and used in the development of the diagnostic strategies. The different data-driven methods used for HVAC system fault diagnosis that are found in the literature are summarized in Table 1.3.

In [29], a sensor FD algorithm for the AHU is discussed employing a Machine Learning technique systematically based on analyzing the system's behavior and com-

paring it against faulty conditions described by a set of behavioral patterns. Common faults are considered which are stuck damper/valve, coil fouling, and damper leakage. A novel fault diagnostic architecture for variable refrigerant flow (VRF) in AHUs is proposed in [30] based on Bayesian Belief Network (BBN) employing expert rule-based technique. The type of evaluated faults are refrigerant under-change and overcharge. The diagnosis method is tested using a practical VRF system under different cooling modes and operation conditions.

In [31], an ICA-BPNN-based fault diagnostic method is presented for refrigerant charge faults in the VRF systems combining Independent Component Analysis (ICA) and Back-Propagation Neural Network (BPNN) methods. The ICA is used for fault detection and the BPNN is employed to develop the fault diagnosis model. A data-driven fault diagnostic strategy for refrigerant charge amount (RCA) faults in the VRF systems is proposed in [32] by combining SVM mRMR-based feature selection and Wavelet Denoising (WD) algorithm. In addition, an enhanced Back-Propagation Neural Network (BPNN). While in [33], a fault diagnosis strategy is proposed for refrigerant charge amount faults -overcharge or undercharge- in the VRF systems utilizing PCA and dual ANN model.

In [34], an improved Decision Tree (DT)-based diagnostic approach is proposed for the VRF system. It combines a data-driven model and a virtual sensor-based fault indicator for faults detection. In [35], a fault diagnosis strategy is presented based on PCA and pattern matching using two factors for characterization of the similarity degree between the historical data and real-time data. While in [36] a FD method is proposed by integrating D-matrices and PCA, and in [37] an enhanced and novel sensor fault diagnosis strategy is presented using the Satizky-Golay (SG) and the PCA methods for

the VRF system, namely SG-PCA method. The SG algorithm is used for smoothing the data based on the least-squares polynomial approximation and the PCA model is trained for fault diagnosis using the smoothed data. In [38], an improved PCA-based with Joint Angle Analysis (JAA) approach is proposed for sensor faults diagnosis in the VAV system considering both fixed bias and drifting faults.

A novel HVAC fan machinery fault diagnosis approach is developed in [39] and [40] by combining Kernel Principal Component Analysis (KPCA) and Least Square Support Vector Machine (LSSVM). In [41], an unsupervised diagnostic strategy is presented employing cluster analysis for AHU sensor fault detection assuming that the faulty data is spatially and temporally separated from the normal data. The recorded history data of the system is pre-processed using PCA for dimension reduction. The clustering algorithm ordering points to identify the clustering structure (OPTICS) is used for data separation and type identification. The sensor faults are modeled as constant offset faults with different severity levels and the FD method is validated using TRNSYS simulation tool. In [42], a dual Neural Network (NN) strategy is proposed for AHU sensor faults detection in which the basic and auxiliary neural networks are developed then combined through allocating the weighting factors of the two neural networks using PCA. Two types of sensor faults are examined which are positive/negative fixed offset and drifting bias in addition to the case of complete sensor failure and a stuck water valve actuator fault. The FD method is validated using TRNSYS simulation tool. While in [43], a robust fault diagnosis method is developed by combining neural networks and subtractive clustering analysis for the same fault types as in [42]. The dual neural networks are used for fault detection and the subtractive clustering analysis is employed for fault diagnosis through adaptive classification of the faults' sources.

In [44], a FD method is presented employing SVM for common faults in the variable air volume conditioning (VAVC) systems in the HVAC system. The evaluated sensor and actuator faults are stuck chilled water valve, stuck VAV box damper fixed bias temperature sensor, and stuck recirculation damper. The FD strategy is tested using MATLAB/Simulink simulation model. While in [45], a novel fault diagnostic approach is developed based on the combination of Gaussian Process (GP) and SVM algorithms for AHUs. The GP algorithm is used to provide a probabilistic model to estimate the AHUs measurements as a function the external variables; time, occupancy, and the ambient temperature. Prediction error and variance are computed then its ratio is used in the SVM algorithm to develop the data-driven model that can be used to detect the faults. It is tested using TRNSYS simulation tool for a 46-zone building.

In [46], a self-adaptive sensor FD approach for an AHU is presented. It consists of two FD models, one for the AHU control loop including two Back-Propagation Neural Networks (BPNNs), and the other model is developed using both wavelet analysis method - for measurement data processing - and Elman neural network for sensor fault identification. The considered sensor faults are fixed bias, drifting bias, and complete sensor failure and the proposed FD strategy is validated using simulation. In [47] and [48], a dual FD approach is presented for VAV boxes in the AHU employing both ANN and fuzzy logic. The latter is utilized for fault detection, and the ANN is used for fault identification. The typical faults in AHU are examined which are damper fault, coil water valve error, sensor offset fault, and the fan fault. The validation of the FD method is carried out by simulating a multi-zone system in MATLAB/Simulink.

In addition, an automated FD strategy for sensors failure in the AHU is proposed in [49] employing Radius Basis Function (RBF) network and a combination of Genetic

Algorithm (GA) and pseudo-inverse matrix algorithm for Gaussian function parameters selection. A Deep Bayesian Networks (DBNs)-based approach is proposed in [50] and [51] for the diagnosis of common faults in AHUs. It uses the diagnostic information about the HVAC system in an information fusion way utilizing fault patterns recorded from multiple identical systems. A semi-supervised fault diagnostic algorithm is developed in [52] and [53] based on PCA statistical model along with a reconstruction-based approach. The PCA model is used for identifying healthy and faulty operation, and the reconstruction-based contribution approach is used for variables isolation, and decision tables for fault diagnosis.

In [54], a FD strategy is presented for chiller systems using Bayesian network classifier with probabilistic boundary in order to eliminate the problem of high false alarm rate. Typical faults in chiller systems are considered in the proposed FD method which are, the reduced condenser water flow, the reduced evaporator water flow, the refrigerant leakage, the refrigerant overcharge, the condenser fouling, the non-condensable gas in the refrigerant, and the excess oil for different severity levels and operating conditions. Additionally, a novel on-line FD method is developed in [55] based on Tree-structured Fault Dependence Kernel (TFDK) method for fault classification for the same types of faults as in [54]. While in [56], a two-stage fault diagnosis strategy is proposed adopting Linear Discriminant Dnalysis (LDA) for data dimension reduction and a distance-based classifier for the common faults in chiller systems.

In [57] and [58], the development of fault diagnosis approach is discussed using neural networks and statistical tools for HVAC chillers. The GLRT is used for faults detection while SVM, PCA, and PLS, are used for faults isolation. In addition, PLS is employed for faults severity estimation and chiller system modeling. Moreover, in

[57] an algorithm for optimal sensor suite selection for maximizing diagnosability is proposed. The same fault diagnosis strategy developed in [49] is proposed and applied for refrigeration units in the HVAC system in [59].

In [60], a fault diagnostic strategy using Artificial Immune Recognition System (AIRS) is proposed based on history and simulation classified system data for 13 common faults in the HVAC system. The method is validated using TRNSYS for a 10-zone building equipped with a full HVAC system. Where in [61], an online fault diagnosis framework is proposed based on the lattice-valued fuzzy automate such that the system is modeled as a fuzzy discrete event system. An online fault detection algorithm is presented in [62] utilizing Machine Learning techniques and HMM model such that the probabilistic relationships among variables of interest are analyzed and encoded. In addition, a sensor fault detection algorithm is proposed in [63] in which a combination of Probabilistic Principal Component Analysis (PPCA) models are used.

In [64], a FD approach is developed for HVAC systems employing Multi-Objective Clustering Rapid Centroid Estimation (MOC-RCE) technique which combines Rapid Centroid Optimization (RCE) algorithm and the Multi-Objective Clustering (MOC) paradigm. A clustering-based distributed sensor fault diagnosis algorithm is presented in [65] for Wireless Sensor Networks (WSN) monitoring HVAC systems, allowing the diagnosis of multiple sensor faults under the presence of disturbances and uncertainties.

In addition, in [66] a fault detection method is presented employing spectral clustering and PCA for better root cause fault detection and reduction of false alarms. Authors in [67] describe the application of a robust dual fault detection approach for HVAC systems under real-time working conditions using online SVM classifier combined with an ANN model. A data-driven FD method is developed in [68] combining BBN and rule-

based analysis for fault diagnosis in AHUs. The types of faults examined are process faults such as fouling, control signal error, sensor bias faults, and actuator faults, such as a stuck valve, stuck damper, etc.

In terms of the knowledge-based methods, an automated expert rule-based FD method is described in [69] for common faults in VAV boxes using mass and energy balance equations of various subsystems of the AHU. In addition, authors in [70] developed a novel semantic knowledge-based fault detection approach utilizing physical system knowledge encoded in a semantic graph to identify potential causes of faults.

Table 1.3: Data-driven FD methods developed for HVAC systems.

Ref.	Fault type	Approach
[45]	Sensor and process - AHU	SVM
[71]	Process - VRF	BPNN and ANN
[33]	Process - VRF	PCA and ANN
[42]	Sensor and actuator - AHU	NN and PCA
[43]	Sensor and actuator - AHU	NN and subtractive clustering
[46]	Sensor - AHU	BPNN and NN
[38]	Sensor - VAV boxes	PCA and JAA
[54]	Process - Chiller system	Bayesian network
[55]	Process - Chiller system	Tree-structured TFDK method
[35]	Sensor, process, and actuator - AHU	PCA and pattern matching
[41]	Sensor - AHU	PCA and clustering analysis
[29]	Process and actuator - AHU	Bayesian network
[57], [58]	Process - Chiller system	PCA, NN, SVM, PLC, GLRT
[52]	Process - Chiller system	PCA
[53]	Process - Chiller system	PCA and RBC
[39], [40]	Process - AHU	KPCA and LSSVM
[70]	Process and actuator - AHU	Expert-rule
[66]	Sensor and process - AHU	PCA and spectral clustering

Ref.	Fault type	Approach
[44]	Sensor and Actuator - VAV boxes	SVM
[64]	Process and actuator - AHU	MOC-REC clustering
[49]	Sensor and process - AHU	GA and RBF methods
[59]	Process - Refrigeration unit	GA and RBF methods
[47], [48]	Sensor, process and actuator - VAV boxes	ANN and Fuzzy logic
[67]	Process -AHU	SVM and ANN
[61]	Sensor, process, and actuator - Overall system	Fuzzy logic
[36]	Sensor, process, and actuator - AHU	PCA and D-matrices
[62]	Sensor, process, and actuator - AHU	DBN, HMM, and clustering analysis
[50], [51]	Sensor, process, and actuator - AHU	Bayesian networks
[37]	Sensor - VRF	SG-PCA method
[56]	Process - chiller system	LDC and distance classifier
[30]	Process - VRF	BBN
[31]	Process - VRF	ICA and BPNN
[32]	Process - VRF	SVM and WD

Ref.	Fault type	Approach
[63]	Sensor - Overall system	Parity relation and PCA
[34]	Process - VRF	Decision tree and VR-FI-based
[65]	Sensor - Overall system	Distributed clustering-based
[68]	Sensor, process, and actuator - AHU	BBN and rule-based analysis

1.5.4 Hybrid Methods

Several applications of hybrid fault diagnosis strategies for HVAC systems combining model-based and data-driven methods are proposed by researchers. Table 1.4 summaries the hybrid FD methods used for HVAC systems in the literature. In [72], SVM is applied to the model parameters that are found recursively by an autoregressive time series model with exogenous variables (ARX). The types of faults considered in the paper are actuator faults and process faults which are a stuck exhaust air damper, a stuck speed return fan, complete failure of return fan, an unstable cooling coil valve control, a stuck outside air damper (fully open or fully closed), a stuck cooling coil valve (partially open, or fully open or fully closed), outside air damper leak, and duct leak.

A hybrid FD strategy is proposed in [73] for cooling coil faults using Unscented Kalman Filter and Statistical Process Control (SPC). The types of faults tested are process faults due to fouling and accumulation of dust particles. The diagnosis method is evaluated using simulation tools. Similar to [73] but on the system-level, UKF and SPC are used in [74] for robust fault detection with potentials for large-scale implementation.

In [75], a novel hybrid FD technique in the AHU is proposed considering fault propagation impacts among components. The fault diagnostic strategy combines both filtering technique - Kalman filter and Unscented Particle Filter (UPF)-, and Hidden Markov Model (HMM) which is a statistical model representing the relationships between hidden states and observations. Statistical Process Control (SPC) is employed for process monitoring and control. Sixteen typical types of faults occurring in AHUs are considered including process and actuator faults which are, a stuck exhaust air damper (closed/open), a stuck outside air damper (closed/leakage), cooling coil tube fouling, dust on cooling coil fins, a stuck water valve (closed/open), duct leakage at different parts of the system (before/after supply fan), complete fans failure, and reduced fan efficiency. The FD method is validated using a simple two-zone HVAC model simulated using MATLAB/Simulink.

In [76], a FD strategy is proposed for the AHU and multiple VAV boxes. The developed algorithm utilizes dynamic HMMs, UPF, and SPC for fault identification in the cooling coil, fans, and VAV boxes. The difference between [75] and [76] is that the former emphasis is on fault propagation impact while in [76], the proposed strategy highlights diagnosing faults of many VAV boxes with minimum computational effort and the algorithm used for sensor faults detection.

In [77], a robust FD strategy is presented for multiple faults in AHU by using a rule-based and residual-based Exponentially Weighted Moving Average (EWMA) control chart method. EWMA is used for fault detection and the expert rule-based method is used for simultaneous multiple faults isolation in the VAV boxes. The types of faults considered are pump fault, cooling coil fouling, cooling coil valve leaking, a stuck cooling coil valve, cooling coil valve controller failure, filter failure, fan failure, temperature

sensor fault, pressure sensor fault, and incorrect chilled water temperature settings. It is validated using operational data of a 36-office building that is fully equipped with an automated energy management and control system.

In [78], a hybrid approach is proposed for on-line fault detection in the chiller plants. It is based on Particle Filter (PF), joint nonlinear Bayesian-based estimation, and kernel-based partial least square method. Moreover, a hybrid fault diagnostic approach is proposed in [79] combining EKF with coupled Hidden Markov Model (CHMM) in the chiller plants. The typical fault is taken into consideration which is the pipe fouling and the leakage. The fault diagnostic method is tested using EnergyPlus simulation tool for a simple two-zone building equipped with a full HVAC system.

In [80], a hybrid fault diagnostic technique is proposed in which the interdependencies among the different HVAC subsystems is considered and the algorithm can easily be adapted to dynamical operational conditions and different building configurations. The proposed system-level FD approach utilizes knowledge about the HVAC system's modules interdependency, the energy and mass conservation laws, and the historical data of the system in developing the diagnostic model.

In [81], a combined system-level and sensor fault FD strategy is proposed for the HVAC systems comprising of two schemes. A system-level fault diagnosing scheme performance indices (PIs) are introduced to determine the performance status of each system as faulty or healthy. The other scheme is for bias sensor fault diagnosis using PCA. Similarly, a hybrid system-level fault diagnostic approach is proposed in [82] for the low delta-T syndrome in the AHU and heat exchanger due to performance degradation in complex HVAC systems. The HVAC system measurement data are processed online to generate the performance indices and the reference models are developed

offline using the regression model which are used in determining the benchmarks of the PIs. Adaptive residuals are calculated using the t-statistic approach to identify the healthy conditions of the performance indices. Additionally, in [83] a fault diagnosis method is proposed for low delta-T syndrome resulted from the cooling coil fouling in heat exchangers. The method uses defined performance indices and adaptive thresholds in order to improve the prediction uncertainty of the reference models. The proposed method is validated using simulation tool.

A novel fault diagnosis strategy is developed in [84] based on parameter estimation utilizing a recursive least-squares-based algorithm to detect faults in the HVAC system operation. A time-varying ARX/ARMAX model is formulated based on the parametric model found from the recursive identification for the types of fault studied. The algorithm is tested for the common fault type occurring in different components of the HVAC system using simulation tools.

In [85], a fault detection and isolation architecture for the VAV boxes is developed based on an integrated statistical and linear model-based framework. A statistical model is developed employing PCA and joint angle analysis for determining benchmarks and a linear model-based framework is designed for FDI of multiple actuator and multiple sensor faults. Additionally, the algorithm includes a safe parking strategy for energy savings purposes. A cross-level FD technique is proposed in [86] for complex HVAC systems. It is based on energy performance monitoring of the HVAC system units, and temporal and spatial partition. Energy utilization is observed in groups and at different levels over well-defined time periods and consequently suitable thresholds are defined for the purpose of fault detection and identification. PCA is used to identify the correlations between system variables.

In [87], a hybrid FD method is proposed combining a rule-based classifier with signal-based CUSUM control chart for sensor faults, process faults, and actuator faults in VAV boxes. In [88], a novel FD strategy is presented based on exponentially weighted moving average charts and Shewhart charts that are compared to a breakout detection algorithm to diagnose faults in the HVAC system. In addition, the method is tested using simulation tools and artificial faults are injected and used for validation of the statistical techniques used.

Table 1.4: Hybrid FD methods developed for HVAC systems.

Ref.	Fault type	Approach
[84]	Process - Overall system	RLS and parameter estimation
[80]	Process and actuator - AHU	Bayesian network and model-based method
[72]	Process and actuator - AHU	ARX model and SVM
[75]	Process and actuator - AHU	KF,UPF, HMM, and SPC
[76]	Process and actuator - AHU	UPF, HMM, and SPC
[74], [89]	Sensor and process - Overall system	UKF and SPC
[73]	Process - AHU	UKF and SPC
[79]	Process - Chiller system	EKF and CHMM
[78]	Process - Chiller system	PF, parameter estimation, and KPLS
[77]	Sensor, process, and actuator - AHU	(EWMA) control chart method and expert rule-based method
[81]	Sensor, process and actuator - Overall system	PCA and performance indices
[82], [83]	Process - AHU	SPC and parameter estimation
[85]	Sensor and actuator - VAV boxes	PCA, JAA, KF
[86]	Process - Overall system	Signal-based and PCA
[87]	Sensor, process, and actuator - VAV boxes	Rule-based classifier with CUSUM control chart
[88]	Process - AHU	EWMA and Shewhart charts

By reviewing the previous works on fault diagnosis in HVAC systems, limited research works cover sensor and actuator faults diagnosis and the matter of sensor data validation and reconciliation is not investigated thoroughly. The focus of the majority of the previous works is on the process faults occurring in the chiller system specifically. This is justified given the impact of the chiller system-related faults on the HVAC system performance and efficiency. Nevertheless, we think that further research in sensor and actuator faults diagnosis is indispensable for the following reasons. The sensors measurements are crucial for the HVAC system operation (i.e. the closed-loop control, the execution of supervision scheme, etc.) and hence the sensors data must be reliable and fault-free. In addition, the actuator faults can alter the HVAC system in terms of maintaining occupants comfort as a result of failing to fulfill the thermal load requirement, system maintenance (i.e. ducts damage or leak due to increased pressure due to damper blockage), and potentially the energy consumption.

Most of the model-based approaches found in the literature are developed for sensor and actuator fault diagnosis. They can diagnose unknown faults utilizing limited knowledge about the system (i.e. the mathematical model), but they require an accurate, detailed, and explicit model depicting the system's operation. Hence, the performance of the fault diagnostic algorithm depends on the model's accuracy. This limits the scalability and adaptability of the diagnosis methods when considering larger and variants buildings. On the other hand, even though the signal-based HVAC fault diagnosis approach does not require the system mathematical model, it is concerned with the major characteristics of the output signals for diagnosis with less attention to system dynamics resulting in potential degradation of the diagnostic approach performance in the presence of unknown disturbances.

On the other hand, the rule-based FD method requires the expert knowledge and it may perform poorly under undefined conditions by the set of the diagnosis rules. Moreover, data-driven HVAC fault diagnosis methods are commonly used as shown in the previous sections. Their main advantage is that the system's model is not required. Hence, they better suit applications where systems models are unavailable, complicated, or challenging to derive such as complex HVAC systems but their performance is tied to the amount and quality of the training data used. More specifically, the use of Machine Learning algorithms such as PCA, SVM, neural networks, etc. in developing HVAC fault diagnosis approaches is evolving. In addition, the newly emerging field of deep learning demonstrates notable capabilities in pattern recognition and health monitoring. It has not yet been applied for HVAC system fault diagnosis, but it has a wide range of similar applications with promising performance. That is, the learning-based methods are characterized by the following features: scalability, reliability, high performance accuracy, and low false alarms rate which are the desired characteristics in the HVAC fault diagnosis system. They can be developed by making use of historical data from the building management system.

1.6 Thesis Objectives

Researchers have been investigating various solutions for fault diagnosis in the HVAC system as demonstrated by the literature survey. However, the improvements are still not keeping up with the growing sector and the increasing global demand for energy. The main objective of this work is to develop an effective fault diagnosis method as to improve the efficiency and reliability of the HVAC system that can be applied to variants of the systems and making use of the availability of system historical data from

the BMS.

This is tackled by proposing two data-driven fault diagnosis methodologies for sensor and actuator faults. Firstly, a sensor data validation and fault diagnosis strategy is developed given that sensors measurements are crucial in the closed-loop control operation of the HVAC system. It aims to promote the HVAC system operation becoming invulnerable under sensor faults and degradation. In addition, a novel and robust CNN-based actuator fault diagnosis is developed to reliably diagnose dampers and valves faults in the HVAC system with high accuracy.

1.7 Thesis Contribution

The contribution of this thesis work can be summarized as follows:

- Two data-driven fault diagnosis strategies are developed for HVAC system sensor and actuator fault diagnosis that do not require the knowledge of the system mathematical model and hence avoiding the complications that can arise due to modeling approximation and uncertainties. The methods are developed utilizing the building's historical data that can be obtained from the modern building management systems.

- A semi-supervised on-line sensor fault diagnosis method for HVAC systems is proposed using an Auto-Associative Neural Network (AANN) for both single and multiple faults utilizing only the healthy sensors data. Due to the network structure, the method is capable of sensor data validation in terms of replacing missing sensor measurements with the validated data, correcting inaccurate or faulty data, and filtering out measurement noise.

- The proposed AANN-based sensor data validation and fault diagnosis technique serves as an effective method to validate sensor measurements for the closed-loop control of the HVAC system to maintain its reliable performance. It can be applied for large-scale buildings with additional zones and it is expected that the performance will be improved with the increase in the HVAC system size due to a higher correlation between the inputs. The proposed AANN-based approach is suitable for applications where sensors measurements are broadly steady over the operation period which is the case with HVAC systems.

- A novel supervised on-line actuator fault diagnosis for HVAC systems using 2D Convolutional Neural Networks (CNNs) is proposed based on a data transformation to obtain the 2D representation of the raw 1D measurements data of the HVAC system variables without advanced data pre-processing requirement. The CNN-based actuator fault diagnosis method demonstrates an improved performance when compared with the commonly used methods and it addresses the limitations of the past works in terms of the method's accuracy, reliability, and precision.

1.8 Thesis Organization

The remainder of this thesis report is organized as follows. In Chapter 2, the details of the building and the HVAC system under this study are presented along with the description of the simulator design using TRNSYS in Sections 2.1 and 2.2 respectively. The simulator details are demonstrated in Section 2.2.1 and simulation results are presented in Sections 2.2.2 and 2.2.3 for the normal operation and in case of actuator faults. In addition, in Chapter 3 the proposed sensor data validation and fault diagnosis

approach using the Auto-Associative Neural Network is described. An overview of the theory of Back-Propagation Neural Networks in provided in Section 3.1 and in Section 3.2, the theory of the Auto-Associative Neural Network is presented in addition to the details of the network training procedure and the performance evaluation metrics used. Section 3.3 presents the evaluation results and discussion of the sensor data validation method while the performance of the proposed approach in sensor fault diagnosis is demonstrated in Section 3.4 and a comparison between the proposed AANN-based scheme and a PCA-based method is presented in Section 3.5. A summary is presented in Section 3.6.

In Chapter 4, a Convolutional Neural Network-based fault diagnosis scheme is developed for actuator faults in HVAC systems. Section 4.1 presents the theory of the Convolutional Neural Networks and in Section 4.2 the training specification, procedures, and performance metrics are described. The details of the proposed diagnosis framework are explained in Section 4.3 and the evaluation results of the developed scheme are demonstrated in Section 4.4. In addition, a comparison between the convolutional neural network-based approach and other commonly used methods for HVAC systems fault diagnosis is conducted is Section 4.5. A summary is presented in Section 4.6. Finally, Chapter 5 provides a summary of the final conclusions of this thesis work presented in this report in addition to future work.

Chapter 2: HVAC System Description and Simulation

This chapter presents the details of the system under this study which is a 3-zone office building equipped with a simple HVAC system for the cooling application. The detailed description of the building is presented in Section 2.1. As discussed in Section 2.2, the HVAC system simulation is carried out using TRNSYS. The simulator details are demonstrated in Section 2.2.1 and the simulation results are presented in Section 2.2.2. In addition, the actuator fault emulation method is presented in Section 2.2.3 and a demonstration of actuator faults results is provided.

2.1 Building Description

The building considered for this work is a one-floor office building presented in [1]. It is assumed to be located in Doha city in Qatar and operating from 6 AM to 6 PM during weekdays (Sunday to Thursday). It is composed of three zones with a total floor area of 200 m^2 as shown in Fig.2.1. Zones 1 and 3 are of a volume of 240 m^3 and a floor area of 80 m^2 each and the first one includes a meeting room and a reception room while the latter has four identical office rooms. The main hall is considered as a separate zone with a volume of 120 m^3 and a floor area of 40 m^2. The zoning of the building is based on the thermal load requirement of the adjacent rooms. That is, the rooms that have approximately the same cooling requirement are considered to be within a distinct zone. In the simulator, each room has a separate temperature sensor and the temperature of each zone is the average of the rooms' temperatures in that zone. In addition, the building's structure and geometry, occupancy, internal equipment, and lighting are considered in the simulator design with the details presented in Table 2.1 and Table 2.2.

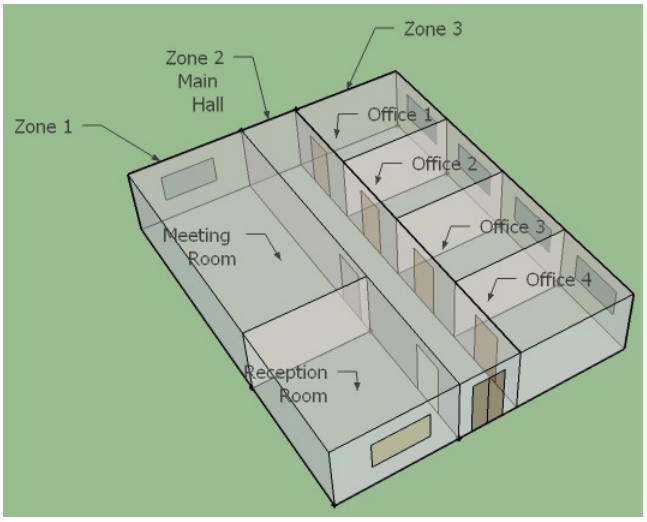


Figure 2.1: Sketch of the simulated 3-zone building.

Table 2.1: Characteristics of the building wall materials [1].

Component	Layer	Thickness(cm)	Thermal conductivity (kJ/h/m/K)	Density (kg/m^3)	Thermal capacity (kJ/kg/K)(kJ/h/m/K)	U-value (W/m^2/K)
Exterior wall	Mortar	2	4.152	2000	0.84	5.338
	Hollow brick	20	0.75	664	0.74	0.200
	Cement plaster	2	4.152	1700	1	5.26
Ground	Stone	20	6.12	2095	1	3.476
	Concrete	12	7.2	2450	1	4.348
	Cement screed	7	3.6	1700	1	4.167
	Tiles	1	4.68	2300	0.84	5.686
Roof	Mortar	2	4.152	2000	0.84	5.338
	Concrete	14	7.2	2450	1	5.650
	Cement screed	3	3.6	1700	1	5.000
Interior wall	Mortar	2	4.152	2000	0.84	5.338
	Hollow brick	7	0.705	938	0.741	1.897
	Mortar	2	4.152	2000	0.84	5.338

Table 2.2: Internal heat gain sources.

Room	Number of occupants	Occupation time	Details
Meeting room	4	Weekdays: 8AM-9AM	Person: Seated, very light work, 120 W
			Lights: 5 W/m^2
Reception room	1	Weekdays: 6AM-6PM	Person: Seated, light work, typing, 150 W
			Computer: 140 W
			Lights: 5 W/m^2
Main hall	4	Weekdays: 6AM-6PM	Person: Standing, light work, 120 W
			Lights: 5 W/m^2
Office room	1	Weekdays: 6AM-6PM	Person: Seated, light work, typing, 150 W
			Computer: 140 W
			Lights: 5 W/m^2

2.2 HVAC System Simulation

The system is simulated using Transient System Simulation Tool (TRNSYS) which is a graphical software environment that allows the simulation of transient systems behavior through energy and mass balance equations [90]. The simulation tool is used to estimate actual dynamics of the HVAC system components while taking into consideration the building geometry and structure, internal loads, weather conditions, and ventilation rate. TRNSYS has been widely used for HVAC systems simulation for research and development purposes as in [41–43, 45, 60, 81] and it is found to be a reliable tool to simulate the system operation.

The main variables of the system are as follows, the zones temperatures T_{z1}, T_{z2}, and T_{z3}, the chilled water tank temperature T_t, the AHU supply air temperature T_{ao}, the return water temperature from the cooling coil to water tank T_{wo}, the ambient temperature T_{amb}, the zones VAV boxes control signals U_1, U_2 and U_3, and the water tank valve control signal U_4.

2.2.1 Simulator Design

The building is equipped with a simple HVAC system for cooling application in which only the temperature at each zone is controlled using proportional integral derivative (PID) controllers. The zones are supplied from the AHU with cold air of constant temperature of 13 °C and a variable flow rate controlled by the VAV box terminals. The water chiller and the cooling coil are connected via the chilled water tank. The water tank temperature is controlled at 11 °C via a water valve regulating the flow of chilled water from the chiller to the tank.

In the simulation software, the components of the HVAC system are simulated as shown in Figure 2.2. The cold output air of the AHU is supplied through supply air ducts and a supply air fan to the zones ducts with the VAV boxes terminals. The zones temperature controllers modulate the position of the air dampers according to the thermal load of the zones and the variation in weather conditions to achieve the desired setpoints. The zones return air is fed back to the AHU through the return air ducts using the return fan. The exhaust air dampers (EA), outside air (OA) dampers, and return air (RA) dampers are operated simultaneously as to control the fraction of the recirculated air and the ventilation air in order to maintain the indoor air quality.

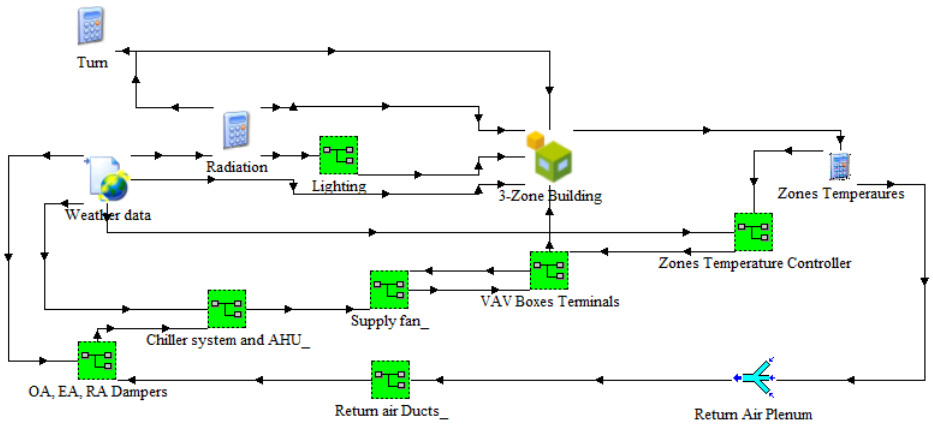


Figure 2.2: HVAC system simulation using TRNSYS software.

2.2.1.1 VAV Boxes

The VAV boxes are modeled as variable speed fans as shown in Fig.2.3. The output of the supply air fan is fed to the plenum and the plenum air proportion are determined based on the control signals emulating the amount of the VAV openings.

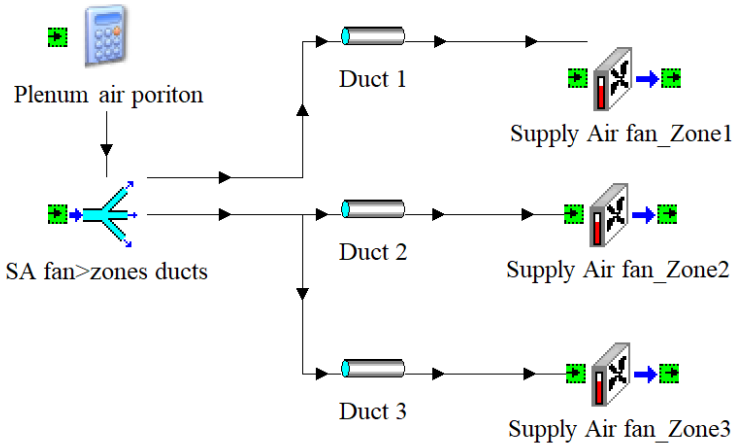


Figure 2.3: Simulation block of the VAV boxes terminals.

2.2.1.2 PID Controllers

As shown in Fig.2.4, three PID controllers are used to control the three zones temperatures based on the specified setpoints. Table 2.3 presents the PID parameters used in the simulation. As per specified by the developers of TRNSYS, there are special considerations associated with the PID controller model [91]. Unlike most of TRNSYS's controller components, this PID controller model may be used in heating, cooling, and combined heating and cooling applications based on the setting of the proportional gain parameter. A negative proportional gain must be used if the controller is used for cooling-only or combined heating and cooling applications. For instance, given that the zone temperature is the controlled variable and the flow rate of the fan is the modulated parameter, the flow rate should increase when the tracking error (the difference between the setpoint and the controlled variable) decreases meaning that more cooling should be provided when the temperature rises above the setpoint. Hence, the setting of the controller should be a negative gain constant and the minimum and maximum control values of the control signal should be set to 0 and 1 that is proportional to the maximum flow rate value.

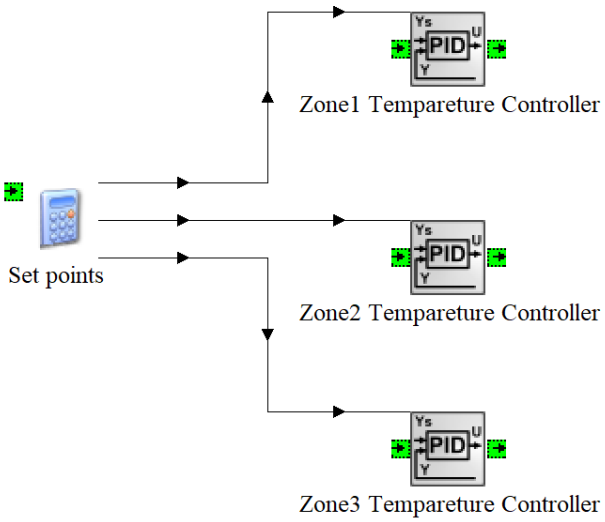


Figure 2.4: Simulation block of the zones temperature controllers.

Table 2.3: Zones temperature PID controllers parameters used in the simulator.

Gain constant	Integral time (hours)	Derivative time (hours)
-0.3	1.5	0.001

2.2.1.3 Chiller System and AHU

Figure 2.5 presents the chiller system and the AHU simulation block. The chiller has a capacity of 300 kW and a coefficient of performance (COP) of 2. The temperature of the chiller outlet water is 7 °C. The amount of water flowing to the chilled water tank is controlled by a variable speed pump simulating the water valve to regulate the tank temperature at the desired setpoint. The tank temperature PID controller specification is presented in Table 2.4. A constant speed pump is used to maintain the water flow rate through the cooling coil at 3 kg/s. The cooling coil model in TRNSYS has an inherent outlet air temperature controller such that the desired value can be set among the other model parameters.

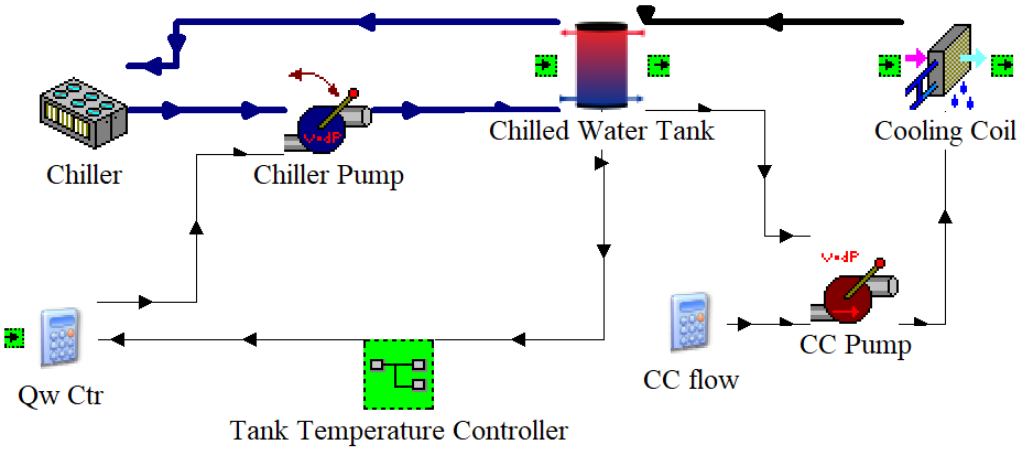


Figure 2.5: Simulation block of the chiller system and AHU.

Table 2.4: Tank temperature PID controller parameters used in the simulator.

Gain constant	Integral time (hours)	Derivative time (hours)
-0.5	1	0.001

2.2.2 Simulation Results

The controlled variables of the system are the zones temperatures upon the set-points. The zones setpoints are ranging between 18 - 26 °C based on the cooling load requirement which depends on the operation's time and day such that it is low in the daytime during working days and it is set at a higher value at night and during week-ends. As mentioned previously, the AHU output air temperature is constant at 13 °C and the water tank temperature is controlled at 11 °C by modulating the control signal of the chiller pump. Figures 2.6 and 2.7 demonstrate the simulation results of the normal system operation for one working day in August in which the three zones temperatures are controlled at 22 °C, 20 °C, and 20 °C at daytime, and at 26 °C, 24 °C, and 25 °C during the night respectively. The zones temperatures are well controlled according to

their setpoints with an average settling time of 40 minutes. At night, the control signals are at their minimum values because the cooling load is low and they peak at 6 AM at which the cooling load requirement increases before the zones temperatures are stabilized at the desired values. That is, when the temperature setpoint is high, the VAV air damper becomes fully closed (i.e. almost zero control signal) and vice versa.

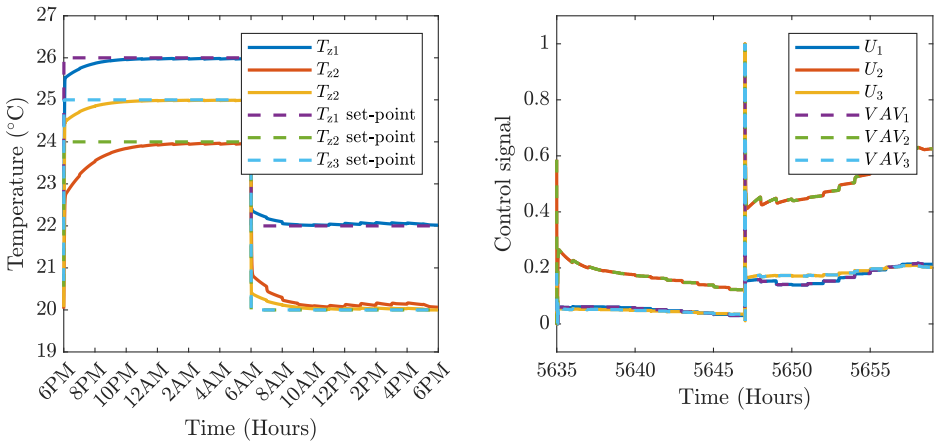


Figure 2.6: System simulation results of the zones temperatures and the VAV boxes control signals under normal operation.

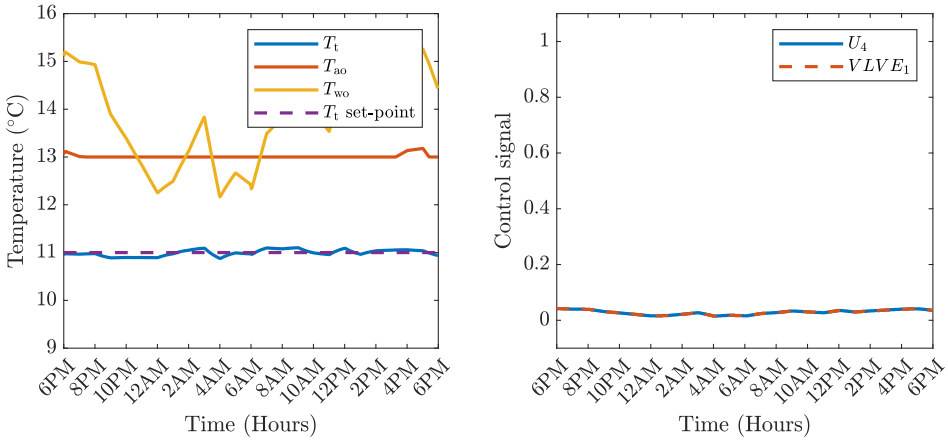


Figure 2.7: System simulation results of the tank, AHU output air, and coil return water temperatures and the valve control signal under normal operation.

2.2.3 Actuator Fault Emulation

An actuator is a component responsible for moving or controlling a mechanism or system such as a valve, a damper, etc. As shown in Fig.2.8, the controller determines the required control input $u(t)$ based on the real-time measurement of the state $x(t)$ and the desired setpoint x_{d} to achieve the output $y(t)$ from the damper or the valve. The control input $u(t)$ is received by the actuator which actuates the damper or valve opening accordingly.

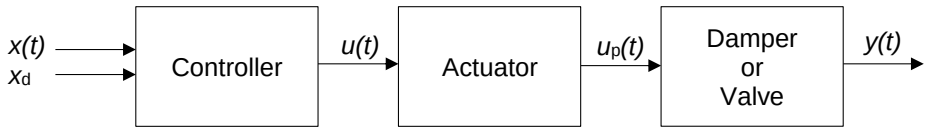


Figure 2.8: Block diagram of a healthy actuation system.

A fault in the actuator results in wrong positioning of the damper or the valve as shown in Fig.2.9 provided that $u_{\mathrm{p}}(t)$ is the damper actuating signal under normal operation, $\overline{u}_{\mathrm{p}}(t)$ is the faulty damper actuating signal, and $f(t)$ is the amount of fault introduced to the actuating signal.

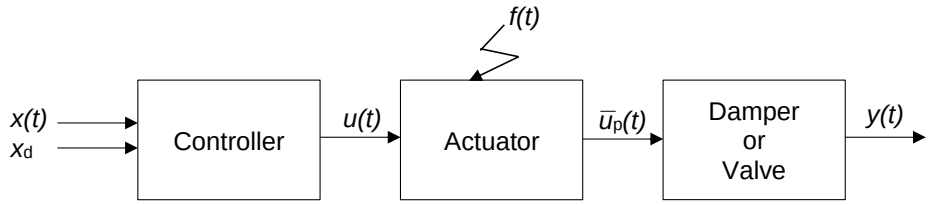


Figure 2.9: Block diagram of an actuation system subjected to actuator fault.

Let the time at which the fault occurs be t_{f}, then the stuck fault in the air damper (or water valve) actuator can be expressed as:

$$\overline{u}_{\mathrm{p}}(t) = u_{\mathrm{p}}(t_{\mathrm{f}}), \quad \text{for} \quad t \geq t_{\mathrm{f}}. \tag{2.1}$$

In terms of simulating the actuator fault, assuming that the valve has linear opening characteristics, the relation between the flow rate and the damper position (or valve opening) can be assumed to be linear [92] as presented in Fig.2.10.

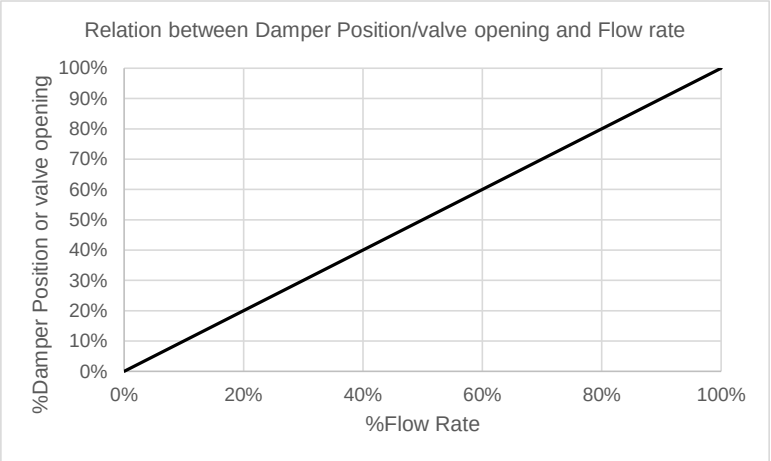


Figure 2.10: The relation between the flow rate percentage and the percentage of damper position (or valve opening).

Moreover, the relation between the control input $u(t)$ and the damper position (or valve opening) can be represented by a linear relation as in equation (2.2) given the specification provided by the manufacturer on the magnitude of the control signal as:

$$u(t) = A\frac{Q^{\mathrm{d}}(t)}{Q_{\mathrm{rated}}(t)}, \tag{2.2}$$

where A is the amplitude of the control signal, $Q^{\mathrm{d}}(t)$ is the desired flow rate, and $Q_{\mathrm{rated}}(t)$ is the rated flow rate.

Hence, the actuator faults are emulated by modifying the VAV damper (or the valve) opening value. That is, in the normal system operation the VAV damper (or valve) opening corresponds to its control signal. Under a stuck actuator fault, the VAV damper (or valve) opening freezes at the control signal value at the instant of the fault occurrence. Two fault scenarios are considered as illustrated in Figures 2.11 to 2.14. The

variables VAV_1, VAV_2, VAV_3, $VLVE_1$ represent the four actuators' openings. The first one shown in Figures 2.11 and 2.12 is a low impact fault occurring in the VAV 1 damper that is stuck at 10% opening position during the night operation at 12 AM. It can be observed that the system is unable to control zone 1 temperature at the desired setpoint and the effect of the fault increases during the daytime in which the cooling load is higher. The control signal U_1 is at its maximum value but the damper opening is stuck in the same position since the fault occurred (12 AM). The second actuator fault case demonstrated in Figures 2.13 and 2.14 is the stuck water valve at a fully closed position at the end of the night operation (a little before 6AM). This fault is of high severity level such that it impacts all of the system's variables. It is observed that the zones temperatures, AHU output air temperature, coil return water temperature, and water tank temperature are diverging from their normal values and they follow the ambient temperature. On the other hand, the control signals are all at their maximum values to attempt mitigating the wrongful behavior of the system.

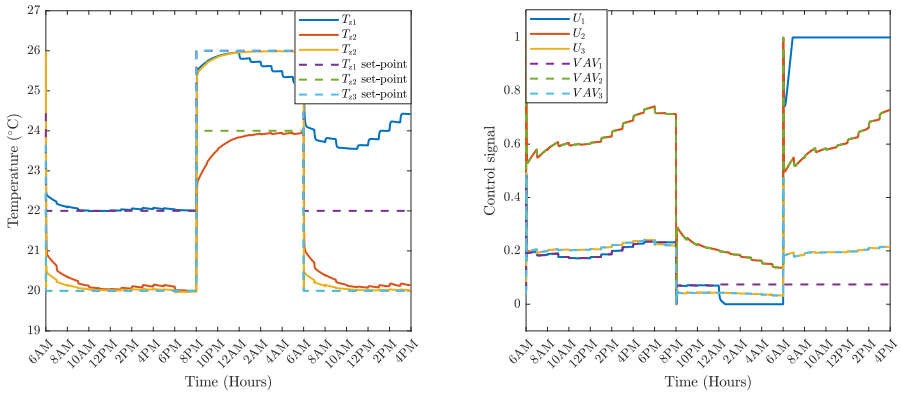


Figure 2.11: System simulation results of the zones temperatures and the VAV boxes control signals under fault in VAV 1 damper.

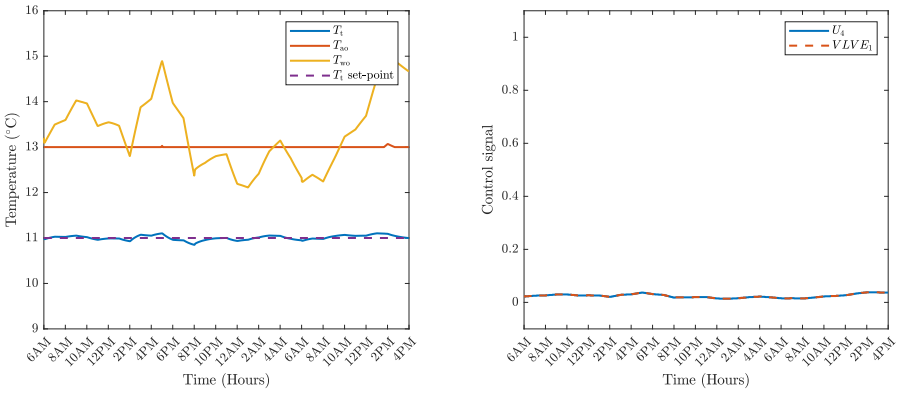


Figure 2.12: System simulation results of the tank, AHU output air, and coil return water temperatures and the valve control signal under fault in VAV 1 damper.

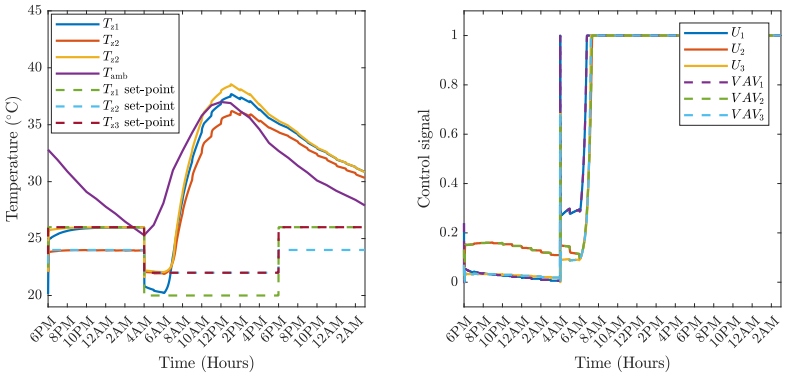


Figure 2.13: System simulation results of the zones temperatures and the VAV boxes control signals under fault in the water valve.

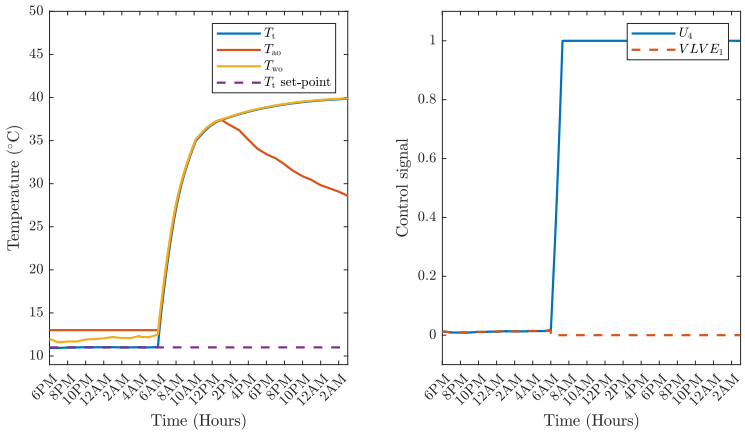


Figure 2.14: System simulation results of the tank, AHU output air, and coil return water temperatures and the valve control signal under fault water valve.

Chapter 3: Sensor Data Validation and Fault Diagnosis using AANN

Sensors measurements are the fundamental inputs of fault diagnosis methods and their reliability is crucial for the closed-loop control of the HVAC systems. The performance of the HVAC control system can be perturbed due to sensor faults or degradation. Hence, sensor data validation and fault diagnosis is a key preliminary step prior to the closed-loop control scheme in which contaminated data are identified and replaced with estimated reliable data.

There are several sensor fault diagnosis methods that have been proposed and applied for HVAC systems. They can be categorized mainly as model-based [9] and data-driven [35–38, 41–51, 60, 63, 65, 81, 85] methods. As mentioned previously, model-based FD methods need comprehensive mathematical modeling which can be costly, time-consuming, computationally demanding, and inaccurate for complex HVAC systems. The FD method performance can be degraded due to modeling approximation and uncertainties.

The main shortcoming of the aforementioned data-driven methods proposed in the literature is that the availability of sufficient amount of faulty data is essential for sensor fault identification and isolation as in [35, 38, 41–45, 47, 48, 60] which is generally not available in the practical systems. While the methods that do not require the faulty data [37, 63, 65, 81, 85] work under the assumption that the healthy and faulty data are spatially and temporally separated which might not be valid in complex sensor fault scenarios. In [46], even though only the data of the normal operation of the system is required to develop the neural network-based sensor fault diagnosis approach, it is based on a multi-model neural network to identify and isolate sensor faults which increases

the computational requirement. Moreover, there is no highlight of the possibility of using the methods for sensor data validation and correction.

This chapter presents a data-driven method for both sensor data validation and fault diagnosis in HVAC systems using an Auto-Associative Neural Network (AANN) to overcome the aforementioned shortcomings. The algorithm is based on applying the theory of Back-Propagation Neural Networks along with nonlinear dimensionality reduction to construct an input-output mapping model in which the network inputs are the sensors measurement data. Table 3.1 summaries the features of the sensor data validation and fault diagnosis methods found in the literature and our proposed AANN-based approach.

Table 3.1: Summary of the literature review in data-driven sensor fault diagnosis for HVAC systems.

[#]	Method	Data source	Features							
			HDT	FD	FI	MF	MDR	FC	NF	INC
[45]	SVM	Simulation (TRNSYS)	✓	✓						
[93]	SVDD	Operational	✓	✓	✓					
[94]	ICA	Operational	✓	✓	✓			✓		
[63]	Probabilistic PCA	-		✓						
[38]	PCA and JAA	Simulation	✓	✓	✓	✓				
[35]	PCA & pattern matching	Experimental		✓		✓				
[41]	PCA & clustering analysis	Simulation (TRNSYS)	✓	✓						
[37]	SG-PCA	Operational	✓	✓	✓					
[81]	PCA	Simulation (TRNSYS)	✓	✓	✓			✓		
[95]	PCA and AFT	Simulation (Modelica)	✓	✓	✓					
[96]	PCA & clustering	Operational	✓	✓	✓					
[97]	PCA	-	✓	✓	✓			✓		
[98]	PCA	Operational	✓	✓				✓		
[46]	BPNN & Elman NN	Simulation		✓	✓	✓				
[42]	NNs	Simulation (TRNSYS)		✓	✓					
[43]	NNs & clustering	Simulation (TRNSYS)		✓	✓					
[48]	Fuzzy NN	Simulation (Simulink)		✓	✓					
[99]	NN & fractal analysis	Simulation		✓						
-	AANN (Proposed method)	Simulation (TRNSYS)	✓	✓	✓	✓	✓	✓	✓	✓

HDT: Training with healthy data only

FD: Fault detection

FI: Fault isolation

MF: Multiple faults

MDR: Missing data replacement

FC: Fault correction

NF: Noise filtering

INC: Inaccuracy correction

The use of Auto-Associative Neural Networks for sensor data validation and fault diagnosis is introduced for the first time in [100] and later on applied to different complex systems [101–107], for measurement noise filtering, sensor fault diagnosis,

and sensor data validation and correction. In [101, 102], a reliable and enhanced on-line AANN-based monitoring method is presented for simulated boiling water reactor (BWR) component and in [104] an AANN is used for sensor data validation and for multiple sensor and component faults diagnosis in gas turbine engines. A sensor fault diagnosis and reconstruction method using AANN is presented [103] for the engine control system. In the Bio-engineering field, AANN-based online fault detection in Virginiamycin production process is proposed in [105] and an online fault diagnosis of α-Amylase production process is presented in [106] using an Auto-Associative Neural Network. In addition, AANN is used for industrial bioprocesses condition monitoring in [107].

The chapter is organized as follows, in Section 3.1, an overview of the theory of neural networks in presented. While in Section 3.2, the theory, mathematical principles and training details of the Auto-Associative Neural Networks is described. The evaluation of the AANN-based data validation scheme is discussed in Section 3.3 and the results are demonstrated. The sensor fault diagnosis strategy using the AANN is presented in Section 3.4 and the performance analysis of the proposed method is carried out. A comparison between the AANN-based and PCA-based is conducted in Section 3.5. A summary of the chapter is presented in Section 3.6.

3.1 Overview of Neural Networks

Layers of the neural network are composed of a number of nodes (or neurons) and they have full pairwise connections with the adjacent ones. Those connection are represented by a set of parameters known as the weights $\mathbf{W}$ and biases $\mathbf{b}$ which are to be adjusted through an iterative training procedure such that the network objective is achieved.

For an input $\mathbf{a}^{[l-1]} \in \mathbb{R}^{n_{l-1}}$ to layer l, its activation $\mathbf{a}^{[l]} \in \mathbb{R}^{n_l}$ is computed as:

$$\mathbf{a}^{[l]} = f\left(\mathbf{W}^{[l]}\mathbf{a}^{[l-1]} + \mathbf{b}^{[l]}\right), \tag{3.1}$$

where f is the activation function, n_{l-1} is the input $\mathbf{a}^{[l-1]}$ dimension, n_l is the activation $\mathbf{a}^{[l]}$ dimension, $\mathbf{W}^{[l]} \in \mathbb{R}^{n_l \times n_{l-1}}$ is the weights matrix of layer l, and $\mathbf{b}^{[l]} \in \mathbb{R}^{n_l}$ is the bias.

3.1.1 Activation Functions

Activation functions enable the network to learn and approximate complex functional mappings between the inputs and output targets. For every node in the layer, a transformation function is applied to the weighted sum vector x to produce the output y. There are several types of activation functions and the most commonly used are:

Sigmoid (σ)

It is expressed as:

$$y = \sigma(x) = \frac{1}{1 + \exp(-x)}. \tag{3.2}$$

Hyperbolic tangent (Tanh)

The Hyperbolic tangent produces an output with values between -1 and 1 and is expressed as:

$$y = \tanh(x) = \frac{2}{1 + \exp(-2x)} - 1. \tag{3.3}$$

3.2 Auto-Associative Neural Networks

Auto-Associative Neural Networks (AANNs) are Back-Propagation Neural Networks with the aim to capture the input-output model from known data samples [108]. The network is trained using the back-propagation algorithm to learn identity mapping from the input to the output. That is, the weights and biases of the network units are adjusted as to reproduce the input at the output. The architecture of an AANN is composed of the input layer, three hidden layers known as mapping layer, bottleneck layer, and de-mapping layer respectively, and the output layer as illustrated in Fig.3.1. The activation functions must be non-linear in the mapping and the de-mapping layers to improve the nonlinear dimensional mapping. However, linear or non-linear activation functions can be used in the bottleneck and output layers [109].

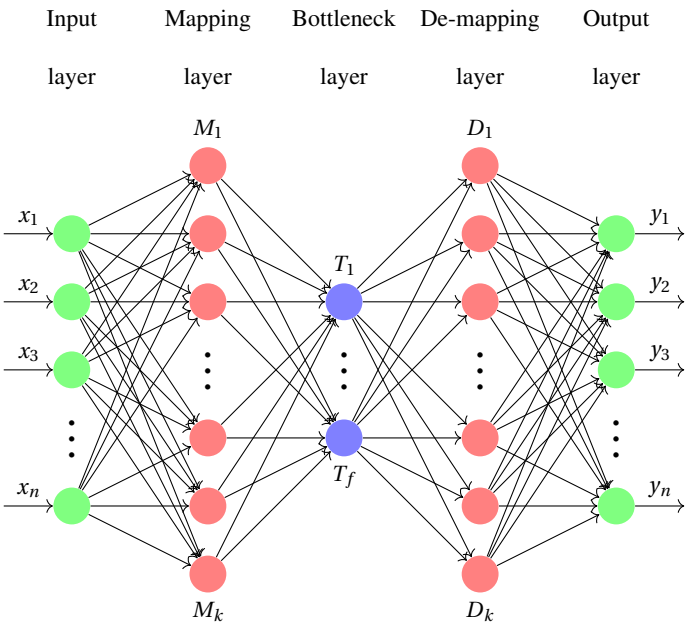


Figure 3.1: Diagram of AANN, x_i's are the network inputs, y_i's are the network outputs, n is the number of inputs/outputs, k is the number of nodes in the mapping/de-mapping layers, and f is the number of nodes in the bottleneck layer.

Both the input and the output layers have the same number of units corresponding to the dimension of the input of interest. In the mapping layer, the input is nonlinearly mapped to a higher dimensional space such that the data can be better separated than in the original space which helps to perform better at the preceding layer. The bottleneck layer aims to compress the input representation to the least possible dimension that promotes eliminating any random variations in the data. That is, the number of units in the bottleneck layer is desired to be as minimum as possible and less than the dimension of the input. The objective of the last two layers together is to reconstruct the inputs from the compressed representation obtained from the bottleneck layer and to bring the reconstructed data to the original spatial dimension.

3.2.1 Mathematical Description

The theory behind Auto-Associative Neural Networks can be described mathematically by two cascaded nonlinear vector transfer functions $\mathbf{G} = [G_1, G_2, \ldots, G_f]^{\mathrm{T}}$ and $\mathbf{H} = [H_1, H_2, \ldots, H_n]^{\mathrm{T}}$ where the first two layers represent the transfer vector function $\mathbf{G}$:$\mathbb{R}^n \rightarrow \mathbb{R}^f$ - provided that $f < n$ - in which the nth dimensional input $\mathbf{X} = [x_1, x_2, \ldots, x_n]^{\mathrm{T}}$ is mapped to a lower dimensional space $\mathbb{R}^f$ as follows:

$$T_i = G_i(\mathbf{X}), \quad i = 1, 2, \ldots, f, \tag{3.4}$$

where T_i is the output of the ith bottleneck unit and G_i is ith nonlinear factor defined by the parameters (weights and biases) on the connection from the input $\mathbf{X}$ to the output T_i. The output of the bottleneck layer $\mathbf{T} = [T_1, T_2, \ldots, T_f]^{\mathrm{T}}$ is then fed as the input to the second transfer vector function $\mathbf{H}$:$\mathbb{R}^f \rightarrow \mathbb{R}^n$ to produce the network output in the original nth dimensional space as:

$$y_j = H_j(\mathbf{T}), \quad j = 1, 2, \ldots, n, \tag{3.5}$$

where y_j is the output of the jth network output, and H_j is jth nonlinear factor defined by the parameters (weights and biases) on the connection from the bottleneck output $\mathbf{T}$ to the output y_i.

The process of dimensional mapping and de-mapping is described by equations (3.4) and (3.5) representing the implementation of a nonlinear principal component analysis (NLPCA) provided that the activation functions in the mapping and de-mapping layers are nonlinear (e.g. Sigmoid, Tanh, etc.).

3.2.2 Network Training and Performance Metrics

The AANN model is developed using MATLAB on data collected from the simulation model in TRNSYS for several modes of the HVAC system cooling operation. The data set contains 3 months of the normal system operation data sampled every 1 min in the summer season from June to August with a total size of 144K samples. The operating modes are day time and night time in both weekdays and weekends for different temperature setpoints of the zones (in the range of 18 - 26 °C). Normalization is performed on the sensors data to have values between 0 and 1 and it is based on the practical measurement range of the temperature sensor which is 10 - 40 °C. The training set and the validation set are 85% and 15% of the total data set, respectively allocated randomly.

The AANN has 11 inputs ($n = 11$) which are the normalized measurement vectors of the 7 sensors, the three control signals of the zones VAV boxes, and the control signal of the water valve. It is worth mentioning that we are interested in the data validation and fault diagnosis of the 6 sensors measuring T_{z1}, T_{z2}, T_{z3}, T_t, T_{ao}, and T_{wo} since they are vital for the HVAC system operation and its closed-loop control. However, the ambient temperature T_{amb} sensor data and the control signals U_1 - U_4 are used to enhance the learning of the AANN given the existing correlation between the 11 variables which promotes developing a proficient input-output mapping model. The cost function is the mean-square-error described as:

$$J = \frac{1}{m \times n} \sum_{i=1}^{n} \sum_{j=1}^{m} \left(x_i^j - y_i^j \right)^2, \tag{3.6}$$

where x_i^j is the ith network input of the jth sample, y_i^j is the ith network output of the

jth sample, and m is the number of samples.

Different network architectures are built by varying the number of neurons in the mapping/de-mapping layers and the bottleneck layer. Once the network is structured, it is trained using the Adams backpropagation algorithm to optimize the cost function. The training is carried out using a computer with 16 GB RAM and processor Intel(R) Core(TM) i7-4510U CPU at 3 GHz speed. The network performance evaluation is based on two metrics in addition to the cost function. The first metric is the network reconstruction accuracy (REC-ACC) which is measured by its ability to produce reliable sensors data at the output -close to the expected healthy data- when faulty inputs are applied. That is, the faulty data are generated as described in Section 3.2.3 using a bias fault of 10 °C. The reconstruction accuracy is determined within 2% difference error margin as:

$$\text{REACC} = \frac{1}{m \times n} \sum_{i=1}^{n} \sum_{j=1}^{m} \text{Count if} \left(\left| \left(y_i^j \right)_{\text{H}} - \left(y_i^j \right)_{\text{F}} \right| < 0.02 \right), \tag{3.7}$$

where $\left(y_i^j \right)_{\text{H}}$ is the healthy ith output of the network of the jth sample due to a healthy input, and $\left(y_i^j \right)_{\text{F}}$ is the actual ith output of the network due to a faulty input of the same sample.

The second evaluation metric is the reduced noise level (RNL) that is measured by the noise removed from the input by the AANN as follows:

$$\text{RNL} = \left(1 - \frac{\sigma_{\text{O}}^2}{\sigma_{\text{I}}^2} \right) \times 100, \tag{3.8}$$

where σ_{O}^2 is the average noise variance of the output vector, and σ_{I}^2 is the average noise variance of the input vector of the network over the test set.

The network that achieves the best performance in terms of these metrics is selected. Non-linear activation is used in the bottleneck layer, namely the sigmoid function which is also used in the mapping and the de-mapping layers while the output layer's activation is linear. Table 3.2 presents a summary of the evaluation results for some structures among all the evaluated network's architectures (with Sigmoid activation in bottleneck layer).

Table 3.2: Results of the performance analysis of different network's architectures.

#	Structure	REC-ACC	Cost	RNL
1	11-18-3-18-11	54.10%	0.0052	85.42%
2	11-36-3-36-11	55.71%	0.0045	79.98%
3	11-48-3-48-11	65.10%	0.0049	76.67%
4	11-15-4-15-11	49.10%	0.0033	76.10%
5	11-42-4-42-11	52.21%	0.0029	79.26%
7	11-27-5-27-11	35.37%	0.0015	65.38%
8	11-48-5-48-11	33.44%	0.0014	65.20%
9	11-45-6-45-11	20.67%	0.0011	64.50%
10	11-30-7-30-11	14.32%	0.0002	58.08%

It is observed that mostly as the number of the neurons in the bottleneck layer increases, the mean-square-error between the network input and output decreases meaning that the network ability to produce outputs matching inputs increases and the cost function value decreases. However, the network ability to filter out noise, and to reconstruct healthy outputs reduces. That is, the bottleneck layer is desired to be as small as possible in order to compress the input's dimension to the minimum possible mak-

ing the network useful for fault diagnosis and noise reduction. Otherwise, the network would just learn the exact mapping of the input at the output as the case with network structures 4 to 10. It is found that the best network architecture is Network 3 with the structure 11-48-3-48-11 with reconstruction accuracy of 65.1% and average noise removal of 76.1%. It consists of 11 neurons in the input and output layers, 48 neurons in the mapping and de-mapping layers, and 3 neurons in the bottleneck layer.

3.2.3 Sensor Fault Models

The performance of the AANN in sensor fault detection and isolation is examined for bias and drift faults with different severity levels. The sensor fault is injected using the normal operation validation dataset as follows:

$$\begin{aligned} x_{\mathrm{f}}(t) &= x_{\mathrm{h}}(t) \pm f_{\mathrm{bias}}, && \text{(biased fault)} \\ x_{\mathrm{f}}(t) &= x_{\mathrm{h}}(t) \pm f_{\mathrm{drift}}\, t, && \text{(drift fault)} \end{aligned} \tag{3.9}$$

where $x_{\mathrm{f}}(t)$ are the faulty sensor data, $x_{\mathrm{h}}(t)$ are the healthy sensor data, f_{bias} is the bias fault magnitude, f_{drift} is the drift gain magnitude, and t is the time.

3.3 AANN-based Sensor Data Validation Approach

The proposed AANN-based scheme is capable of recovering the healthy data from a faulty or inaccurate sensor since the network is trained to reproduce the healthy version at the output and hence it can be used primarily in sensor error correction and missing sensors replacement. Additionally, the AANN helps in data noise reduction and inaccuracy correction due to the data dimensionality compression implemented in the bottleneck layer which serves in eliminating uncorrelated variation in the data such as

measurement noise. The evaluation analysis of the applied data validation strategy is presented in the following subsections.

3.3.1 Sensor Error Correction

The unreliable sensor data due to fault occurrence (bias fault or drift fault) are replaced by their corrected values utilizing the other valid sensors data. Nevertheless, since the network variables are interrelated, a deviation is expected in the fault-free sensors data at the output due to the occurred fault. The network performance is evaluated to investigate the network capability in an effective sensor error correction due to the fault and the robustness of the healthy sensors under sensor fault occurrence. The evaluation metrics for the faulty sensor data correction and the healthy sensors data deviation per data sample are as follows [104]:

$$\begin{aligned} \text{Recovery Rate} &= \left(1 - \frac{|Y_\text{o} - Y_\text{f}|}{Y_\text{h}}\right) \times 100, \\ \text{Deviation Rate} &= \left(\frac{|Y_\text{o} - Y_\text{f}|}{Y_\text{h}}\right) \times 100, \end{aligned} \tag{3.10}$$

where Y_o is the actual output of the AANN and Y_h is the expected healthy output.

The results of the AANN capability in sensor fault correction are shown in Figures 3.2 to 3.4. These figures depict the average recovery rate of the faulty sensors data and the average deviation rate of the other healthy ones for different bias faults in the range of 1-30 °C over the test data set. The results show significant recovery rates for the sensors in the realistic bias fault range of 1-10 °C with an average recovery rate of around 93% on the faulty sensors while the deviation percentage in most of the healthy sensors data is maintained below 7%.

The capability of the AANN in fault correction decays for extreme bias faults. The

return water temperature sensor validated data T_{wo} is mostly sensitive to faults in the other sensors which is apparent from the average deviation rate plots. The larger the bias magnitude on the faulty sensor, the higher the deviation rate on T_{wo} data.

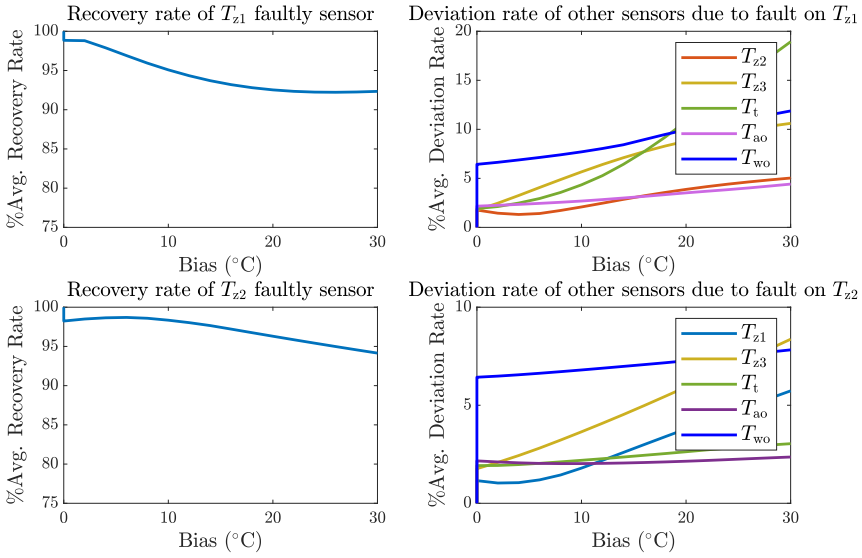


Figure 3.2: The average recovery rate and the average deviation rate for different bias faults in sensor error correction performance of the AANN on T_{z1} and T_{z2} sensors data.

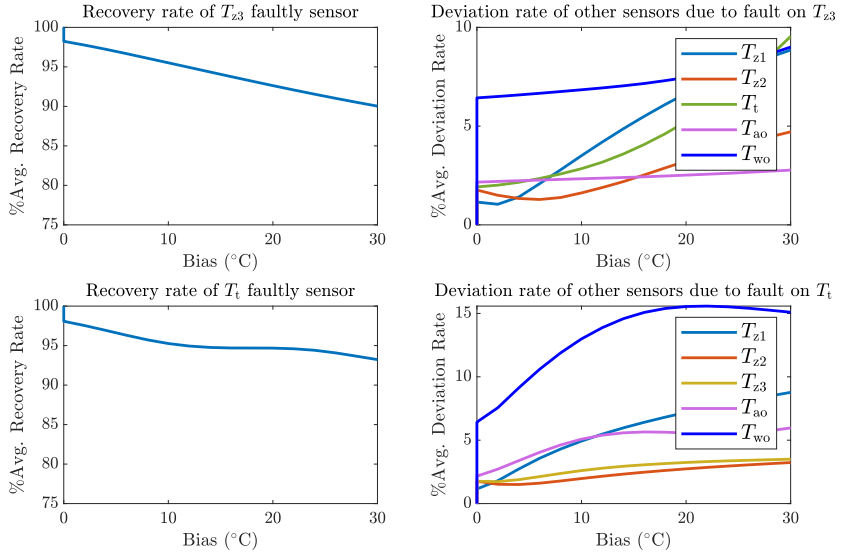


Figure 3.3: The average recovery rate and the average deviation rate for different bias faults in sensor error correction performance of the AANN on T_{z3} and T_t sensors data.

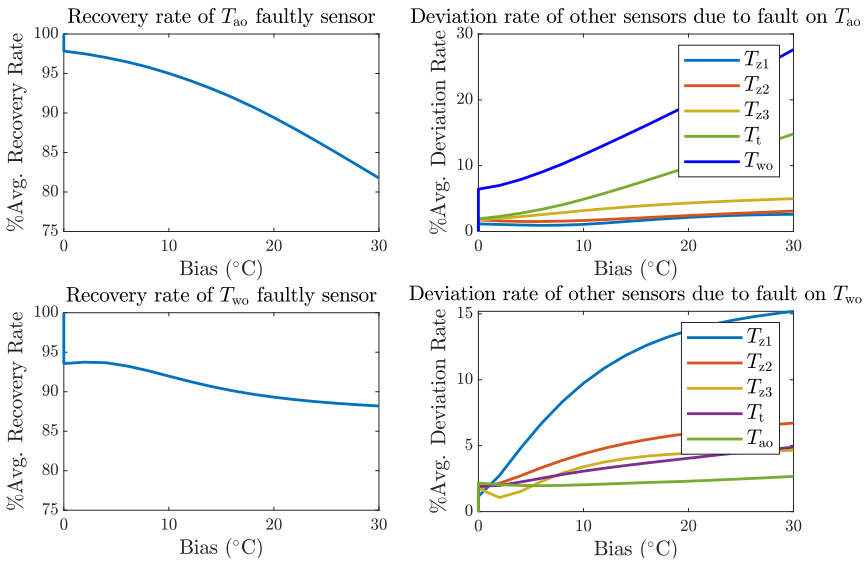


Figure 3.4: The average recovery rate and the average deviation rate for different bias faults in sensor error correction performance of the AANN on T_{ao} and T_{wo} sensors data.

3.3.2 Missing Sensor Replacement

Sensors functionality may be interrupted due to maintenance activities, complete failure, etc. [100]. The AANN retains sufficient information making it possible to replace missing sensors readings with estimated values using the data of the available sensors. This feature allows estimating the missing sensors data if they become unavailable and hence not interfere with other performance or safety-related processes within the system that rely on the continuous availability of those sensors data. Figure 3.5 demonstrates the missing sensor data recovery for the six sensors in the scenario of a single sensor complete failure at 12:00 PM. Tables 3.3 and 3.4 present the average recovery rate of the missing sensor and the average percentage deviation in the available remaining sensors over the test dataset. It is found that the network is remarkably capable of producing reliable data for the missing sensor measurement with an average recovery rate of 90% while the average deviation rate on the remaining sensors is

around 4.5%.

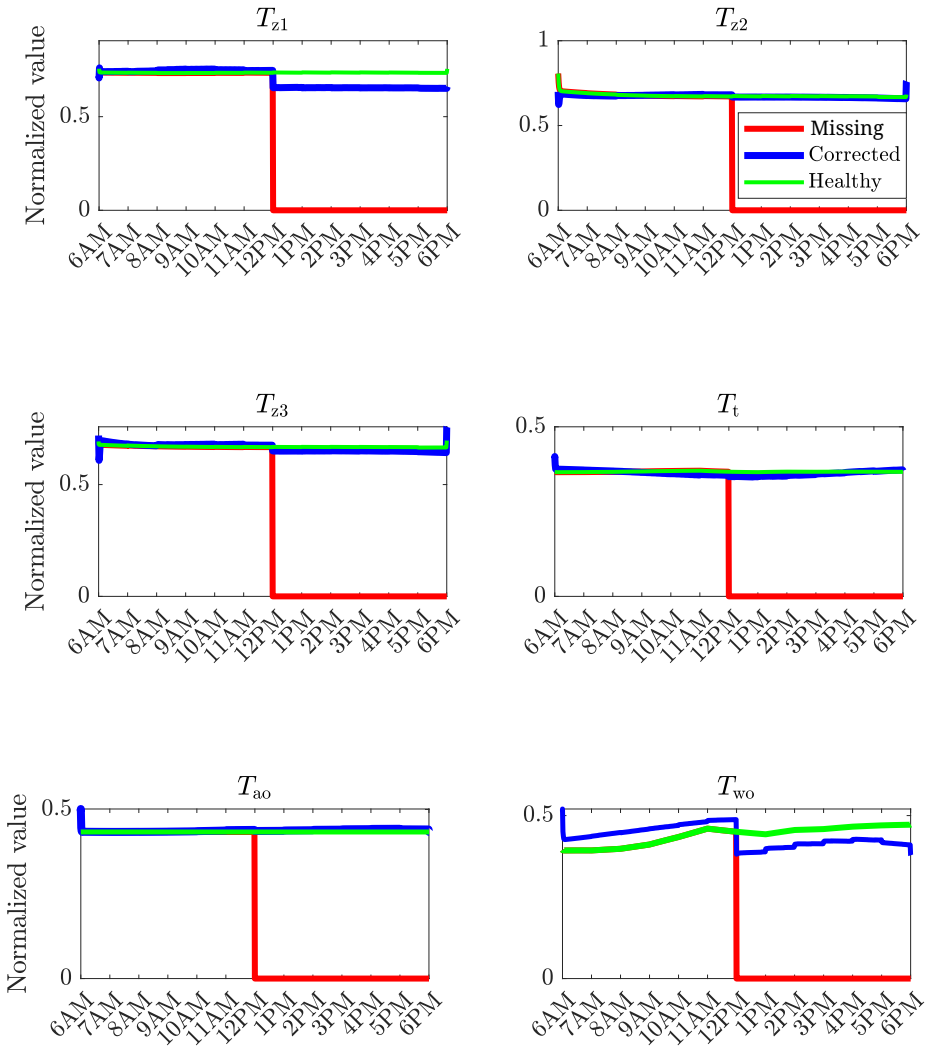


Figure 3.5: The missing, the corrected, and the healthy sensors data of the AANN due to a single complete sensor failure for each of the six sensors.

Table 3.3: Percentage of data recovery of the missing sensors.

Missing sensor	T_{z1}	T_{z2}	T_{z3}	T_t	T_{ao}	T_{wo}
Avg. Recovery	89.0%	94.2%	95.5%	97.4%	98.4%	88.2%

Table 3.4: Average percentage of data deviation of the remaining available sensors.

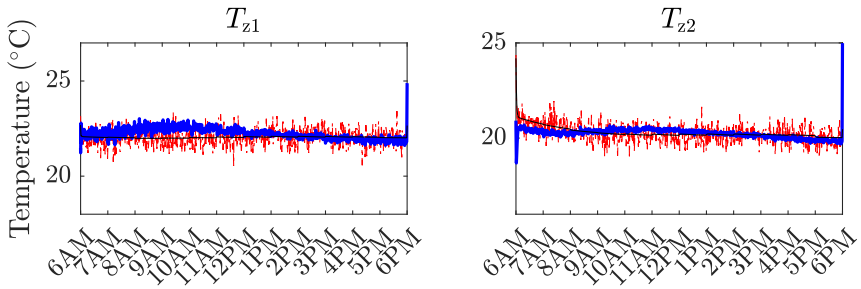
Missing sensor	T_{z1}	T_{z2}	T_{z3}	T_t	T_{ao}	T_{wo}
Avg. deviation	5.00%	3.57%	3.94%	4.42%	3.92%	5.12%

3.3.3 Noise Filtering

Additionally, the AANN is found capable of filtering the measurement noise in the sensors data. The performance of the network in measurement noise reduction is noticeable as listed in Table 3.5 and illustrated in Fig.3.6 with an average of 92% noise removal. The analysis is conducted under the assumption that the HVAC system units -zones, AHU, water tank, etc.- are equipped with the same type of temperature sensor which has a measurement range of 5-50 °C and accuracy of ±0.5°C. The nominal noise level is assumed to be 1% of the full range.

Table 3.5: Percentage of noise reduction in each sensor for the noise reduction evaluation.

Sensor	T_{z1}	T_{z2}	T_{z3}	T_t	T_{ao}	T_{wo}
RNL	71.78%	65.74%	66.10%	92.57%	89.52%	32.61%



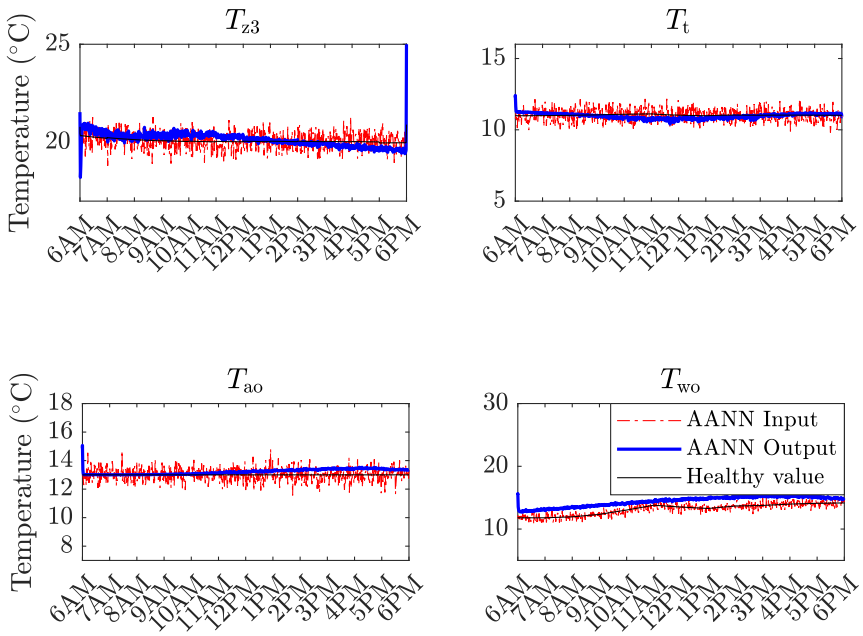


Figure 3.6: The noisy and the filtered sensors data in noise reduction performance of the AANN on the six sensors data.

3.3.4 Sensor Inaccuracy Correction

Sensors are subjected to loss of accuracy in which the absolute difference between the sensor output and the actual value is increased. Maintaining sensors accuracy is important for a reliable and successful system operation. The sensor inaccuracy is modeled by increasing the level of the noise in the sensors measurements to be 10% of the full range [104]. The performance results of the network in sensors inaccuracy correction are presented in Table 3.6 and are shown in Fig.3.7. It is clear that the network substantially suppressed the random variations due to the loss of accuracy for the sensors by an average of 94.5%.

Table 3.6: Percentage of noise reduction in each sensor for sensor inaccuracy correction evaluation.

Sensor	T_{z1}	T_{z2}	T_{z3}	T_t	T_{ao}	T_{wo}
RNL	89.78%	96.81%	95.79%	97.34%	98.06%	88.42%

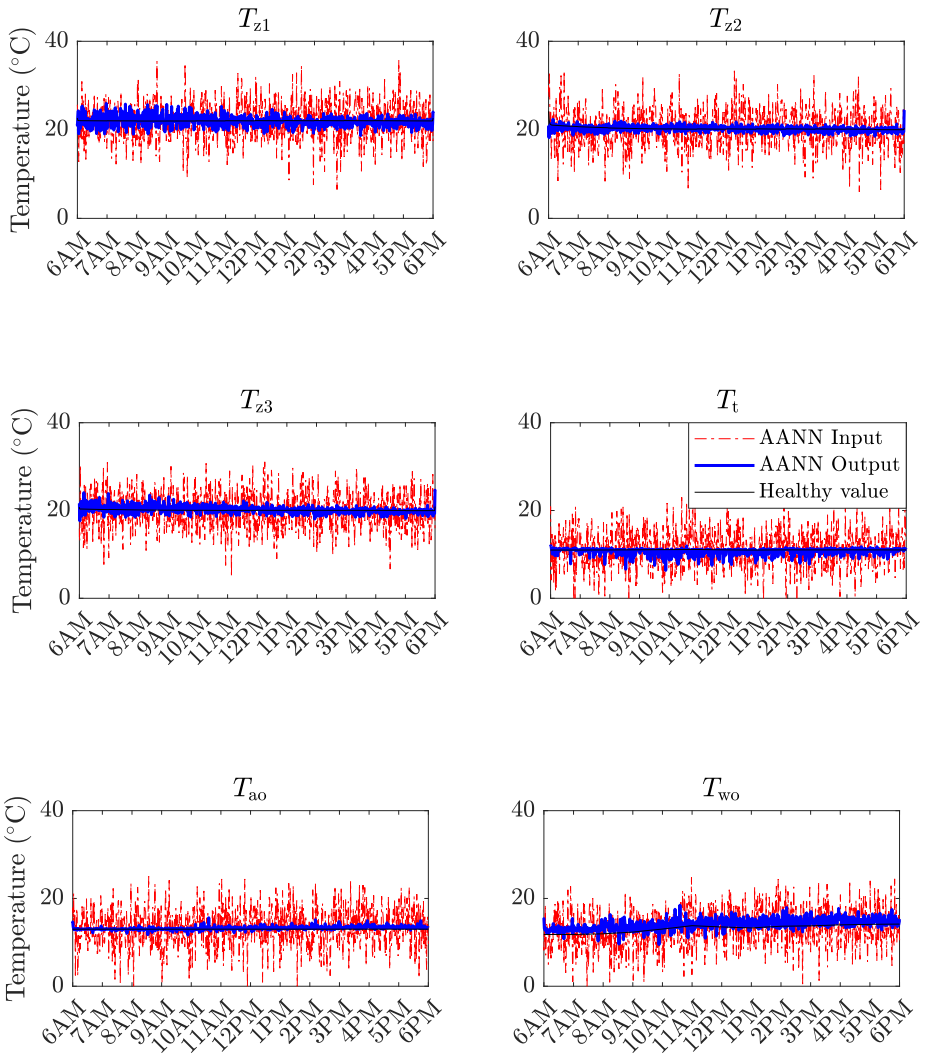


Figure 3.7: Inaccurate and accurate sensors data in sensors inaccuracy correction performance of the AANN on the six sensors data.

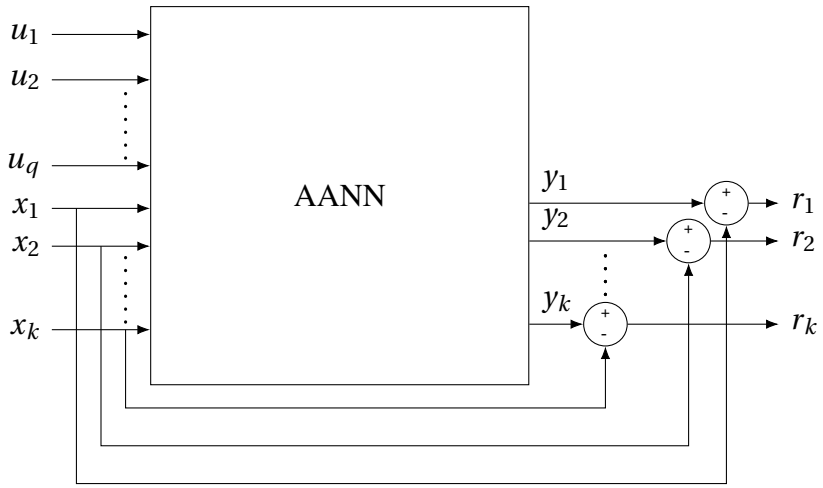


Figure 3.8: Diagram of sensor fault diagnosis scheme using AANN given q is the number of control signals and k is the number of sensors.

3.4 AANN-based Sensor Fault Diagnosis Approach

The Auto-Associative Neural Network can be used for sensor fault diagnosis based on observing the difference between the input x_i and the output y_i of the network with proper selection of the detection threshold ε_i as shown in Fig.3.8. Ideally, in fault-free scenarios, the generated residuals r_i are expected to be almost zero given that the network is constructing the input at the output layer. However, practically that is not the case since the network accuracy will never be 100%. Hence, it is a common practice to assign a threshold ε_i which is a safe margin the network residuals must exceed to indicate the fault occurrence as follows:

$$r_i = \left| y_i - x_i \right| \tag{3.11}$$

$$\text{The } i\text{th sensor fault indicator} = \begin{cases} 0, & \text{if } r_i < \varepsilon_i \\ & \\ 1, & \text{if } r_i \geq \varepsilon_i \end{cases}, \quad \text{for} \quad i = 1, \ldots k. \tag{3.12}$$

The detection threshold is closely tied to the diagnosis method characteristics which are the detection time, sensitivity, reliability, robustness, and isolability. These aspects need to be compromised when selecting the detection threshold. If the threshold value is set low, quick and sensitive fault detection is achieved but FD method reliability, robustness, and isolability may be degraded because false alarms rate will be increased due to modeling approximation and noise degrading. If the threshold value is high, the diagnosis algorithm will be robust to noise, but slow in fault detection, insensitive to low impact faults, and unreliable since missed alarms rate will be increased.

The performance of the AANN in sensor fault diagnosis is examined for bias and drift faults with different severity levels based on the selected detection threshold in Table 3.7. The detectable and isolable bias range for each sensor is presented in Table 3.8. It is determined by injecting a single bias fault with different magnitudes and assessing the True Positives Rate (TPR) for the faulty sensor indicating the percentage of correctly detected faults to be 85% minimum, and by False Positives Rate (FPR) for the remaining healthy sensors representing the percentage of the false alarms that is to be maintained below 10%.

Table 3.7: Detection threshold selected for sensor fault diagnosis.

Sensor	T_{z1}	T_{z2}	T_{z3}	T_t	T_{ao}	T_{wo}
Threshold ε	0.06	0.06	0.06	0.05	0.05	0.1

The percentage of correctly identified faults increases as the severity of the fault becomes higher as in Fig.3.9 while the average percentage of false alarms -triggered due to the fault - on the remaining healthy sensors increases as shown in Fig.3.10. For example, if sensor T_{z1} is subjected to $\pm 6^{\circ}$C, around 98% of actual fault instances are

detected and an average of 3% of time instances are falsely triggered alarms on the other fault-free sensors due to the network output deviation caused by the fault.

The capability of the proposed sensor fault diagnosis approach in diagnosing the sensor faults in the cooling coil output water temperature sensor is quite limited, unlike the others. This is due to the fact that this variable T_{wo} fluctuates according to the time-varying cooling load while the rest are generally steady over the air-conditioning period and hence it is easier to diagnose such sensors faults when their readings deviate from the normal values.

Table 3.8: Detectable and isolable bias fault for sensor fault detection and isolation.

Sensor	T_{z1}	T_{z2}	T_{z3}	T_t	T_{ao}	T_{wo}
Range (°C)	3-7	2-7	3-7	3-10	3-10	3.5-4.5

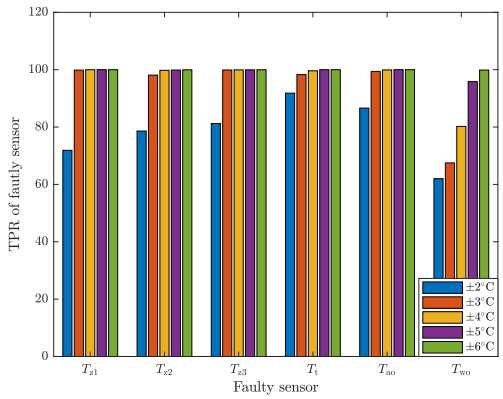


Figure 3.9: The percentage of the true positive rate under single bias sensor fault with different severity levels.

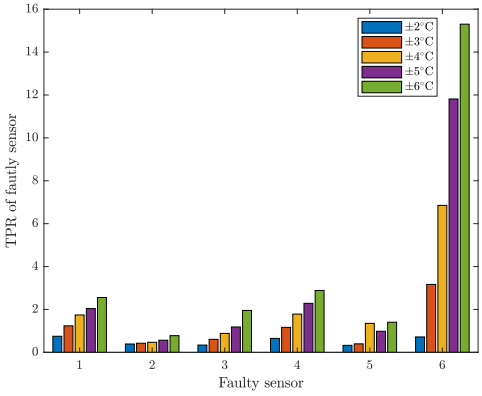


Figure 3.10: The average percentage of the false positive rate of the healthy sensors under single bias sensor fault with different severity levels.

As been demonstrated above, the sensitively of the diagnosis scheme is fairly acceptable with a detectable and isolable bias fault range of ± 5 to 15% of the full sensor measurement range, an average TPR of 85% on the faulty sensors, and an average FPR of 3% on the healthy sensors. Faults in T_{wo} sensor mainly causes extreme deviation in T_{z1} sensor output causing it's residual to exceed the detection threshold. This results in increased FPR on the fault-free sensor and consequently, the detectable and isolable bias range on T_{wo} sensor is narrow.

The evaluation of the sensor diagnosis scheme is based on looking at a one-day window assuming that the fault source will be eliminated within that time period. In addition, the case studies presented in the following subsections are evaluated under the presence of measurement noise with a variation level of 0.5% of the full sensor measurement range. It is worth mentioning that the AANN-based sensor fault diagnosis scheme is robust under low noise level operation such that the noise variance is below 1 °C. However, the approach cannot be used in the presence of high measurement noise because the difference between the noisy input and the filtered output can be interpreted falsely as a fault incident.

3.4.1 Single Sensor Fault

Single bias and drift faults are examined on each of the six sensors and the proposed method demonstrates an effective and reliable performance in diagnosing the sensor faults. Examples are presented in Fig.3.11 to Fig.3.14. Figure 3.11 shows 0.6 °C/h drift fault in sensor T_{z1} at 2 PM. The fault is detected within 2 hours after the fault magnitude exceeds the minimum detectable bias while the residuals of the healthy sensors are well below their detection thresholds. In Fig.3.12, sensor T_{z2} is subjected to a drift fault of 0.3 °C/h at 9 AM. The detection time is longer since the severity of the fault is lower. The detection time for bias faults is immediate as illustrated in Fig.3.13 and Fig.3.14.

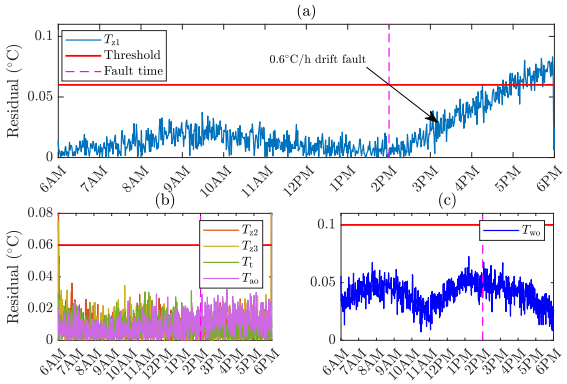


Figure 3.11: Single drift fault of 0.6°C/h in T_{z1} sensor. (a) Faulty sensor residual. (b) and (c) Residuals of the fault-free sensors.

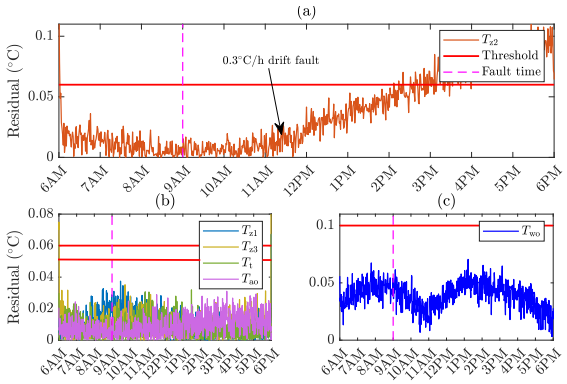


Figure 3.12: Single drift fault of 0.3°C/h in T_{z2} sensor. (a) Faulty sensor residual. (b) and (c) Residuals of the fault-free sensors.

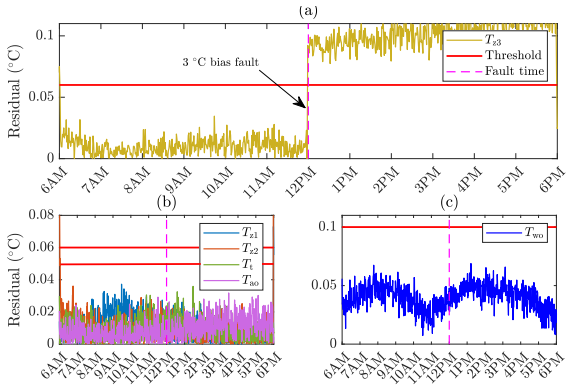


Figure 3.13: Single bias fault of 3°C in T_{z3} sensor. (a) Faulty sensor residual. (b) and (c) Residuals of the fault-free sensors.

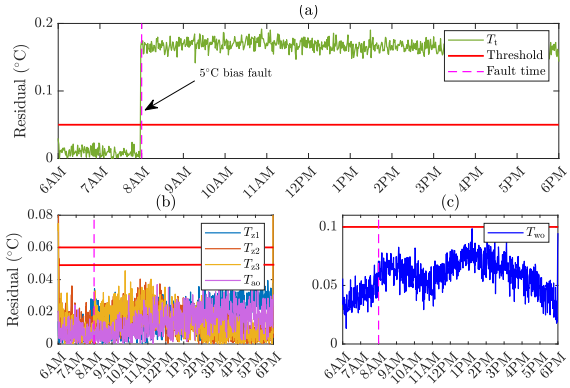


Figure 3.14: Single bias fault of 5°C in T_t sensor. (a) Faulty sensor residual. (b) and (c) Residuals of the fault-free sensors.

3.4.2 Multiple Sensor Fault

The occurrence of multiple sensor faults is also investigated thoroughly by injecting several sensor faults and observing the sensor residuals and the proposed approach is found capable of diagnosing multiple sensor faults as long as the fault amount is within the diagnosable range. For instance, Figure 3.15 shows 0.5 °C/h and 0.9 °C/h drift faults in two sensors, T_{z1} and T_t occurring at 9 AM and 1 PM respectively. The scenario of two bias faults of 7 °C and 4 °C in T_{z3} and T_{wo} sensors is shown in Fig.3.16. In addition, Figure 3.17 presents two drift faults and one bias fault in T_{ao}, T_{z1}, and T_t occurring simultaneously.

As shown in these figures, the residuals of the faulty sensors exceed the detection threshold while the others remain well below it. Unlike the majority of the previous works in sensor fault diagnosis schemes, the AANN-based method is able to detect and isolate multiple sensors faults whether they are interrelated (e.g., T_{wo} and T_{z3} in Fig.3.16) or not (e.g., T_{ao} and T_{z1} in Fig.3.17).

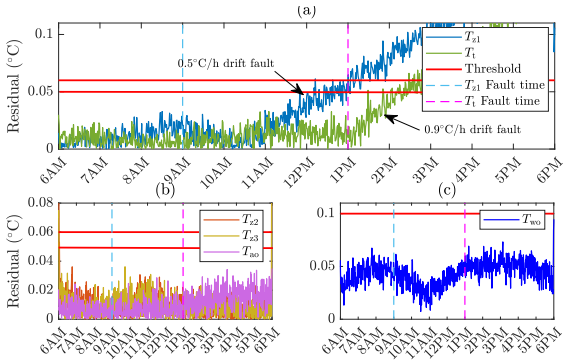


Figure 3.15: Multiple drift faults of 0.5 °C/h and 0.9 °C/h in T_{z1} and T_t sensors. (a) Faulty sensors residuals. (b) and (c) Residuals of the fault-free sensors.

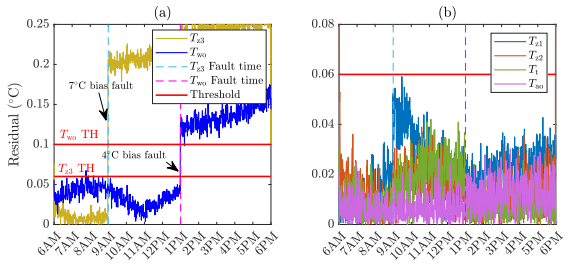


Figure 3.16: Multiple bias faults of 7 °C and 4 °C in T_{z3} and T_{wo} sensors. (a) Faulty sensors residuals. (b) Residuals of the fault-free sensors.

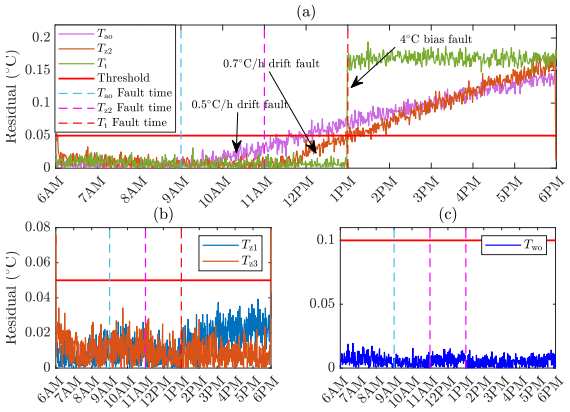


Figure 3.17: Multiple drift and bias faults of 0.5 °C/h, 0.7 °C/h, and 4 °C in T_{ao}, T_{z2}, and T_t sensors. (a) Faulty sensors residuals. (b) and (c) Residuals of the fault-free sensors.

3.5 Comparison with PCA-based Method

The performance of the proposed AANN-based sensor data validation and fault diagnosis scheme is compared with the PCA-based approach in [81]. The number of chosen principal components is 3 with a cumulative variance contribution of 95%. In terms of data validation, the PCA method performance in noise filtering and sensor inaccuracy correction is insignificant. However, the approach is capable of replacing missing sensor data with an average recovery rate of 77%.

For example, Fig.3.18 shows the recovery of T_{z1} sensor data after it is lost at 12

PM. The AANN-based method surpasses the PCA-based by 22% in missing sensor data replacement. The efficiency of these two schemes in fault correction is evaluated by injecting a sudden single 5°C and 10°C bias faults. It is found that the faulty data is validated by an average recovery percentage of 97% and 95% using the AANN-based approach and 93% and 86% via the PCA-based method for the 5 °C and 10 °C faults respectively on the six sensors. The fault correction capability of the two methods is approximately the same for low severity level faults but the AANN-based method performance is better as the fault severity increases.

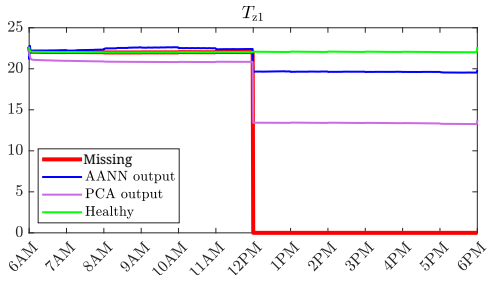


Figure 3.18: The missing, the healthy, and the corrected sensors data of the AANN and PCA due to a single complete T_{z1} sensor failure.

For the PCA-based sensor fault diagnosis algorithm, the threshold Q_a is set at 95% confidence interval and found to be 5.15. Tested on a 3°C bias fault, the accuracy of the PCA-based diagnosis method is lower with an average diagnosis accuracy of 59%. For instance, Fig.3.19 presents a 0.5 °C/h drift fault on T_{z2} occurring at 8 AM. The fault is detected and isolated at 12:30 PM using the AANN provided that the residuals of the remaining healthy sensors are maintained below the detection threshold as shown in Fig.3.20. With the PCA model, the fault is detected at 12:30 PM as well but only correctly isolated at 4 PM as shown from the Q-contribution plot in Fig.3.21. Table 3.9 presents a summary of the comparison between the two approaches.

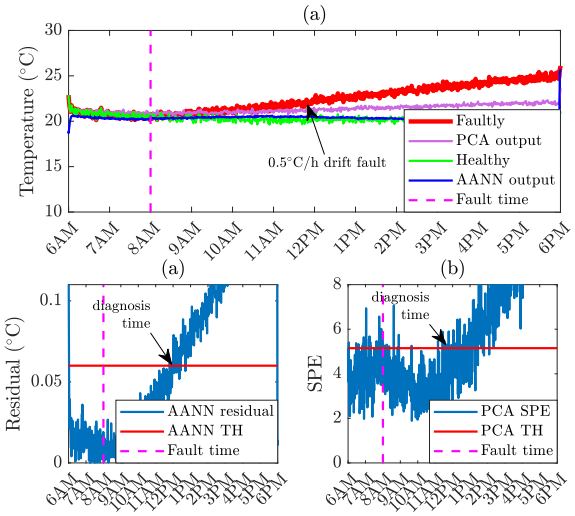


Figure 3.19: (a) Output of AANN and PCA models due to a single drift fault of 0.5°C/h in T_{z2} sensor. (b) The AANN residual of the faulty sensor. (c) The Squared Prediction Error (SPE) of the PCA output.

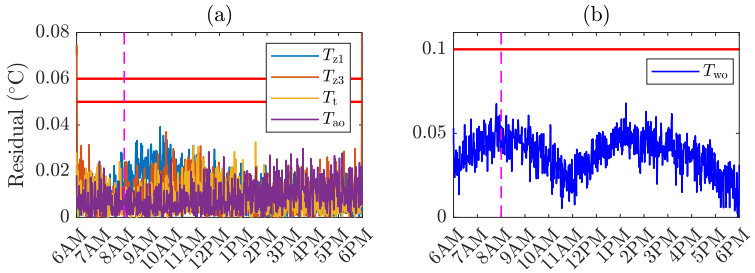


Figure 3.20: AANN-based method residuals of the fault-free sensors due to a single drift fault of 0.5°C/h in T_{z2} sensor.

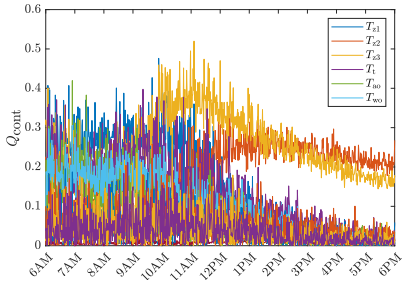


Figure 3.21: The Q-contribution plot of the PCA model due to a single drift fault of 0.5°C/h in T_{z2} sensor.

Table 3.9: Summary of the comparison between the performance of the AANN-based and the PCA-based sensor data validation and fault diagnosis approaches.

Performance metric		AANN	PCA
Average diagnosis accuracy		80%	59%
Average fault correction	4 °C	97%	93%
	10 °C	95%	86%
Average missing data recovery		94%	77%

3.6 Summary

This chapter presented an application of the Auto-Associative Neural Network in sensor data validation and fault diagnosis for a 3-zone HVAC system. The method was developed and tested using simulation data generated from TRNSYS. The AANN-based solution has demonstrated effective performance in sensor data error correction, data replacement of unavailable sensors, measurement noise reduction, as well as in both single and multiple sensor faults diagnosis. The performance of the diagnostic method was found very satisfactory with a detectable and isolable bias range of 5 - 15% of the full measurement range.

This approach was compared with a PCA-based algorithm and the results showed a notable improvement by 35% in diagnosis accuracy, 22% in missing data recovery, and 10% in fault correction. Moreover, unlike the PCA-based approach, it was capable of diagnosing multiple and simultaneous sensor faults, as well as sensor noise filtering and inaccuracy correction.

The proposed algorithm can be applied for large-scale buildings with additional zones and it is expected that the performance will be improved with the increase in the HVAC system size due to a higher correlation between the inputs. Its computational requirement can be compromised by exploiting parallelism and deploying a kind of a multi-agent diagnosis scheme as the building size grows larger. In addition, this method can be used as a pre-preliminary step to the HVAC closed-loop control system. The AANN sensor data validation and fault diagnosis approach is suitable for applications where sensors measurements are broadly steady over the operation period which is the case with HVAC systems. It is robust under low noise level and detection threshold re-assignment must be carried out for higher measurement noise variance to avoid misinterpreting the difference between the noisy input and the filtered output as a fault occurrence.

Chapter 4: Actuator Fault Diagnosis of HVAC System using CNNs

Researchers have been investigating and proposing various solutions for actuator fault diagnosis given that actuators are mostly prone to faults resulting in thermal discomfort and energy consumption inefficiency in buildings. As summarized in Table 4.1, different data-driven approaches such as SVM [44], PCA and pattern matching [35], Bayesian Networks [51, 80, 110], neural networks [43, 48], and control chart and rule-based [77] have been developed for HVAC actuator fault diagnosis. Development in this field is still progressing but it is still not keeping up with the growing building sector and the increasing global demand for energy. For instance, some developed approaches have limited performance capability for large-scale buildings [44] and others do not address the fault isolation matter [35]. The limitation in the proposed approach in [77] is its dependency on a set of expert rules while the other methods suffer from limited diagnosis accuracy due to the relatively high rate of false alarms (or missed alarms) [43, 48, 51, 80, 110].

This chapter presents a novel supervised on-line actuator fault diagnosis for HVAC systems using 2D Convolutional Neural Networks. It aims to address the limitations found in the previous works in terms of the diagnosis accuracy, and adaptability to larger buildings by adopting the recently evolving topology of the Convolutional Neural Networks.

Table 4.1: Summary of HVAC system actuator fault diagnosis methods in the literature.

Ref.	Method	Data source	Fault Type
[44]	SVM	Simulation (MATLAB/Simulink)	Stuck damper/valve
[35]	PCA and Pattern Matching	Experimental	Stuck damper
[51]	Bayesian networks	Experimental	Stuck damper
[80]	Bayesian networks	Simulation	Stuck valve
[110]	Bayesian network	Simulation	Stuck damper/valve
[48]	ANN and Fuzzy logic	Simulation (MATLAB/Simulink)	Stuck valve
[43]	NNs and clustering analysis	Simulation (TRNSYS)	Stuck valve
[77]	Control chart and rule-based	Operational	Stuck valve

Convolutional Neural Networks (CNNs) are commonly used for computer vision and image processing applications due to their computation efficiency, high performance accuracy, and feature extraction and classification fused characteristic when dealing with high-dimensional data [3]. In addition, they are used with 1D data for natural language processing [111–113], biomedical engineering applications [62, 114–128], and condition monitoring and fault diagnosis [129–152].

One-dimensional Convolutional Neural Network (1D CNN) has been used for bear-

ing fault detection in [129–132], structural damage detection in [133–135], hydraulic pump fault diagnosis in [136], gearbox condition monitoring in [137], and modular multilevel converter (MMC) circuits fault diagnosis in [138]. 1D CNNs are effective when dealing with relatively fast signals in which particular features or patterns can be captured in short fixed-length segments.

In some research works, 2D CNNs are used with 1D signals where the raw data are converted to a 2-dimensional configuration using various conversion approaches. For instance, in [139–141] 2D CNN-based method is proposed for bearing fault diagnosis using vibration images produced by signal amplitude to pixel intensity mapping while [142] uses the actual images of the vibration signals segments as CNN inputs. Process fault diagnosis frameworks are developed in [143, 144] in which time-series process variables are configured into a matrix where the x-axis is time and the y-axis represents different process variables.

Moreover, signals processing techniques are applied to the raw signals to obtain their 2D representations as in [145] using Discrete Fourier Transform (DFT), in [146–148] using time-frequency representation, in [149] using Continuous Wavelet Transform Scalogram (CWTS), and in [150] using Spectral Energy Maps (SEMs) for bearing and rotary machinery fault diagnosis. In [151], planetary gearbox fault diagnosis method is presented using Discrete Wavelet Transform (DWT) on vibration signals and in [152], a CNN-based stability monitoring system for power systems is developed using the heatmap representation of the measurements.

The limitations with the aforementioned CNN-based algorithms proposed in the literature are that they require advanced data preprocessing and/or a sequence of input data to make the diagnosis decision. For slow signals such as temperature, the time

frame window should be large enough for the network to learn to capture the fault occurrence which consequently leads to long detection time and increased computational requirements.

The work presented in this chapter eliminates those drawbacks found in the FD CNN-based approaches by proposing the use of 2D CNN with efficient and real-time data transformation to obtain the 2D representation of the raw 1D measurements data of the HVAC system variables without advanced data pre-processing requirement. The proposed diagnosis scheme demonstrates improved performance overly when compared with other commonly used methods.

This chapter is organized as follows. Section 4.1 presents the theory of the Convolutional Neural Networks and in Section 4.2 the network training details and the evaluation metrics are presented. The proposed CNN-based HVAC actuator fault diagnosis approach is presented in Section 4.3. Firstly, in Section 4.3.1 the proposed 1D data representation transformation is demonstrated and in Section 4.3.2 the evaluated CNN-based fault diagnosis models are described. In Section 4.4, the results of the evaluated models are presented while the comparison results between the actuator fault diagnosis using CNN, ANN, and SVM are discussed in Section 4.5. A summary is presented in Section 4.6.

4.1 Overview of Convolutional Neural Networks

The Convolutional Neural Network is a special type of Back-Propagation Neural Networks with an input layer, hidden layers, and an output layer and it is characterized by its internal structure which contains the Convolution (Conv) layers, the pooling layers, and the Fully Connected (FC) layers as shown in Fig.4.1. Similar to the stan-

dard neural network, it is trained using the backpropagation algorithm to optimize a cost function. However, CNN exploits spatially local correlation by enforcing a local connectivity pattern between neurons of adjacent layers [3].

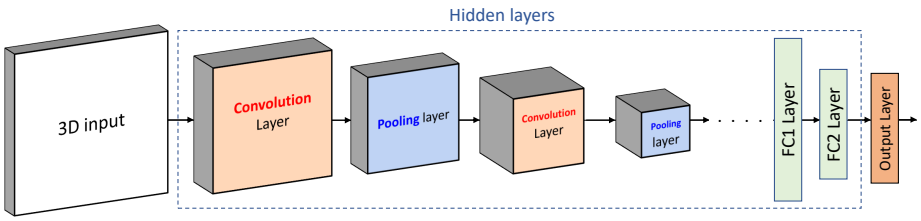


Figure 4.1: Typical Convolutional Neural Network architecture.

4.1.1 Convolution Layer

The convolution layer is the core building block of the CNN where most of the computation is carried out. Convolution denoted by the operator '$*$' is a mathematical operation that uses both addition and multiplication and it is used during both forward and backward pass during training. The convolution layer consists of a number of convolution windows called filters or kernels that slide over the input preserving its spatial structure to produce the feature map (output) as illustrated in Fig.4.2.

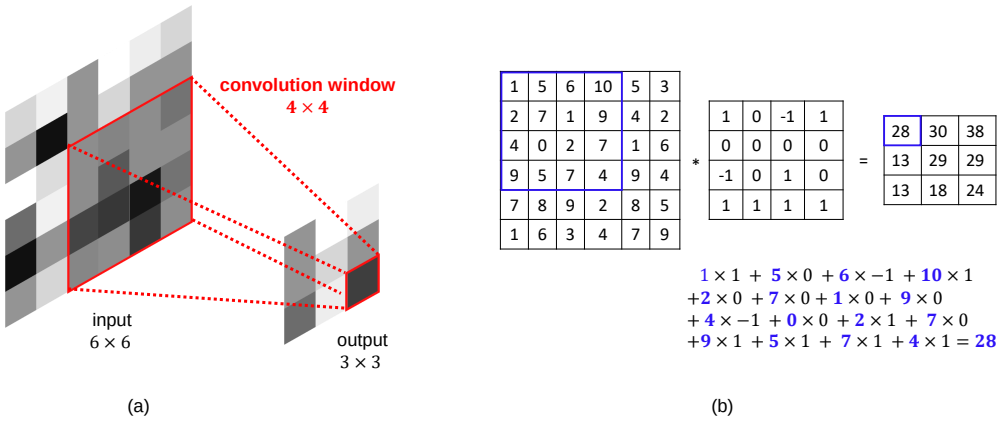


Figure 4.2: Convolution layer operation. (a) The sliding of the convolution window (kernel) over the input to compute the output. (b) The convolution mathematical operation (valid convolution).

Given the illustration in Fig.4.3, the layers activations (outputs) size is denoted by $n \times m$ with c number of feature maps. Let the ith feature mapping of the activations in layer l be $\mathbf{A}_i^{[l]} \in \mathbb{R}^{n_l \times m_l}$, the jth feature mapping of the activations of the preceding layer are $\mathbf{A}_j^{[l-1]} \in \mathbb{R}^{n_{l-1} \times m_{l-1}}$, the kernel is $w_{ij}^{[l]} \in \mathbb{R}^2$, i.e, weights of the connection between the ith feature mapping at the layer $l-1$ and the jth feature mapping at the layer l, the bias is $b_i^{[l]} \in \mathbb{R}$, and the activation function is f, then [3]

$$\mathbf{A}_i^{[l]} = f\left(\sum_j w_{ij}^{[l]} * \mathbf{A}_j^{[l-1]} + b_i^{[l]}\right). \tag{4.1}$$

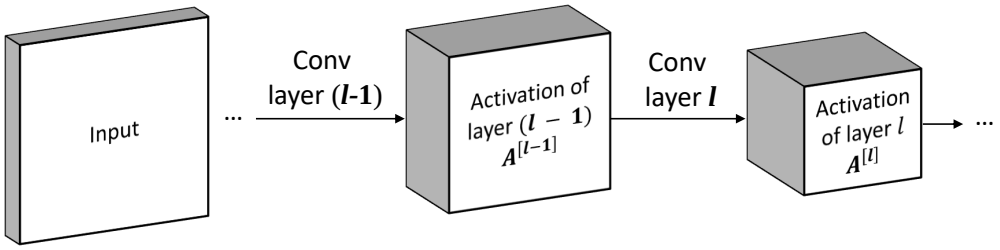


Figure 4.3: Illustration of outputs of the Convolutional Neural Network layers.

Convolution layers are characterized by neurons' local connectivity determined by the kernel size, spatial arrangement which is how the filter slides over the input, and parameter sharing [2].

There are a number of hyper-parameters in the convolution layer which are the activation function, the size of kernel $f \in \mathbb{R}^2$ which defines the field of view of the convolution, the number of filters $F \in \mathbb{R}$, i.e., the number of feature maps, the stride $s \in \mathbb{R}$ representing the size of the convolution step, i.e. if it is one, the kernel is moved by one step in each direction, and the type of convolution which can be same or valid. In the same convolution, zero-padding is performed on the input in order for the output to have the same dimension as the input whereas a valid convolution is a type of convolution operation that does not use any padding on the input.

4.1.2 Pooling Layer

The pooling layer which is also known as the sub-sampling layer aims to reduce the spatial size of the convolution layer output in order to reduce the number of parameters and the amount of computation. There are two commonly used types of pooling, which are max pooling and average pooling as shown in Fig.4.4 and their fundamental principle is merging similar local features into a single dominant one. That is, the pooling layer transforms the representation $h \in \mathbb{R}^{M \times M}$ to a pooled representation $g \in \mathbb{R}^{\frac{M}{P} \times \frac{M}{P}}$ by segmenting h into non-overlapping temporal regions $r_k \in \mathbb{R}^{P \times P}$, for $k = 1 \ldots, \left(\frac{M}{P}\right)^2$ and finding the maximum or the average of the regions as follows:

Max pooling:

$$g_i^k = \max_{j \in r_k} \left(h_j^k \right), \tag{4.2}$$

Average pooling:

$$g_i^k = \frac{1}{P^2} \sum_{j \in r_k} h_j^k, \tag{4.3}$$

where for the kth region r_k, h_j^k is the jth element of the original representation and g_i^k is the ith element of the pooled representation computed for that region. The pooling layer does not have any learnable parameters.

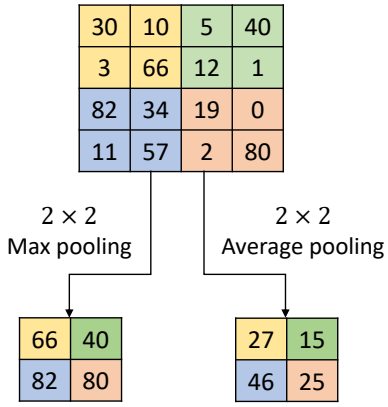


Figure 4.4: Max and average pooling operation with $M = 4$ and $P = 2$.

4.1.3 Fully Connected Layer

Nodes of the fully connected layer have full pairwise connections with the adjacent layers. The number of neurons in the FC layer is a hyper-parameter to be set in addition to the type of activation function. Multiple fully connected layers are usually placed before the output layer for the purpose of feature classification. For an input $\mathbf{a}^{[l-1]} \in \mathbb{R}^{n_1}$ to layer l, its activation $\mathbf{a}^{[l]} \in \mathbb{R}^{n_2}$ is computed as:

$$\mathbf{a}^{[l]} = f\left(\mathbf{W}^{[l]}\mathbf{a}^{[l-1]} + \mathbf{b}^{[l]}\right), \tag{4.4}$$

where f is the activation function, $\mathbf{W}^{[l]} \in \mathbb{R}^{n_2 \times n_1}$ is the weights matrix of layer l, and $\mathbf{b}^{[l]} \in \mathbb{R}^{n_2}$ is the bias.

FC layer takes 1D input and hence the output of the previous layers is flattened into a one-dimensional array before feeding it to the FC layer as shown in Fig.4.5. Let $\mathbf{Q} = [q_{ijk}] \in \mathbb{R}^{w \times v \times c}$ be the 3-dimensional output of the last convolutional layer where $w \times v$ is size of the output, and c is the number of feature maps. Then, the flattening produces the vector $\mathbf{Q}_{\text{flat}} = [q_{111}, q_{121}, q_{131}, \ldots, q_{wvc}] \in \mathbb{R}^N$ where $N = w \times v \times c$ is the number of elements of the output $\mathbf{Q}$.

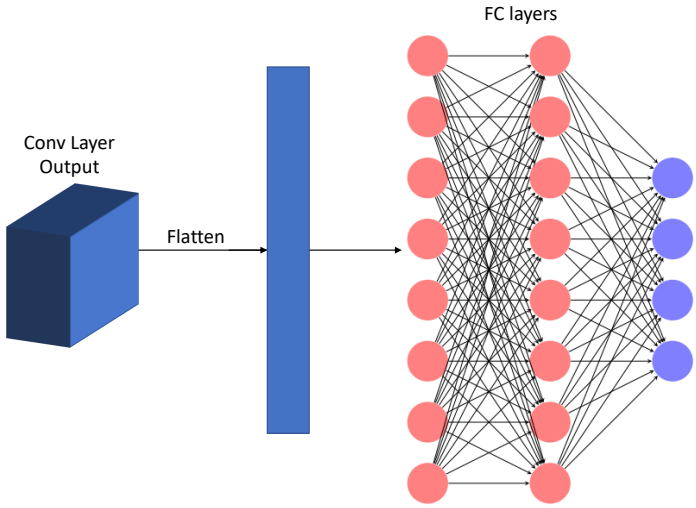


Figure 4.5: Flattening convolution layer (and pooling layer) output before feeding it in to the FC layers.

4.1.4 Output Layer

The output layer in a typical CNN is a fully connected layer with a number of nodes equivalent to the number of classes and using a Softmax activation function. The Softmax function is applied to the output of the fully connected layer $\mathbf{z} = [z_1, z_2, \ldots, z_k]$ to generate the output $\mathbf{y} = [y_1, y_2, \ldots, y_k]$ corresponding to the probability of belonging to each of the classes over the predicted output classes. The ith output y_i is computed

as:

$$y_i = \frac{\exp(z_i)}{\sum_{j=1}^{k} \exp(z_j)}, \quad i = 1, \ldots, k, \tag{4.5}$$

where k is the number of classes, $y_i \in [0, 1]$, and $\sum_{j=1}^{k} y_j = 1$.

4.1.5 Activation Functions

As mentioned in the Chapter 3, Section 3.1.1, activation functions are used to promote the network learning capability. In addition to the activation functions described in the previous chapter, the ReLU function is a commonly used with CNNs. It returns 0 if the input is negative and its mathematical form is:

$$y = \max(0, x). \tag{4.6}$$

Another version is the leaky ReLU such that for negative input values, it returns a proportion of the input based on the scaling factor α and it expressed as:

$$y = \max(\alpha x, x), \tag{4.7}$$

where $\alpha < 1$.

4.2 Convolutional Neural Network Training

The training of CNN is a highly iterative process and it can be challenging due to the several hyper-parameters involved. In addition, the choice of the optimization algorithm is crucial which can be batch gradient decedent (GD), mini-batch gradient descent, stochastic gradient descent with momentum (SGDM), adaptive moment estimation (Adam), etc. For a k-class classification problem, given a training set $\{\mathbf{x}_i, \mathbf{t}_i\}_{i=1}^{m}$, where m is the number of samples, $\mathbf{x}_i \in \mathbb{R}^n$ denotes the observation vector and $\mathbf{t}_i \in \{0,1\}^k$ is the class indicator vector with one-of-k encoding, i.e., for a class k, only the kth element in the vector $\mathbf{t}_i$ is 1 and all the other elements are 0, the cost function E to

be optimized is the cross-entropy function given by [3]:

$$E = \sum_{i=1}^{m} \sum_{j=1}^{k} \mathrm{t}_{ij} \ln y_{ij}, \tag{4.8}$$

where t_{ij} is the jth element of the target vector $\mathbf{t}_i$, and y_{ij} is the jth element of the prediction output vector $\mathbf{y}_i \in \mathbb{R}^k$.

In addition to the hyper-parameters associated with the layers discussed in the previous section, there are ones related to the training procedures which are the learning rate α, the mini-batch size, the number of epochs which is the number of full passes through the training set during training, etc. One can always achieve better results by hyper-parameter tuning. Moreover, other techniques can be applied to optimize the training process and to avoid over-fitting such as batch normalization, regularization, and dropout as described in the following subsections.

4.2.1 Batch Normalization

Batch Normalization (BN) aims to eliminate the internal covariate shift problem given that the distribution of each layer's inputs changes continuously during training. It ensures that the range of the output of a layer is within a small interval before feeding it to the subsequent layer. The normalization is performed with the running average of the mean–variance statistics of each mini-batch. For instance, given the input to the layer l, x_i, the mean μ_B and variance σ_B^2 are calculated over the mini-batch and for each input channel. Then, the normalized activations are calculated as:

$$\widehat{x}_i = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}}, \tag{4.9}$$

where ϵ is used to ensure numerical stability when the mini-batch variance is very small. The final BN output is calculated as:

$$y_i = \gamma \hat{x}_i + \beta, \tag{4.10}$$

where γ is the scale factor and β is the offset factor. BN allows the network to be more stable. However, it usually slows down the training process and requires careful initialization [153].

4.2.2 Regularization

Regularization aims to improve the generalization of the network to different data and to solve the problem of over-fitting by limiting the magnitude of the network's parameters. That is, over-fitting occurs when the weights of the neural network are trained such that the network performs very well on the training dataset with poor generalization on other data. Figure 4.6 demonstrates the cases of under-fitting and over-fitting with respect to the error (cost) and the function order.

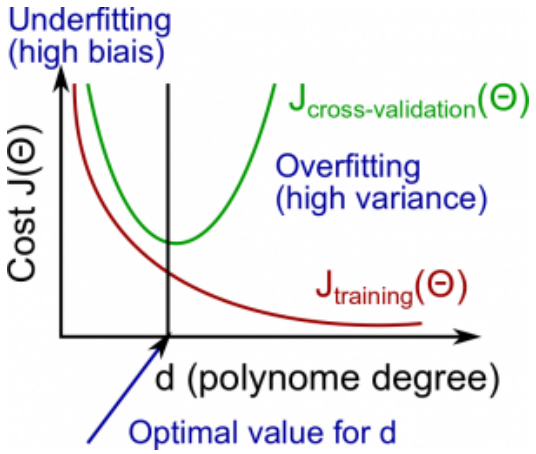


Figure 4.6: Under-fitting and over-fitting problems [2].

4.2.2.1 ℓ_p Regularization

Classical regularization techniques are ℓ_1 and ℓ_2 regularizations which work by imposing a constraint on the weights Θ in the cost function as:

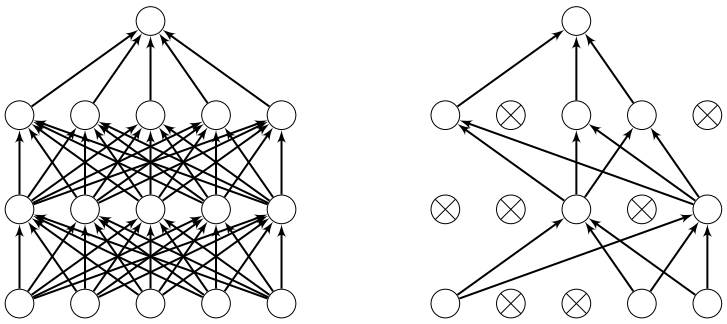
$$E_{R_{\ell_1}} = E + \lambda \sum_i \Theta_i, \tag{4.11}$$

$$E_{R_{\ell_2}} = E + \lambda \sum_i \Theta_i^2, \tag{4.12}$$

where λ is the regularization factor (coefficient).

4.2.2.2 Dropout

Dropout prevents over-fitting by randomly turning off some neurons in the forward-propagation pass during training at each iteration. It works by setting nodes outputs to zero randomly with a probability p as shown in Fig.4.7.



(a) Standard neural network (b) After applying Dropout

Figure 4.7: Dropout operation [3].

4.2.3 Performance Evaluation Metrics

The confusion matrix or the error matrix is typically used to evaluate the performance of the CNN. It is a form of contingency table with two dimensions identified as True and Predicted, and a set of classes in both dimensions as presented in Table 4.2. The following performance metrics are derived from the confusion matrix [154].

Table 4.2: Table of confusion for 2-class problem.

		Predicted	
		Positive	Negative
True	Positive	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

4.2.3.1 Accuracy and Error

Accuracy (ACC) is a measure of the closeness between the predicted result and the true value while the error is the measure of deviation of the predicted value from the true one. Accuracy is given as:

$$\text{ACC} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}. \tag{4.13}$$

4.2.3.2 Precision

It is also called the positive predictive value (PPV) which is a measure of the closeness of the set of predicted results and it is expressed as:

$$\text{PPV} = \frac{\text{TP}}{\text{TP} + \text{FP}}. \tag{4.14}$$

4.2.3.3 Sensitivity

It is called true positive rate (TPR) or recall and is calculated by,

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}. \tag{4.15}$$

4.2.3.4 Specificity

It is also known as true negative rate (TNR) and is expressed as:

$$\mathrm{TNR} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}}. \tag{4.16}$$

4.2.3.5 F_1-Score

It is the harmonic average of the precision and recall, where it is at its best at a value of 1 meaning perfect precision and recall and it is given as:

$$\mathrm{F}_1 = 2 \times \frac{\mathrm{PPV} \times \mathrm{TPR}}{\mathrm{PPV} + \mathrm{TPR}}. \tag{4.17}$$

4.2.3.6 Matthews Correlation Coefficient (MCC)

It provides a balanced measure to evaluate the performance of binary classification algorithms such that a value of 1 means perfect prediction and it is expressed as:

$$\mathrm{MCC} = \frac{\mathrm{TP} \times \mathrm{TN} - \mathrm{FP} \times \mathrm{FN}}{\sqrt{(\mathrm{TP} + \mathrm{FP})(\mathrm{TP} + \mathrm{FN})(\mathrm{TN} + \mathrm{FP})(\mathrm{TN} + \mathrm{FN})}}. \tag{4.18}$$

4.2.4 *k*-Fold Cross Validation

This method is used to gain a sense of the network ability to generalize well on different data especially when the amount of data is limited [155]. Firstly, the data set is divided equally into k random subsets, also called folds. Then, the CNN is trained k times and in each time one fold is used as the validation set and the remaining ones for training as illustrated in Fig.4.8. The performance evaluation of the CNN is determined by averaging the performance results of the k trained models.

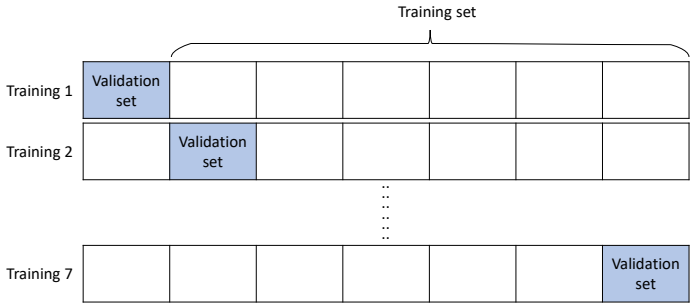


Figure 4.8: 7-fold cross validation.

4.3 Proposed CNN-based Actuator Fault Diagnosis Framework

The framework of the proposed CNN-based diagnosis method as illustrated in Fig.4.9 consists of an off-line stage in which the CNN is trained using the historical building data, and an on-line stage such that the real-time measurements of the system variables are acquired by the building management system (BMS) and used to determine the diagnosis decision. Various network architectures and several diagnosis models are proposed for actuator fault diagnosis using CNN as presented in the following subsections.

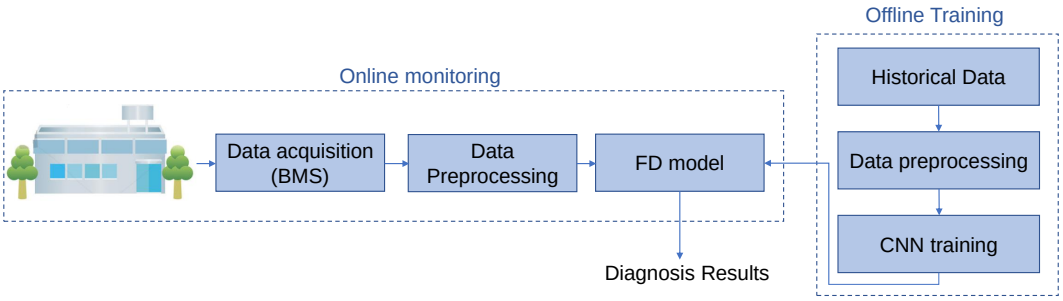


Figure 4.9: The CNN-based HVAC system actuator fault diagnosis method framework.

4.3.1 Proposed 1D to 2D Data Conversion Technique

The variables of the system are time-series signals, i.e. temperature, control signals. They need to be reformed before being used in the CNN-based diagnosis model because the CNN input is 2-dimensional. The first step of the signal-to-matrix conversion is the normalization of the temperature data to be ranging between 0 and 1. It is done based on the practical measurement range of the temperature sensor of 10 - 40 °C. The normalized data and the control signals are configured -with proper zero-padding - into a 3D matrix from each data sample such that three different 2D configurations are produced as illustrated in Fig.4.10.

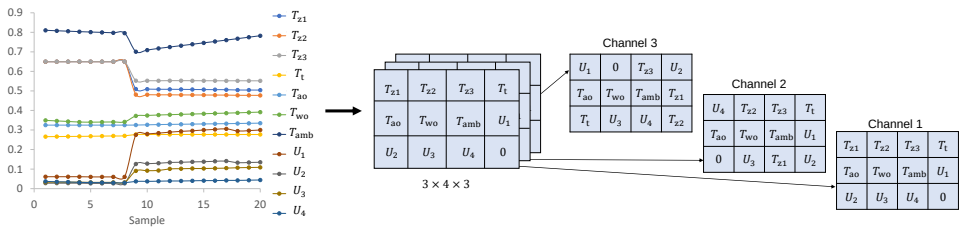


Figure 4.10: Reshaping the 11 system's 1-dimensional data samples into 3D configuration of size 3×4×3.

The arrangement of the system's variables into the 3D matrix is another hyper-parameter to be decided. That is, there are enormous possibilities of how the variables

are configured into a 2D representation and it corresponds to the number of its permutations. For example, for h different elements, the number of possible arrangements corresponds to $h!$. This hyper-parameter can be managed by training the network several times for different permutations generated randomly and then the arrangement that promotes the network for the top performance is chosen.

4.3.2 Convolutional Neural Network Architectures

Networks can be designed shallow or deep based on the number of hidden layers, and thin or wide depending on the kernel size which determines the extent of the local interactions [153]. As presented in Fig.4.11, nine possible CNN architectures are designed. For example, Structures 1-3 use different kernel sizes of 2×1, 2×2, and 2×3 respectively and have one convolution layer and two FC layers. Structures 4-7 are deeper with one convolution layer and three FC layers in structures 4 and 5, and two convolution layers and two FC layers in structures 6 and 7.

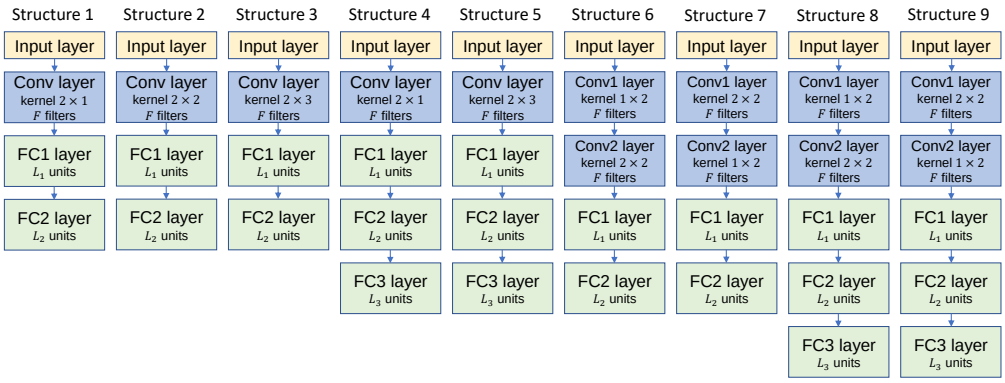


Figure 4.11: Potential network architectures ranging from shallow and thin (left) to deep and wide (right).

4.3.3 CNN-based Actuator Fault Diagnosis Models

The CNN-based actuator fault diagnosis framework is aimed to identify the system condition which has 5 possible classes that are healthy, faulty in actuator 1, faulty in actuator 2, faulty in actuator 3, and faulty in actuator 4. The evaluated models are presented in the following subsections.

4.3.3.1 Model-1

In this model shown in Fig.4.12, a multi-class CNN is trained to diagnose the system state, whether it is fault-free, actuator 1 fault, actuator 2 fault, actuator 3 fault, or actuator 4 fault.

Figure 4.12: FD Model-1 using a multi-class CNN.

4.3.3.2 Model-2

The model shown in Fig.4.13 consists of two Convolutional Neural Networks in a hierarchical framework in which a two-class CNN is trained to identify whether the system is healthy or not and the other multi-class CNN is used to isolate the fault source; actuator 1, actuator 2, actuator 3, or actuator 4.

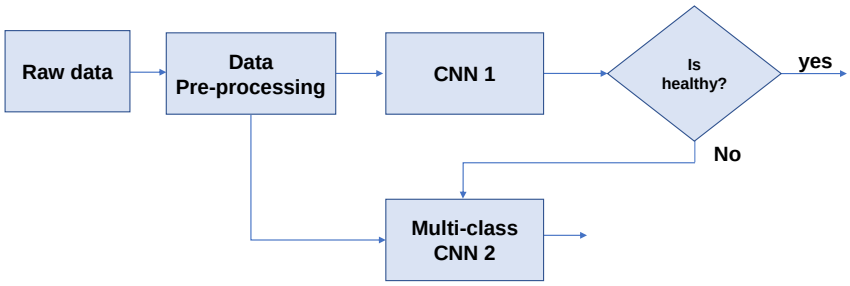


Figure 4.13: FD Model-2 consisting of CNN 1 to determine the system health state and CNN 2 to isolate the fault source.

4.3.3.3 Model-3

It is a multi-model framework composed of 5 two-class CNNs as shown in Fig.4.14. CNN 1 determines whether the system is fault-free or not. The remaining CNNs are trained to diagnose each of the 4 faults independently.

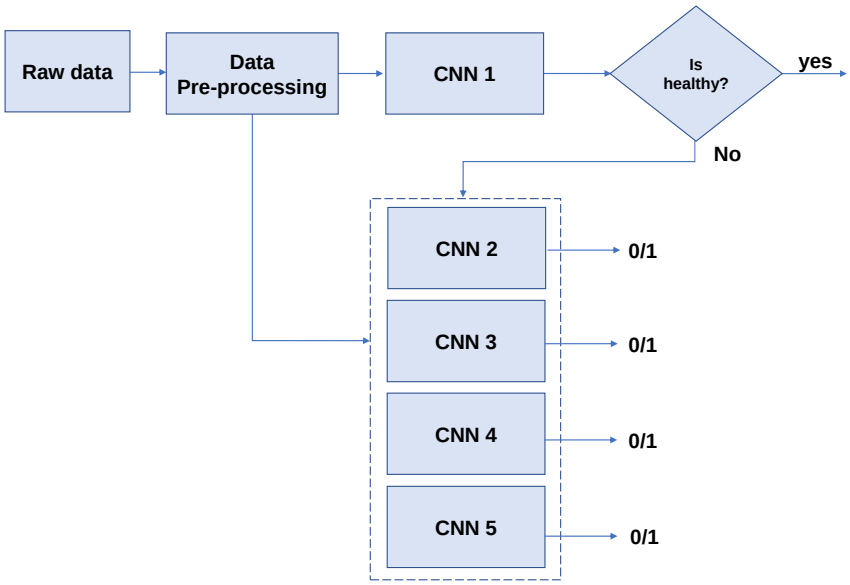


Figure 4.14: FD Model-3 consisting of 5 two-class CNNs for each class of the system state.

4.3.3.4 Model-4

This model is very similar to Model-3 except that CNNs 2-4 share the same convolution layer as presented in Fig.4.15 in an attempt to reduce the computational requirement. The convolution layer is pre-trained to learn the basic features and later on the fully connected layers are trained for each network individually to diagnose each fault type.

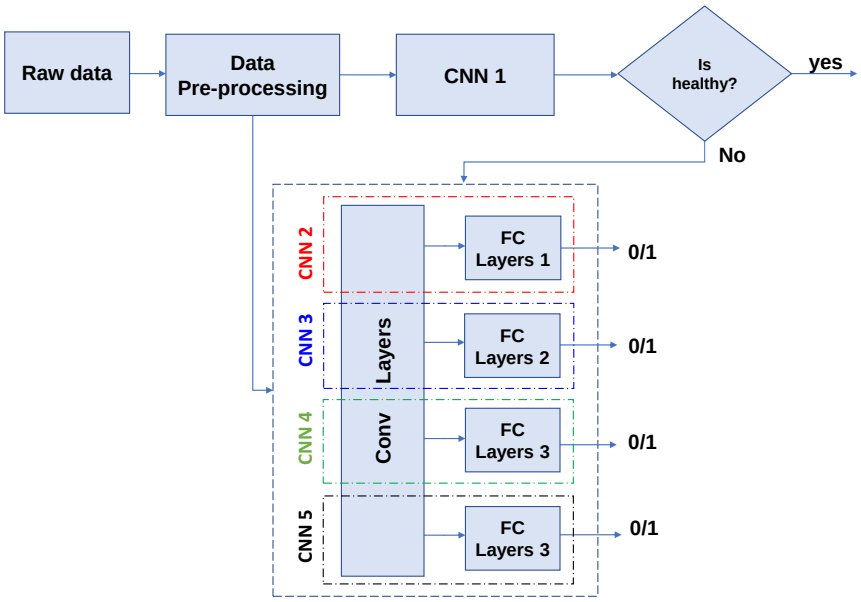


Figure 4.15: FD Model-4 consisting of 5 two-class CNNs for each class of the system state while the last 4 CNNs share the same convolution layer.

4.4 Results of the Proposed CNN-based Actuator FD Models

There are numerous CNNs that can be designed for each of the nine structures in Fig.4.11 based on the hyper-parameter choices, i.e., activation function, number of convolution layer filters, number of FC layer units, BN, regularization, etc. Random hyper-parameter search has demonstrated effective performance [156] and hence it is used in

the design process of the CNNs for the evaluated models. Using MATLAB, the CNN training algorithm used is Adam with piece-wise learning decay and the settings are presented in Table 4.3. The training is carried out using a computer with 8 GB RAM and processor Intel(R) Core(TM) i7-4510U CPU at 2.8 GHz speed. The CNN training is conducted using 7-folds cross-validation and the average folds performance results are demonstrated.

Table 4.3: The settings used for the CNN training.

Number of epochs	Batch size	Initial learning rate
5-15	256	0.05

The trained network is identified by the following: its structure i.e., structure 1, 2,..., or 9, and the dimension of the classification problem, i.e., Two-class, 4-class, or 5-class. For example, CNN-S8-5C-1 means its CNN number 1 of structure 8 for 5-class classification problem. In addition, for two-class CNN, the classes are identified as presented in Table 4.4.

Table 4.4: Labels of classes according to the system state.

System State	Class Identification
Healthy	Class A
Faulty at actuator 1	Class B
Faulty at actuator 2	Class C
Faulty at actuator 3	Class D
Faulty at actuator 4	Class E

4.4.1 Dataset

The data used for Convolutional Neural Networks training and testing are collected from the system simulator using TRNSYS. The total dataset size is 230k samples with 150k samples corresponding to the normal system operation and 20k samples of each of the 4 types of the actuator fault. The types of actuators are 3 VAV box dampers and one water valve. The data are collected at a sampling time of 1 min for 2 months duration for the healthy system operation. The faults data are acquired for 13 days per fault type for different fault scenarios which are stuck at fully closed (FC) position, stuck at fully open (FO) position, and stuck at partially open (PO) position as summarized in Table 4.5.

Table 4.5: Dataset used for Convolutional Neural Network training.

System State	Train dataset (samples)	Test dataset (samples)	Description
Healthy	125607	20934	-
Actuator 1 fault	18175	3029	Stuck damper: FC, FO, PO at 10%, 40%
Actuator 2 fault	18175	3029	Stuck damper: FC, FO, PO at 10%, 50%
Actuator 3 fault	18175	3029	Stuck damper: FC, FO, PO at 20%, 40%
Actuator 4 fault	18175	3029	Stuck valve: FC, FO, PO at 10%, 50%

In data-driven classification approaches, a common issue is encountered with such dataset referred to as data imbalance. When data of some classes are scarce (e.g. faulty data) while for others, data are abundant (e.g. healthy data), the classification performance of the network is altered such that a bias is introduced to the network predictions since the network model will pay more attention to the majority class. The effect of this issue can be minimized by modifying the sampling rate which can be done by majority under-sampling, minority over-sampling, or a combination of both with proper

design of the network structure to avoid over-fitting problems. It is worth noting that data balancing is performed only on the training dataset.

4.4.2 Model-1

In this model, majority-downsampling is used to balance the training dataset such that the sampling time of the healthy data is modified to be 7 min. The performance of the CNN is moderate as presented in Table 4.6 with a maximum accuracy of 94.95% and F_1-score of 92.99% achieved by CNN-S9-5C-2 with the details presented in Fig.4.16. The details of the rest of the evaluated networks are presented in Appendix 5.1. The network ability to correctly identify data classes is inadequate which is interpreted by the CNN's sensitivity of around 93%. This can be attributed to the insufficient amount of data to train the network for the multi-classification problem with this level of complexity.

Figure 4.16: CNN-S9-5C-2 structure.

Table 4.6: Performance results of the trained CNN for Model-1.

Network	Accuracy	Error	Sensitivity	Specificity	Precision	F_1-score
CNN-S8-5C-1	0.9292	0.0708	0.9304	0.9775	0.9147	0.9169
CNN-S9-5C-2	0.9495	0.0505	0.9328	0.9825	0.9340	0.9299
CNN-S7-5C-3	0.9410	0.0590	0.9366	0.9811	0.9200	0.9238
CNN-S6-5C-4	0.9328	0.0672	0.9306	0.9777	0.9154	0.9188
CNN-S5-5C-5	0.9364	0.0636	0.9321	0.9793	0.9205	0.9199
CNN-S4-5C-6	0.9108	0.0892	0.9173	0.9737	0.8794	0.8904
CNN-S3-5C-7	0.9413	0.0587	0.9232	0.9785	0.9322	0.9221
CNN-S2-5C-8	0.9385	0.0615	0.9224	0.9780	0.9370	0.9217
CNN-S1-5C-9	0.9221	0.0779	0.9209	0.9743	0.9089	0.9061

4.4.3 Model-2

The 4-class CNN for Model-2 demonstrates improved results as presented in Table 4.7. The network with the architecture presented in Fig.4.17 is capable of approximating the functional mapping between the input and output up to an accuracy of 96.51%, F_1-score of 97.21%, and sensitivity of 97.21%. The details of the rest of the evaluated networks are presented in Appendix 5.2.

Figure 4.17: CNN-S5-4C-5 structure.

Table 4.7: Performance results of the trained CNN2 for Model-2.

Network	Accuracy	Error	Sensitivity	Specificity	Precision	F_1-score
CNN-S8-4C-1	0.9488	0.0512	0.9590	0.9863	0.9617	0.9586
CNN-S9-4C-2	0.9506	0.0494	0.9604	0.9868	0.9631	0.9598
CNN-S7-4C-3	0.9570	0.0430	0.9656	0.9885	0.9679	0.9654
CNN-S6-4C-4	0.9632	0.0368	0.9705	0.9902	0.9725	0.9704
CNN-S5-4C-5	0.9651	0.0349	0.9721	0.9907	0.9748	0.9721
CNN-S4-4C-6	0.9636	0.0364	0.9709	0.9903	0.9725	0.9709
CNN-S3-4C-7	0.9612	0.0388	0.9612	0.9871	0.9641	0.9607
CNN-S2-4C-8	0.9563	0.0437	0.9563	0.9854	0.9581	0.9558
CNN-S1-4C-9	0.9612	0.0388	0.9612	0.9871	0.9641	0.9607

4.4.4 Model-3

The training of each CNN in Model-3 is carried out using all the classes data such that the targets are labeled positive for the concerned class, and negative otherwise with proper data balancing. For example, for a CNN trained to identify Class A, the data samples of Classes B to E are labeled as negative outputs and the healthy data are identified with positive labels and down-sampled to avoid data imbalance, such that the number of negative and positive samples are equal. For training the CNNs for Classes B to E, both majority down-sampling for the healthy data and minority up-sampling for the concerned class data are used to balance the dataset.

Satisfactory network performance is achieved as summarized in Table 4.8 and the best results are obtained with the simple CNN structures 2 and 3 as highlighted in light gray color, with an average accuracy and F_1-score of 98%. For the limited available

data, the network capability in identifying the classes is improved since the classification problem is simpler with only two possibilities for the output (1 or 0). The detailed architecture of the networks is illustrated in Fig.4.18 and presented in Table 4.9. The details of the rest of the evaluated networks are presented in Appendix 5.3.

Table 4.8: Performance results of the trained CNNs for Model-3.

	Network	Accuracy	Error	Sensitivity	Specificity	Precision	F_1-score
Class A	CNN-S8-1C-A1	0.9572	0.0428	0.9814	0.9814	0.9829	0.9817
	CNN-S7-1C-A2	0.9531	0.0469	0.9791	0.9791	0.9807	0.9797
	CNN-S4-1C-A3	0.9523	0.0477	0.9782	0.9782	0.9815	0.9794
	CNN-S3-1C-A4	0.9831	0.0169	0.9614	0.9956	0.9922	0.9765
	CNN-S2-1C-A5	0.9833	0.0167	0.9632	0.9949	0.9909	0.9769
Class B	CNN-S7-1C-B1	0.9666	0.0334	0.9833	0.9833	0.9738	0.9733
	CNN-S3-1C-B2	0.9871	0.0129	0.9945	0.9139	0.9914	0.9929
	CNN-S2-1C-B3	0.9915	0.0085	0.9980	0.9275	0.9928	0.9954
	CNN-S2-1C-B4	0.9882	0.0118	0.9954	0.9165	0.9917	0.9935
Class C	CNN-S7-1C-C1	0.9690	0.0310	0.9794	0.9794	0.9634	0.9682
	CNN-S3-1C-C2	0.9850	0.0150	0.9941	0.8961	0.9896	0.9918
	CNN-S2-1C-C3	0.9913	0.0087	0.9982	0.9233	0.9923	0.9952
	CNN-S2-1C-C4	0.9845	0.0155	0.9929	0.9026	0.9902	0.9915
Class D	CNN-S7-1C-D1	0.9949	0.0051	0.9942	0.9942	0.9937	0.9939
	CNN-S3-1C-D2	0.9966	0.0034	0.9984	0.9794	0.9979	0.9981
	CNN-S2-1C-D3	0.9967	0.0033	0.9982	0.9816	0.9981	0.9982
	CNN-S2-1C-D4	0.9973	0.0027	0.9987	0.9840	0.9984	0.9985
Class E	CNN-S7-1C-E1	0.9774	0.0226	0.9807	0.9807	0.9730	0.9744
	CNN-S3-1C-E2	0.9841	0.0159	0.9908	0.9174	0.9918	0.9913
	CNN-S2-1C-E3	0.9839	0.0161	0.9902	0.9218	0.9923	0.9912
	CNN-S2-1C-E4	0.9825	0.0175	0.9888	0.9199	0.9921	0.9903

Figure 4.18: Architecture of the CNN with best performance for Model-3.

Table 4.9: Details of the selected CNN architectures for Model-3.

Network	Conv1 layer		FC1 layer
	Kernel size	Number of Filter	Number of units
CNN-S2-1C-A5	2×2	6	7
CNN-S2-1C-B3	2×2	6	9
CNN-S2-1C-C3	2×2	5	10
CNN-S2-1C-D4	2×2	8	7
CNN-S3-1C-E2	2×3	10	6

4.4.5 Model-4

The training of the CNN for this model is conducted using the convolution layer from the pre-trained networks obtained from Model-3. Table 4.10 presents the evaluation results of the best CNNs for the 4 fault classes and the specification of the design of the network is presented in Fig.4.19. The kernel of the convolution layer is of size $2 \times 2 \times 8$ and the number of nodes in the FC layer is 15 units for CNN-Class C and CNN-Class D, and 25 units for CNN-Class B and CNN-Class E. The advantage of this model is reducing the amount of the required computation while achieving satisfactory classification performance.

Figure 4.19: Architecture of the CNN with best performance for Model-4.

Table 4.10: Performance results of the trained CNNs for Model-4.

Network	Accuracy	Error	Sensitivity	Specificity	Precision	F_1-score
CNN-Class B	0.9900	0.0100	0.9930	0.9602	0.9960	0.9945
CNN-Class C	0.9823	0.0178	0.9865	0.9405	0.9940	0.9902
CNN-Class D	0.9924	0.0076	0.9962	0.9550	0.9955	0.9958
CNN-Class E	0.9794	0.0206	0.9887	0.8871	0.9888	0.9886

4.4.6 Comparison Between Evaluated CNN-based Models

According to the evaluation results of the four models, the best performance is achieved by Model-3 which encompasses a number of CNNs corresponding to the number of classes and each CNN is trained to perform binary classification of "belongs to the class or not". However, it can be computationally demanding especially for large-scale buildings. This issue can be minimized by Model 4 while still maintaining just about the same performance.

Even though Models 1 and 2 demand less processing requirements, their classification capabilities are found to be limited and insufficient. Nevertheless, their performance is expected to improve for larger buildings because the increase in the number of correlated variables shall promote the network learning capacity.

Depending on the building size and design specifications -i.e. zoning, etc.-, a combination of the evaluated models can be deployed in the final fault diagnosis scheme.

For example, a two-class CNN can be designed to determine whether the system is fault-free or not and dedicated CNNs using Model-2 or Model-4 can be designed for the fault types with similar attributes.

4.4.7 Evaluation of the Proposed CNN-based Actuator FD Scheme

This subsection presents case studies demonstrating the performance of the CNN-based diagnosis scheme for actuator faults. The presented evaluation is carried out using a dedicated test dataset that is generated for the case studies in which increased measurement noise is introduced to the data. The noise is emulated by an additive random uniformly distributed variable with a maximum level of 0.5 °C. Table 4.11 presents the evaluated faults scenarios used for testing the proposed CNN-based diagnosis approach.

The first case study evaluating the faults F-VAV1-High to F-Valve-High presents an actuator fault with a high severity level. For instance, the VAV damper is stuck at a fully closed position at the night during which the load is minimal causing the HVAC system to fail to meet the cooling load at the daytime and vice versa. The second case study presents a low impact fault event such that the actuator is stuck at a position that does not severely impede the HVAC system operation in fulfilling the thermal load requirement. The performance evaluation is conducted on the CNN-based framework using Model-3 presented in Section 4.4.4. Table 4.12 summarizes the evaluation results of the average performance of the five CNNs in Model-3 under each fault scenario. The evaluation is made upon the following metrics, the diagnosis accuracy, the false alarms rate representing the false positive rate, and the missed alarms rate indicating the false negative rate. It is based on an observation window of 1 day under the assumption that the fault source will be abolished within that time period.

Table 4.11: Actuator fault scenarios used for the evaluation of the proposed CNN-based FD scheme.

Fault identifier	Description
F-VAV1-High	Stuck damper at FO position at the beginning of daytime
F-VAV2-High	Stuck damper at FC position during night
F-VAV3-High	Stuck damper at FC position during night
F-Valve-High	Stuck valve at FC position during minimum thermal load
F-VAV1-Low	Stuck damper at PO at 18% opening
F-VAV2-Low	Stuck damper at PO at 56% opening

For the high severity level fault events demonstrated in Figures 4.20 to 4.23, it is observed that the diagnosis is quick and accurate in most of the cases with about 100% average diagnosis accuracy of the 5 CNNs and almost no false or missed alarms. However, for the F-VAV2-High fault, an increased rate of false alarms is observed of around 5% with an average accuracy of 96% as shown in Fig.4.21. The effect of those outliers on the reliability of the diagnosis approach can be mitigated by applying statistical analysis on the CNN outputs similar to the one proposed in [157]. It aims to determine the probability of the fault occurrence based on the frequency of positive flags raised by the CNN within a specified time period with careful attention to avoid compromising the diagnosis time. It is worth noting that the plots of CNNs outputs are down-sampled for demonstration purposes.

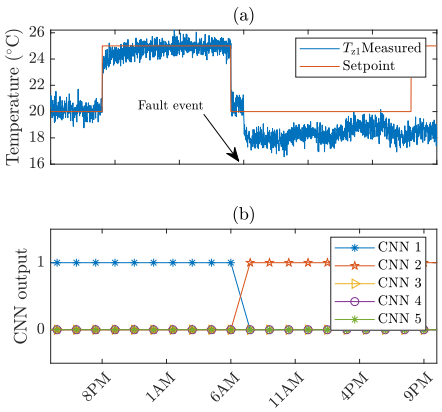


Figure 4.20: CNN-based diagnosis method performance due to stuck F-VAV1-High fault at 100% open position at the beginning of the day. (a) Zone 1 temperature. (b) CNNs outputs.

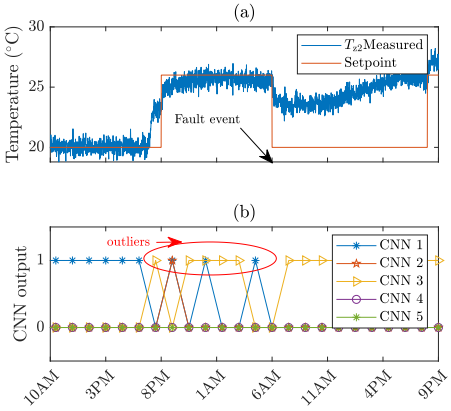


Figure 4.21: CNN-based diagnosis method performance due to stuck F-VAV2-Hight fault at 100% closed position at night. (a) Zone 2 temperature. (b) CNNs outputs.

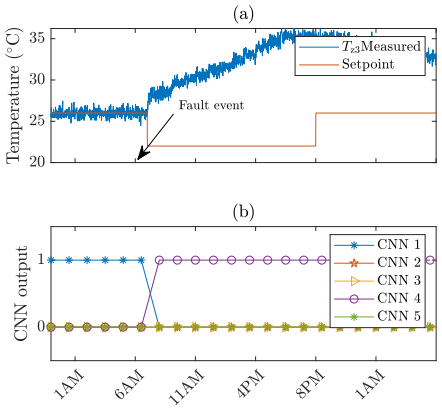


Figure 4.22: CNN-based diagnosis method performance due to F-VAV3-High fault at 100% closed position at night. (a) Zone 3 temperature. (b) CNNs outputs.

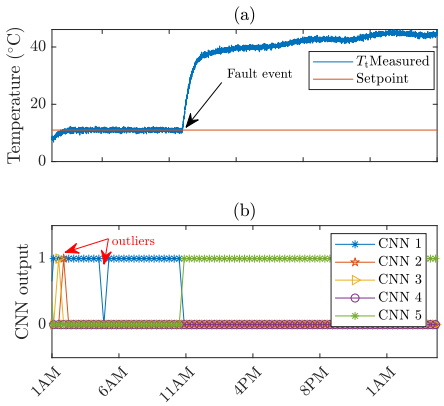


Figure 4.23: CNN-based diagnosis method performance due to F-Valve-High fault at 100% closed position. (a) Tank temperature. (b) CNNs outputs.

Figures 4.24 and 4.25 represent the diagnosis results of fault incidents with low impact level. The damper in each of the two scenarios is stuck at a non-critical time of the operation. For instance, in F-VAV1-Low fault, the damper is stuck at 18% partially open position which was sufficient to meet the low thermal load requirement at

the night time. Hence, it is observed that the diagnosis time is long since the fault is identified when it impacts the system states which results in the observed increase in the missed alarms rate of around 6% and the lower diagnostic accuracy with an average of 89%. The overall performance of the CNN-based proposed diagnosis approach is found stationarity such that it is quick, robust to measurement noise, reliable, and accurate.

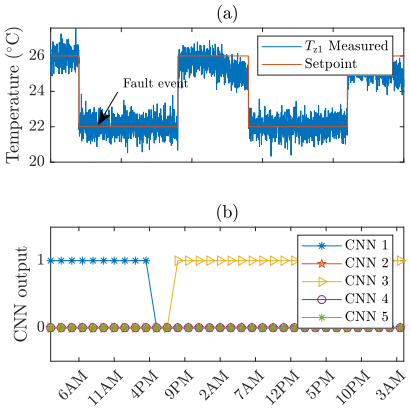


Figure 4.24: CNN-based diagnosis method performance due to stuck F-VAV1-Low fault at 18% open position. (a) Zone 1 temperature. (b) CNNs outputs.

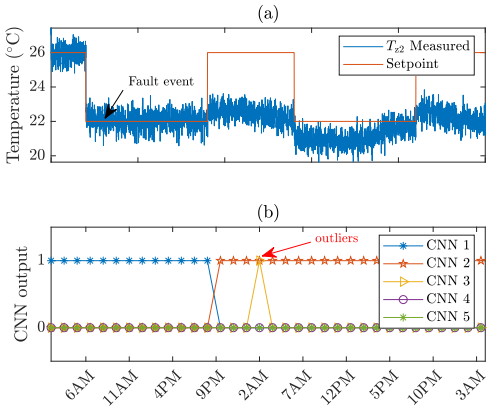


Figure 4.25: CNN-based diagnosis method performance due to stuck F-VAV2-Low fault at 56% open position. (a) Zone 2 temperature. (b) CNNs outputs.

Table 4.12: Evaluation results of the proposed CNN-based FD scheme for the different actuator fault testing scenarios.

Fault identifier	Diagnosis accuracy	False alarms rate	Missed alarms rate	Diagnosis time
F-VAV1-High	99.99%	0.01%	0.00%	30 min
F-VAV2-High	96.67%	5.01%	2.12%	60 min
F-VAV3-High	100.00%	0.00%	0.00%	2 min
F-Valve-High	99.85%	0.14%	0.00%	23 min
F-VAV1-Low	91.57%	4.49%	7.11%	8 hr
F-VAV2-Low	88.01%	6.54%	5.89%	7 hr

4.5 Comparison Between the Proposed CNN-based Actuator FD Method and Others

This section presents the performance evaluation of the proposed CNN-based actuator fault diagnosis method against SVM and neural network. The analysis is carried out for Model-3 and the results are presented in Table 4.13.

It is found that the CNN-based FD diagnosis scheme demonstrates improved performance when compared with the NN-based and SVM-based approaches. That is, the average sensitivity (or recall) of Model-3 using the CNN is about 99% while it is 95% for SVM and NN with an overall performance improvement of around 4%. The value of recall is inversely proportional to the missed alarms rate and hence this means that using CNNs elevates the reliability of the fault diagnosis. In addition, the average diagnosis accuracy is higher using CNN with around 99% while it is 96% and 95% for SVM and

NN respectively. The average diagnosis precision of the CNN-based approach is 99% with an increase of 18% and 2% compared to SVM and NN respectively. Moreover, the binary classification quality represented by the MCC value is improved by about 11% with an average score of 95% using the CNN while it is 86% for SVM and NN.

Table 4.13: Comparison results between using NN, SVM, and CNN for Model-3 actuator fault diagnosis scheme.

Method	Network	Accuracy	Error	Sensitivity	Specificity	Precision	F_1-score	MCC
NN	CNN1	0.9010	0.0990	0.8516	0.9404	0.9194	0.8842	0.7997
	CNN2	0.9823	0.0177	0.9874	0.9418	0.9927	0.9900	0.9124
	CNN3	0.9838	0.0162	0.9913	0.9236	0.9905	0.9909	0.9177
	CNN4	0.9816	0.0184	0.9994	0.8393	0.9803	0.9898	0.9042
	CNN5	0.9298	0.0702	0.9211	0.9994	0.9999	0.9589	0.7508
SVM	CNN1	0.9258	0.0742	0.9832	0.8539	0.8940	0.9365	0.8534
	CNN2	0.9794	0.0206	0.9560	0.9823	0.8711	0.9115	0.9012
	CNN3	0.9703	0.0297	0.9431	0.9737	0.8174	0.8757	0.8618
	CNN4	0.9867	0.0133	0.8855	0.9993	0.9940	0.9366	0.9312
	CNN5	0.9389	0.0611	0.9846	0.9332	0.6478	0.7815	0.7699
CNN	CNN1	0.9833	0.0167	0.9632	0.9949	0.9909	0.9769	0.9640
	CNN2	0.9915	0.0085	0.9980	0.9275	0.9928	0.9954	0.9485
	CNN3	0.9913	0.0087	0.9982	0.9233	0.9923	0.9952	0.9461
	CNN4	0.9973	0.0027	0.9987	0.9840	0.9984	0.9985	0.9841
	CNN5	0.9841	0.0159	0.9908	0.9174	0.9918	0.9913	0.9054

4.6 Summary

A CNN-based actuator fault diagnosis strategy is developed in this chapter for stuck dampers and water valve faults in the HVAC system. Different models (e.g. multi-model, and single model) and CNN architectures of the proposed scheme are tested and compared. This analysis exploits most of the possible configurations of the CNN-based diagnosis method that can be deployed for HVAC systems. That is, there is a trade-off

between the performance of the model and the computational requirement. For instance, Model-3 demonstrates the best option in terms of accuracy, precision, and recall while Model-4 is a less computationally demanding version with some performance accuracy compromise. In terms of the Convolutional Neural Network architecture, depending on the availability of training data, there is a wide range of network designs that can be used with careful attention to avoid over-fitting problem.

In addition, the CNN-based approach is compared with SVM and Neural Network and the proposed method demonstrates improved performance in terms the diagnosis accuracy, precision, and recall. The use of the Convolutional Neural Network for HVAC system actuator fault diagnosis is proven to be successful in improving the accuracy, reliability, and precision of the fault diagnosis by an average of 4%, 11%, and 10% respectively when compared to standard Neural Networks and Support Vector Machine which are commonly used algorithms.

Chapter 5: Summary and Future Work

Given the fact that the HVAC systems account for 40% of the total building energy usage, the work presented in this report is concerned with the development of fault diagnosis for HVAC systems to optimize their performance and maximize their efficiency given the increasing global energy demand. Considering the underlying constraints and complexity associated with the HVAC system models, this work presented two data-driven fault diagnosis strategies for HVAC system sensor and actuator fault diagnosis that do not require the knowledge of the system mathematical model but rather utilize the building historical data that can be obtained from the modern building management systems.

The first part of the thesis presented the proposed AANN-based sensor data validation and fault diagnosis technique that can be used to validate sensor measurements in terms of missing sensor data replacement, sensor fault and inaccuracy correction, and sensor data noise reduction for the closed-loop control of the HVAC system to maintain its reliable performance. Moreover, the AANN-based sensor fault diagnosis method showed a capability in diagnosing both single and multiple bias and drift faults effectively. It addressed the limitations of the previous works and showed notable overall performance improvements when compared with the commonly used method that is PCA-based. The second part presented an actuator fault diagnosis approach for HVAC systems using 2D Convolutional Neural Networks by proposing an efficient and real-time data transformation to obtain the 2D representation of the raw 1D measurement data of the HVAC system variables. The CNN-based actuator fault diagnosis method demonstrated a successful deployment of the newly evolving field of Machine Learning for condition monitoring and fault diagnosis applications by showing an improved per-

formance when compared with the commonly used methods which are NN and SVM. It addressed the limitations of the past works in terms of the accuracy, reliability, and precision.

Both methods are robust and they can be applied for large-scale buildings with additional zones and it is expected that the performance will be improved with the increase in the HVAC system size due to a higher correlation between the inputs. The use of a multi-agent fault diagnosis framework is suggested for large-scale buildings in order to compensate for the increased computational requirement of the proposed algorithms. There are two limitation associated with the proposed approaches. Firstly, data-driven methods require sufficient volumes of representative data of the system of interest to ensure a reliable performance. The second limitation is regarding the training phase which is time consuming and computationally demanding.

The future work would be considering the integration of multiple simultaneous actuator faults as well as the development of data-driven fault diagnosis strategy for simultaneous sensor and actuator faults diagnosis. In addition, energy performance assessment of the HVAC system can be conducted to demonstrate the diagnosis approach capability in improving the system efficiency. Moreover, for the AANN-based sensor fault diagnosis method, a Machine Learning-based technique can be developed for threshold selection.

REFERENCES

[1] S. Hamdaoui, M. Mahdaoui, A. Allouhi, R. E. Alaiji, T. Kousksou, and A. E. Bouardi, "Energy demand and environmental impact of various construction scenarios of an office building in morocco," *Journal of Cleaner Production*, vol. 188, pp. 113 – 124, 2018.

[2] M. Sewak, M. R. Karim, and P. Pujari, *Practical Convolutional Neural Networks: Implement Advanced Deep Learning Models Using Python*. Packt Publishing, 2018.

[3] H.-I. Suk, "Deep learning for medical image analysis," Academic Press, 2017.

[4] I. E. Agency, "Energy technology perspectives," 2017.

[5] O. F. E. COOPERATION and DEVELOPMENT, *ENERGY TECHNOLOGY PERSPECTIVES 2017: catalysing energy technology transformations*. OECD, 2017.

[6] R. McDowall, *Fundamentals of HVAC systems*. American Society of Heating, Refrigerating and Air-Conditioning Engineers, 2007.

[7] C. F. Alcala and S. J. Qin, "Analysis and generalization of fault diagnosis methods for process monitoring," *Journal of Process Control*, vol. 21, no. 3, pp. 322 – 330, 2011. Thomas McAvoy Festschrift.

[8] F. S. Merritt and J. Ambrose, *Building Engineering and Systems Design*. Boston, MA: Springer, 1989.

[9] V. Reppa, P. Papadopoulos, M. M. Polycarpou, and C. G. Panayiotou, "A dis-

tributed architecture for HVAC sensor fault detection and isolation," *IEEE Transactions on Control Systems Technology*, vol. 23, pp. 1323–1337, July 2015.

[10] Z. Gao, C. Cecati, and S. X. Ding, "A survey of fault diagnosis and fault-tolerant techniques; part I: Fault diagnosis with model-based and signal-based approaches," *IEEE Transactions on Industrial Electronics*, vol. 62, pp. 3757–3767, June 2015.

[11] S. Simani, C. Fantuzzi, and R. Patton, *Model-based fault diagnosis in dynamic systems using identification techniques*. Springer, 2003.

[12] Z. Gao, C. Cecati, and S. X. Ding, "A survey of fault diagnosis and fault-tolerant techniques; part II: Fault diagnosis with knowledge-based and hybrid/active approaches," *IEEE Transactions on Industrial Electronics*, vol. 62, pp. 3768–3774, June 2015.

[13] T. Darure, J. J. Yamé, and F. Hamelin, "Fault-adaptive control of VAV damper stuck in a multizone building," in *2016 3rd Conference on Control and Fault-Tolerant Systems (SysTol)*, pp. 170–176, Sept 2016.

[14] T. Darure, J. J. Yamé, and F. Hamelin, "Model-based fault-tolerant control of VAV damper lock-in place failure in a multizone building," in *2016 14th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pp. 1–6, Nov 2016.

[15] K. Srinivasarengan, J. Ragot, D. Maquin, and C. Aubrun, "Nonlinear joint state-parameter observer for VAV damper position estimation," in *2016 3rd Conference on Control and Fault-Tolerant Systems (SysTol)*, pp. 164–169, Sept 2016.

[16] B. T. Thumati, M. A. Feinstein, J. W. Fonda, A. Turnbull, F. J. Weaver, M. E. Calkins, and S. Jagannathan, "An online model-based fault diagnosis scheme for HVAC systems," in *2011 IEEE International Conference on Control Applications (CCA)*, pp. 70–75, Sept 2011.

[17] N. Tudoroiu and M. Zaheeruddin, "Joint UKF estimator for fault detection and isolation in heating ventilation and air conditioning systems," in *2011 5th International Symposium on Computational Intelligence and Intelligent Informatics (ISCIII)*, pp. 53–58, Sept 2011.

[18] M. Zaheeruddin and N. Tudoroiu, "Dual EKF estimator for fault detection and isolation in heating ventilation and air conditioning systems," in *IECON 2012 - 38th Annual Conference on IEEE Industrial Electronics Society*, pp. 2257–2262, Oct 2012.

[19] M. Bonvini, M. D. Sohn, J. Granderson, M. Wetter, and M. A. Piette, "Robust on-line fault detection diagnosis for HVAC components based on nonlinear state estimation techniques," *Applied Energy*, vol. 124, no. Supplement C, pp. 156 – 166, 2014.

[20] N. Tudoroiu, M. Zaheeruddin, E. R. Tudoroiu, and V. Jeflea, "Fault detection and diagnosis (FDD) in heating ventilation air conditioning systems (HVAC) using an interactive multiple model augmented unscented kalman filter (IMMAUKF)," in *2008 Conference on Human System Interactions*, pp. 334–339, May 2008.

[21] N. Tudoroiu, M. Zaheeruddin, C. Chiru, M. Grigore, and E. R. Tudoroiu, "Unscented kalman filter (UKF) and frequency analysis (FA) techniques used for fault detection, diagnosis and isolation (FDDI) in heating ventilation air condi-

tioning systems (HVAC)-comparison results," in *2009 2nd Conference on Human System Interactions*, pp. 442–449, May 2009.

[22] P. M. Papadopoulos, V. Reppa, M. M. Polycarpou, and C. G. Panayiotou, "Distributed adaptive estimation scheme for isolation of sensor faults in multi-zone HVAC systems," *IFAC-PapersOnLine*, vol. 48, no. 21, pp. 1146 – 1151, 2015. 9th IFAC Symposium on Fault Detection, Supervision andSafety for Technical Processes SAFEPROCESS 2015.

[23] P. M. Papadopoulos, V. Reppa, M. M. Polycarpou, and C. G. Panayiotou, "Distributed diagnosis of actuator and sensor faults in HVAC systems," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 4209 – 4215, 2017.

[24] P. Wang, X. Tang, and R. Gao, "Automated performance tracking for heat exchangers in HVAC," in *2015 IEEE International Conference on Automation Science and Engineering (CASE)*, pp. 949–954, Aug 2015.

[25] P. Wang and R. X. Gao, "Automated performance tracking for heat exchangers in HVAC," *IEEE Transactions on Automation Science and Engineering*, vol. 14, pp. 634–645, April 2017.

[26] D. He, C. Jiang, R. G. Harley, T. G. Habetler, and R. Ding, "A new electrical method on airflow fault detection of air handling unit (AHU)," in *2015 Clemson University Power Systems Conference (PSC)*, pp. 1–5, March 2015.

[27] C. Jiang, T. G. Habetler, and W. p. Cao, "Improved condition monitoring of the faulty blower wheel driven by brushless DC motor in air handler unit (AHU)," in *2016 IEEE Energy Conversion Congress and Exposition (ECCE)*, pp. 1–5, Sept 2016.

[28] N. Tudoroiu and M. Zaheeruddin, "Fault detection and diagnosis of the valve actuators in HVAC systems, using frequency analysis," in *2005 International Conference on Industrial Electronics and Control Applications*, pp. 4 pp.–4, 2005.

[29] M. Najafi, D. M. Auslander, P. L. Bartlett, P. Haves, and M. D. Sohn, "Application of machine learning in the fault diagnostics of air handling units," *Applied Energy*, vol. 96, pp. 347 – 358, 2012. Smart Grids.

[30] M. Hu, H. Chen, L. Shen, G. Li, Y. Guo, H. Li, J. Li, and W. Hu, "A machine learning bayesian network for refrigerant charge faults of variable refrigerant flow air conditioning system," *Energy and Buildings*, 2017.

[31] S. Sun, G. Li, H. Chen, Q. Huang, S. Shi, and W. Hu, "A hybrid ICA-BPNN-based FDD strategy for refrigerant charge faults in variable refrigerant flow system," *Applied Thermal Engineering*, vol. 127, no. Supplement C, pp. 718 – 728, 2017.

[32] K. Sun, G. Li, H. Chen, J. Liu, J. Li, and W. Hu, "A novel efficient SVM-based fault diagnosis method for multi-split air conditioning system's refrigerant charge fault amount," *Applied Thermal Engineering*, vol. 108, no. Supplement C, pp. 989 – 998, 2016.

[33] S. Shi, G. Li, H. Chen, Y. Hu, X. Wang, Y. Guo, and S. Sun, "An efficient VRF system fault diagnosis strategy for refrigerant charge amount based on PCA and dual neural network model," *Applied Thermal Engineering*, vol. 129, pp. 1252 – 1262, 2018.

[34] G. Li, H. Chen, Y. Hu, J. Wang, Y. Guo, J. Liu, H. Li, R. Huang, H. Lv, and J. Li, "An improved decision tree-based fault diagnosis method for practical vari-

able refrigerant flow system using virtual sensor-based fault indicators," *Applied Thermal Engineering*, 2017.

[35] S. Li and J. Wen, "Application of pattern matching method for detecting faults in air handling unit system," *Automation in Construction*, vol. 43, pp. 49 – 58, 2014.

[36] Y. Yan, P. B. Luh, and K. R. Pattipati, "Fault diagnosis framework for air handling units based on the integration of dependency matrices and PCA," in *2014 IEEE International Conference on Automation Science and Engineering (CASE)*, pp. 1103–1108, Aug 2014.

[37] Y. Guo, G. Li, H. Chen, Y. Hu, H. Li, L. Xing, and W. Hu, "An enhanced PCA method with savitzky-golay method for VRF system sensor fault detection and diagnosis," *Energy and Buildings*, vol. 142, no. Supplement C, pp. 167 – 178, 2017.

[38] Z. Du, X. Jin, and L. Wu, "Fault detection and diagnosis based on improved PCA with JAA method in VAV systems," *Building and Environment*, vol. 42, no. 9, pp. 3221 – 3232, 2007.

[39] L. Xuemei, S. Ming, D. Lixing, X. Gang, and L. Jibin, "Novel HVAC fan machinery fault diagnosis method based on KPCA and SVM," in *2009 International Conference on Industrial Mechatronics and Automation*, pp. 492–496, May 2009.

[40] L. Xuemei, D. Lixing, L. Jincheng, and X. Gang, "Combining KPCA and LSSVM for HVAC fan machinery fault recognition," in *2009 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 1241–1246, Dec 2009.

[41] R. Yan, Z. Ma, G. Kokogiannakis, and Y. Zhao, "A sensor fault detection strategy for air handling units using cluster analysis," *Automation in Construction*, vol. 70, pp. 77 – 88, 2016.

[42] Z. Du, B. Fan, J. Chi, and X. Jin, "Sensor fault detection and its efficiency analysis in air handling unit using the combined neural networks," *Energy and Buildings*, vol. 72, pp. 157 – 166, 2014.

[43] Z. Du, B. Fan, X. Jin, and J. Chi, "Fault detection and diagnosis for buildings and HVAC systems using combined neural networks and subtractive clustering analysis," *Building and Environment*, vol. 73, no. Supplement C, pp. 1 – 11, 2014.

[44] A. Beghi, L. CecChinato, L. Corso, M. Rampazzo, and F. Simmini, "Process history-based fault detection and diagnosis for VAVAC systems," in *2013 IEEE International Conference on Control Applications (CCA)*, pp. 1165–1170, Aug 2013.

[45] P. M. V. Every, M. Rodriguez, C. B. Jones, A. A. Mammoli, and M. Martínez-Ramón, "Advanced detection of HVAC faults using unsupervised SVM novelty detection and gaussian process models," *Energy and Buildings*, vol. 149, pp. 216 – 224, 2017.

[46] B. Fan, Z. Du, X. Jin, X. Yang, and Y. Guo, "A hybrid FDD strategy for local system of AHU based on artificial neural network and wavelet analysis," *Building and Environment*, vol. 45, no. 12, pp. 2698 – 2708, 2010.

[47] W. H. Allen and A. Rubaai, "Fuzzy-neuro health monitoring system for HVAC

system variable-air-volume unit," in *2013 IEEE Industry Applications Society Annual Meeting*, pp. 1–8, Oct 2013.

[48] W. H. Allen, A. Rubaai, and R. Chawla, "Fuzzy neural network-based health monitoring for HVAC system variable-air-volume unit," *IEEE Transactions on Industry Applications*, vol. 52, pp. 2513–2524, May 2016.

[49] Y. Huang, N. Li, Y. Huang, and Y. Shi, "Automated fault detection and diagnosis for an air handling unit based on a GA-trained RBF network," in *2006 International Conference on Communications, Circuits and Systems*, vol. 3, pp. 2038–2041, June 2006.

[50] Y. Zhao, J. Wen, F. Xiao, X. Yang, and S. Wang, "Diagnostic bayesian networks for diagnosing air handling units faults – Part I: Faults in dampers, fans, filters and sensors," *Applied Thermal Engineering*, vol. 111, no. Supplement C, pp. 1272 – 1286, 2017.

[51] Y. Zhao, J. Wen, and S. Wang, "Diagnostic bayesian networks for diagnosing air handling units faults – Part II: Faults in coils and sensors," *Applied Thermal Engineering*, vol. 90, no. Supplement C, pp. 145 – 157, 2015.

[52] A. Beghi, R. Brignoli, L. Cecchinato, G. Menegazzo, M. Rampazzo, and F. Simmini, "Data-driven fault detection and diagnosis for HVAC water chillers," *Control Engineering Practice*, vol. 53, pp. 79 – 91, 2016.

[53] A. Beghi, R. Brignoli, L. Cecchinato, G. Menegazzo, and M. Rampazzo, "A data-driven approach for fault diagnosis in HVAC chiller systems," in *2015 IEEE Conference on Control Applications (CCA)*, pp. 966–971, Sept 2015.

[54] S. He, Z. Wang, Z. Wang, X. Gu, and Z. Yan, "Fault detection and diagnosis of chiller using bayesian network classifier with probabilistic boundary," *Applied Thermal Engineering*, vol. 107, pp. 37 – 47, 2016.

[55] D. Li, Y. Zhou, G. Hu, and C. J. Spanos, "Fault detection and diagnosis for building cooling system with a tree-structured learning method," *Energy and Buildings*, vol. 127, no. Supplement C, pp. 540 – 551, 2016.

[56] D. Li, G. Hu, and C. J. Spanos, "A data-driven strategy for detection and diagnosis of building chiller faults using linear discriminant analysis," *Energy and Buildings*, vol. 128, pp. 519 – 529, 2016.

[57] S. M. Namburu, M. S. Azam, J. Luo, K. Choi, and K. R. Pattipati, "Data-driven modeling, fault diagnosis and optimal sensor selection for HVAC chillers," *IEEE Transactions on Automation Science and Engineering*, vol. 4, pp. 469–473, July 2007.

[58] K. Choi, S. M. Namburu, M. S. Azam, J. Luo, K. R. Pattipati, and A. Patterson-Hine, "Fault diagnosis in HVAC chillers," *IEEE Instrumentation Measurement Magazine*, vol. 8, pp. 24–32, Aug 2005.

[59] Y. Huang and J. Zhan, "GA and RBF based real-time FDD for refrigeration units," in *2009 International Symposium on Intelligent Ubiquitous Computing and Education*, pp. 22–25, May 2009.

[60] L. Chang, H. Wang, and L. Wang, "Fault detection and diagnosis of an HVAC system using artificial immune recognition system," in *2013 IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC)*, pp. 1–5, Dec 2013.

[61] S. Wang, Y. Ji, W. Dong, and X. Sun, "A novel fault diagnosis framework of computing with words based on lattice-valued fuzzy automata," in *2013 Fifth International Conference on Measuring Technology and Mechatronics Automation*, pp. 62–66, Jan 2013.

[62] Y. Guo, J. Wall, J. Li, and S. West, "Intelligent online sensor monitoring and fault alarm system in heating ventilation and air conditioning systems," in *2015 9th International Conference on Sensing Technology (ICST)*, pp. 789–794, Dec 2015.

[63] R. Sharifi and R. Langari, "Nonlinear sensor fault diagnosis using mixture of probabilistic PCA models," *Mechanical Systems and Signal Processing*, vol. 85, pp. 638 – 650, 2017.

[64] M. Yuwono, S. W. Su, Y. Guo, J. Li, S. West, and J. Wall, "Automatic feature selection using multiobjective cluster optimization for fault detection in a heating ventilation and air conditioning system," in *2013 1st International Conference on Artificial Intelligence, Modelling and Simulation*, pp. 171–176, Dec 2013.

[65] F. Boem, R. Reci, A. Cenedese, and T. Parisini, "Distributed clustering-based sensor fault diagnosis for HVAC systems," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 4197 – 4202, 2017.

[66] B. Littooy, S. Loire, M. Georgescu, and I. Mezić, "Pattern recognition and classification of HVAC rule-based faults in commercial buildings," in *2016 IEEE International Conference on Big Data (Big Data)*, pp. 1412–1421, Dec 2016.

[67] D. Dehestani, S. Su, H. Nguyen, and Y. Guo, "Robust fault tolerant application for HVAC system based on combination of online SVM and ANN black

box model," in *2013 European Control Conference (ECC)*, pp. 2976–2981, July 2013.

[68] D. Dey and B. Dong, "A probabilistic approach to diagnose faults of air handling units in buildings," *Energy and Buildings*, vol. 130, pp. 177 – 187, 2016.

[69] D.-W. Han and Y.-S. Chang, "Automated fault diagnosis method for a variable air volume air handling unit," in *2009 ICCAS-SICE*, pp. 2012–2017, Aug 2009.

[70] J. Ploennigs, M. Maghella, A. Schumann, and B. Chen, "Semantic diagnosis approach for buildings," *IEEE Transactions on Industrial Informatics*, vol. PP, no. 99, pp. 1–1, 2017.

[71] Y. Guo, G. Li, H. Chen, J. Wang, M. Guo, S. Sun, and W. Hu, "Optimized neural network-based fault diagnosis strategy for VRF system in heating mode using data mining," *Applied Thermal Engineering*, vol. 125, pp. 1402 – 1413, 2017.

[72] T. Mulumba, A. Afshari, K. Yan, W. Shen, and L. K. Norford, "Robust model-based fault diagnosis for air handling units," *Energy and Buildings*, vol. 86, pp. 698 – 707, 2015.

[73] Y. Yan, P. B. Luh, and B. Sun, "Fault detection of cooling coils based on unscented kalman filters and statistical process control," in *2013 IEEE International Conference on Automation Science and Engineering (CASE)*, pp. 237–242, Aug 2013.

[74] B. Sun, P. B. Luh, Z. O'Neill, and F. Song, "Building energy doctors: SPC and kalman filter-based fault detection," in *2011 IEEE International Conference on Automation Science and Engineering*, pp. 333–340, Aug 2011.

[75] Y. Yan, P. B. Luh, and K. R. Pattipati, "Fault diagnosis of HVAC air-handling systems considering fault propagation impacts among components," *IEEE Transactions on Automation Science and Engineering*, vol. 14, pp. 705–717, April 2017.

[76] Y. Yan, P. B. Luh, and K. R. Pattipati, "Fault diagnosis of HVAC: Air handling units and variable air volume boxes," in *2016 IEEE International Conference on Automation Science and Engineering (CASE)*, pp. 960–965, Aug 2016.

[77] H. Wang and Y. Chen, "A robust fault detection and diagnosis strategy for multiple faults of VAV air handling units," *Energy and Buildings*, vol. 127, no. Supplement C, pp. 442 – 451, 2016.

[78] P. Wang and R. Gao, "On-line fault detection and diagnosis for chiller system," in *2016 IEEE International Conference on Automation Science and Engineering (CASE)*, pp. 1313–1318, Aug 2016.

[79] Y. Yan, P. B. Luh, and K. R. Pattipati, "Chiller plant fault diagnosis considering fault propagation," in *2015 International Conference on Complex Systems Engineering (ICCSE)*, pp. 1–6, Nov 2015.

[80] K. Verbert, R. Babuška, and B. D. Schutter, "Combining knowledge and historical data for system-level fault diagnosis of HVAC systems," *Engineering Applications of Artificial Intelligence*, vol. 59, pp. 260 – 273, 2017.

[81] S. Wang, Q. Zhou, and F. Xiao, "A system-level fault detection and diagnosis strategy for HVAC systems involving sensor faults," *Energy and Buildings*, vol. 42, no. 4, pp. 477 – 490, 2010.

[82] D. ce Gao, S. Wang, K. Shan, and C. Yan, "A system-level fault detection and diagnosis method for low delta-t syndrome in the complex HVAC systems," *Applied Energy*, vol. 164, no. Supplement C, pp. 1028 – 1038, 2016.

[83] D. ce Gao, S. Wang, F. Xiao, and kui Shan, "A fault detection and diagnosis method for low delta-T syndrome in a complex air-conditioning system," *Energy Procedia*, vol. 61, pp. 2514 – 2517, 2014. International Conference on Applied Energy, {ICAE2014}.

[84] W. Turner, A. Staino, and B. Basu, "Residential HVAC fault detection using a system identification approach," *Energy and Buildings*, vol. 151, pp. 1 – 17, 2017.

[85] H. Shahnazari, P. Mhaskar, J. M. House, and T. I. Salsbury, "Heating, ventilation and air conditioning systems: Fault detection and isolation and safe parking," *Computers and Chemical Engineering*, vol. 108, no. Supplement C, pp. 139 – 151, 2018.

[86] S. Wu and J.-Q. Sun, "Cross-level fault detection and diagnosis of building HVAC systems," *Building and Environment*, vol. 46, no. 8, pp. 1558 – 1566, 2011.

[87] H. Wang, Y. Chen, C. W. Chan, and J. Qin, "An online fault diagnosis tool of VAV terminals for building management and control systems," *Automation in Construction*, vol. 22, pp. 203 – 211, 2012. Planning Future Cities-Selected papers from the 2010 eCAADe Conference.

[88] M. Horrigan, W. J. Turner, and J. O'Donnell, "A statistically-based fault detec-

tion approach for environmental and energy management in buildings," *Energy and Buildings*, pp. –, 2017.

[89] B. Sun, P. B. Luh, Q. S. Jia, Z. O'Neill, and F. Song, "Building energy doctors: An SPC and kalman filter-based method for system-level fault detection in HVAC systems," *IEEE Transactions on Automation Science and Engineering*, vol. 11, pp. 215–229, Jan 2014.

[90] S. e. a. Klein, "Trnsys 17: A transient system simulation program, solar energy laboratory," 2017.

[91] S. E. Laboratory, "Trnsys 17: 05-mathematicalreference," 2017.

[92] E. ToolBox, "Control valves and flow characteristics," 2003.

[93] G. Li, Y. Hu, H. Chen, H. Li, M. Hu, Y. Guo, S. Shi, and W. Hu, "A sensor fault detection and diagnosis strategy for screw chiller system using support vector data description-based D-statistic and DV-contribution plots," *Energy and Buildings*, vol. 133, pp. 230 – 245, 2016.

[94] M. Kim, H. Liu, J. T. Kim, and C. Yoo, "Sensor fault identification and reconstruction of indoor air quality (IAQ) data using a multivariate non-gaussian model in underground building space," *Energy and Buildings*, vol. 66, pp. 384 – 394, 2013.

[95] M. Padilla and D. Choinière, "A combined passive-active sensor fault detection and isolation approach for air handling units," *Energy and Buildings*, vol. 99, pp. 214 – 219, 2015.

[96] G. Li and Y. Hu, "Improved sensor fault detection, diagnosis and estimation for screw chillers using density-based clustering and principal component analysis," *Energy and Buildings*, vol. 173, pp. 502 – 515, 2018.

[97] Y. Chen and L. Lan, "Fault detection, diagnosis and data recovery for a real building heating/cooling billing system," *Energy Conversion and Management*, vol. 51, no. 5, pp. 1015 – 1024, 2010.

[98] Y. Hu, G. Li, H. Chen, H. Li, and J. Liu, "Sensitivity analysis for PCA-based chiller sensor fault detection," *International Journal of Refrigeration*, vol. 63, pp. 133 – 143, 2016.

[99] Y. Zhu, X. Jin, and Z. Du, "Fault diagnosis for sensors in air handling unit based on neural network pre-processed by wavelet and fractal," *Energy and Buildings*, vol. 44, pp. 7 – 16, 2012.

[100] M. Kramer, "Autoassociative neural networks," *Computers & Chemical Engineering*, vol. 16, no. 4, pp. 313 – 328, 1992. Neutral network applications in chemical engineering.

[101] M. Marseguerra and A. Zoia, "The autoassociative neural network in signal analysis: II. application to on-line monitoring of a simulated bwr component," *Annals of Nuclear Energy*, vol. 32, no. 11, pp. 1207 – 1223, 2005.

[102] M. Marseguerra and A. Zoia, "The autoassociative neural network in signal analysis: I. the data dimensionality reduction and its geometric interpretation," *Annals of Nuclear Energy*, vol. 32, no. 11, pp. 1191 – 1206, 2005.

[103] X. hua HUANG, "Sensor fault diagnosis and reconstruction of engine control

system based on autoassociative neural network," *Chinese Journal of Aeronautics*, vol. 17, no. 1, pp. 23 – 27, 2004.

[104] Z. N. S. Vanini, N. Meskin, and K. Khorasani, "Multiple-model sensor and components fault diagnosis in gas turbine engines using autoassociative neural networks," *Journal of Engineering for Gas Turbines and Power*, vol. 136, p. 091603, May 2014.

[105] J. Huang, H. Shimizu, and S. Shioya, "Data preprocessing and output evaluation of an autoassociative neural network model for online fault detection in virginiamycin production," *Journal of Bioscience and Bioengineering*, vol. 94, no. 1, pp. 70 – 77, 2002.

[106] H. Shimizu, K. Yasuoka, K. Uchiyama, and S. Shioya, "On-line fault diagnosis for optimal rice α-amylase production process of a temperature-sensitive mutant of saccharomyces cerevisiae by an autoassociative neural network," *Journal of Fermentation and Bioengineering*, vol. 83, no. 5, pp. 435 – 442, 1997.

[107] J. Glassey, M. Ignova, G. Montague, and A. Morris, "Autoassociative neural networks in bioprocess condition monitoring," *IFAC Proceedings Volumes*, vol. 27, no. 2, pp. 445 – 449, 1994. IFAC Symposium on Advanced Control of Chemical Processes, Kyoto, Japan, 25-27 May 1994.

[108] C. V. Kropas-Hughes, M. E. Oxley, S. K. Rogers, and M. Kabrisky, "Autoassociative–heteroassociative neural networks," *Engineering Applications of Artificial Intelligence*, vol. 13, no. 5, pp. 603 – 609, 2000.

[109] M. A. Kramer, "Nonlinear principal component analysis using autoassociative neural networks," *AIChE Journal*, vol. 37, no. 2, pp. 233–243.

[110] S. C. Bengea, P. Li, S. Sarkar, S. Vichik, V. Adetola, K. Kang, T. Lovett, F. Leonardi, and A. D. Kelman, "Fault-tolerant optimal control of a building HVAC system," *Science and Technology for the Built Environment*, vol. 21, no. 6, pp. 734–751, 2015.

[111] O. Abdel-Hamid, A. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, "Convolutional neural networks for speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, pp. 1533–1545, Oct 2014.

[112] "Hybrid convolutional neural networks for articulatory and acoustic information based speech recognition," *Speech Communication*, vol. 89, pp. 103 – 112, 2017.

[113] R. Ptucha, F. P. Such, S. Pillai, F. Brockler, V. Singh, and P. Hutkowski, "Intelligent character recognition using fully convolutional neural networks," *Pattern Recognition*, vol. 88, pp. 604 – 613, 2019.

[114] S. Kiranyaz, T. Ince, and M. Gabbouj, "Real-time patient-specific ECG classification by 1-D convolutional neural networks," *IEEE Transactions on Biomedical Engineering*, vol. 63, pp. 664–675, March 2016.

[115] R. D. Labati, E. Muñoz, V. Piuri, R. Sassi, and F. Scotti, "Deep-ECG: Convolutional neural networks for ECG biometric recognition," *Pattern Recognition Letters*, 2018.

[116] S. M. Mathews, C. Kambhamettu, and K. E. Barner, "A novel application of deep learning for single-lead ECG classification," *Computers in Biology and Medicine*, vol. 99, pp. 53 – 62, 2018.

[117] U. R. Acharya, H. Fujita, S. L. Oh, Y. Hagiwara, J. H. Tan, and M. Adam, "Appli-

cation of deep convolutional neural network for automated detection of myocardial infarction using ECG signals," *Information Sciences*, vol. 415-416, pp. 190 – 198, 2017.

[118] Y. Li, Y. Pang, J. Wang, and X. Li, "Patient-specific ECG classification by deeper CNN from generic to dedicated," *Neurocomputing*, vol. 314, pp. 336 – 346, 2018.

[119] Özal Yıldırım, P. Pławiak, R.-S. Tan, and U. R. Acharya, "Arrhythmia detection using deep convolutional neural network with long duration ECG signals," *Computers in Biology and Medicine*, vol. 102, pp. 411 – 420, 2018.

[120] M. A. Rahhal, Y. Bazi, H. AlHichri, N. Alajlan, F. Melgani, and R. Yager, "Deep learning approach for active classification of electrocardiogram signals," *Information Sciences*, vol. 345, pp. 340 – 354, 2016.

[121] G. Sannino and G. D. Pietro, "A deep learning approach for ECG-based heartbeat classification for arrhythmia detection," *Future Generation Computer Systems*, vol. 86, pp. 446 – 455, 2018.

[122] S. W. Smith, B. Walsh, K. Grauer, K. Wang, J. Rapin, J. Li, W. Fennell, and P. Taboulet, "A deep neural network learning algorithm outperforms a conventional algorithm for emergency department electrocardiogram interpretation," *Journal of Electrocardiology*, vol. 52, pp. 88 – 95, 2019.

[123] "Deep convolutional neural network for the automated detection and diagnosis of seizure using EEG signals," *Computers in Biology and Medicine*, vol. 100, pp. 270 – 278, 2018.

[124] U. R. Acharya, S. L. Oh, Y. Hagiwara, J. H. Tan, H. Adeli, and D. P. Subha,

"Automated EEG-based screening of depression using deep convolutional neural network," *Computer Methods and Programs in Biomedicine*, vol. 161, pp. 103 – 113, 2018.

[125] M. Atzori, M. Cognolato, and H. Müller, "Deep learning with convolutional neural networks applied to electromyography data: A resource for the classification of movements for prosthetic hands," *Frontiers in Neurorobotics*, vol. 10, 2016.

[126] Z. Ding, C. Yang, Z. Tian, C. Yi, Y. Fu, and F. Jiang, "sEMG-based gesture recognition with convolution neural networks," *Sustainability*, vol. 10, no. 6, p. 1865, 2018.

[127] R. Schirrmeister, L. Gemein, K. Eggensperger, F. Hutter, and T. Ball, "Deep learning with convolutional neural networks for decoding and visualization of EEG pathology," *2017 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*, 2017.

[128] G. Vrbancic and V. Podgorelec, "Automatic classification of motor impairment neural disorders from EEG signals using deep convolutional neural networks," *Elektronika ir Elektrotechnika*, vol. 24, no. 4, 2018.

[129] T. Ince, S. Kiranyaz, L. Eren, M. Askar, and M. Gabbouj, "Real-time motor fault detection by 1-D convolutional neural networks," *IEEE Transactions on Industrial Electronics*, vol. 63, pp. 7067–7075, Nov 2016.

[130] P. Honghu, H. Xingxi, T. Sai, and M. Fanming, "An improved bearing fault diagnosis method using one-dimensional CNN and LSTM," *Strojniski Vestnik / Journal of Mechanical Engineering*, vol. 64, no. 7/8, pp. 443 – 452, 2018.

[131] B. Zhang, W. Li, J. Hao, X.-L. Li, and M. Zhang, "Adversarial adaptive 1-D convolutional neural networks for bearing fault diagnosis under varying working condition," 05 2018.

[132] B. Yang, Y. Lei, F. Jia, and S. Xing, "An intelligent fault diagnosis approach based on transfer learning from laboratory bearings to locomotive bearings," *Mechanical Systems and Signal Processing*, vol. 122, pp. 692 – 706, 2019.

[133] O. Abdeljaber, O. Avci, S. Kiranyaz, M. Gabbouj, and D. J. Inman, "Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks," *Journal of Sound and Vibration*, vol. 388, pp. 154 – 170, 2017.

[134] O. Abdeljaber, O. Avci, M. S. Kiranyaz, B. Boashash, H. Sodano, and D. J. Inman, "1-D CNNs for structural damage detection: Verification on a structural health monitoring benchmark data," *Neurocomputing*, vol. 275, pp. 1308 – 1317, 2018.

[135] O. Avci, O. Abdeljaber, S. Kiranyaz, M. Hussein, and D. J. Inman, "Wireless and real-time structural damage detection: A novel decentralized method for wireless sensor networks," *Journal of Sound and Vibration*, vol. 424, pp. 158 – 172, 2018.

[136] L. Jing, M. Zhao, P. Li, and X. Xu, "A convolutional neural network based feature learning and fault diagnosis method for the condition monitoring of gearbox," *Measurement*, vol. 111, pp. 1 – 10, 2017.

[137] J. Yan, H. Zhu, X. Yang, Y. Cao, and L. Shao, "Research on fault diagnosis of hydraulic pump using convolutional neural network," *Journal of Vibroengineering*, vol. 18, pp. 5141–5152, 12 2016.

[138] S. Kiranyaz, A. Gastli, L. Ben-Brahim, N. Alemadi, and M. Gabbouj, "Real-time fault detection and identification for MMC using 1D convolutional neural networks," *IEEE Transactions on Industrial Electronics*, pp. 1–1, 2018.

[139] L. Wen, X. Li, L. Gao, and Y. Zhang, "A new convolutional neural network-based data-driven fault diagnosis method," *IEEE Transactions on Industrial Electronics*, vol. 65, pp. 5990–5998, July 2018.

[140] D.-T. Hoang and H.-J. Kang, "Rolling element bearing fault diagnosis using convolutional neural network and vibration image," *Cognitive Systems Research*, vol. 53, pp. 42 – 50, 2019. Advanced Intelligent Computing.

[141] C. Lu, Z. Wang, and B. Zhou, "Intelligent fault diagnosis of rolling bearing using hierarchical convolutional network based health state classification," *Advanced Engineering Informatics*, vol. 32, pp. 139 – 151, 2017.

[142] Y. Fu, Y. Zhang, Y. Gao, H. Gao, T. Mao, H. Zhou, and D. Li, "Machining vibration states monitoring based on image representation using convolutional neural networks," *Engineering Applications of Artificial Intelligence*, vol. 65, pp. 240 – 251, 2017.

[143] K. B. Lee, S. Cheon, and C. O. Kim, "A convolutional neural network for fault classification and diagnosis in semiconductor manufacturing processes," *IEEE Transactions on Semiconductor Manufacturing*, vol. 30, pp. 135–142, May 2017.

[144] H. Wu and J. Zhao, "Deep convolutional neural network model based chemical process fault diagnosis," *Computers and Chemical Engineering*, vol. 115, pp. 185 – 197, 2018.

[145] O. Janssens, V. Slavkovikj, B. Vervisch, K. Stockman, M. Loccufier, S. Verstockt, R. V. de Walle, and S. V. Hoecke, "Convolutional neural network based fault detection for rotating machinery," *Journal of Sound and Vibration*, vol. 377, pp. 331 – 345, 2016.

[146] X. Li, W. Zhang, and Q. Ding, "Deep learning-based remaining useful life estimation of bearings using multi-scale feature extraction," *Reliability Engineering and System Safety*, vol. 182, pp. 208 – 218, 2019.

[147] D. Verstraete, A. Ferrada, E. Droguett, V. Meruane, and M. Modarres, "Deep learning enabled fault diagnosis using time-frequency image analysis of rolling element bearings," *Shock and Vibration*, vol. 2017, pp. 1–17, 10 2017.

[148] Z. Zhu, G. Peng, Y. Chen, and H. Gao, "A convolutional neural network based on a capsule network with strong generalization for bearing fault diagnosis," *Neurocomputing*, vol. 323, pp. 62 – 75, 2019.

[149] S. Guo, T. Yang, W. Gao, and C. Zhang, "A novel fault diagnosis method for rotating machinery based on a convolutional neural network," *Sensors*, vol. 18, p. 1429, 05 2018.

[150] V. Tra, J. Kim, S. A. Khan, and J.-M. Kim, "Bearing fault diagnosis under variable speed using convolutional neural networks and the stochastic diagonal Levenberg-Marquardt algorithm," *Sensors*, vol. 17, p. 2834, Jun 2017.

[151] R. Chen, X. Huang, L. Yang, X. Xu, X. Zhang, and Y. Zhang, "Intelligent fault diagnosis method of planetary gearboxes based on convolution neural network and discrete wavelet transform," *Computers in Industry*, vol. 106, pp. 48 – 59, 2019.

[152] A. Gupta, G. Gurrala, and S. P.S, "An online power system stability monitoring system using convolutional neural networks," *IEEE Transactions on Power Systems*, pp. 1–1, 2018.

[153] J. Salvador, "Chapter 8 - deep learning," in *Example-Based Super Resolution* (J. Salvador, ed.), pp. 113 – 127, Academic Press, 2017.

[154] K. M. Ting, *Confusion Matrix*, pp. 209–209. Boston, MA: Springer US, 2010.

[155] A. Sosnovshchenko, *MACHINE LEARNING WITH SWIFT: artificial intelligence for ios*. PACKT Publishing Limited, 2018.

[156] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *Journal of Machine Learning Research*, vol. 13, no. Feb, pp. 281–305, 2012.

[157] "On reducing false alarms in multivariate statistical process control," *Chemical Engineering Research and Design*, vol. 88, no. 4, pp. 430 – 436, 2010.

APPENDIX

This section presents the details of the evaluated Convolutional Neural Network architectures in terms of the number of layers, sizes of the layers (e.g. size of the kernel, number of filters, number of nodes), type of activation functions, and the use of training enhancement methods (e.g. batch normalization, dropout, etc.).

5.1 Structures of the trained CNNs for Model-1

This section presents the detailed structures of the networks in Table 4.6 in Chapter 4, Section 4.4.2.

CNN-S8-5C-1

	NAME
1	Input 3x4x3 images with 'zerocenter' normalization
2	Conv1_1 6 1x2x3 convolutions with stride [1 1] and padding [0 0 0 0]
3	LeakyRELU1_1 Leaky ReLU with scale 0.01
4	Conv1_2 12 2x2x6 convolutions with stride [1 1] and padding [0 0 0 0]
5	BN Batch normalization with 12 channels
6	LeakyRELU1_2 Leaky ReLU with scale 0.01
7	FC1 30 fully connected layer
8	LeakyRELU_1 Leaky ReLU with scale 0.01
9	FC2 15 fully connected layer
10	LeakyRELU_2 Leaky ReLU with scale 0.01
11	FC3 5 fully connected layer
12	Softmax softmax
13	Classify crossentropyex

CNN-S7-5C-3

	NAME
1	Input 3x4x3 images with 'zerocenter' normalization
2	Conv1_1 6 2x2x3 convolutions with stride [1 1] and padding [0 0 0 0]
3	LeakyRELU1_1 Leaky ReLU with scale 0.01
4	Conv1_2 12 1x2x6 convolutions with stride [1 1] and padding [0 0 0 0]
5	LeakyRELU1_2 Leaky ReLU with scale 0.01
6	BN Batch normalization with 12 channels
7	drop1 35% dropout
8	FC1 10 fully connected layer
9	Sigmoid SigmoidLayer
10	FC2 5 fully connected layer
11	Softmax softmax
12	Classify crossentropyex

CNN-S6-5C-4

	NAME
1	Input 3x4x3 images with 'zerocenter' normalization
2	Conv1_1 6 1x2x3 convolutions with stride [1 1] and padding [0 0 0 0]
3	LeakyRELU1_1 Leaky ReLU with scale 0.01
4	Conv1_2 18 2x2x6 convolutions with stride [1 1] and padding [0 0 0 0]
5	LeakyRELU1_2 Leaky ReLU with scale 0.01
6	BN Batch normalization with 18 channels
7	drop1 35% dropout
8	FC1 20 fully connected layer
9	LeakyRELU Leaky ReLU with scale 0.01
10	FC2 5 fully connected layer

CNN-S5-5C-5

1	Input 3x4x3 images with 'zerocenter' normalization
2	Conv1 10 2x3x3 convolutions with stride [1 1] and padding [0 0 0 0]
3	BN Batch normalization with 10 channels
4	LeakyRELU1 Leaky ReLU with scale 0.01
5	drop1 35% dropout
6	FC1 10 fully connected layer
7	LeakyRELU_1 Leaky ReLU with scale 0.01
8	FC2_1 10 fully connected layer
9	LeakyRELU_2 Leaky ReLU with scale 0.01
10	FC2_2 5 fully connected layer
11	Softmax softmax
12	Classify crossentropyex

CNN-S4-5C-6

	NAME
1	Input 3x4x3 images with 'zerocenter' normalization
2	Conv1 10 2x1x3 convolutions with stride [1 1] and padding [0 0 0 0]
3	BN Batch normalization with 10 channels
4	Sigmoid SigmoidLayer
5	FC1 20 fully connected layer
6	LeakyRELU_1 Leaky ReLU with scale 0.01
7	FC2_1 8 fully connected layer
8	LeakyRELU_2 Leaky ReLU with scale 0.01
9	FC2_2 5 fully connected layer
10	Softmax softmax
11	Classify crossentropyex

CNN-S3-5C-7

	NAME
1	Input 3x4x3 images with 'zerocenter' normalization
2	Conv1 6 2x3x3 convolutions with stride [1 1] and padding [0 0 0 0]
3	BN Batch normalization with 6 channels
4	LeakyRELU Leaky ReLU with scale 0.01
5	drop1 35% dropout
6	FC1 18 fully connected layer
7	LeakyRELU1 Leaky ReLU with scale 0.01
8	FC2 5 fully connected layer
9	Softmax softmax
10	Classify crossentropyex

CNN-S2-5C-8

	NAME	
1	Input	3×4×3 images with 'zerocenter' normalization
2	Conv1	8 2×2×3 convolutions with stride [1 1] and padding [0 0 0 0]
3	BN	Batch normalization with 8 channels
4	Sigmoid	SigmoidLayer
5	drop1	35% dropout
6	FC1	10 fully connected layer
7	LeakyRELU	Leaky ReLU with scale 0.01
8	FC2	5 fully connected layer
9	Softmax	softmax
10	Classify	crossentropyex

CNN-S1-5C-9

	NAME	
1	Input	3×4×3 images with 'zerocenter' normalization
2	Conv1	9 2×1×3 convolutions with stride [1 1] and padding [0 0 0 0]
3	BN	Batch normalization with 9 channels
4	Sigmoid	SigmoidLayer
5	drop1	35% dropout
6	FC1	10 fully connected layer
7	LeakyRELU	Leaky ReLU with scale 0.01
8	FC2	5 fully connected layer
9	Softmax	softmax
10	Classify	crossentropyex

5.2 Structures of the trained CNNs for Model-2

This section presents the detailed structures of the networks in Table 4.7 in Chapter 4, Section 4.4.3.

CNN-S8-5C-1

	NAME	
1	Input	3×4×3 images with 'zerocenter' normalization
2	Conv1_1	8 1×2×3 convolutions with stride [1 1] and padding [0 0 0 0]
3	LeakyRELU1_1	Leaky ReLU with scale 0.01
4	Conv1_2	12 2×2×8 convolutions with stride [1 1] and padding [0 0 0 0]
5	BN	Batch normalization with 12 channels
6	LeakyRELU1_2	Leaky ReLU with scale 0.01
7	FC1	30 fully connected layer
8	LeakyRELU_1	Leaky ReLU with scale 0.01
9	FC2	15 fully connected layer
10	LeakyRELU_2	Leaky ReLU with scale 0.01
11	FC3	5 fully connected layer
12	Softmax	softmax
13	Classify	crossentropyex

CNN-S7-5C-3

	NAME	
1	Input	3×4×3 images with 'zerocenter' normalization
2	Conv1_1	6 2×2×3 convolutions with stride [1 1] and padding [0 0 0 0]
3	LeakyRELU1_1	Leaky ReLU with scale 0.01
4	Conv1_2	12 1×2×6 convolutions with stride [1 1] and padding [0 0 0 0]
5	LeakyRELU1_2	Leaky ReLU with scale 0.01
6	BN	Batch normalization with 12 channels
7	drop1	35% dropout
8	FC1	10 fully connected layer
9	Sigmoid	SigmoidLayer
10	FC2	5 fully connected layer
11	Softmax	softmax
12	Classify	crossentropyex

CNN-S6-5C-4

	NAME	
1	Input	3×4×3 images with 'zerocenter' normalization
2	Conv1_1	6 1×2×3 convolutions with stride [1 1] and padding [0 0 0 0]
3	LeakyRELU1_1	Leaky ReLU with scale 0.01
4	Conv1_2	18 2×2×6 convolutions with stride [1 1] and padding [0 0 0 0]
5	LeakyRELU1_2	Leaky ReLU with scale 0.01
6	BN	Batch normalization with 18 channels
7	drop1	35% dropout
8	FC1	20 fully connected layer
9	LeakyRELU	Leaky ReLU with scale 0.01
10	FC2	5 fully connected layer

CNN-S5-5C-5

1	Input	3×4×3 images with 'zerocenter' normalization
2	Conv1	10 2×3×3 convolutions with stride [1 1] and padding [0 0 0 0]
3	BN	Batch normalization with 10 channels
4	LeakyRELU1	Leaky ReLU with scale 0.01
5	drop1	35% dropout
6	FC1	10 fully connected layer
7	LeakyRELU_1	Leaky ReLU with scale 0.01
8	FC2_1	10 fully connected layer
9	LeakyRELU_2	Leaky ReLU with scale 0.01
10	FC2_2	5 fully connected layer
11	Softmax	softmax
12	Classify	crossentropyex

CNN-S4-5C-6

	NAME	
1	Input	3×4×3 images with 'zerocenter' normalization
2	Conv1	10 2×1×3 convolutions with stride [1 1] and padding [0 0 0 0]
3	BN	Batch normalization with 10 channels
4	Sigmoid	SigmoidLayer
5	FC1	20 fully connected layer
6	LeakyRELU_1	Leaky ReLU with scale 0.01
7	FC2_1	8 fully connected layer
8	LeakyRELU_2	Leaky ReLU with scale 0.01
9	FC2_2	5 fully connected layer
10	Softmax	softmax
11	Classify	crossentropyex

CNN-S3-5C-7

	NAME	
1	Input	3×4×3 images with 'zerocenter' normalization
2	Conv1	6 2×3×3 convolutions with stride [1 1] and padding [0 0 0 0]
3	BN	Batch normalization with 6 channels
4	LeakyRELU	Leaky ReLU with scale 0.01
5	drop1	35% dropout
6	FC1	18 fully connected layer
7	LeakyRELU1	Leaky ReLU with scale 0.01
8	FC2	5 fully connected layer
9	Softmax	softmax
10	Classify	crossentropyex

CNN-S2-5C-8

	NAME	
1	Input	3x4x3 images with 'zerocenter' normalization
2	Conv1	9 2x2x3 convolutions with stride [1 1] and padding [0 0 0 0]
3	BN	Batch normalization with 9 channels
4	Sigmoid	SigmoidLayer
5	drop1	35% dropout
6	FC1	10 fully connected layer
7	LeakyRELU	Leaky ReLU with scale 0.01
8	FC2	5 fully connected layer
9	Softmax	softmax
10	Classify	crossentropyex

CNN-S1-5C-9

	NAME	
1	Input	3x4x3 images with 'zerocenter' normalization
2	Conv1	9 2x1x3 convolutions with stride [1 1] and padding [0 0 0 0]
3	BN	Batch normalization with 9 channels
4	Sigmoid	SigmoidLayer
5	drop1	35% dropout
6	FC1	10 fully connected layer
7	LeakyRELU	Leaky ReLU with scale 0.01
8	FC2	5 fully connected layer
9	Softmax	softmax
10	Classify	crossentropyex

5.3 Structures of the trained CNNs for Model-3

This section presents the detailed structures of the networks in Table 4.8 in Chapter 4, Section 4.4.4.

CNN-S8-1C-A1

	NAME	
1	Input	3x4x3 images with 'zerocenter' normalization
2	Conv1_1	8 1x2x2 convolutions with stride [1 1] and padding [0 0 0 0]
3	LeakyRELU_1	Leaky ReLU with scale 0.01
4	BN_1	Batch normalization with 8 channels
5	Conv1_2	16 2x2x8 convolutions with stride [1 1] and padding [0 0 0 0]
6	Sigmoid	SigmoidLayer
7	BN_2	Batch normalization with 16 channels
8	FC1	20 fully connected layer
9	LeakyRELU_2	Leaky ReLU with scale 0.01
10	FC2	5 fully connected layer
11	LeakyRELU_3	Leaky ReLU with scale 0.01
12	FC3	2 fully connected layer
13	Softmax	softmax
14	Classify	crossentropyex

CNN-S7-1C-A2

ANALYSIS RESULT

	NAME	
1	Input	3x4x3 images with 'zerocenter' normalization
2	Conv1_1	6 2x2x3 convolutions with stride [1 1] and padding [0 0 0 0]
3	LeakyRELU1_1	Leaky ReLU with scale 0.01
4	Conv1_2	12 1x2x6 convolutions with stride [1 1] and padding [0 0 0 0]
5	LeakyRELU1_2	Leaky ReLU with scale 0.01
6	BN	Batch normalization with 12 channels
7	drop1	35% dropout
8	FC1	10 fully connected layer
9	Sigmoid	SigmoidLayer
10	FC2	2 fully connected layer
11	Softmax	softmax
12	Classify	crossentropyex

CNN-S4-1C-A3

	NAME	
1	Input	3x4x3 images with 'zerocenter' normalization
2	Conv1	10 2x1x3 convolutions with stride [1 1] and padding [0 0 0 0]
3	Sigmoid	SigmoidLayer
4	BN	Batch normalization with 10 channels
5	FC1	20 fully connected layer
6	LeakyRELU_1	Leaky ReLU with scale 0.01
7	FC2_1	4 fully connected layer
8	LeakyRELU_2	Leaky ReLU with scale 0.01
9	FC2_2	2 fully connected layer
10	Softmax	softmax
11	Classify	crossentropyex

CNN-S3-1C-A4

	NAME	
1	Input	3x4x3 images with 'zerocenter' normalization
2	Conv1	10 2x3x3 convolutions with stride [1 1] and padding [0 0 0 0]
3	BN	Batch normalization with 10 channels
4	LeakyRELU1	Leaky ReLU with scale 0.01
5	drop1	35% dropout
6	FC1	5 fully connected layer
7	LeakyRELU	Leaky ReLU with scale 0.01
8	FC2	2 fully connected layer
9	Softmax	softmax
10	Classify	crossentropyex with classes '0' and '1'

CNN-S7-1C-B1

	NAME	
1	Input	3x4x3 images with 'zerocenter' normalization
2	Conv1_1	6 2x2x3 convolutions with stride [1 1] and padding [0 0 0 0]
3	LeakyRELU1_1	Leaky ReLU with scale 0.01
4	Conv1_2	12 1x2x6 convolutions with stride [1 1] and padding [0 0 0 0]
5	LeakyRELU1_2	Leaky ReLU with scale 0.01
6	BN	Batch normalization with 12 channels
7	drop1	35% dropout
8	FC1	10 fully connected layer
9	Sigmoid	SigmoidLayer
10	FC2	2 fully connected layer
11	Softmax	softmax
12	Classify	crossentropyex

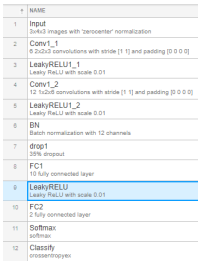
CNN-S3-1C-B2

	NAME	
1	Input	3x4x3 images with 'zerocenter' normalization
2	Conv1	8 2x2x3 convolutions with stride [1 1] and padding [0 0 0 0]
3	BN	Batch normalization with 8 channels
4	Sigmoid	SigmoidLayer
5	drop1	35% dropout
6	FC1	10 fully connected layer
7	LeakyRELU	Leaky ReLU with scale 0.01
8	FC2	2 fully connected layer
9	Softmax	softmax
10	Classify	crossentropyex with classes '0' and '1'

CNN-S2-1C-B4

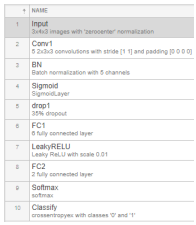
	NAME	
1	Input	3x4x3 images with 'zerocenter' normalization
2	Conv1	9 2x2x3 convolutions with stride [1 1] and padding [0 0 0 0]
3	BN	Batch normalization with 9 channels
4	LeakyRELU1	Leaky ReLU with scale 0.01
5	drop1	35% dropout
6	FC1	8 fully connected layer
7	LeakyRELU	Leaky ReLU with scale 0.01
8	FC2	2 fully connected layer
9	Softmax	softmax
10	Classify	crossentropyex with classes '0' and '1'

CNN-S7-1C-C1



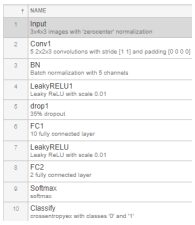
	NAME
1	Input 3x4x3 images with 'zerocenter' normalization
2	Conv1_1 6 2x2x3 convolutions with stride [1 1] and padding [0 0 0 0]
3	LeakyRELU1_1 Leaky ReLU with scale 0.01
4	Conv1_2 12 1x2x6 convolutions with stride [1 1] and padding [0 0 0 0]
5	LeakyRELU1_2 Leaky ReLU with scale 0.01
6	BN Batch normalization with 12 channels
7	drop1 35% dropout
8	FC1 10 fully connected layer
9	LeakyRELU Leaky ReLU with scale 0.01
10	FC2 2 fully connected layer
11	Softmax softmax
12	Classify crossentropyex

CNN-S3-1C-C2



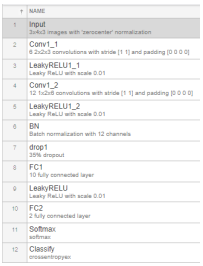
	NAME
1	Input 3x4x3 images with 'zerocenter' normalization
2	Conv1 5 2x3x3 convolutions with stride [1 1] and padding [0 0 0 0]
3	BN Batch normalization with 5 channels
4	Sigmoid SigmoidLayer
5	drop1 35% dropout
6	FC1 6 fully connected layer
7	LeakyRELU Leaky ReLU with scale 0.01
8	FC2 2 fully connected layer
9	Softmax softmax
10	Classify crossentropyex with classes '0' and '1'

CNN-S2-1C-C4



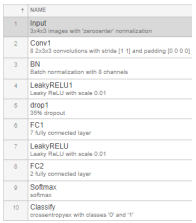
	NAME
1	Input 3x4x3 images with 'zerocenter' normalization
2	Conv1 5 2x2x3 convolutions with stride [1 1] and padding [0 0 0 0]
3	BN Batch normalization with 5 channels
4	LeakyRELU1 Leaky ReLU with scale 0.01
5	drop1 35% dropout
6	FC1 10 fully connected layer
7	LeakyRELU Leaky ReLU with scale 0.01
8	FC2 2 fully connected layer
9	Softmax softmax
10	Classify crossentropyex with classes '0' and '1'

CNN-S7-1C-D1



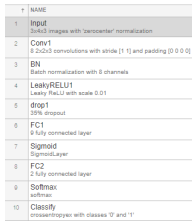
	NAME
1	Input 3x4x3 images with 'zerocenter' normalization
2	Conv1_1 6 2x2x3 convolutions with stride [1 1] and padding [0 0 0 0]
3	LeakyRELU1_1 Leaky ReLU with scale 0.01
4	Conv1_2 12 1x2x6 convolutions with stride [1 1] and padding [0 0 0 0]
5	LeakyRELU1_2 Leaky ReLU with scale 0.01
6	BN Batch normalization with 12 channels
7	drop1 35% dropout
8	FC1 10 fully connected layer
9	LeakyRELU Leaky ReLU with scale 0.01
10	FC2 2 fully connected layer
11	Softmax softmax
12	Classify crossentropyex

CNN-S3-1C-D2



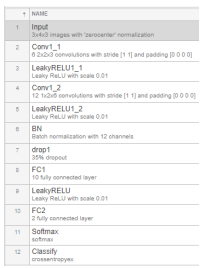
	NAME
1	Input 3x4x3 images with 'zerocenter' normalization
2	Conv1 8 2x3x3 convolutions with stride [1 1] and padding [0 0 0 0]
3	BN Batch normalization with 8 channels
4	LeakyRELU1 Leaky ReLU with scale 0.01
5	drop1 35% dropout
6	FC1 7 fully connected layer
7	LeakyRELU Leaky ReLU with scale 0.01
8	FC2 2 fully connected layer
9	Softmax softmax
10	Classify crossentropyex with classes '0' and '1'

CNN-S2-1C-D3



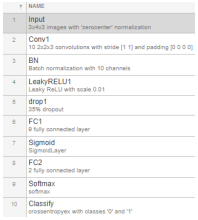
	NAME
1	Input 3x4x3 images with 'zerocenter' normalization
2	Conv1 8 2x2x3 convolutions with stride [1 1] and padding [0 0 0 0]
3	BN Batch normalization with 8 channels
4	LeakyRELU1 Leaky ReLU with scale 0.01
5	drop1 35% dropout
6	FC1 9 fully connected layer
7	Sigmoid SigmoidLayer
8	FC2 2 fully connected layer
9	Softmax softmax
10	Classify crossentropyex with classes '0' and '1'

CNN-S7-1C-E1



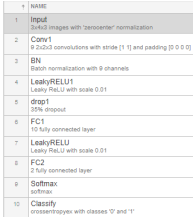
	NAME
1	Input 3x4x3 images with 'zerocenter' normalization
2	Conv1_1 6 2x2x3 convolutions with stride [1 1] and padding [0 0 0 0]
3	LeakyRELU1_1 Leaky ReLU with scale 0.01
4	Conv1_2 12 1x2x6 convolutions with stride [1 1] and padding [0 0 0 0]
5	LeakyRELU1_2 Leaky ReLU with scale 0.01
6	BN Batch normalization with 12 channels
7	drop1 35% dropout
8	FC1 10 fully connected layer
9	LeakyRELU Leaky ReLU with scale 0.01
10	FC2 2 fully connected layer
11	Softmax softmax
12	Classify crossentropyex

CNN-S2-1C-E3



	NAME
1	Input 3x4x3 images with 'zerocenter' normalization
2	Conv1 10 2x2x3 convolutions with stride [1 1] and padding [0 0 0 0]
3	BN Batch normalization with 10 channels
4	LeakyRELU1 Leaky ReLU with scale 0.01
5	drop1 35% dropout
6	FC1 9 fully connected layer
7	Sigmoid SigmoidLayer
8	FC2 2 fully connected layer
9	Softmax softmax
10	Classify crossentropyex with classes '0' and '1'

CNN-S2-1C-E4



	NAME
1	Input 3x4x3 images with 'zerocenter' normalization
2	Conv1 9 2x2x3 convolutions with stride [1 1] and padding [0 0 0 0]
3	BN Batch normalization with 9 channels
4	LeakyRELU1 Leaky ReLU with scale 0.01
5	drop1 35% dropout
6	FC1 10 fully connected layer
7	LeakyRELU Leaky ReLU with scale 0.01
8	FC2 2 fully connected layer
9	Softmax softmax
10	Classify crossentropyex with classes '0' and '1'