Journal Pre-proofs

Non-Intrusive Classification of Gas-Liquid Flow Regimes in an S-shaped Pipeline Riser Using a Doppler Ultrasonic Sensor and Deep Neural Networks

Somtochukwu Godfrey Nnabuife, Boyu Kuang, James F. Whidborne, Zeeshan Rana

PII: S1385-8947(20)32529-8
DOI: https://doi.org/10.1016/j.cej.2020.126401
Reference: CEJ 126401

To appear in: *Chemical Engineering Journal*

Received Date: 22 December 2019
Revised Date: 20 July 2020
Accepted Date: 22 July 2020

Please cite this article as: S. Godfrey Nnabuife, B. Kuang, J.F. Whidborne, Z. Rana, Non-Intrusive Classification of Gas-Liquid Flow Regimes in an S-shaped Pipeline Riser Using a Doppler Ultrasonic Sensor and Deep Neural Networks, *Chemical Engineering Journal* (2020), doi: https://doi.org/10.1016/j.cej.2020.126401

This is a PDF file of an article that has undergone enhancements after acceptance, such as the addition of a cover page and metadata, and formatting for readability, but it is not yet the definitive version of record. This version will undergo additional copyediting, typesetting and review before it is published in its final form, but we are providing this version to give early visibility of the article. Please note that, during the production process, errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

© 2020 Published by Elsevier B.V.

Chemical Engineering Journal, Volume 403, January 2021 Article number 126401
DOI: 10.1016/j.cej.2020.126401

Journal Pre-proofs

Non-Intrusive Classification of Gas-Liquid Flow Regimes in an S-shaped Pipeline Riser Using a Doppler Ultrasonic Sensor and Deep Neural Networks

Somtochukwu Godfrey Nnabuife[a], Boyu Kuang[b], James F. Whidborne[c*], Zeeshan Rana[b]
[a] Geo-Energy Engineering Centre, Cranfield University, Cranfield, MK43 0AL, United Kingdom
[b] Centre for Computational Engineering Sciences, Cranfield University, Cranfield, MK43 0AL, United Kingdom
[c] Dynamics Simulation and Control Group, Cranfield University, Cranfield, MK43 0AL, United Kingdom
*Corresponding author, email address: j.f.whidborne@cranfield.ac.uk

Abstract

The problem of predicting the regime of a two-phase flow is considered. An approach is proposed that classifies the flow regime using Deep Neural Networks (DNNs) operating on features extracted from Doppler ultrasonic signals of the flow using the Fast Fourier Transform (FFT) is proposed. The features extracted are categorised into one of the four flow regime classes: the annular, churn, slug, and bubbly flow regimes. The scheme was tested on signals from an experimental facility. To increase the number of samples without losing key classification information, this paper proposes a Twin-window Feature Extraction (TFE) technique. To further distinguish the performance of the proposed approach, the classifier was compared to four conventional machine learning classifiers: namely, the AdaBoost classifier, bagging classifier, extra trees classifier, and decision tree classifier. Using the TFE features, the DNNs classifier achieved a higher recognition accuracy of 99.01% and greater robustness for the overfitting challenge, thereby showing the superiority of the DNNs in flow regime classification when compared to the four conventional machine-learning classifiers, which had classification accuracies of 55.35%, 86.21%, 82.41%, and 80.03%, respectively. This approach demonstrates the application of DNNs for flow regime classification in chemical and petroleum engineering fields, using a clamp-on Doppler ultrasonic sensor. This appears to be the first known successful attempt to identify gas-liquid flow regimes in an S-shaped riser using Continuous Wave Doppler Ultrasound (CWDU) and DNNs.

Keywords: Doppler ultrasound, Deep Neural Networks (DNNs), Fast Fourier Transform (FFT), S-shaped riser, Machine Learning (ML), Twin-window Feature Extraction (TFE).

1 Introduction

Multiphase flows are encountered in a wide range of engineering and industrial processes, such as those in petrochemical, petroleum, thermal, and chemical engineering [1]. Determining the distribution of constitutive phases during the simultaneous gas-liquid two-phase flow through a pipe relies on various parameters, such as pipe orientation with respect to gravity, phase flow rate, pipe size and shape, magnitude and direction of

Published by Elsevier. This is the Author Accepted Manuscript issued with: Creative Commons Attribution Non-Commercial No Derivatives License (CC:BY:NC:ND 4.0).
The final published version (version of record) is available online at DOI: 10.1016/j.cej.2020.126401. Please refer to any applicable publisher terms of use.

individual phase velocities, flow properties, and operating conditions. All these distributions, often known as flow regimes, can be quite complex. The flow can be turbulent or laminar, unsteady or steady, liquid and gas can be segregated, gas can flow as bubbles within the liquid, or liquid can flow as droplets within the gas [2]. When liquid and gas flow simultaneously in an S-shaped riser or a vertical pipeline, different flow regimes can occur, such as annular, bubbly, churn, or slug flows. The characteristics of flow pressure drop, flow resistance, and heat transfer vary with the flow regime.

The flow regime is one of the major factors affecting the reliable and efficient online measurement of flow rate, phase fraction, and other multiphase flow parameters. Moreover, for proper operation and design of two-phase flow facilities, accurate prediction of the system's pressure drop is required, and that prediction relies on proper understanding and knowledge of the nature of the flow regimes in the system [3].

Even though accurate and reliable objective identification of flow regime is desirable, flow regime definitions are often based on visual descriptions such as graphics or photographic illustrations, with a corresponding element of subjectivity [4]. The complex nature of a two-phase flow, which is characterised by gas-phase compressibility, phase interaction, phase slip, turbulence, and a deformable phase interface, makes it challenging to obtain a reliable flow model. Identification of flow regime can be performed either visually by inspecting the flow through a transparent section of the pipe or by measuring and quantitatively defining the flow parameter fluctuations such as pressure or void fraction, which reflect the flow structure [5].

Since flow regime characteristics are normally hidden within apparently random signal fluctuations obtained using a measurement instrument, signal analysis plays a vital role

in flow regime identification. The analysed signal's extracted features are interpreted for flow regime identification using a rule-based system. Although many efforts have been made to make flow regime identification perception bias-free, total objective flow regime predictions are yet to be achieved [2]. In a quest to reduce flow regime identification subjectivity, different signal processing methods, machine learning [7], soft computing methods [8], and statistical methods [9] are widely used.

The application of wavelet analysis for flow regime classification has been proven to be efficient in feature extraction of multiphase flow behaviour. Wavelet analysis chacterises the particle fluidisation complex structure[10]. Wavelet analysis has attracted a lot of interest in various fields of engineering applications in combination with mathematics [11]. Qiang et al. used a signal processing approach for flow regime classification including peak points count, amplitude spectra, and Probability Density Distribution (PDD). In their work, they adopted a fuzzy system classification method in combination with wavelet transform analysis to develop a unique method to identify the dense suspension vertical flow, turbulent, bubbling, Circulating Turbulent Fluidized bed (CTFB) and Fast Fluidisation (FF) flow patterns [10]. Zhou et al applied the acoustic emission technique for two-phase gas-solid flow regime transition identification. They proposed the combination of multiscale and standard deviation analysis to examine the flow transitions among fast fluidisation, bubbling fluidisation, dense phase pneumatic conveying and turbulent fluidisation.[12]

various signal processing methods, machine learning [7], soft computing methods [8], and statistical methods [9] are widely used.

The application of wavelet analysis for flow regime classification has proven to be efficient in extracting features of multiphase flow behaviour. Wavelet analysis chacterises the particle fluidisation's complex structure [10] and has attracted significant interest in various fields of engineering applications and mathematics [11]. Qiang et al. used a signal processing approach for flow regime classification, including peak points count, amplitude spectra, and probability density distribution. In their work, they adopted a fuzzy classification method in combination with wavelet transform analysis to develop a unique method to identify the dense suspension vertical flow, turbulent, bubbling, circulating turbulent fluidized bed, and fast fluidisation flow patterns [10]. Zhou et al. applied the acoustic emission technique for two-phase gas-solid flow regime transition identification. They proposed a combination of multiscale and standard deviation analysis to examine the flow transitions among fast fluidisation, bubbling fluidisation, dense phase pneumatic conveying, and turbulent fluidisation [12].

The difficulties in flow regime identification are distinct. Neural net methods solve the task by learning from known similar flow patterns. The interest in neural networks started in the 1980s because they were seen as an efficient tool for modelling human thoughts. Flow regime prediction using neural networks has proved capable of objectively classifying two-phase flow regimes, clustering features for large numbers of samples when training processes are properly implemented [14, 15]. Over the years, different types of neural networks have been used by many researchers for flow regime identification: namely, Feed-Forward Neural Networks (FFNN), Self-Organizing Neural Networks (SONN), and Probabilistic Neural Networks (PNNs). Also, different statistical parameters such as skewness, the mean and standard deviation of void impedance signals

[15], the void impedance signals of the Probability Density Function (PDF) [16], the void fraction Cumulative Probability Density Function (CPDF) [17], the local pressure variation's Power Spectral Density (PSD) [18], and the local and global bubble chord length CPDF [19] have been used as input in the neural net systems. An integral parameter, CPDF is found to be more reliable and stable than other statistical parameters. Furthermore, CPDF requires less input data, and it is widely used as a flow regime prediction indicator alongside neural networks [20].

Tsoukalas et al. identified air-water flow regimes in an upward flow from area average void fraction fluctuations, impedance signals of PSDs, and PDFs using a neuro-fuzzy system [21]. Mi et al. successfully classified flow regimes in a vertical conduit using electrical capacitance probe signals and neural networks [22]. Xie et al. predicted the flow regimes in a gas-liquid-fibre three-phase flow vertical pipe system using an Artificial Neural Network (ANN) approach by designing a feed-forward three-layer ANN that deployed seven inputs to represent the PSD distribution characteristics of normalized pressure signal variations [23]. Sharma et al. worked to develop an objective air-water flow regime identification based on data from literature using ANNs. Three different ANNs were examined. A PNN characterized by the Bayes-Parzen classification theory yielded accurate flow regime predictions for various channel inclinations and diameters [24]. The Feed-Forward Back Propagation (FFBP) method also gave accurate flow regime predictions but was unsuccessful with flow transition regions. All the results Sharma et al. generated were validated with theoretical and experimental models obtainable in the literature. Nandagopal et al. studied the confluence angle effects on a liquid-liquid flow regime system for a circular microchannel. In their work, they aimed

at finding a reliable objective liquid-liquid flow regime indicator for fluctuating confluence angle. They studied different radial basis networks, adaptive neuro-fuzzy interface systems, and FFBPs for better flow regime prediction [7]. Other recently reported work on neural networks for two-phase flow regime identification can be found in [7], [25], [26], [27], and [28].

Recently, DNNs have achieved outstanding classification performance and become a prominent tool in various pattern classification applications, such as document analysis [29], image recognition [30], object detection, and video understanding [31].

Despite the numerous efforts being made to unravel the characteristics of flow regimes, most of the identification methods proposed in the literature to date cannot provide a reliable and efficient solution for industrial online flow regime identification. This is due to their inconsistent repeatability and reliability, as described in the work of [32]. However, developing new methods and principles for flow regime identification is of great technological and scientific benefit.

For safety and long-term industrial system performance of applications such as nuclear and chemical reactors, biomedical applications, and petroleum processing systems, it is important to monitor system flow regimes during transients and normal operations. Hence, for accurate design, operation, and analysis, it is essential to have full knowledge of a system's flow regimes [33]. Many efforts have been made towards two-phase gas-liquid flow regime classification through experimentation and analysis [22]. However, most of these efforts have adopted conventional machine learning methods [26]. Flow regime classification using deep learning methods has not been explored.

In this paper, a non-intrusive and non-radioactive method for the identification of a two-phase gas-liquid flow regime in an S-shaped pipeline-riser system is proposed that uses continuous wave Doppler ultrasound (CWDU) signals and a DNNs approach. The proposed method is cost-effective, reliable, repeatable, and promising for online flow regime monitoring. In this method, the FFT is employed to extract influential features from the CWDU signals. The FFT decreases the dimensions without losing the key information for the flow regimes classification.

In summary, the main contributions of this paper are as follows: (i) This appears to be the first trial using CWDU and DNNs to automatically identify the flow regime in an S-shaped riser. It proposes a non-contact, end-to-end approach that is more flexible and user-friendly than conventional approaches; the industrial customer can use it as a black box instead of having to acquire professional-level knowledge. (ii) This paper proposes a Twin-window Feature Extraction (TFE) algorithm based on the FFT [34] that can expand the data space and extract the key features from raw samples. The TFE algorithm provides an alternative to manually carrying out multiple two-phase flow experiments to expand the data space for deep learning training. This significantly increases the overall research efficiency and decreases the cost. The paper also offers a promising idea for two-phase flow data acquisition. By using this technique for two-phase flow regime identification, industries can achieve enhanced production, better process performance, and hence, economic advantages.

This paper is organised as follows. Section 2 presents the sensor principle and the algorithm for the CWDU along with the data acquisition process, feature extractions, DNN architecture, and mathematical expressions. In Section 3, the experimental method

used in this study is described. Section 4 presents the results and a discussion of the analysed data, and finally, conclusions are drawn in Section 5.

2 Methodology

This paper proposes a new framework, illustrated in Figure 2-1, that can classify and analyse gas-liquid flow patterns using a non-contact method. Ultrasonic Doppler sensors are used for data acquisition, and the data collected in each experiment is a sample. To prevent incomplete classification information, each sample consists of 1.3 million discrete values. However, the classification information in each sample is highly dispersed and noisy. Therefore, this paper proposes an FFT-based TFE algorithm that extracts features from the raw samples to reduce the sample length (or dimension) without losing key information for the flow regime identification. Finally, this paper trains a deep learning network classifier to obtain a highly accurate gas-liquid flow classification model.

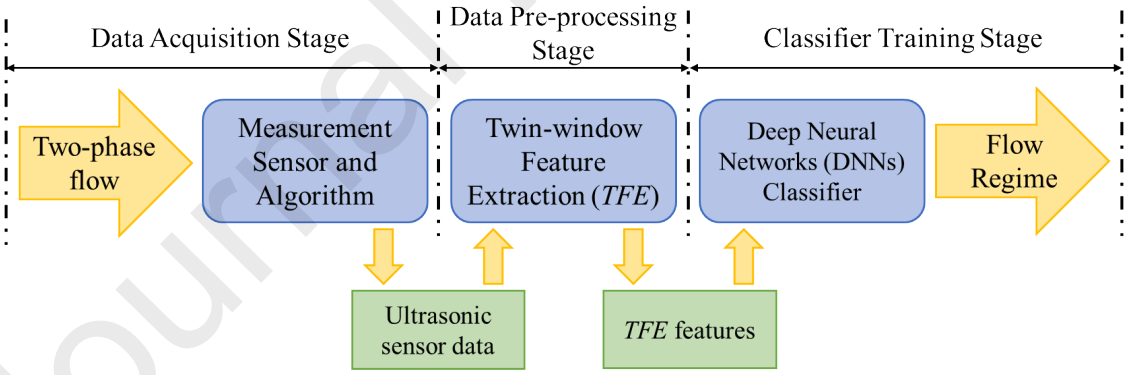


Figure 2-1: The schematic diagram of the overall framework. The schematic diagram of the overall framework. The yellow arrows denote the data flow inside the framework, the blue squares refer to the three main steps, and the green squares indicate data communicated between the main steps. The general framework can be divided into three stages (data acquisition, data pre-processing, and classifier training stage), which have been identified with dot-dash lines.

2.1 Measurement sensor and algorithm

The Doppler effect or Doppler shift is the change in frequency of an acoustic wave when there exists relative movement between the source and the acoustic receiver, with the frequency variation proportional to the acoustic source's velocity [35]. This velocity is acquired by estimating the shift in frequency between the source and the receiver. In the CWDU flowmeter, shown in Figure 2-2, an acoustic beam is transmitted continuously from the transmitting transducer into the flow line. The beam is then reflected by the fluid's moving scatterers, which could be in the form of bubbles [36]. The receiving transducer receives the acoustic beam that was scattered in the flow-line, and the flow velocity is obtained from the shift in the frequency [37]. The principle behind this sensor is discussed next.

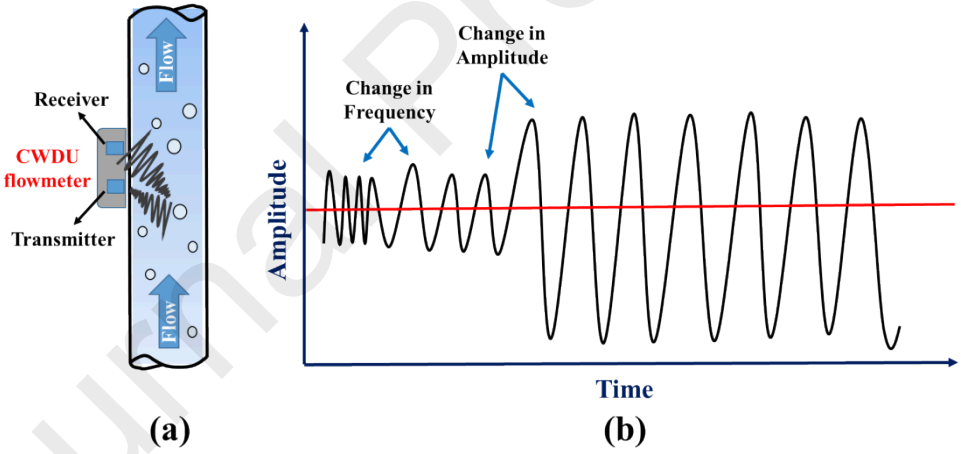


Figure 2-2: Ultrasound Doppler principle [37].

The signal transmitted is assumed to be

$$x_t(t) = \varepsilon_t \cos(\omega_t t), \tag{1}$$

and the signal received from one of the scatterers is

$$x_r(t) = \varepsilon_r \cos\left(\{\omega_t + \omega_d\}t + \theta_1\right), \tag{2}$$

where $\omega_t = 2\pi f_t$ is the transmitted signal's angular frequency, $\omega_d = 2\pi f_d$ is the angular frequency shift, and θ_1 is the phase shift based on the relative location of the scatterer between the transducer and receiver [38].

Multiplying the two signals gives

$$x_t(t)x_r(t) = \varepsilon_t\varepsilon_r \cos\left(\omega_t t\right)\cos\left(\{\omega_t + \omega_d\}t + \theta_1\right) \tag{3}$$

$$= \frac{\varepsilon_t\varepsilon_r}{2}[\cos\left(\omega_d t + \theta_1\right) + \cos\left(\{2\omega_t + \omega_d\}t + \theta_1\right)]. \tag{4}$$

Applying a low-pass filter to eliminate the source frequency $2f_t$, leaves only the Doppler signal [38]:

$$x_d(t) = \frac{\varepsilon_t\varepsilon_r}{2}\cos\left(\omega_d t + \theta_1\right). \tag{5}$$

Finally, the relationship between the velocity of the scatterer and the Doppler shift f_d can be determined by [39]:

$$f_d = 2f_t\frac{v}{c}\cos\theta, \tag{6}$$

where f_t is the ultrasound frequency transmitted, f_d is the shift in Doppler frequency, θ is the angle between the ultrasound beam and the flow velocity, and v is the flow velocity average.

The CWDU adopted in this paper is the same as that used in [37]. It acquires the Doppler signals, estimates the shift in Doppler frequency of the reflected ultrasonic signals from

the flow discontinuities, and then calculates the flow velocity. In this paper, the data acquisition system NI PCI-6040E was used to sample the analogue signals.

2.2 Ultrasonic sensor data acquisition and pre-processing

A continuous-wave Doppler ultrasound with voltage excitation of ±10 V, running at a 500 kHz carrier frequency was connected to the multiphase flow S-shaped riser topside as shown in Figure 2-3. The ultrasound beam incident angle was 58° with respect to the S-shaped riser flow direction. For easy transmission of ultrasound energy, a gel coupling agent was used to hold the Doppler transducer firmly to the flow pipe. The CWDU electronics flow meter was used to acquire the Doppler frequency shift voltage signals for further analysis. The data used for this paper is available on Cranfield Online Research Data (CORD) [40].

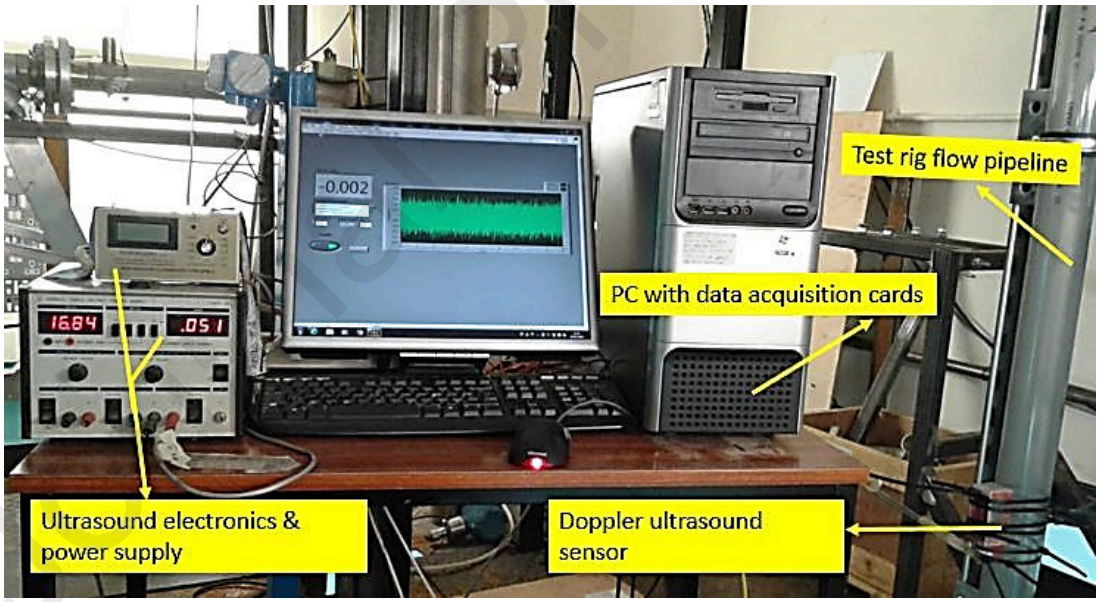


Figure 2-3: Doppler ultrasonic sensor and its auxiliary instruments.

2.3 The twin-window feature extraction (TFE) process

As mentioned in Section 2.2, the length of each raw data sample is 1.3 million float digits, which is a lengthy and highly noisy signal. This paper proposes a novel feature extraction algorithm, the TFE, to reduce the length and eliminate the noise. The TFE method can be understood as a composite operation which is implemented by two segmentation windows (*window A* and *window B*) and one FFT. This article explains the TFE process along with the short, intermediate, and global operations (or from local to global) below.

The short operation is directly implemented using the FFT. More specifically, this article uses the nonequispaced fast Fourier transform method [41]. To achieve the best Fourier transform efficiency, all *N*-orders (*N*) of FFTs take integer powers of two. For example, the *N* values in this study are 128, 256, 512, 1024, 2048, and 4096, and their orders respectively are 7, 8, 9, 10, 11, and 12. Section 4 specifically discusses the experimental results corresponding to the different *N*-orders. Furthermore, the result of the FFT is symmetrical on the positive and negative axis, so the positive and negative values of the FFT result have the same contribution to the classification task. Therefore, this paper uses only the positive part of the FFT result as the final feature extraction result.

Journal Pre-proofs

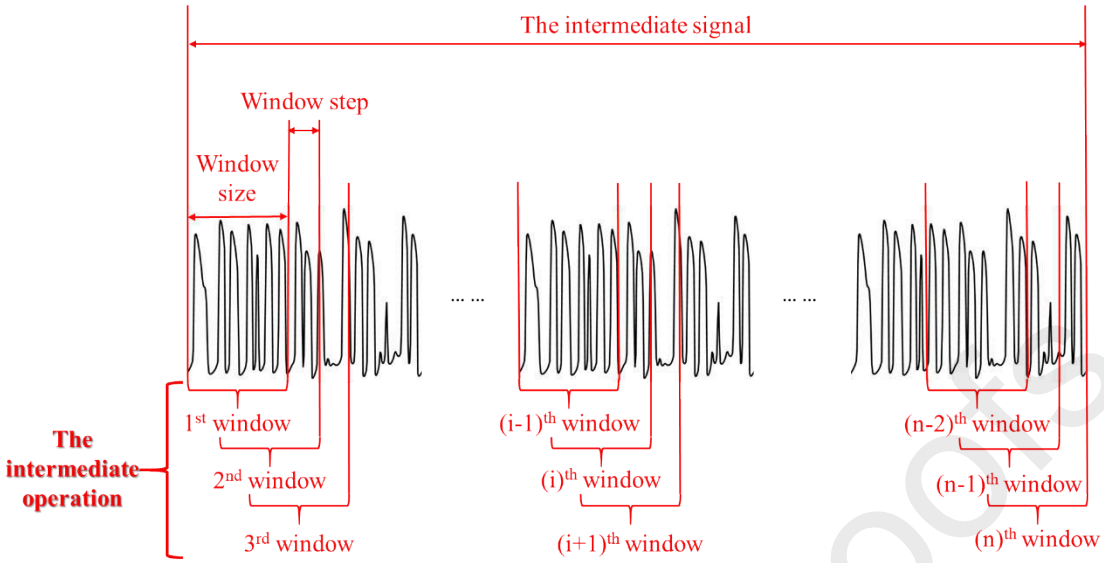


Figure 2-4: The intermediate operation process. The red line and comments correspond to the intermediate operations. i represents a given window index, and n represents the last window index.

The intermediate operation is performed by multiple local operations. As shown in Figure 2-4, a given intermediate signal is segmented into multiple short windows of equal size. Then, the FFT is performed on each small window, and the arithmetic average from all FFT results is calculated. More specifically, the step length of each window corresponds to the window step, thus *(window size – window step)* is the overlapping length.

Journal Pre-proofs

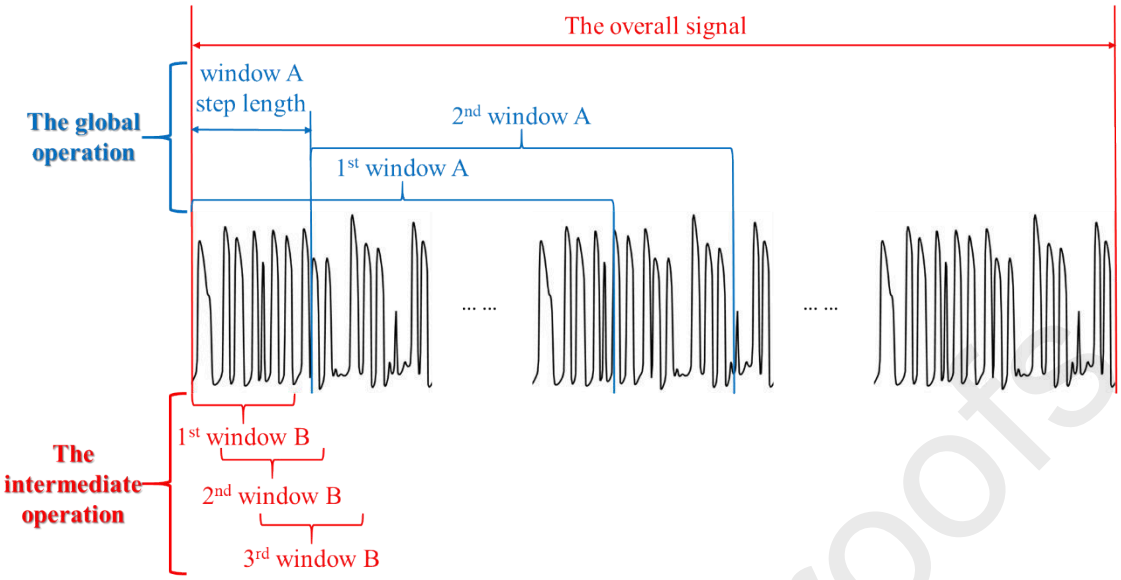


Figure 2-5: Flow chart of the proposed Twin-window Feature Extraction (TFE). The blue line and comments correspond to the global operation (or outside window operations). The red line and comments correspond to the intermediate operation (or inner window operations).

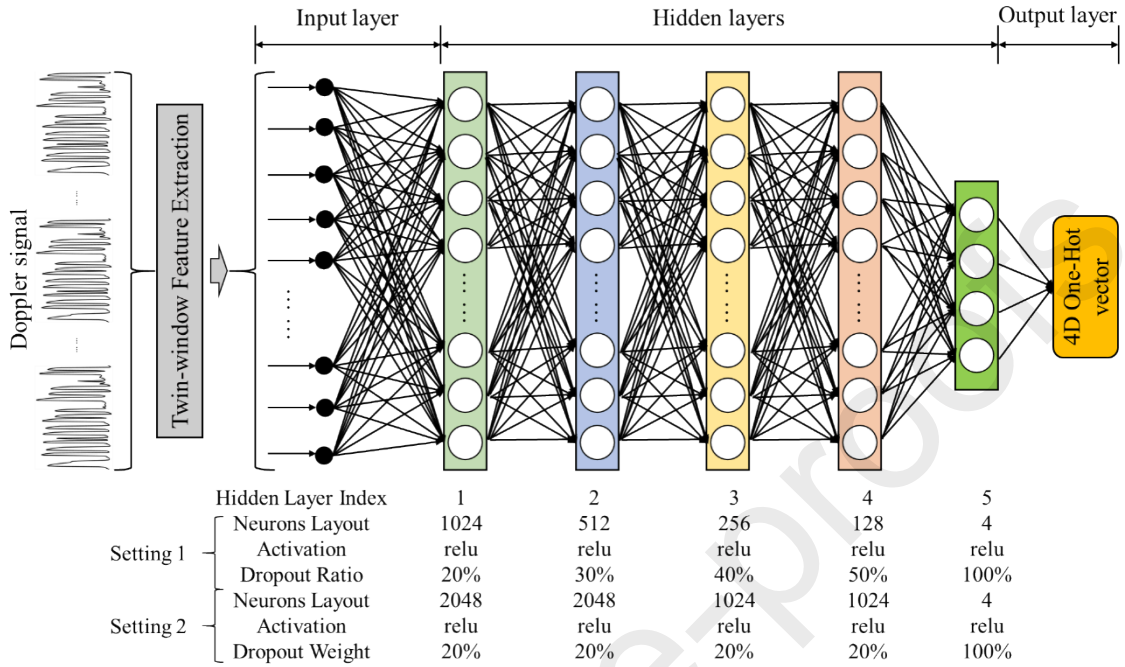
According to Sections 2.1 and 2.2, the complete classification signal for each flow regime is shorter than 2 seconds (s), while the entire sample is 130 s, which means the information per sample is theoretically far more than necessary. According to the Nyquist criterion, the segmentation length of each sample should be 5 times longer than the complete signal which can ensure the integrity of the information. Thus, the length of the intermediate window should technically be longer than or equal to 10,000 values. To facilitate the discussion, the TFE has been illustrated in Figure 2-5, where the intermediate window is *window B*, and a larger *window A* lies outside the multiple *window B*s. *Window A* splits the sample into multiple sub-segments. *Window A step length* represents the moving step between two adjacent sub-segments. To ensure the continuity of the sub-samples, *(window A length – window A step length)* corresponds to the overlap between different *windows A*s. In a given *window A*, the *intermediate window* corresponds to a

given *window B*. The *window B step length* represents the step length, and *(window B length – window B step length)* represents the overlap length. Then, the FFT operation is performed with each *window B*. This paper uses the arithmetic mean from all *window B*s as the extracted feature vector corresponding to *window A*. Therefore, a sample can be divided into an integer number of *(sample length/window A)*, which increases the total number of samples.

According to the above explanations, the TFE can expand the sample space without losing much classification accuracy. It is noteworthy that the TFE algorithm makes a significant impact on solutions that use deep learning methods for the problem of long and complicated one-dimensional signals, which also can serve as inspiration for other similar challenges. One of the important constraints for deep learning solutions is the size of the labelled (i.e., the ground-truth) sample-space [42]. The ultrasonic Doppler signal in this study is difficult not only to obtain but also to manually label. The TFE can significantly expand the sample-space, which can improve the performance of the DNNs classifier. (In Algorithm 1 in the appendix, this paper provides pseudo-code for the proposed TFE algorithm.)

Journal Pre-proofs

2.4 Architectures and mathematical expressions of the DNNs classifier



	Hidden Layer Index	1	2	3	4	5
Setting 1	Neurons Layout	1024	512	256	128	4
	Activation	relu	relu	relu	relu	relu
	Dropout Ratio	20%	30%	40%	50%	100%
Setting 2	Neurons Layout	2048	2048	1024	1024	4
	Activation	relu	relu	relu	relu	relu
	Dropout Weight	20%	20%	20%	20%	100%

Figure 2-6: Deep Neural Networks (DNNs) model architecture. The left-hand side of the DNNs model represents the input Doppler signal. The grey box represents the TFE pre-processing algorithm. The light green, blue, light orange, and light red rectangles represent the four hidden layers.

The essential task for this classifier is classification, and this paper uses a DNN architecture to perform the classification task. The inputs are vectors that contain all the information. A DNNs classifier can directly input all information into neural networks and automatically identify the implicit connections within the data. With a deep and wide neural network structure, the DNNs classifier has substantial information analysis abilities [43]. The process of extracting feature vectors using the TFE algorithm in Section 2.3 can be essentially understood as a process of mapping the raw data to high-dimensional vector space. This paper uses a DNNs classifier to train the flow regimes classification model in this high-dimensional feature vector space.

As shown in Figure 2-6, the DNNs classifier proposed in this paper can be divided into two settings corresponding to different TFE vector lengths. This paper represents these two architectures as two settings of the DNNs classifier and names them Setting 1 and Setting 2.

Setting 1 and Setting 2 are both "narrow-down" architectures. The inputs have relatively long lengths, but the outputs are always four-dimensional (4D) vectors. The overall process of the DNNs classifier can be understood as another feature extraction process, which eventually summarizes all the information from a long input to a short output. More specifically, the calculations between adjacent hidden layers are also implicit feature extraction processes. It is noteworthy that the essence of feature extraction is to summarize the key information and abandon other information. Therefore, Setting 1 halves the vector length in every hidden layer, which provides a smoothed architecture to prevent losing too much information in a single calculation. Setting 2 is designed to handle the case of very long input vectors, but it significantly increases the computational complexity and is only necessary if the input vector is far longer than the first hidden layer.

This DNNs classifier has four hidden layers. There is no specific design rule for DNNs, and this paper uses the integer power of two as the number of neural nodes for each hidden layer. Therefore, setting-1 layouts with 1024, 512, 256, and 128 neural nodes, and setting-2 layouts with 2048, 2048, 1024, and 1024 neural nodes. It is clear that the number of hidden nodes of Setting 2 is 256 times the number in Setting 1.

Each hidden layer consists of one fully connected layer, one relu activation layer [42] [43], and one dropout layer. The dropout ratios for Setting 1 are 20%, 30%, 40%, and

Journal Pre-proofs

50%. As mentioned previously, each layer is an implicit feature extraction process, thus the shallow hidden layer contains a higher ratio of noisy information. The function of the dropout layer is to prevent overfitting, therefore Setting 1 retains fewer nodes in shallow hidden layers but more nodes in deep hidden layers. Setting 2 keeps all the dropout ratios at 20%. Setting 2 is designed to handle conditions with long inputs which contain far more information than needed. The solution in this paper is to increase the complexity of the DNNs, investing more computational power to address the information increase.

The output layer is a "one-hot" format label, which means the label is a vector. The four flow regimes are labelled in a *4D* vector, (1,0,0,0), (0,1,0,0), (0,0,1,0) and (0,0,0,1), and the multiple classification is changed to a binary classification, which prevents the fuzziness of an intermediate value. The cost function used in the DNNs classifier is the mean square error.

3 Test Rig and Experimental Procedure

3.1 Two-phase flow test rig set-up

The experimental test was conducted at Cranfield University on a 2-inch S-shaped riser of the multiphase flow facility. The S-shaped 2-inch flow test facility consists of a 3.5-m topside section, a 5.7-m vertical upper section, a 1.5-m down-comer, a 5.5-m vertical lower section, and a 40-m horizontal pipeline. The flow loop is managed with DeltaV supervisory control and data acquisition programming software. The air flow rate utilised was provided from a bank of parallel-connected compressors. When the parallel-connected compressors are operated, 1410 m^3/hr free air delivery maximum air flow rate at 7 bar can be dispensed. An 8-m^3 capacity receiver is used to accumulate the air from the parallel-connected compressors to minimise variations in pressure. Air flow rate from

the receiver is passed through fine, medium, and coarse filters before passing through a cooler to filter out the condensates and debris in the air before it goes to the flow meters. The water flow was supplied from a 12.5-m^3 capacity water tank. Two multistage Grundfos CR90-5 pumps with a duty of 100 m^3/hr at 10 bar delivered water into the flow loop. The DeltaV software was used to automate operation of the water pumps. An approximately 30 kg/s 3-inch Foxboro CFT50 Coriolis meter and 7.36 l/s 1-inch Rosemount 8742 magnetic flow meter were used to measure the water flow rate.

An 11.12-m^3 horizontal multiphase gravity separator was used to separate the water and air at the end of the experiments. The water flowed into a 1.6-m^3 coalescer, where it was cleaned before being stored in the repository tank, while the air was released into the atmosphere after cleaning.

3.2 The S-shaped riser multiphase flow loop

The 2-inch multiphase flow S-shaped riser test facility used for this paper had a 1.5-m down-comer, an internal diameter of 54.8 mm, and a 40-m length. To allow visual observations of the flow regime, the S-shaped pipeline riser had a transparent section along the vertical riser. To achieve the flow regime desired, the air-water flow rate in the pipeline was controlled automatically using the DeltaV software [44]. A schematic diagram of the S-shaped riser is displayed in Figure 3-1.

Journal Pre-proofs

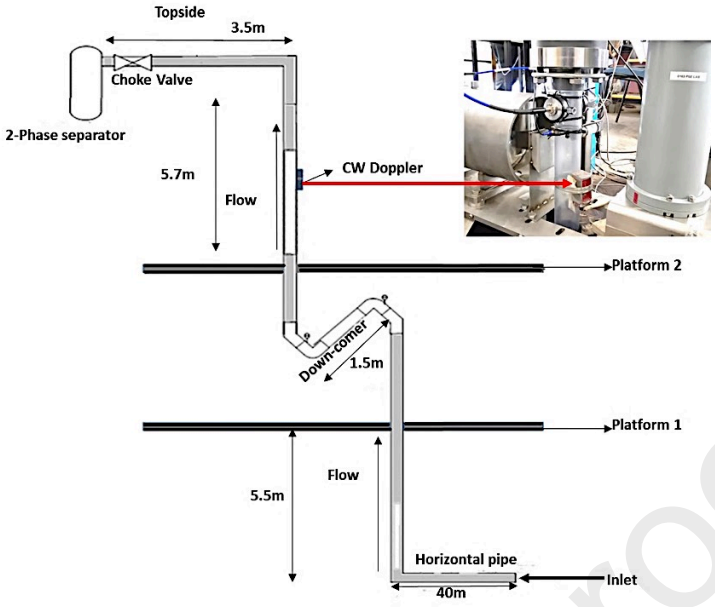


Figure 3-1: Schematic diagram of the S-shaped rig.

3.3 Platform and hardware for training the DNNs classifier

This paper used Tensorflow as the training platform, which integrates Compute Unified Device Architecture (CUDA) based Graphics Processing Unit (GPU) acceleration [45]. The sklearn library [46] was used to label the ground-truth data, and the training processes and results were visualized using the matplotlib library. In terms of hardware, the workstation central processing unit was an Intel Core i7-7700, the memory size was 32 GB, and the GPU was an NVIDIA GTX1080.

4 Results and Discussions

4.1 Data pre-processing

The feature extraction method used in this paper was discussed in Section 3. To evaluate the influence of all relevant parameters – for example, the window B length, the window B step length, and the FFT sampling rate value (N) – the raw samples were divided into 16 datasets corresponding to different control parameters. The detailed technical

specifications are listed in Table 1. It is noteworthy that even though this paper uses the word "dataset" to indicate the 16 experiments conducted, the actual meaning of dataset is the different outcomes of different TFE settings, but the original data (the 125 samples) is the same.

Table 1: DNNs architecture experimental settings

idx	*NSS (samples)*	*WAL (digits)*	*WAS (digits)*	*WBL (digits)*	*WBS (digits)*	*N (digits)*	*TVR*	*BS (samples)*	*CB*
0_1	125	1,300,000	325,000	10,000	¼ (2500)	1024	20%	8	93%
Control variable: *WBL (length of window B)*									
1_1	125	1,300,000	325,000	**20,000**	¼ (5000)	1024	20%	8	93%
1_2	125	1,300,000	325,000	**5,000**	¼ (1250)	1024	20%	8	93%
Control variable: *WBS (window B step length)*									
2_1	125	1,300,000	325,000	10,000	**½ (5000)**	1024	20%	8	93%
2_2	125	1,300,000	325,000	20,000	**½ (10,000)**	1024	20%	8	93%
Control variable: *N (N-orders value of the fast Fourier transform)*									
3_1	125	1,300,000	325,000	10,000	¼ (2500)	**2048**	20%	8	93%
3_2	125	1,300,000	325,000	10,000	¼ (2500)	**4096**	20%	8	93%
3_3	125	1,300,000	325,000	10,000	¼ (2500)	**512**	20%	8	93%
3_4	125	1,300,000	325,000	10,000	¼ (2500)	**256**	20%	8	93%
3_5	125	1,300,000	325,000	10,000	¼ (2500)	**128**	20%	8	93%
3_6	125	1,300,000	325,000	10,000	¼ (2500)	**64**	20%	8	93%
Control variable: *WAL (length of window A)*									
4_1	625	**650,000**	162,500	10,000	¼ (2500)	1024	20%	8	93%
4_2	2125	**260,000**	65,000	10,000	¼ (2500)	1024	20%	8	93%
4_3	4625	**130,000**	32,500	10,000	¼ (2500)	1024	20%	8	93%
4_4	6125	**100,000**	25,000	10,000	¼ (2500)	1024	20%	8	96%
4_5	12,625	**50,000**	12,500	10,000	¼ (2500)	1024	20%	8	96%

idx identifies the various experimental adjustments: the first digit represents different control variables, and the second represents the specific test. NSS represents the number of samples in the new sample space. WAL is the length of window A. WAS is the window A step length. WBL is the length of window B. WBS is the window B step length. N is the N-orders value of the fast Fourier transform. TVR represents the ratio of the testing set over the sample space. BS is the batch size of the DNNs classifier. CB is the call-back point, which is the stopping point of the DNNs classifier.

The 0_1 dataset directly inputs each sample with a length of 1.3 million digits into the *TFE* intermediate operation, the number of the original sample is 125. The length of window B for the *FFT* is 10,000, the window B step length is 1/4 window, the number of

samples is 2048, and the training-set and the testing-set are divided along the scale of 80%:20% after shuffled. Eighty percent of the data (100 samples) is used to train the classifier, and 20% (25 samples) is used to test it.

Here are all other dataset specifications:

1) Compared with 0_1, the window sizes (the length of window B, WBL) of datasets 1_1 and 1_2 for the *FFT* are different, which are 20,000 and 5000 digits. Other parameters remain the same as in 0_1.
2) Datasets 2_1 and 2_2 separately correspond to 0_1 and 1_1, and window B step lengths are ¼ (2500 digits) and ½ (5000 digits). Other parameters remain the same.
3) Datasets 3_1, 3_2, 3_3, 3_4, 3_5, and 3_6 uses the window B step length of ¼ (2500 digits) and the window B length of 10,000 digits. The experimental parameter is the *N*-orders number for the *FFT*. These are 4096, 8192, 1024, 512, 256, and
4) 128 digits, respectively. In other words, the corresponding *TFE* feature vectors are the lengths of 2048, 4096, 512, 256, 128, and 64. It is noteworthy that changing the *N*-orders value for the *FFT* changed the reserved information in the *TFE* feature vector. The above datasets all used the overall sample as the intermediate signal, meaning the length of window B equalled the length of window A; thus they had the same number of samples, which was 125. This meant the training sample space was 100, and the testing sample space was 25. In other words, the testing accuracy had a very low confidence resolution. More specifically, 25 wrong tests corresponded to a 0% testing accuracy, whereas 25 right tests

corresponded to a 100% accuracy, and thus the confidence resolution was 4%. It is clear that the testing accuracy had low reliability and stability. The following experiments creatively overcame such issues based on the proposed TFE algorithm.

According to the deduction in Section 2.3, the TFE theoretically should expand the sample space without losing classification accuracy. To verify this important deduction, datasets 4_1, 4_2, 4_3, 4_4, and 4_5 set the length of window A to 650,000, 260,000, 130,000, 100,000, and 50,000, respectively. The window A step length was 1/4 for these five datasets, thus the sample spaces correspondingly expanded to 625, 2125, 4625, 6125, and 12,625. Therefore, the testing sample spaces were 125, 425, 925, 1225 and 2525, respectively, and the corresponding confidence resolutions were 0.8%, 0.235%, 0.108%, 0.082%, and 0.040%. Thus, the testing accuracies for these datasets were more reliable and stable, which is shown by the smoothness of the training and testing graphs in Section 4.2.

4.2 Hyperparameters of the proposed DNNs classifier

The detailed structure of the DNNs classifier was described in detail in Section 2. This section focuses on the hyperparameters of the DNNs classifier. In this paper, the DNNs classifier is an example of supervised learning. For the four flow regimes, this paper used a one-hot vector [46] to label the data, with the four labels being [1,0,0,0], [0,1,0,0], [0,0,1,0], and [0,0,0,1]. The one-hot label format is essential for the classification task, which transfers each type of flow regime to binary classification. The 0 and 1 in classification are significantly different, which can avoid the issue of label uncertainty. The “loss” (loss function) used in this paper was the mean square error, which was the

training criteria for the gradient descent. The optimizer used was the Adam optimizer, and the learning rate was set to 0.0001. The batch size was 8 samples per batch, and the initial training round was limited to 10,000 times, but the bounce point was also set to terminate the training when the training accuracy reached a certain value. To better observe the training trend, the training accuracy, training loss, verification accuracy, and verification loss were reported for every epoch, and the corresponding confusion matrix was also output.

4.3 Testing and analysis of pre-processing datasets

This section focuses on the impact of using the TFE algorithm to increase the number of samples. This paper uses the term "dataset" to refer to the outputs of the TFE pre-processing algorithm. This is done because the source data is from the Doppler signals; different TFE hyperparameters were used to pre-process them and they produce different outputs (as shown in Figure 4-1). They are different data (in both the number of samples and values), therefore this paper uses "dataset" to refer to them to avoid confusion.

The results for dataset 0_1 are shown in Figure 4-1. First, when the training accuracy reaches the call-back point (93%), the testing accuracy reaches 87.55%. Second, in Figure 4-1(B), the training loss is 0.0455 and the verification loss is 0.0448. Third, in the confusion matrix, shown in Figure 4-1(C), the testing accuracies of categories 3 and 4 are very high, but the accuracies of categories 1 and 2 are less than 80%, thus the right-true ratios of categories 1 and 2 are relatively poor.

The results of dataset 0_1 seem good, indicating the DNNs classifier can work efficiently on this flow regime identification. However, there are many problems with dataset 0_1. First, although both training and testing accuracy are relatively good, the accuracy line is

discontinuous. It is noteworthy that its overall trend still generally indicates the approximate accuracy, but the accuracy resolution is insufficient, which means the accuracy of 87.55% cannot represent the model accuracy. Second, the training and testing loss is relatively low, however, the training loss keeps decrease while the verification loss stays stable, which indicates that the training model is overfitting. In other words, the model might fit the training data well, but the generalization-ability is not good.

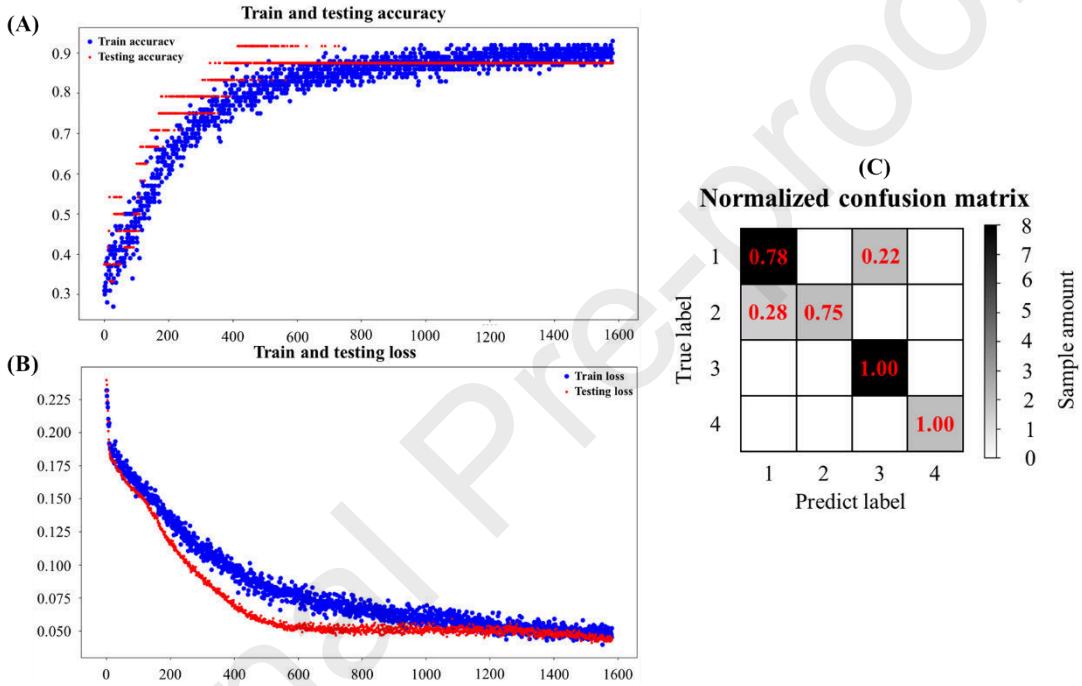


Figure 4-1: The DNNs classifier performance for data set 0_1 showing the training and testing accuracy and loss and the overall confusion matrix. (A) indicates the curves of training and testing accuracies, (B) indicates the curves of training and testing losses, (C) indicates the normalized confusion matrix. In (A) and (B), the blue curves correspond to the training results, and the red curves correspond to the testing accuracy. The horizontal axis corresponds to the training loops, the vertical axis corresponds to the metric (percentage for accuracy and L2 distance for loss). In (C), the horizontal axis represents predictions of the four flow regimes, and the vertical axis represents the ground-truth labels. The grey-scales for blocks represent the number of samples corresponding to that block. The following figures have similar displays, thus the notation is explained only once.

Table 2 shows the results of datasets 1_1, 1_2, 2_1, 2_2, 3_1, 3_2, 3_3, 3_4, 3_5, and 3_6.

1) It can be seen in 1_1 and 1_2 that changing the length of window B has a weak impact on the classification accuracy. Compared to the window B length of 10,000, the training and testing losses are even increased. Therefore, 10,000 is a reasonable window B length setting for this study.

2) The results for 2_1 and 2_2 indicate the testing loss almost doubled, and the testing accuracy decreased, which indicates that when the window B step length increases, classification information is lost. Therefore, this paper keeps the window B step length fixed at 1/4 in the following experiments.

Table 2: The learning rate, training, and testing accuracies

idx	*LR*	Training Accuracy	Training Loss	Testing Accuracy	Testing Loss
1_1	0.0001	93.0%	0.0593	87.50%	0.0543
1_2	0.0001	94.0%	0.0553	83.33%	0.0713
2_1	0.0001	94.0%	0.0477	75.00%	0.0988
2_2	0.0001	94.0%	0.0526	75.00%	0.1122
3_1	0.0001	96.0%	0.0386	91.67%	0.0501
3_2	0.0001	93.0%	0.0592	91.67%	0.0466
3_3	0.0001	93.0%	0.0488	91.67%	0.0554
3_4	0.0001	93.0%	0.0481	87.50%	0.0556
3_5	0.0001	93.0%	0.0525	87.50%	0.0555
3_6	0.0001	93.0%	0.039	87.50%	0.0573

idx represents the index of datasets. LR is the learning ratio of the corresponding DNNs classifier. Interestingly, even when multiple settings have been changed among these 10 datasets, the testing

accuracies stay on three values, 75%, 87.5%, and 97.67%, which means the sample-space has a greater influence on this result than the other parameters do.

3) Datasets 3_1, 3_2, 3_3, 3_4, 3_5, and 3_6 display good accuracy and loss values. However, as shown in Figures 4-2, 4-3, 4-4, 4-5, 4-6, and 4-7, the local oscillation of the accuracy and loss curve is still poor, which indicates that the training result is unstable and unreliable. Moreover, it can be seen from the loss image that the over-fitting condition still exists due to the small sample space.

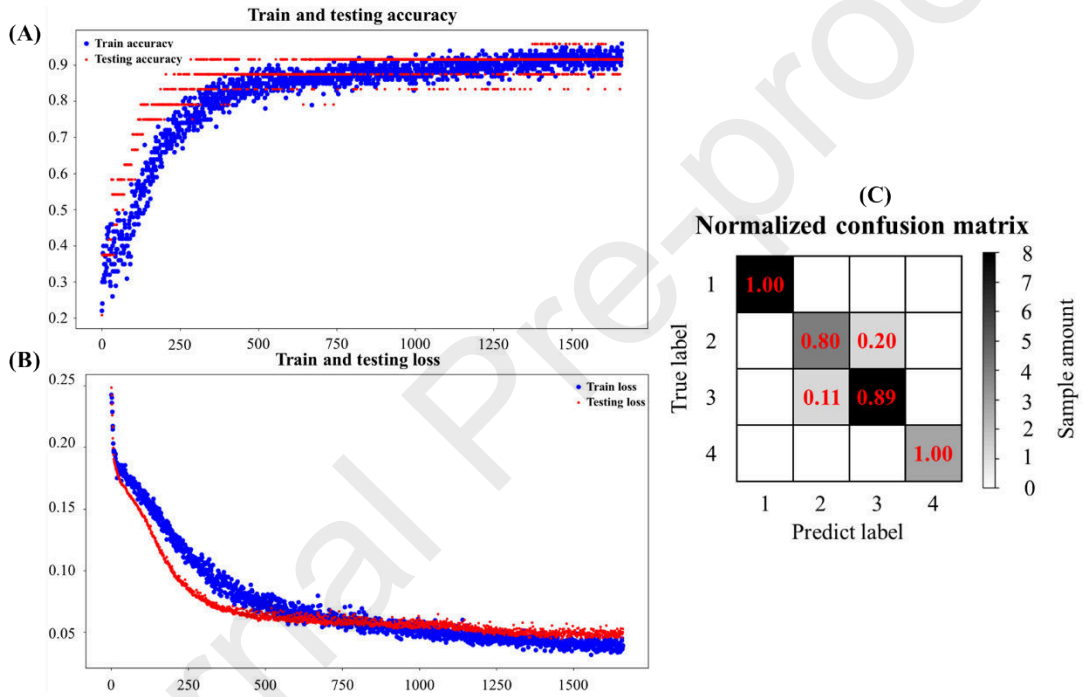


Figure 4-2: The DNNs classifier performance for dataset 3_1 showing the training and testing accuracy and loss and the overall confusion matrix. The situation of discontinuity of testing accuracy is the same as for dataset 0_1. The loss curves also show overfitting occurred.

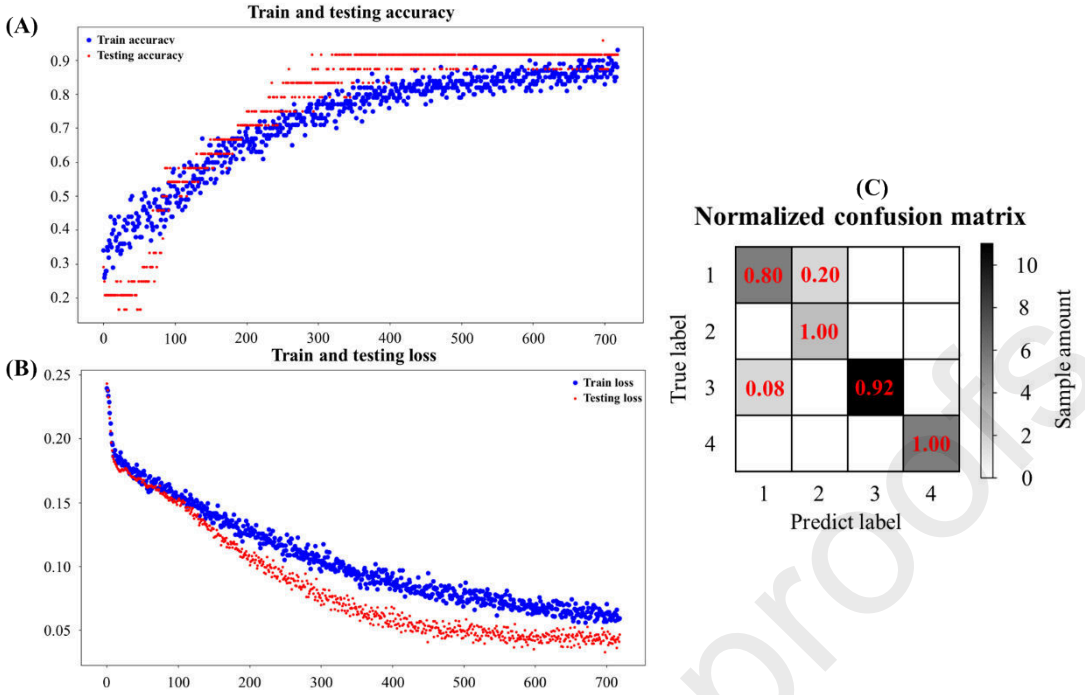


Figure 4-3: The DNNs classifier performance for dataset 3_2 showing the training and testing accuracy and loss and the overall confusion matrix. The situation of discontinuity still exists, but the loss curves become smoother, which indicates that the overfitting seems to disappear. However, it is noteworthy that the testing loss curve moves below the training loss curve, which indicates that the model performs better with new data than with training data. There is something wrong with this result. Dataset 3_2 uses a large N value (4096), but the sample space remains the same as for the previous datasets. In other words, 3_2 tries to fit more features into a small sample-space; overfitting still occurs.

Journal Pre-proofs

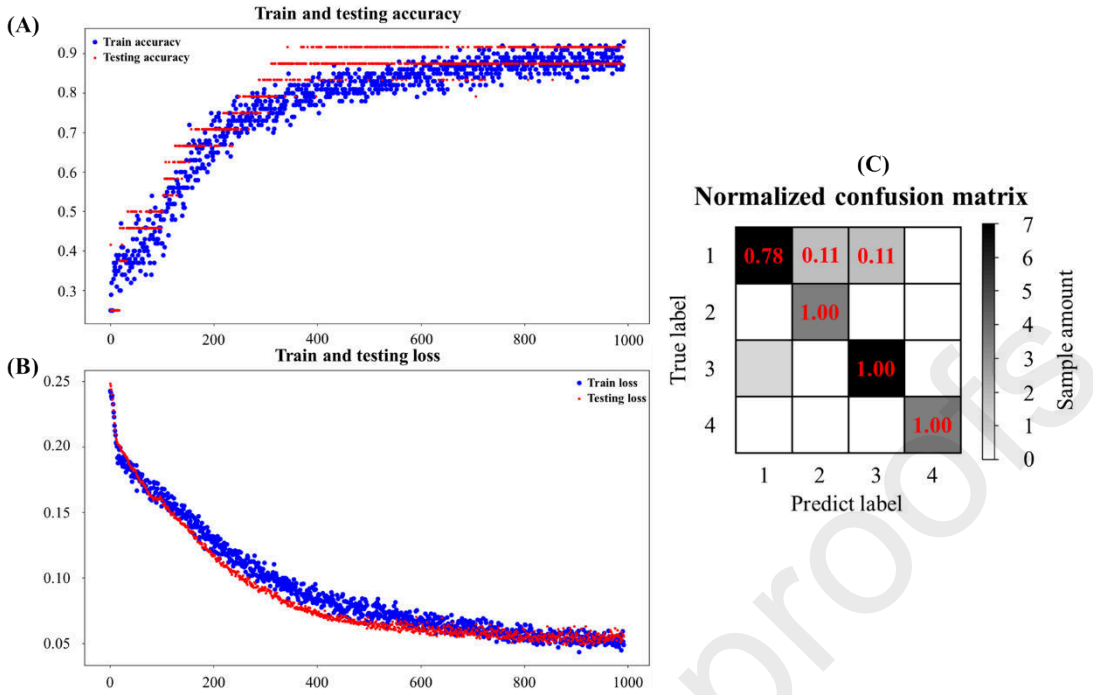


Figure 4-4: The DNNs classifier performance for dataset 3_3 showing the training and testing accuracy and loss and the overall confusion matrix. 3_3 verifies the analysis of 3_2. 3_3 decreases the N value from 4096 digits to 512 digits, and the loss curves become more stable and reasonable. However, as the N value (the number of learning features) decreases, the training and testing-accuracy curves become unsteady. Moreover, the situation of discontinuity still exists. The remaining three datasets, 3_4, 3_5, and 3_6, decrease the N value to highlight its impact and to identify a proper N value for the following experiments.

Journal Pre-proofs

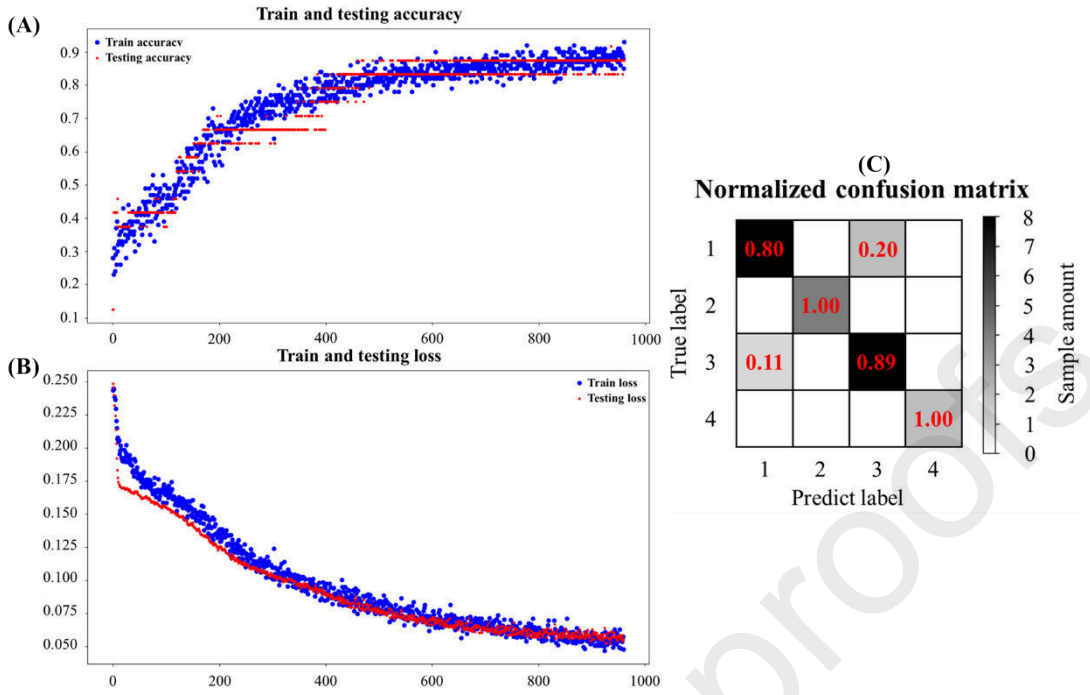


Figure 4-5: The DNNs classifier performance for dataset 3_4 showing the training and testing accuracy and loss and the overall confusion matrix.

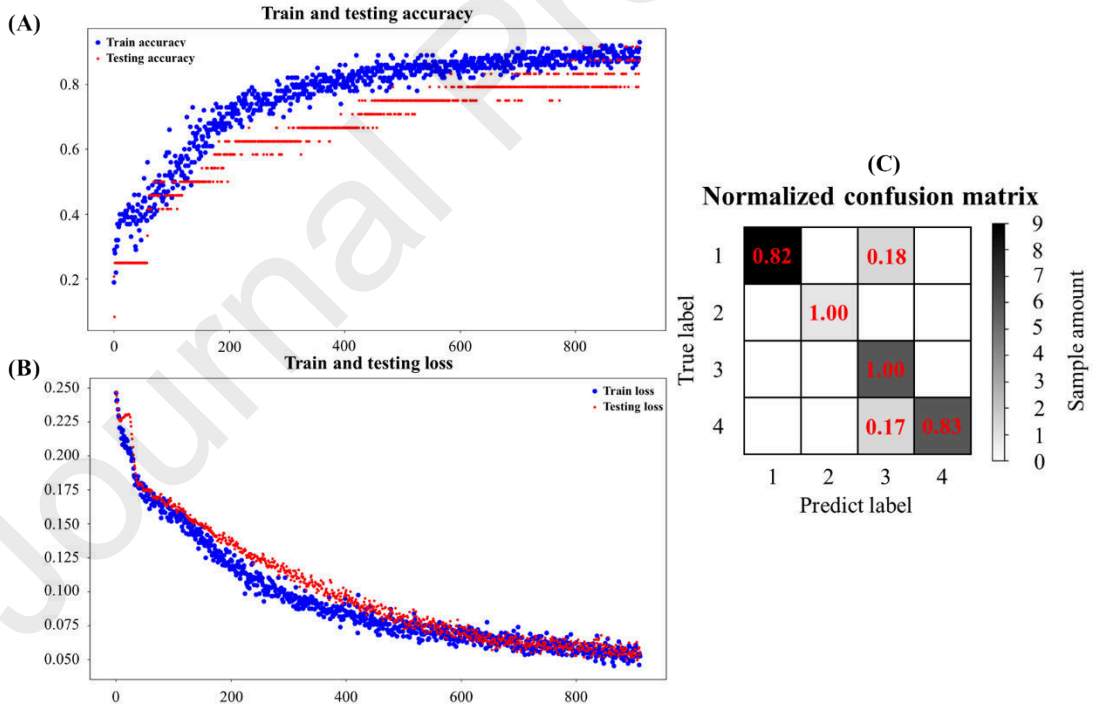


Figure 4-6: The DNNs classifier performance for dataset 3_5 showing the training and testing accuracy and loss and the overall confusion matrix.

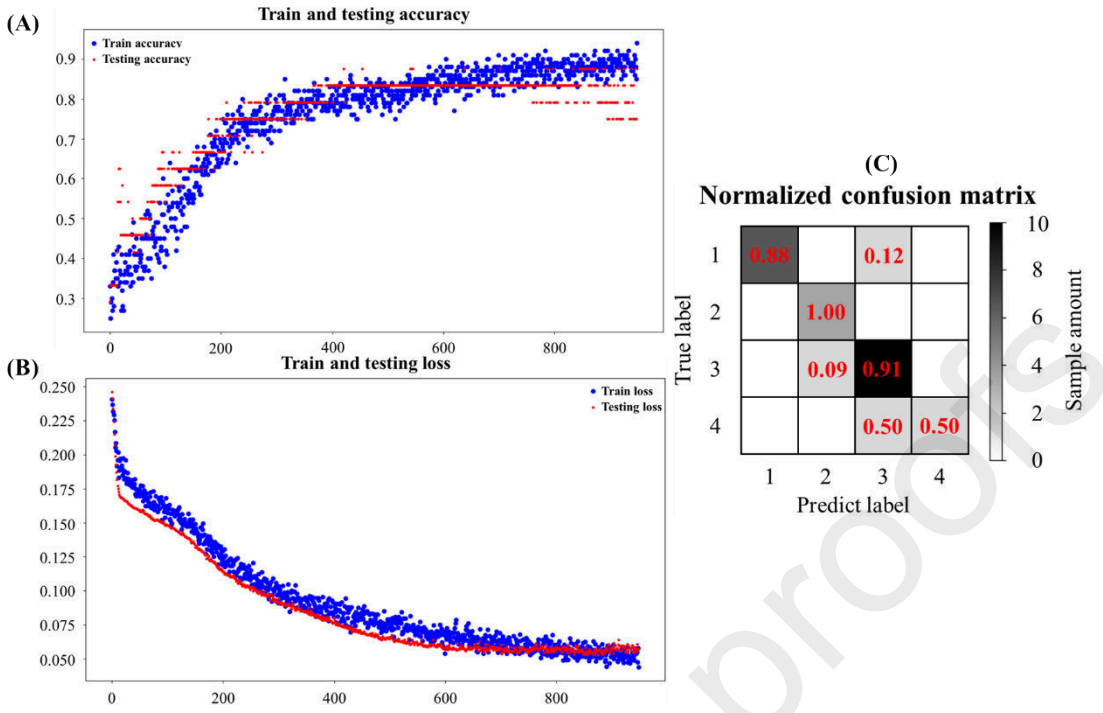


Figure 4-7: The DNNs classifier performance for dataset 3_6 showing the training and testing accuracy and loss and the overall confusion matrix.

As mentioned in Sections 2.1 and 2.2, the raw samples are collected by repeating the experiment 125 times, which is a huge and complicated task. Collecting all the required data by conducting more experiments is unrealistic. Considering the deduction in Section 2.3, the proposed TFE algorithm should expand the sample space without losing classification accuracy. Therefore, in the 4_*x* experiments, the length of window A was adjusted to verify the Section 2.3 deduction.

4) The specific operation flow has been described in Section 4.2. From Figures 4-8, 4-9, 4-10, 4-11, and 4-12, it can be seen that, as the New Sample Space (NSS) expanded, the verification accuracy steadily increased, and the training loss and verification loss decreased significantly. Furthermore, the loss and accuracy curves are quite smooth, which means that the training model is stable and reliable. Moreover, as the NSS

increased, the call-back iteration count decreased from 1600 iterations to less than 200 iterations, which indicates that the training efficiency of individual batch size improved by about 88%. Compared to the state-of-art support vector machine classifier used in [37], the performance of the DNNs classifier is superior by 9%.

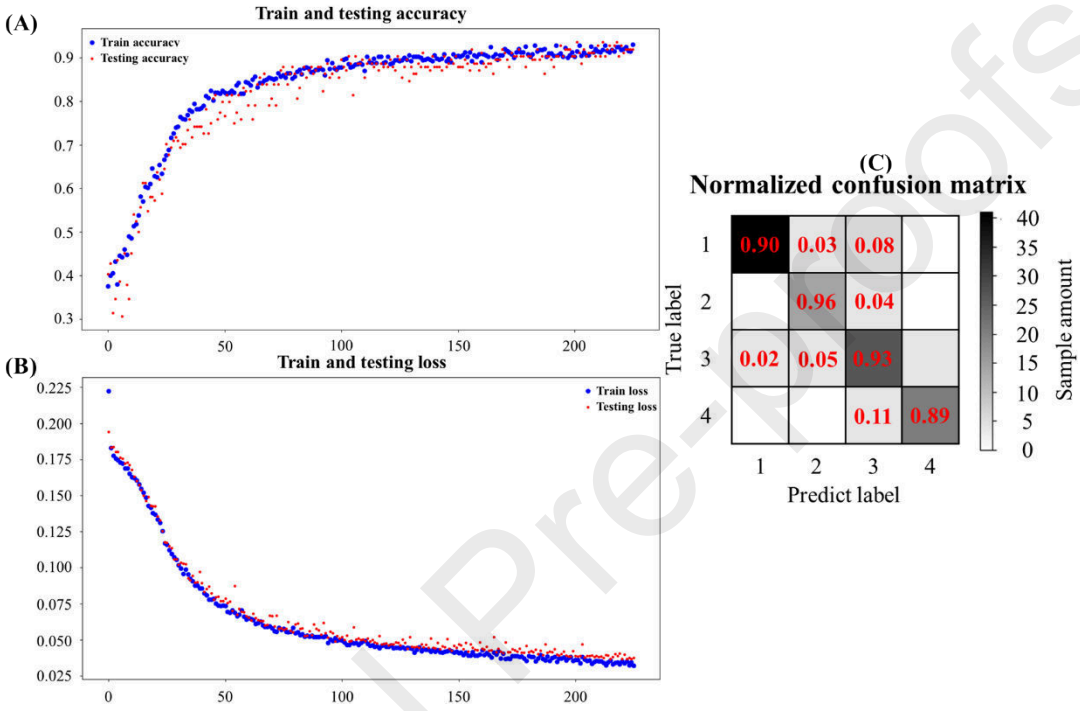


Figure 4-8: The DNNs classifier performance for dataset 4_1 showing the training and testing accuracy and loss and the overall confusion matrix. This result is interesting: as 4_1 uses the TFE method to increase the sample space, both accuracy and loss curves become much smoother. Furthermore, it is noteworthy that the number of training loops sharply decreases from 800+ iterations to just over 200. The distribution of training accuracy and testing accuracy also becomes more continuous, which means the model becomes more stable and reliable. The accuracy resolution of the testing curve becomes more reasonable.

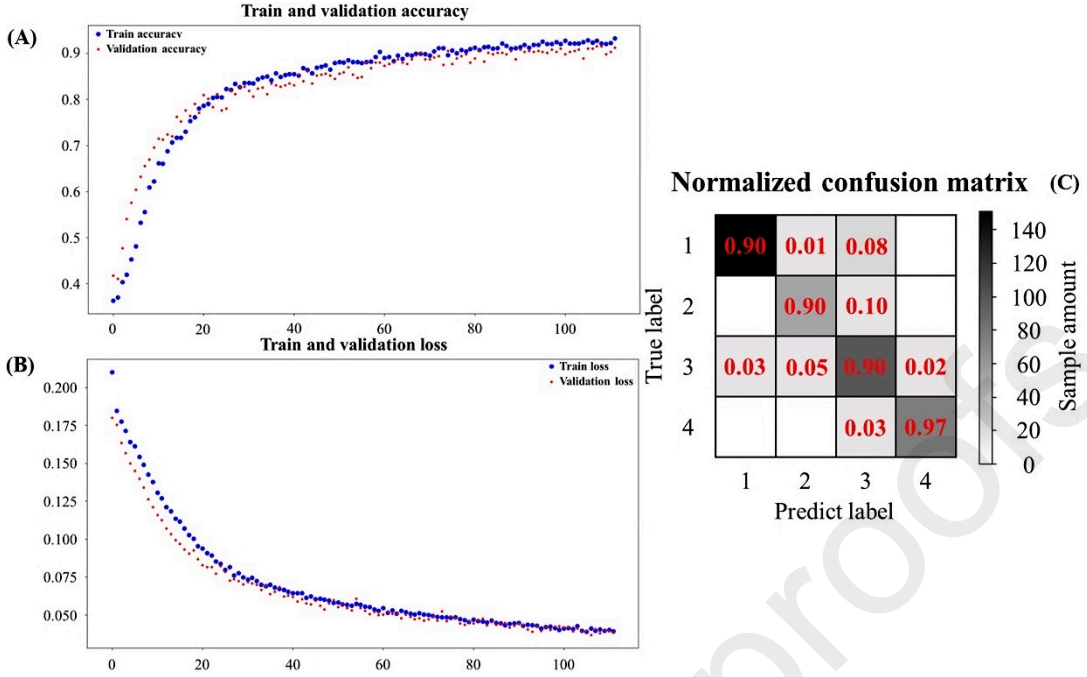


Figure 4-9: The DNNs classifier performance for dataset 4_2 showing the training and testing accuracy and loss and the overall confusion matrix. Even though the 4_1 dataset produced good results, and the curves are stable, this paper used datasets 4_2, 4_3, 4_4, and 4_5 to explore the capacity of the TFE algorithm for the DNNs classifier.

Journal Pre-proofs

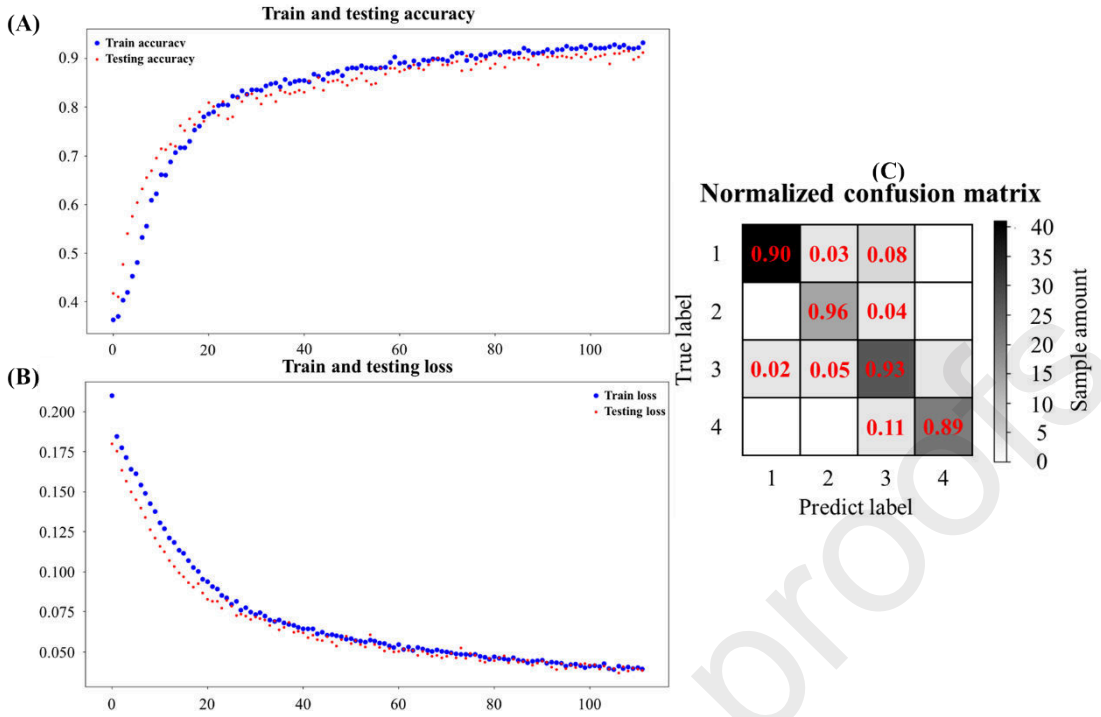


Figure 4-10: The DNNs classifier performance for dataset 4_3 showing the training and testing accuracy and loss and the overall confusion matrix. The number of loops was 50, which significantly increased training efficiency.

Journal Pre-proofs

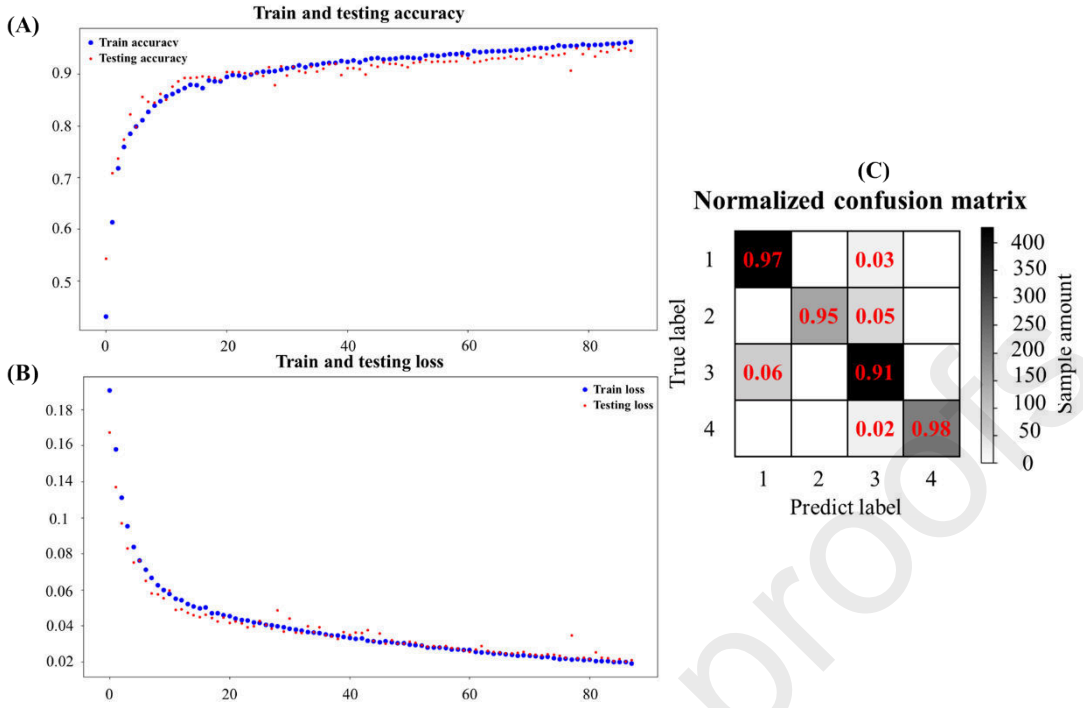


Figure 4-11: The DNNs classifier performance for dataset 4_4 showing the training and testing accuracy and loss and the overall confusion matrix.

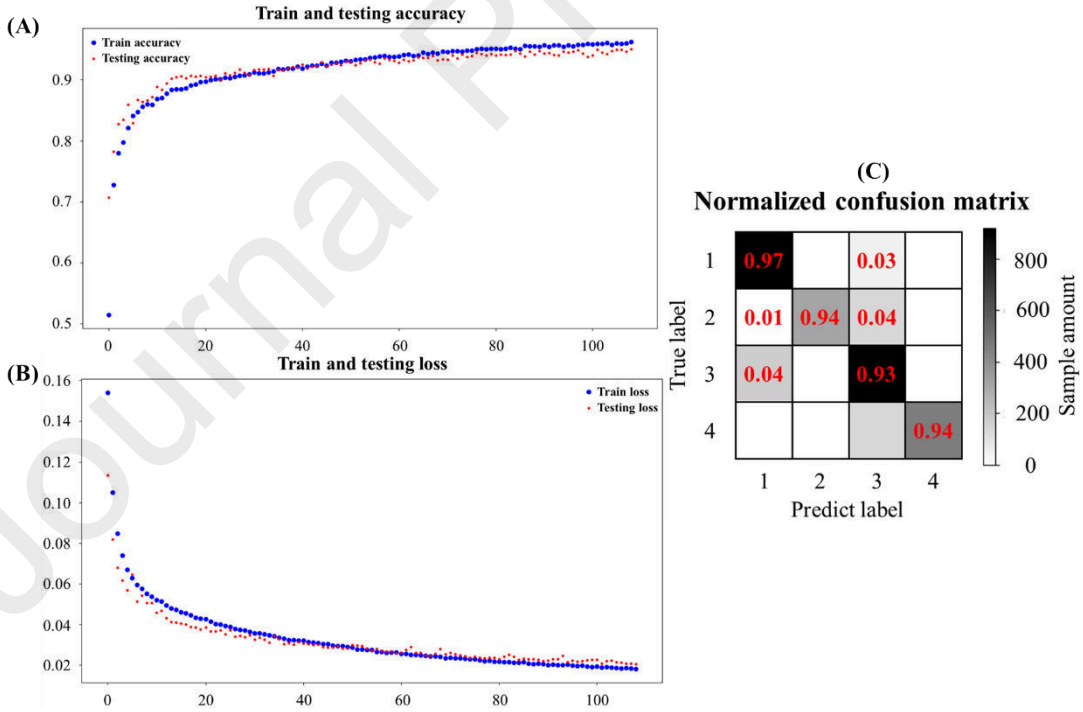


Figure 4-12: The DNNs classifier performance for dataset 4_5 showing the training and testing accuracy and loss and the overall confusion matrix. Within Nyquist's theorem, 4_5 is the largest

sample space the TFE algorithm can increase. The accuracy can increase further before the model goes into overfitting.

4.4 Further evaluation for overfitting challenge

This section further trains the proposed model based on the results reported in Section 4.3 and evaluates its robustness for the overfitting challenge. The risk of overfitting is common in neural networks, even though the project adopted a series of measures to suppress overfitting, as noted in Section 4.2. Section 4.3 discussed in detail the impact of different preprocessing methods on the stability and efficiency of DNNs training, making clear that the TFE algorithm can significantly improve the DNNs' stability and efficiency. Therefore, this section utilizes a pre-processed dataset 4_5, because it showed the highest stability and efficiency. The basic hyperparameter settings were kept generally the same, however, the shuffled 4_5 dataset was divided into a training set (7575 samples), a testing set (2524 samples), and a validation set (2526 samples) according to a 60%:20%:20% ratio. The training set was used in the training process, and the testing set was used for cross-validation. The validation set was used to assess the overfitting of the training model after the training task had been completed.

The goal in this section is to obtain a two-phase flow prediction model with the highest possible accuracy, and which is generalizable. The training call-back was set at 99%. Figure 4-13 presents the training and testing accuracy curve, Figure 4-14 presents the training and testing loss curve, Figure 4-15(A) presents the testing confusion matrix, and Figure 4-15(B) presents the validation confusion matrix.

It is noteworthy that, when the training accuracy reaches 99.01%, the training loss is 0.0060, the testing accuracy is 96.28%, and the testing loss is 0.0152. Furthermore, the validation accuracy is 96.35%, and the validation loss is 0.0159. Figures 4-13 and 4-14 curves are quite smooth. The excellent accuracy and loss results in the validation set also confirm that the model does not show overfitting. In other words, this model shows good generalization ability in the two-phase flow regime classification problem.

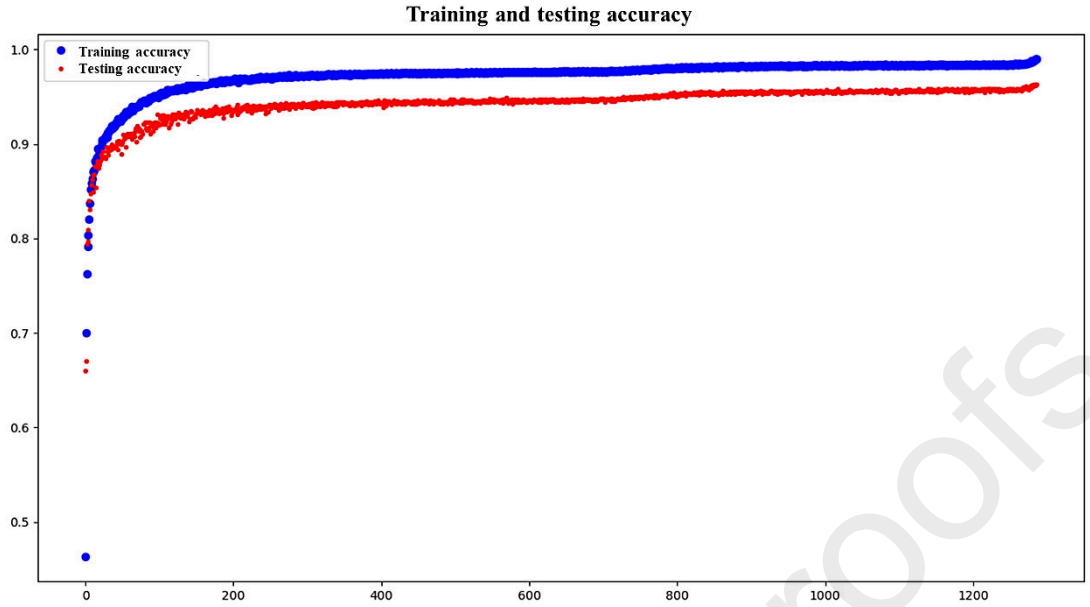


Figure 4-13: The training and testing accuracy curves. The training accuracy reaches 99.01%, and the testing accuracy reaches 96.28%. The overall change is smooth, which means the model and training process is highly stable. The trend continues to increase, which means the model does not show explicit overfitting.

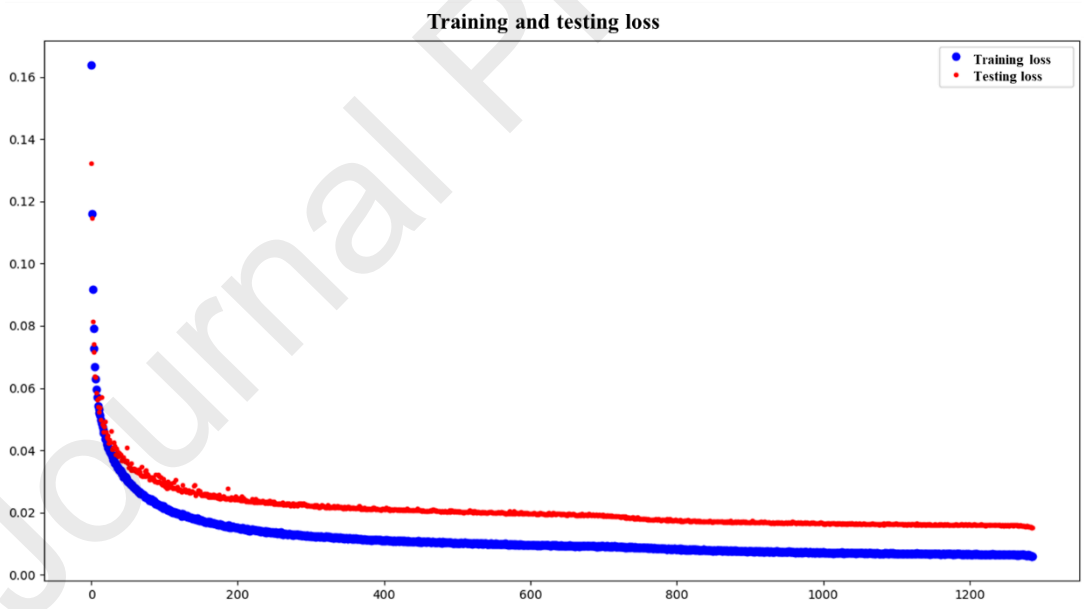


Figure 4-14: The training and testing loss curves. The training loss is 0.0060, while the testing accuracy is 0.0152. The overall change is smooth, which means the model and training process is highly stable. The trend continues to increase, which means the model does not show explicit overfitting.

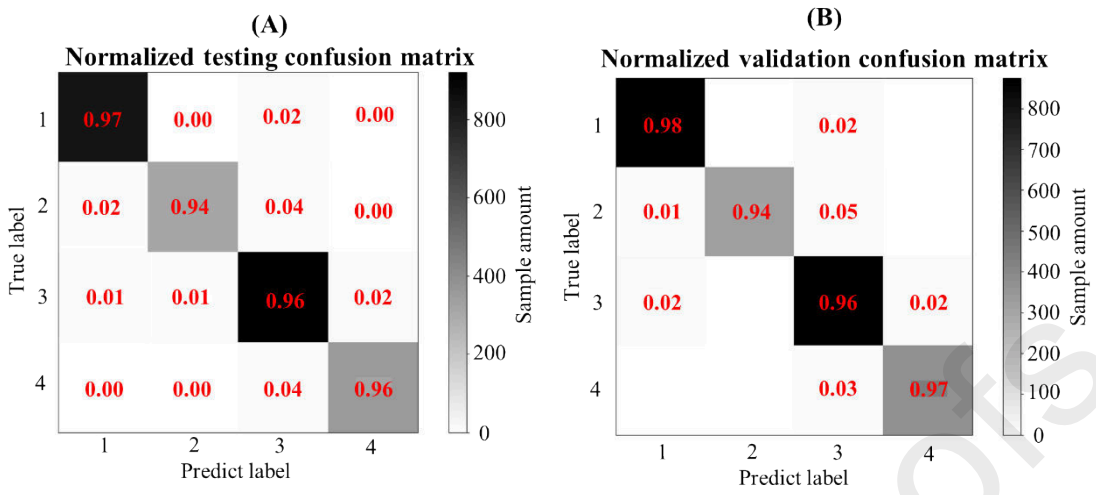


Figure 4-15: The testing and validation confusion matrices. (A) displays the normalised testing confusion matrix, and (B) displays the normalised validation confusion matrix. Both show high true-positive, true-negative, false-positive, and false-negative accuracy. The validation confusion matrix in particular demonstrates the proposed model has high robustness for the overfitting challenge and can be widely generalised to other situations.

4.5 Comparison experiments with machine learning classifiers

To further validate the method, this paper takes datasets 1_1, 4_1, and 4_5 and tests them in four conventional machine learning classifiers: AdaBoost, bagging, extra trees, and decision tree classifiers. The AdaBoost classifier uses the SAMME.R algorithm [47]. The base estimator of the bagging classifier is a decision tree. The criterion used for the extra trees classifier was “gini”. The criterion used for the decision tree classifier was also “gini” [48]. The specific results are shown in Table 3, and the corresponding confusion matrices are presented in Figures 4-16, 4-17, 4-18, and 4-19, respectively. When the NSS was expanded, except for the AdaBoost classifier, the other machine learning classifiers performed relatively well. However, the DNNs classifier proposed in this paper still has superior accuracy. Moreover, the deep learning model used in this paper uses GPU acceleration, which offers great advantages in training efficiency [49].

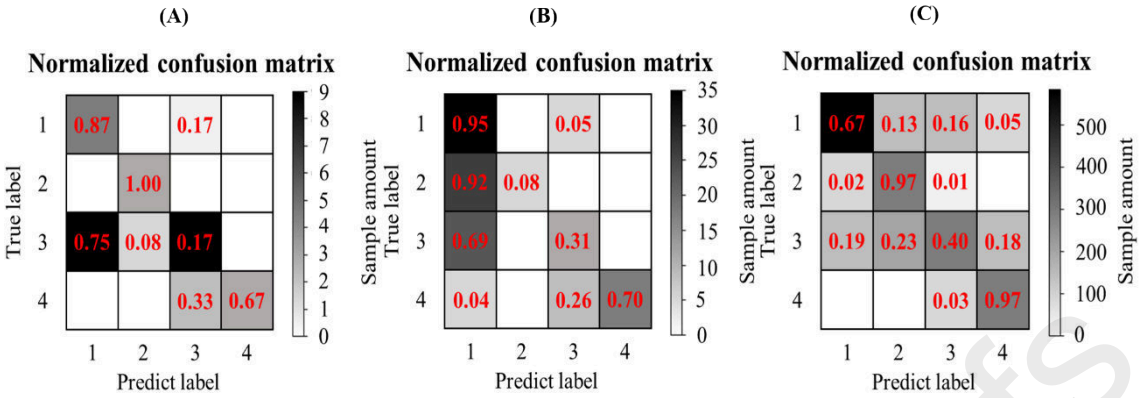


Figure 4-16: Confusion matrix of flow regime classification using the AdaBoost classifier.

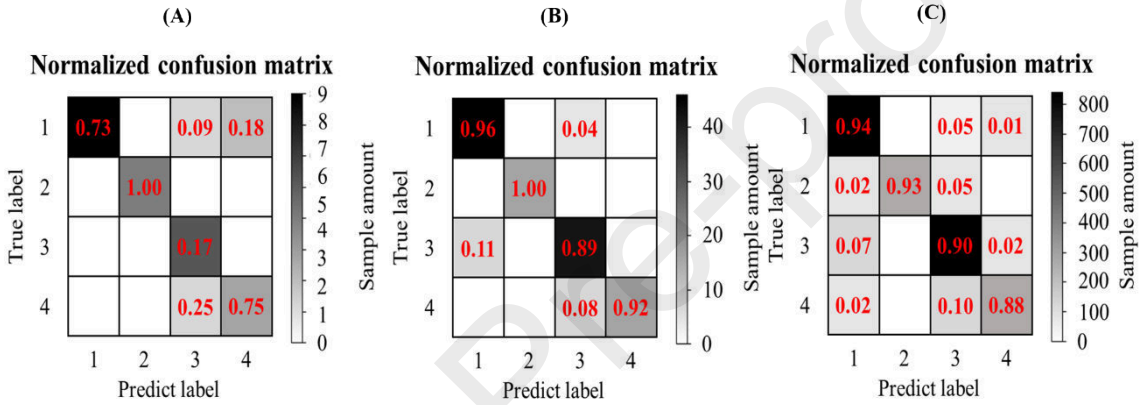


Figure 4-17: Confusion matrix of flow regime classification using the bagging classifier.

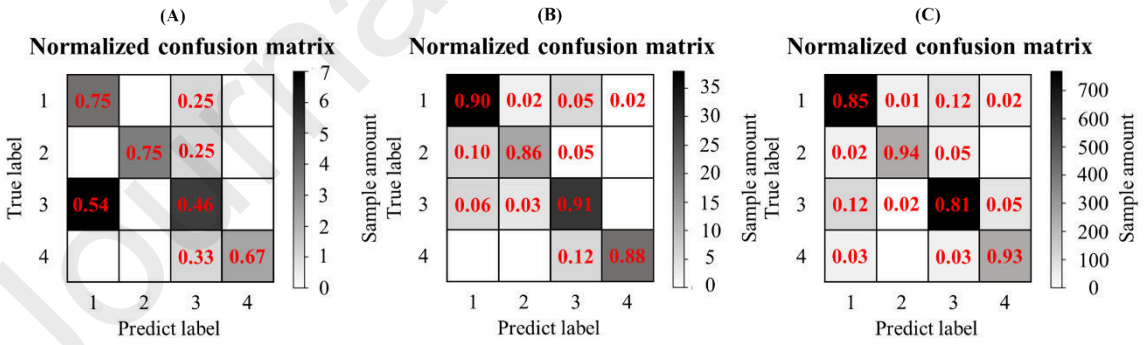


Figure 4-18: Confusion matrix of flow regime classification using the decision tree classifier.

Journal Pre-proofs

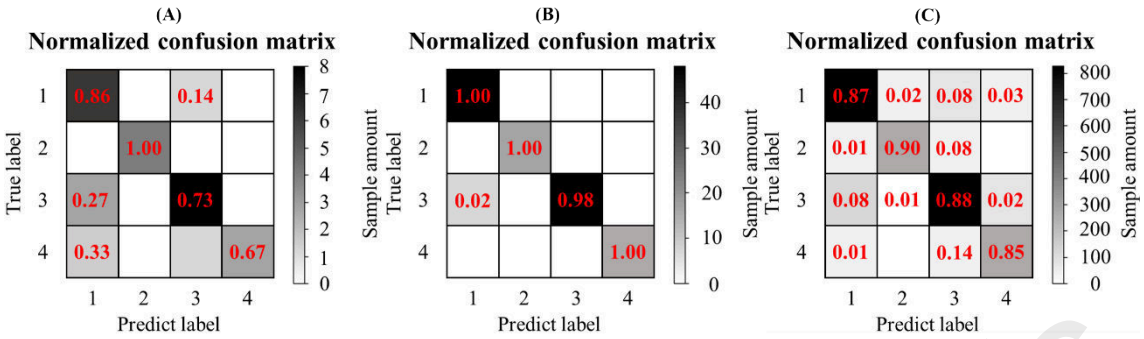


Figure 4-19: Confusion matrix of flow regime classification using the extra trees classifier.

Table 3: Comparison of all classifiers used for flow regime identification

S/N	Classifier	Performance
1	DNNs	94.97%
2	AdaBoost	55.35%
3	Bagging	86.21%
4	Extra trees	82.41%
5	Decision tree	80.03%

Table 3 presents the performance of the DNNs classifier and the conventional machine learning classifiers that were compared to it. The AdaBoost classifier performs well in bubbly and annular flow classification but worse in slug flow and churn flow classification and had an accuracy of 55.35%. The bagging classifier performs well in slug flow, bubbly, and churn flow classification but less well in annular flow classification, although its performance is still within an acceptable range as it yields 86.21% overall accuracy. The extra-trees classifier had an overall performance accuracy of 82.41%. The decision-tree classifier performance in the slug, bubbly, and annular flow classification is good, but its performance in churn flow classification is worse, yielding 80.03% overall accuracy. The DNNs flow regime predictions had an overall accuracy of 94.97%, demonstrating that the method is superior to the other machine learning methods tested.

5 Conclusion

Multiphase flow is a prevalent phenomenon in the chemical and petroleum engineering fields. The movement and multiphase flow interface structures are complex, which distribute heterogeneously and randomly on temporal and spatial scales having flow multivalue state and structures. The mass and energy transportation structure for each flow regime is different. It is important to identify the flow regime of a system to better understand the system's mechanism and physical phenomena.

An approach to two-phase flow regime classification using DNNs and CWDU was proposed and tested by the experiment. The experiment was carried out in an S-shaped pipeline-riser system. A novel TFE algorithm was proposed to extract the feature vectors from the raw ultrasonic signals. The algorithm not only reduced the data dimensions but also expanded the sample space without reducing classification accuracy. Sixteen methods for segmenting the ultrasonic signals were created and tested with the proposed DNNs architecture.

The proposed approach to two-phase gas-liquid flow regime identification in an S-shaped pipeline-riser system was successful and produced significant classification performance. The CWDU, TFE, and DNNs approach is more accurate, reliable, and stable than the four conventional machine-learning algorithms tested.

Journal Pre-proofs

6 Appendix

Algorithm 1: the pseudo-code for *TFE* algorithm

Input: $Data_{raw}$, raw data space with the matrix-shape of (125,1300000). 125 refers to the number of raw samples (N), the unit is *samples*. 1300000 refers to the length of each raw sample (L), the unit is *digits*.

Output: $Data_{TEF}$, the expanded data space.

1. initialise w_A (the length of window A), the unit is *digits*.
2. initialise w_B (the length of window B), the unit is *digits*
3. initialise w_A and w_B to contain window A and window B data segments.
4. $Sample_{TEF}$ refers to the extracted *TFE* sample.
5. initial a contain S_A (the step length of window A), the unit is digits.
6. initial a contain S_B (the step length of window B), the unit is digits.
7. $b = slice\ (a, p1, p2)$ is an operator which inputs a, then slices a from $p1$ to 2, then returns to b.
8. $d = FFT\ (e,val)$ is an operator that conducts *Fast Fourier Transformation* to input e with sampling rate val, and returns to d.
9. For each $i \in [i, N+1]$ do.
10. input raw sample $N\ o.i$ as $sample_i$
11. Input label $N\ o.i$ as $label_i$
12. **For** each $j_{wB} \in [1, w_A + 1]$ **do**
13. $WA = slice\ (sample_{i,}\ j_{wA} * S_A, j_{wA} * S_A + w_A)$
14. **for** each $j_{wB} \in [1, w_B + 1]$ **do**
15. $WB = slice\ (WA,\ j_{wB} * S_B, j_{wB} * S_B + w_B)$
16. $temp_1 = FFT(WB, val)$
17. $temp_2 = temp_2 + temp_1$
18. **end for**
19. $Sample_{TEF} = temp_2/j_{wB}$
20. $Data_{TEF} \leftarrow Sample_{TEF}$
21. reset $temp_1$, $temp_2$ and WB
22. **end for;**
23. rest WA
24. **end for;**
25. Return $Data_{TEF}$

Acknowledgment

Journal Pre-proofs

The authors would like to thank the Process Systems Engineering Group at Cranfield University for advice and support.

References

[1] M. Firouzi, S.H. Hashemabadi, Exact solution of two phase stratified flow through the pipes for non-Newtonian Herschel-Bulkley fluids, Int. Commun. Heat Mass Transf. 36 (2009) 768–775. https://doi.org/10.1016/j.icheatmasstransfer.2009.03.018.

[2] S. Chakraborty, P.K. Das, A unique methodology of objective regime classification for two phase flow based on the intensity of digital images, Exp. Therm. Fluid Sci. 99 (2018) 537–546. https://doi.org/10.1016/j.expthermflusci.2018.07.037.

[3] Y. Zhou, L. Yang, Y. Lu, X. Hu, X. Luo, H. Chen, Flow regime identification in gas-solid two-phase fluidization via acoustic emission technique, Chem. Eng. J. 334 (2018) 1484–1492. https://doi.org/10.1016/j.cej.2017.11.050.

[4] B. Wu, M. Firouzi, T. Mitchell, T.E. Rufford, C. Leonardi, B. Towler, A critical review of flow maps for gas-liquid flows in vertical pipes and annuli, Chem. Eng. J. 326 (2017) 350–377. https://doi.org/10.1016/j.cej.2017.05.135.

[5] L. Liebenberg, J.P. Meyer, Objective classification of two-phase flow regimes, Heat Transf. Eng. 29 (2008) 1–2. https://doi.org/10.1080/09647040701677706.

[6] E. Barrau, N. Rivière, C. Poupot, A. Cartellier, Single and double optical probes in air-water two-phase flows: Real time signal processing and sensor performance, Int. J. Multiph. Flow. 25 (1999) 229–256.

Journal Pre-proofs

https://doi.org/10.1016/S0301-9322(98)00042-1.

[7] M.S.G. Nandagopal, E. Abraham, N. Selvaraju, Advanced neural network prediction and system identification of liquid-liquid flow patterns in circular microchannels with varying angle of confluence, Chem. Eng. J. 309 (2017) 850–865. https://doi.org/10.1016/j.cej.2016.10.106.

[8] Y. Yan, L. Wang, T. Wang, X. Wang, Y. Hu, Q. Duan, Application of soft computing techniques to multiphase flow measurement: A review, Flow Meas. Instrum. 60 (2018) 30–43. https://doi.org/10.1016/j.flowmeasinst.2018.02.017.

[9] D. Peng, B. Merriman, S. Osher, H. Zhao, M. Kang, A PDE-based fast local level set method, J. Comput. Phys. 155 (1999) 410–438. https://doi.org/10.1006/jcph.1999.6345.

[10] Q. Geng, X. Zhu, J. Yang, X. You, Y. Liu, C. Li, Flow regime identification in a novel circulating-turbulent fluidized bed, Chem. Eng. J. 244 (2014) 493–504. https://doi.org/10.1016/j.cej.2014.01.102.

[11] S.H. Park, S.D. Kim, Wavelet transform analysis of pressure fluctuation signals in a three-phase fluidized bed, Korean J. Chem. Eng. 18 (2001) 1015–1019. https://doi.org/10.1007/BF02705635.

[12] Y. Zhou, L. Yang, Y. Lu, X. Hu, X. Luo, H. Chen, Flow regime identification in gas-solid two-phase fluidization via acoustic emission technique, Chem. Eng. J. 334 (2018) 1484–1492. https://doi.org/10.1016/j.cej.2017.11.050.

[13] G.H. Roshani, S.A.H. Feghhi, A. Mahmoudi-Aznaveh, E. Nazemi, A. Adineh-Vand, Precise volume fraction prediction in oil-water-gas multiphase flows by

means of gamma-ray attenuation and artificial neural networks using one detector, Meas. J. Int. Meas. Confed. 51 (2014) 34–41. https://doi.org/10.1016/j.measurement.2014.01.030.

[14] S. Ghosh, D.K. Pratihar, B. Maiti, P.K. Das, Identification of flow regimes using conductivity probe signals and neural networks for counter-current gas-liquid two-phase flow, Chem. Eng. Sci. 84 (2012) 417–436. https://doi.org/10.1016/j.ces.2012.08.042.

[15] A.J. Roman, P.J. Kreitzer, J.S. Ervin, M.S. Hanchak, L.W. Byrd, Flow pattern identification of horizontal two-phase refrigerant flow using neural networks, Int. Commun. Heat Mass Transf. 71 (2016) 254–264. https://doi.org/https://doi.org/10.1016/j.icheatmasstransfer.2015.12.033.

[16] J.Y. Lee, M. Ishii, N.S. Kim, Instantaneous and objective flow regime identification method for the vertical upward and downward co-current two-phase flow, Int. J. Heat Mass Transf. 51 (2008) 3442–3459. https://doi.org/10.1016/j.ijheatmasstransfer.2007.10.037.

[17] J.P. Schlegel, P. Sawant, S. Paranjape, B. Ozar, T. Hibiki, M. Ishii, Void fraction and flow regime in adiabatic upward two-phase flow in large diameter vertical pipes, Nucl. Eng. Des. 239 (2009) 2864–2874. https://doi.org/10.1016/j.nucengdes.2009.08.004.

[18] B. Santoso, Indarto, Deendarlianto, S.W. Thomas, The identification of gas-liquid co-current two phase flow pattern in a horizontal pipe using the power spectral density and the artificial neural network (ANN), Mod. Appl. Sci. 6 (2012) 56–67. https://doi.org/10.5539/mas.v6n9p56.

Journal Pre-proofs

[19] J.E. Julia, B. Ozar, J.J. Jeong, T. Hibiki, M. Ishii, Flow regime development analysis in adiabatic upward two-phase flow in a vertical annulus, Int. J. Heat Fluid Flow. 32 (2011) 164–175. https://doi.org/10.1016/j.ijheatfluidflow.2010.09.003.

[20] Z. Li, G. Wang, M. Yousaf, X. Yang, M. Ishii, Flow structure and flow regime transitions of downward two-phase flow in large diameter pipes, Int. J. Heat Mass Transf. 118 (2018) 812–822. https://doi.org/10.1016/j.ijheatmasstransfer.2017.11.037.

[21] L.H. Tsoukalas, M. Ishii, Y. Mi, A neurofuzzy methodology for impedance-based multiphase flow identification, Eng. Appl. Artif. Intell. 10 (1997) 545–555. https://doi.org/10.1016/s0952-1976(97)00037-7.

[22] Y. Mi, M. Ishii, L.H. Tsoukalas, Vertical two-phase flow identification using advanced instrumentation and neural networks, Nucl. Eng. Des. 184 (1998) 409–420. https://doi.org/10.1016/S0029-5493(98)00212-X.

[23] T. Xie, S.M. Ghiaasiaan, S. Karrila, Artificial neural network approach for flow regime classification in gas-liquid-fiber flows based on frequency domain analysis of pressure signals, Chem. Eng. Sci. 59 (2004) 2241–2251. https://doi.org/10.1016/j.ces.2004.02.017.

[24] R. Antony, M.S.G. Nandagopal, S. Rangabhashiyam, N. Selvaraju, Probabilistic neural network prediction of liquid- liquid two phase flows in a circular microchannel, J. Sci. Ind. Res. (India). 73 (2014) 525–529. https://doi.org/http://nopr.niscair.res.in/handle/123456789/29196.

[25] M.S. Giri Nandagopal, N. Selvaraju, Prediction of liquid-liquid flow patterns in a

Journal Pre-proofs

y-junction circular microchannel using advanced neural network techniques, Ind. Eng. Chem. Res. 55 (2016) 11346–11362. https://doi.org/10.1021/acs.iecr.6b02438.

[26] Q. Xu, S. Ye, W. Liu, Y. Chen, Q. Chen, L. Guo, Intelligent identification of steam jet condensation regime in water pipe flow system by wavelet multiresolution analysis of pressure oscillation and artificial neural network, Appl. Therm. Eng. 147 (2019) 1047–1058. https://doi.org/10.1016/j.applthermaleng.2018.11.005.

[27] N. Giannetti, M.A. Redo, Sholahudin, J. Jeong, S. Yamaguchi, K. Saito, H. Kim, Prediction of two-phase flow distribution in microchannel heat exchangers using artificial neural network, Int. J. Refrig. 111 (2020) 53–62. https://doi.org/10.1016/j.ijrefrig.2019.11.028.

[28] G.H. Roshani, E. Nazemi, M.M. Roshani, Identification of flow regime and estimation of volume fraction independent of liquid phase density in gas-liquid two-phase flow, Prog. Nucl. Energy. 98 (2017) 29–37. https://doi.org/10.1016/j.pnucene.2017.02.004.

[29] F. Gargiulo, S. Silvestri, M. Ciampi, G. De Pietro, Deep neural network for hierarchical extreme multi-label text classification, Appl. Soft Comput. J. 79 (2019) 125–138. https://doi.org/10.1016/j.asoc.2019.03.041.

[30] F.E. Fernandes Junior, G.G. Yen, Particle swarm optimization of deep neural networks architectures for image classification, Swarm Evol. Comput. 49 (2019) 62–74. https://doi.org/10.1016/j.swevo.2019.05.010.

[31] H. Mohsen, E.-S.A. El-Dahshan, E.-S.M. El-Horbaty, A.-B.M. Salem,

Journal Pre-proofs

Classification using deep learning neural networks for brain tumors, Futur. Comput. Informatics J. 3 (2018) 68–71. https://doi.org/10.1016/j.fcij.2017.12.001.

[32] L. Liu, B. Bai, Flow regime identification of swirling gas-liquid flow with image processing technique and neural networks, Chem. Eng. Sci. 199 (2019) 588–601. https://doi.org/10.1016/j.ces.2019.01.037.

[33] H. Yin, Y. Zhou, J. Zhao, Y.P. Du, Q.S. An, Y. Wang, L. Ma, Flow-pattern recognition and dynamic characteristic analysis based on multi-scale marginal spectrum entropy, Appl. Therm. Eng. 146 (2019) 30–38. https://doi.org/10.1016/j.applthermaleng.2018.07.077.

[34] J.W. Cooley, J.W. Tukey, An algorithm for the machine calculation of complex fourier series, Math. Comput. 19 (1965) 297. https://doi.org/10.2307/2003354.

[35] A. Malone, D. Chari, S. Cournane, I. Naydenova, A. Fagan, J. Browne, Investigation of the assessment of low degree (<50%) renal artery stenosis based on velocity flow profile analysis using Doppler ultrasound: An in-vitro study, Phys. Medica. 65 (2019) 209–218. https://doi.org/10.1016/j.ejmp.2019.08.016.

[36] V.A. White, A.A. Mahendru, Doppler ultrasound in obstetrics, Obstet. Gynaecol. Reprod. Med. 29 (2019) 158–163. https://doi.org/10.1016/j.ogrm.2019.03.004.

[37] S.G. Nnabuife, K.E.S. Pilario, L. Lao, Y. Cao, M. Shafiee, Identification of gas-liquid flow regimes using a non-intrusive Doppler ultrasonic sensor and virtual flow regime maps, Flow Meas. Instrum. 68 (2019) 101568. https://doi.org/10.1016/j.flowmeasinst.2019.05.002.

[38] R. Cobbold, Doppler ultrasound: Physics, instrumentation, and clinical applications, J. Biomed. Eng. 11 (1989) 528. https://doi.org/10.1016/0141-5425(89)90051-4.

[39] M.L. Sanderson, H. Yeung, Guidelines for the use of ultrasonic non-invasive metering techniques, Flow Meas. Instrum. 13 (2002) 125–142. https://doi.org/10.1016/S0955-5986(02)00043-2.

[40] S.G. Nnabuife, J. Whidborne, L. Lao, Two-phase gas-liquid flow regimes identification in an S-shape pipeline-riser using Doppler ultrasonic sensor, Cranfield Online Research Data (CORD) Repository, Cranfield, 2019. https://doi.org/10.17862/cranfield.rd.11369379.

[41] M. Kircheis, D. Potts, Direct inversion of the nonequispaced fast Fourier transform, Linear Algebra Appl. 575 (2019) 106–140. https://doi.org/10.1016/j.laa.2019.03.028.

[42] H.A. Ghiassirad, M. Aliyari Shoorehdeli, F. Farivar, Application of constrained learning in making deep networks more transparent, regularized, and biologically plausible, Eng. Appl. Artif. Intell. 85 (2019) 421–428. https://doi.org/10.1016/j.engappai.2019.06.022.

[43] H.S. Heo, B.M. So, I.L.H. Yang, S.H. Yoon, H.J. Yu, Automated recovery of damaged audio files using deep neural networks, Digit. Investig. 30 (2019) 117–126. https://doi.org/10.1016/j.diin.2019.07.007.

[44] S.G. Nnabuife, J. Whidborne, L. Lao, Y. Cao, Venturi multiphase flow measurement based active slug control, in: ICAC 2019 - 2019 25th IEEE Int. Conf. Autom. Comput., Chinese Automation and Computing Society in the UK -

Journal Pre-proofs

CACSUK, Lancaster, UK, 2019: pp. 1–6. https://doi.org/10.23919/IConAC.2019.8895212.

[45] V. Pierro, L. Troiano, E. Mejuto, G. Filatrella, Stochastic first passage time accelerated with CUDA, J. Comput. Phys. 361 (2018) 136–149. https://doi.org/10.1016/j.jcp.2018.01.039.

[46] G. Varoquaux, L. Buitinck, G. Louppe, O. Grisel, F. Pedregosa, A. Mueller, Scikit-learn, Mach. Learn. Res. 19 (2015) 29–33. https://doi.org/10.1145/2786984.2786995.

[47] C. Xiao, N. Chen, C. Hu, K. Wang, J. Gong, Z. Chen, Short and mid-term sea surface temperature prediction using time-series satellite data and LSTM-AdaBoost combination approach, Remote Sens. Environ. 233 (2019) 111358. https://doi.org/10.1016/j.rse.2019.111358.

[48] H. Wang, M. Hong, Online ad effectiveness evaluation with a two-stage method using a Gaussian filter and decision tree approach, Electron. Commer. Res. Appl. 35 (2019). https://doi.org/10.1016/j.elerap.2019.100852.

[49] L. Bach, G. Hasle, C. Schulz, Adaptive large neighborhood search on the graphics processing unit, Eur. J. Oper. Res. 275 (2019) 53–66. https://doi.org/10.1016/j.ejor.2018.11.035.

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

HIGHLIGHTS

- Ultrasonic Doppler sensors are non-radioactive and non-intrusive
- Flow regime classification in an S-shape riser using deep neural networks
- Deep neural networks operate on extracted features from Doppler ultrasonic signals
- Fast Fourier transforms obtain extracted features
- Twin-window feature extraction increases sample amount without information loss

Journal Pre-proofs

https://dspace.org/

Non-intrusive classification of gas-liquid flow regimes in an S-shaped pipeline-riser using doppler ultrasonic sensor and deep neural networks

Nnabuife, Somtochukwu Godfrey; Kuang, Boyu; Whidborne, James F.; Rana, Zeeshan

2020-07-26
Elsevier
https://doi.org/10.1016/j.cej.2020.126401
Nnabuife SG, Kuang B, Whidborne J, et al., (2021) Non-intrusive classification of gas-liquid flow regimes in an S-shaped pipeline-riser using doppler ultrasonic sensor and deep neural networks. Chemical Engineering Journal, Volume 403, January 2021, Article number 126401

Downloaded from a DSpace repository. https://dspace.org/